

CERES: CAPACITY EXPANSION AND RELIABILITY
EVALUATION SYSTEM

Shoichiro Nakamura
Professor of Nuclear Engineering

Spyridon Tzemos
Battelle Memorial Institute

Jeffrey Shih
Institute Researcher in Engineering

Mohammad Harunuzzaman
Graduate Research Associate

THE NATIONAL REGULATORY RESEARCH INSTITUTE
2130 Neil Avenue
Columbus, Ohio 43210

prepared for the

U.S. Department of Energy
Economic Regulatory Administration
Division of Regulatory Assistance

In partial fulfillment of
Grant No. DE-FG01-80RG10268
Modification A003

November 1981

This report was prepared by The National Regulatory Research Institute under a grant from the U.S. Department of Energy (DOE). The views and opinions of the authors do not necessarily state or reflect the views, opinions, or policies of DOE, or The National Regulatory Research Institute.

Reference to trade names or specific commercial products, commodities, or services in this report does not represent or constitute an endorsement, recommendation, or favoring by DOE or The National Regulatory Research Institute of the specific commercial product, commodity, or service.

FOREWORD

This report was prepared by The National Regulatory Research Institute (NRRI) under Grant No. DE-FG-01-80RG10268 from the U.S. Department of Energy (DOE), Economic Regulatory Administration, Division of Regulatory Assistance. The opinions expressed herein are solely those of the authors and do not reflect the opinions nor the policies of either the NRRI or the DOE.

The NRRI is making this report available to those concerned with state utility regulatory issues since the subject matter presented here is believed to be of timely interest to regulatory agencies and to others concerned with utility regulation.

Douglas N. Jones
Director

EXECUTIVE SUMMARY

The Capacity Expansion and Reliability Evaluation System (CERES) is a modular program developed at The National Regulatory Research Institute at The Ohio State University to find the economically optimum generating expansion plan for an electric utility with various constraints.

CERES is an entirely new program and has been developed to overcome the difficulties with the existing programs of the same kind. CERES has the following advantages: (1) the entire code is operated interactively with the user on a time sharing terminal, (2) input preparation and operation of the code are significantly easier than previously available codes, (3) computing time is much shorter, and (4) the accounting procedure adopted in the code is more suitable for analyzing investor-owned electric utilities.

Easy input preparation and code operation are achieved by (1) carefully designed input procedures for a time-sharing terminal, (2) procedures for easily updating data sets while running the program, and (3) an automatic iterative procedure to find the optimum solution. Higher accuracy and significantly shorter computing time are achieved by the use of two different numerical algorithms to simulate electric generating system operations. Two optional accounting systems are available in CERES: one is based on the salvage value of the plants; the other is based on the annualized fixed charge rate. The financial analysis capability of CERES allows the user to evaluate with a minimal effort the effect of the optimized expansion planning on the financial condition of the utility.

TABLE OF CONTENTS

	Page
INTRODUCTION	1
Chapter	
1 WHAT DOES CERES DO?	2
2 MODULES AND SUBMODULES	4
2.1 PLANT Submodule	5
2.2 LOAD Submodule	7
2.3 PREP Submodule	7
2.4 DYNO Submodule	15
2.5 FINAN Module	21
CONCLUSION	25
Appendix	
A PROBABILISTIC SIMULATION OF ELECTRIC GENERATING SYSTEM USING CUMULANTS	28
B PIECEWISE LINEAR POLYNOMIAL EXPANSION OF LOAD FUNCTIONS	39
C DYNAMIC PROGRAMMING OPTIMIZATION OF SYSTEM EXPANSION	40
D PROBABILISTIC SIMULATION WITH HYDROELECTRIC GENERATING UNITS	45
E MATHEMATICAL FORMULAS IN THE FINAN MODULE	51
F PROGRAM LISTING	73
G SAMPLE OUTPUT	257
REFERENCES	307

INTRODUCTION

The Capacity Expansion and Reliability Evaluation System (CERES) is a computer program for analyzing electric power generating systems including hydroelectric units and pump storage units. It is designed to find the economically optimum generating expansion plan for an electric utility with various constraints. In the past, WASP-I and WASP-II [1-4] were the only sophisticated programs in the public domain used for this purpose. WASP-II has been extensively used for a number of expansion studies at The Ohio State University [5-8]. Through the use of WASP-I as well as the effort to modify WASP-II, a need for major changes in WASP-II became obvious in order to make the system expansion study easier, especially for use by state regulatory agencies.

Rather than modifying WASP-II, however, an entirely new code, CERES, has been developed. The CERES is to be used exactly for the same purpose as WASP-II, but it is independent of WASP-II in its program development and has the following significant differences: (a) the entire code is operated on a time-sharing terminal in an interactive manner between the code user and the computer; (b) input preparations and operation of the code are much simpler than WASP-II; (c) running time is significantly reduced while accuracy is improved by adopting new computational techniques; (d) accounting formulas used in the code are more consistent with the traditional practices of investor-owned electric utilities of this country; (e) financial analyses of a utility based on the optimum expansion plan, or even suboptimum plan, may be performed at the user's option; and (f) future modifications of the code will be easier than for WASP-II.

The basic features of CERES are described in the remainder of this report. The details of technical aspects are given in several appendices. The operational procedure of the program is described in a separate report, CERES Operational Procedure.

CHAPTER 1

WHAT DOES CERES DO?

CERES analyzes alternative expansion plans of an electric utility generating system for up to 20 years considering all generating units involved. Those units are classified into two categories: scheduled unit types, and expansion unit types. Scheduled units include the generating units that exist at the beginning of the study period and those units that are firmly scheduled for future addition. Retirement of generating units is also considered. The expansion units refer to those candidate units for future addition that are not firmly scheduled at the beginning of the study period but that can be added to the system if selected for optimum system expansion. Optimization of system expansion is achieved by the forward dynamic programming so as to minimize the total objective function that is the sum of construction, operating, and maintenance costs during the entire period. The effects of discounting and escalating costs in each future year are taken into consideration in evaluating the total cost.

The addition and retirement of generating units are assumed to occur only once a year. If the system must be expanded, all possible combinations of new candidate generating units that satisfy certain criteria are evaluated. This evaluation requires simulation of the system energy production and reliability during each year, considering both the scheduled and expansion units. This evaluation is accomplished by probabilistic simulation [7,8]. Subsequently, these combinations of new candidate generating units are compared, according to the dynamic programming algorithm [1,8], in order to find the optimum solution.

Suppose the generating units for a year are all specified. Each year under consideration is divided into four periods (seasons). With the electric load characteristics provided as input for each period, the loss-of-load probability (LOLP) and total operating costs are calculated

using the probabilistic simulation technique. The effects of maintenance shutdown and forced outage of generating units are incorporated in the probabilistic simulation.

Since generating system expansion is assumed to occur only once a year, the decision making by dynamic programming is done only once a year. The system configurations in each year that do not meet the dynamic programming optimization path or do not meet the system reliability requirements are rejected, and only those expansion histories that satisfy the dynamic programming optimality and reliability requirements are recorded. Costs of the plans are compared if there are multiple acceptable expansion plans. The optimum expansion plan is the one with the minimum cost among all the plans.

CHAPTER 2

MODULES AND SUBMODULES

CERES consists of three major modules as follows:

INPUT Module

 PLANT Submodule

 LOAD Submodule

OPTIM Module

 PREP Submodule

 DYNO Submodule

FINAN Module

The INPUT module is divided into a PLANT submodule and a LOAD submodule. The two submodules may be run either sequentially or concurrently. The PLANT submodule reads the plant data for scheduled and expansion unit types through a time-sharing terminal and creates data files. It can also revise the plant data files that already exist. The LOAD submodule reads reference hourly load data and key data characterizing seasonal load demand; then, it creates the files of seasonal load duration curves as well as the load cumulants. The OPTIM module performs the remainder of the optimization analyses and has the following functions: (a) checks reliability of alternative configurations (generating mixes) and calculates production cost of electricity, (b) finds the least cost expansion plan by means of dynamic programming. Based on the results of the optimization, the FINAN module performs a financial analysis that is similar to that in the RAM model [12] but in a simpler form. The use of the FINAN module is optional, and it is run only after the DYNO submodule.

More details of the module and submodules are explained next.

2.1 PLANT Submodule

This submodule interactively reads the following data through a time-sharing terminal:

- (1) Starting year of the expansion study
- (2) Last year of the expansion study (total number of years of expansion study must not exceed 20)
- (3) Maximum number of expansion unit types declared: if the number declared is N, then the unit type numbers 1 through N are assigned for expansion unit types, and the unit type numbers N + 1 up to 200 are assigned for scheduled unit types
- (4) Individual plant data for thermal units include the following:
 - a. Upper limit and lower limit on the number of units of each expansion unit type in each year
 - b. Number of units of each scheduled unit type in each year
 - c. Capacity of the base block in MW
 - d. Total capacity in MW (base block plus peak block)
 - e. Maintenance requirement in days/year
 - f. Forced outage rate in fraction
 - g. Capital cost of construction in \$/kW
 - h. Fuel cost for the base block in \$/MWh
 - i. Fuel cost for the peak block in \$/MWh
 - j. Fixed operating and maintenance costs in \$/MW/year
 - k. Variable operating and maintenance costs in \$/MWh
 - l. Economic life of the unit in year
- (5) The data requirement for hydro generating units includes the following:
 - a. Upper and lower limits on the number of units of each hydro unit type for expansion candidates

- b. Number of units of each scheduled hydro unit of type i
- c. Base capacity in MW
- d. Maximum rated hydro capacity in MW
- e. Available in-flow hydro energy in each season in GWh
- f. Energy storage limit of reservoir in GWh
- g. Seasonal multipliers of base capacity and in-flow energy
- h. Maintenance required in days/year
- i. Forced outage rate in fraction
- j. Capital cost of construction in \$/KW
- k. Fixed operating and maintenance costs in \$/MW/year
- l. Variable operating and maintenance costs in \$/MWh
- m. Economic life of the unit in year
- n. Capital cost escalation rate in fraction

(6) Pump storage plant data include the following:

- a. Upper and lower bounds on the number of units of each pump storage unit type for expansion candidates
- b. Number of units of each scheduled storage unit in each year
- c. Generating capacity in MW
- d. Energy storage limit of the reservoir in GWh
- e. Pumping efficiency in fraction
- f. Generating efficiency in fraction
- g. Maintenance requirement in days/year
- h. Forced outage rate in fraction
- i. Capital cost of construction \$/KW

- j. Fixed operating and maintenance costs in \$/MW/year
- k. Variable operating and maintenance costs in \$/MWh
- l. Economic life of the unit
- m. Capital cost escalation rate in fraction

Items (1), (2), and (3) can be input only once when a file is created and cannot be changed once fixed. Items (4), (5) and (6) may be revised during creation of a new file. Item (3) may also be revised as often as necessary (see the flowchart in figure 2-1). The input and revision procedure is guided by the computer interactively. After running this submodule, the input data are stored in a file allocated to Unit 11.

2.2 LOAD Submodule

Before running the LOAD submodule, the user must prepare a reference hourly load data file allocated to Unit 20. This module then reads the following data from the time-sharing terminal:

- a. Energy multipliers for four seasons of each year
- b. Load factors for four seasons of each year

The input procedure is guided by the computer with ample chance for the user to correct and check the input data. After the user input procedure is completed, load duration curves and cumulants [9,10] are calculated and written in files for each season and each year (see figure 2-2). Three files are written when a run is completed (Unit 21: energy multiplier and load factors; Unit 25: load duration curves; Unit 26: load cumulants). By using the same program, it is also possible to revise the energy multipliers and load factors.

2.3 PREP Submodule

The PREP submodule performs the following functions: (a) reads and

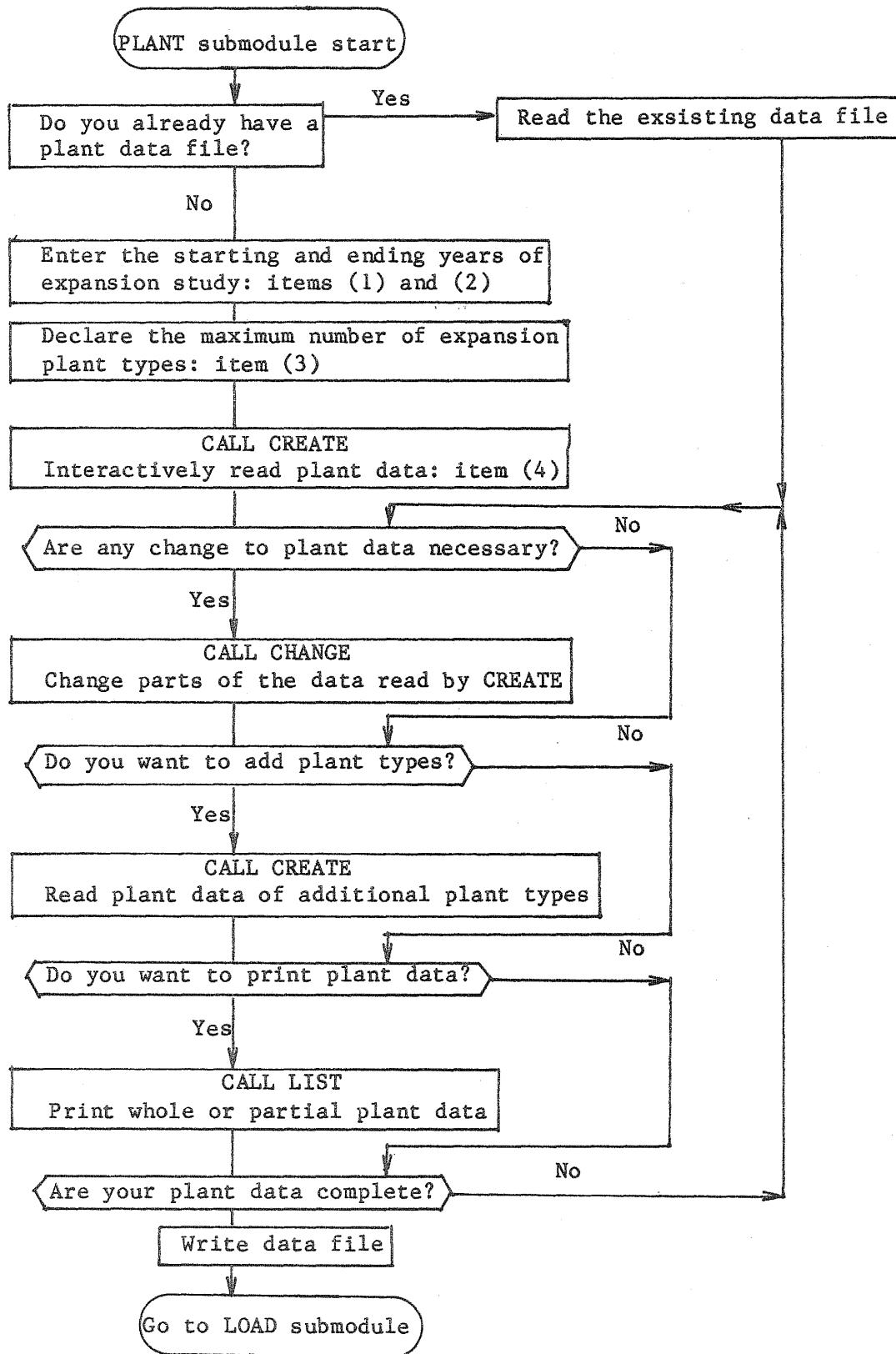


Figure 2-1 PLANT submodule flowchart

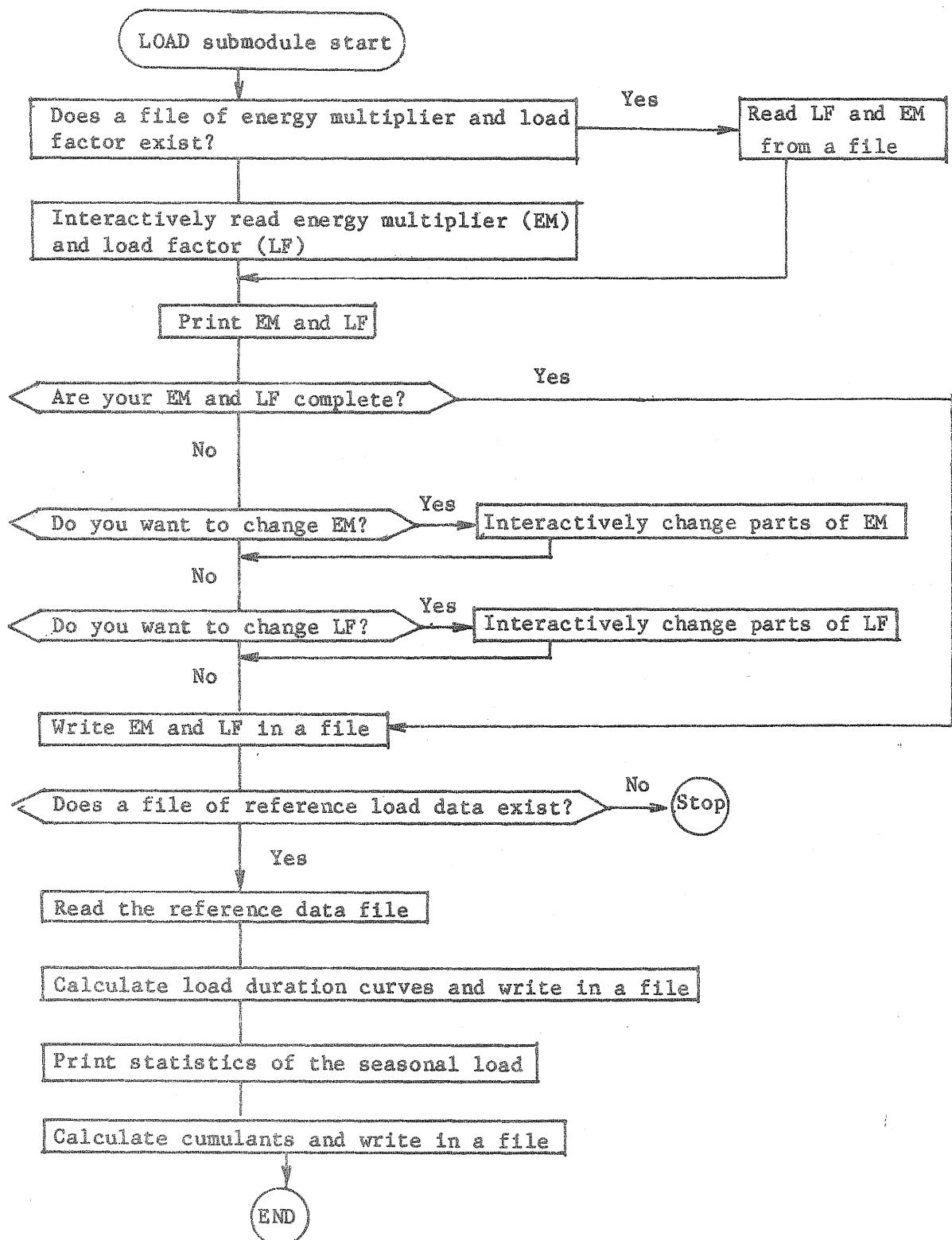


Figure 2-2 LOAD submodule flowchart

restructures the plant and load input data, (b) calculates plant seasonal maintenance schedule, (c) estimates minimum and maximum reserve margins, (d) finds the economic loading order, (e) calculates plant cumulants, f) estimates minimum energy costs, and (g) transfers all the above information to DYNO submodule. The flowchart in figure 2-3 shows the sequence of calculations.

2.3.1 Maintenance Schedule of Scheduled Units

The maintenance requirement for each unit type is specified by input in number of days of maintenance shutdown per year per unit. The PREP submodule determines the season in which the maintenance of each unit type takes place. The maintenance schedule of the expansion units is considered in the DYNO submodule.

The season of maintenance of each unit type is determined first for the largest unit type as follows: the excess capacity of scheduled units for each season j is calculated by

$$EXC_j = \sum_i EC_{ij} - PL_j \quad (1)$$

where

EC_{ij} : the total capacity of scheduled unit of type i in season j

PL_j : peakload demand in season j

EXC_j : excess capacity in season j

Seasonal values of EXC_j in the year are compared and the season in which EXC_j becomes the maximum is found. The maintenance of the largest unit type is assumed to take place in that season. Once the maintenance season is determined, the seasonal maintenance outage rate, MOR, for that unit type in that particular season is calculated by

$$MOR_{ij} = D_i / (365/4) \quad (2)$$

where D_i is the required number of maintenance days per year for the unit of type i . MOR_{ij} for other seasons is set to zero. After

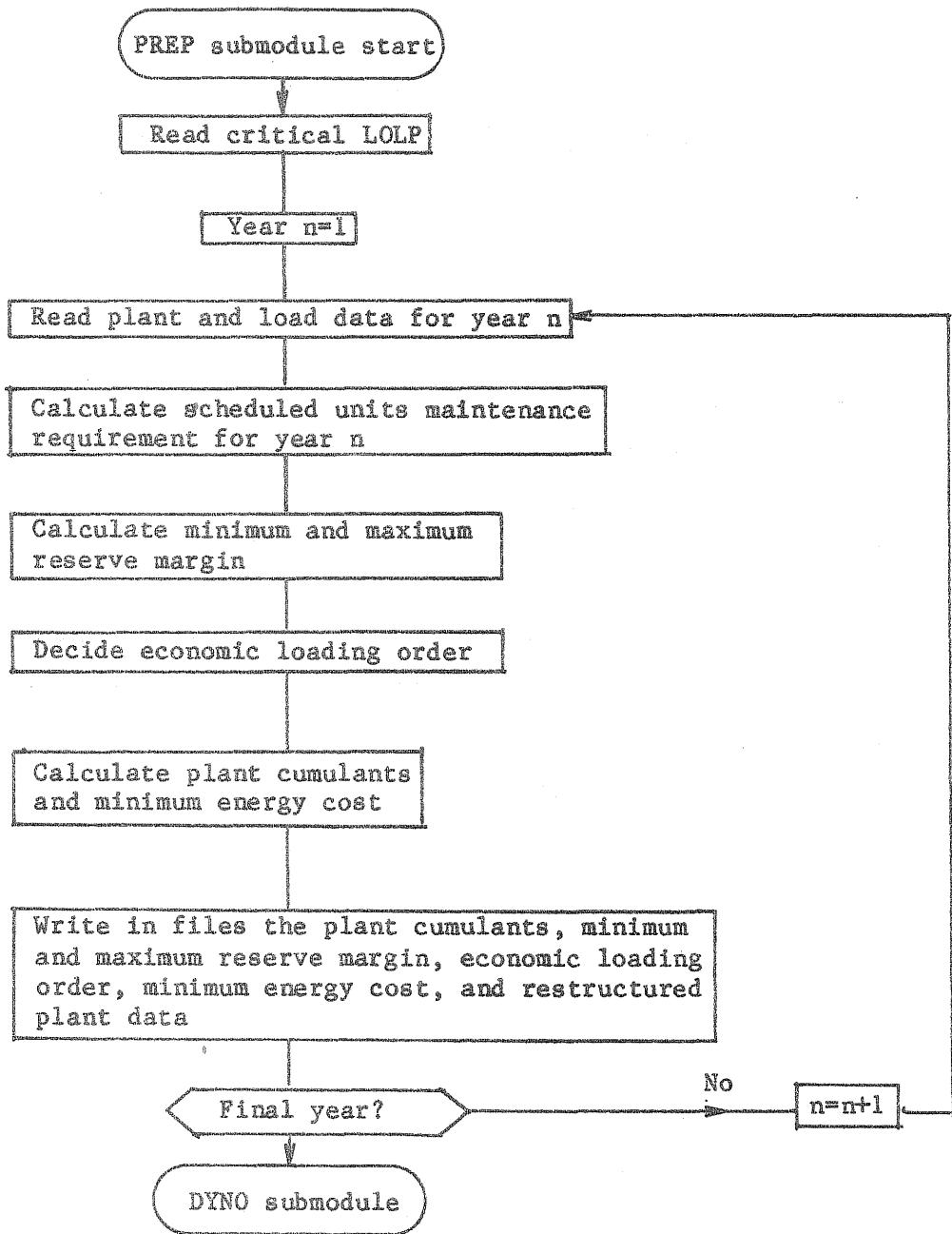


Figure 2-3 PREP submodule flowchart

MOR_{ij} for the unit type is calculated, EC_{ij} in equation (1) for that unit type is redefined, before processing the maintenance schedule of the next unit type, by

$$EC_{ij} = C_i(1-MOR_{ij}) \quad (3)$$

where C_i is the total capacity of a unit of type i . The same procedure is applied to the next largest unit type and repeated until all the scheduled units are considered.

2.3.2 Minimum and Maximum Reserve Margin Requirements

The minimum reserve margin requirement ($RMIN$) for each year is applied in the DYNO submodule to examine the generation configurations. The $RMIN$ is calculated based on the critical loss-of-load probability denoted here by $CLOLP$ in such a way that if a configuration does not satisfy the minimum reserve margin, then the configuration will never satisfy the critical LOLP requirement.

$RMIN$ is found by the following procedure. For each season of year n , $LOLP$ for the system consisting only of scheduled unit types is calculated first. If $LOLP$ calculated is greater than $CLOLP$, the expansion candidate units allowed for the year are hypothetically added one by one to the system until $LOLP$ becomes less than $CLOLP$. The forced outage rate of the added units is assumed to be equal to that of the lowest forced outage rate among all the allowed expansion units for the year. In adding the units, the smallest units available are added first, and then the next smallest units, and so on. This is repeated for each season of the year n . The maximum total capacity of expansion units thus added is found and denoted by CA_n . Then $RMIN_n$ is calculated by

$$RMIN_n = CS_n + CA_n - P_n \quad (4)$$

where

$RMIN_n$: $RMIN$ for year n

CS_n : The total capacity of the scheduled units for year n

CA_n : the total capacity of the expansion units hypothetically added as mentioned above

P_n : the annual peak of the load demand in year n

The generating system consisting of the scheduled units plus expansion units that are provisionally added is by no means related to the optimized generation configuration, but $RMIN$ thus calculated represents the lower bound for the reserve margin of the systems that satisfy the CLOLP criterion. Thus, $RMIN$ is used to exclude the generation configurations that do not satisfy the CLOLP criterion without calculating LOLP.

The maximum reserve margin requirements ($RMAX$) are calculated using the same procedure used for the calculation of $RMIN$ with the following modifications:

- (a) The CLOLP is replaced by a LOLP upper bound
- (b) The capacity of each added unit is assumed to be equal to the capacity of the largest expansion plant type
- (c) The forced outage rate of the scheduled units is assumed to be equal to the largest rate of all the expansion unit types

The $RMAX$ thus calculated serves as an upper bound for the reserve margin of the generating system. Any configuration with a larger reserve margin will be considered to be economically unjustifiable. It should be noted that both $RMIN$ and $RMAX$ may be overridden by the user during the execution of the DYNO submodule.

2.3.3 Plant Cumulant File for Scheduled Plants

The plant cumulants for the scheduled units are calculated from the plant moments (see appendix A-6). The plant moments are defined as

$$M_{ijv} = (1 - FOR_i)(EC_{ij})^v \quad (5)$$

where

v : integer

M_{ijv} : v-th moment of unit type i in season j

FOR_i : forced outage rate of unit type i

EC_{ij} : equivalent capacity of unit type i in season j as defined by equation (3)

For the plants with two capacity blocks, the moments for the first capacity block are calculated by (5), by replacing EC_{ij} with ECB_{ij} , defined as

$$ECB_{ij} = (1 - MOR_{ij})CB_i \quad (6)$$

where

MOR_{ij} : maintenance outage rate of unit type i in season j

CB_i : base block capacity of a unit of type i

2.3.4 Loading Order

The loading order of the base and peak blocks of the units is determined in the strict order of economic merit.

2.3.5 Lower Bound of Operating Cost (LBOC)

LBOC is the operating cost calculated for the hypothetical generating system that consists of the scheduled units and pseudounits for expansion. The pseudounit added in year n has the same capacity as CA_n defined in section 2.3.2, and its operating cost per MWh is equal to the lowest among all the expansion unit types that are allowed for year n. The forced outage rate and maintenance requirements of the pseudounit are assumed to be equal to the lowest in the same sense. In year n, there are at most n pseudounits, the capacities of which are CA_1, CA_2, \dots, CA_n . In this hypothetical system, the pseudounits have a higher loading order than the scheduled units. Therefore, the operating cost of the hypothetical system is lower than any generating

configuration consisting of real expansion units and so represents a lower bound of the operating cost for the system satisfying the minimum reserve margin requirement for the year. Thus, if the calculated annual operating cost of an arbitrary system is lower than LBOC, it will not satisfy the RMIN requirement. This principle is used in the fathoming process in the DYNO submodule.

2.4 DYNO Submodule

This submodule performs the economic evaluation of the alternative expansion plans and determines the best expansion policy for the system. The forward dynamic programming method is used to find the expansion plan that gives the minimum discounted cash flow of compounded capital and operating expenditures over the study period. The value of the objective function (total cost), which is to be minimized through dynamic programming, is calculated for each system configuration in each study year. Each system configuration is designated as a "state," and each year is designated as a "stage." DYNO provides the option of sensitivity analysis whereby the user can study the effects of allowing different expansion types, different discount rates, and different definitions of the objective function. DYNO produces a report of the optimum or a few of the next best suboptimum solutions at the user's request. A flowchart of the DYNO submodule is shown in figure 2-4.

2.4.1 Objective Function

CERES has two definitions for the objective function: traditional definition, and an alternative definition using the levelized fixed charge rate. Either of the two options can be chosen for sensitivity analysis at the time the program is executed.

(a) Traditional Definition

The objective function is defined as the sum of the operating costs and capital costs for construction minus the salvage value of the units,

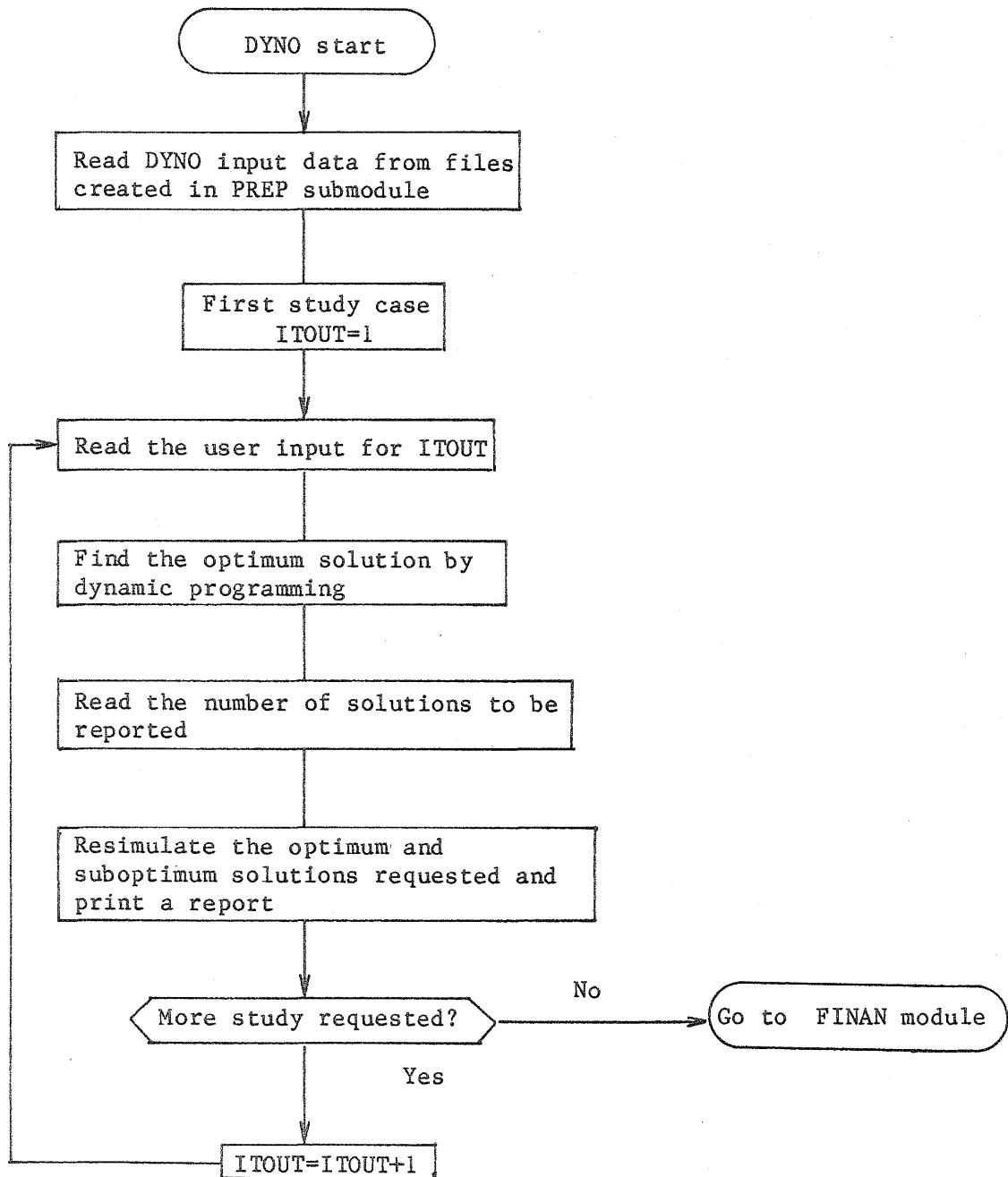


Figure 2-4 DYNO submodule flowchart

both of which are discounted to a specified base year. Discount and escalation rates are applied to each cost component. The salvage value considered in the objective function represents the credit given for the unused portion of the unit life. In other words, it is equivalent to the value of the generating units after depreciation. The salvage value is calculated by the straight-line-depreciation schedule. More details are given in appendix C.

(b) Fixed Charge Rate Option

The fixed charge rate (FCR) is defined as that factor, which when multiplied by the capital cost of a facility, produces a leveled annual fixed charge reflecting return on investment, taxes, insurance, retirement depreciation, and investment credit [11]. In this option, the objective function is the sum of (i) the present worth of the total operating cost for all the generating units during the study period, and (ii) the present worth of the fixed charges of all generating units that were added to the system during the study period.

2.4.2 Constraints of Dynamic Programming

The constraints imposed on the allowable generation configurations (states) for each year are divided into two types as follows:

- (i) User-specified constraints(input)
 - a) $\text{LOLP} < \text{CLOLP}$ (input)
 - b) Maximum reserve margin (input)
 - c) Maximum and minimum number of expansion units allowed in each type (input)
- (ii) Computational constraints
 - d) Minimum and maximum reserve margin limits, RMIN and RMAX
 - e) Tunnel constraints

The total number of states (generation configurations) in a year satisfying constraint (c) can be a large number, some of which would be rejected by constraints (a) and (b). The reserve margin is calculated first, since its calculation is much less expensive than for LOLP. If the reserve margin is less than RMIN, that state is immediately disqualified because constraint (a) will not be satisfied (see section 2.3.2). The same applies to the maximum reserve margin: in this case, the maximum LOLP specified in PREP will not be satisfied.

Tunnel constraints are used to limit temporarily the number of states to be considered in a run of dynamic programming optimization. They are used to reduce the computational time but should not affect the final result of optimization. Tunnel constraints consist of upper and lower bounds for the number of expansion units allowed each year. Because of the artificial tunnel constraints, dynamic programming (DP) optimization is performed with an automated iterative scheme (tunnel iteration). In the first iteration, DP is run with the first guess of tunnels. If the optimized trajectory runs on the tunnel boundary, the tunnel is changed and DP is rerun. This procedure is automatically iterated until the optimum trajectory stays inside the tunnels. The wider the tunnels, the faster the iterative convergence but more core space and computing time are required per iteration.

The constraints are applied in the following hierarchy: (1) maximum and minimum number of expansion units allowed, (2) RMIN, (3) maximum reserve margin, (4) LOLP, and (5) tunnels.

2.4.3 Fathoming Technique

Fathoming is a technique to disqualify some trajectory (history) of system expansion without completing the DP optimization. Suppose the minimum objective function at the final stage (year) among all the previous tunnel iterations is OBJ_L . In the next tunnel iteration, OBJ_L serves as an upper bound of the objective function.

Suppose the DP procedure reached the year n at which the objective function of a trajectory at state g is $\text{OBJ}_{g,n}$. Then, for the trajectories passing the state g , a lower bound for the objective function at the final stage N may be written as

$$\text{LB}_{g,n} = \text{OBJ}_{g,n} + \sum_{i=n+1}^N \text{LOBC}_i + \text{CC}_{g,n} \quad (7)$$

where

$\text{LB}_{g,n}$: the lower bound for the objective function at the final stage estimated at state g in year n

$\text{OBJ}_{g,n}$: the value of the objective function at state g in year n

LOBC_i : the minimum possible operating cost of a system in year n that satisfies the user constraints in stage i

$\text{CC}_{g,n}$: the minimum possible construction cost that satisfies the user constraints after the addition of state g and for all remaining stages $n+1, n+2, \dots, N$

N : the total number of stages (years) in the study period.

LOBC_i is calculated in the PREP module. $\text{CC}_{g,n}$ is estimated by first calculating the MW capacity that should be added in stages $n+1$ to N in order to reach the minimum reserve margin in the last stage (N) of the study period. This MW capacity is then multiplied by the minimum construction cost and is properly adjusted for escalation and discounting to base year. The state g in year n and those trajectories that pass the state will be no longer considered if

$$\text{LB}_{g,n} > \text{OBJ}_L \quad (8)$$

where

LB_n : as defined in equation (7)

OBJ_L : objective function upper bound (the minimum value of the objective function of all previous tunnel iterations).

2.4.4 Sensitivity Analysis

DYNO provides the user with the option of performing a sensitivity analysis on the following parameters.

- (a) The expansion unit types that are allowed in the optimization. A maximum of eight unit types can be considered in each DP optimization. These units are chosen from the list of expansion unit types that are specified in the INPUT module.
- (b) The discount rate may be changed to study its effect on the optimum solution.
- (c) The traditional definition of the objective function may be changed to the fixed charge option, and any fixed charge rate may be specified.
- (d) Critical LOLP may be changed from that used in the PREP submodule. However, consistency with minimum reserve margin calculations requires that the new value for the critical LOLP is larger than that used in the PREP submodule.
- (e) The minimum and maximum reserve margins calculated in PREP may be overridden.

2.4.5 Reports

After each DP optimization, DYNO provides a report for the following information:

- (a) The construction expenditures associated with each unit type for all years of the study period
- (b) Characteristics of the yearly operating costs associated with each unit type for all years of the study period
- (c) Optimum expansion plan:
 - (i) The number of units from each expansion candidate that come on line
 - (ii) The system LOLP and unserved energy
 - (iii) The operating and construction costs
 - (iv) The salvage value (if this option is used)
 - (v) The cumulative objective function up to and including year N

2.5 FINAN MODULE

The financial module of CERES is designed to perform financial analyses of the optimum or suboptimum solution obtained from the DYNO submodule. It uses as input data the capital budget, operating and maintenance costs, and interest rates for each year in the study period. Data based on financial practices and regulatory policies for a given utility are also used. Most of these data are specified by the user, and the rest is transferred from the PLANT and DYNO submodules.

This module produces various financial statements, that is, the income statement, the balance sheet, and the uses-and-sources-of-funds statements. It also calculates a set of financial ratios that may be used to evaluate the projected financial condition of the utility.

2.5.1 Input Data

The input data may be divided into three parts: (1) the user specified data, (2) the data obtained from the DYNO submodule, (3) the data obtained from PLANT submodule.

(1) User Specified Data are the following:

- a. Construction expenditures for scheduled units
- b. Gross plant value of scheduled units
- c. Construction time for expansion unit types
- d. Construction expenditure for each expansion unit type during each year of construction
- e. Interest rates for short and long debts
- f. Federal and state tax rates
- g. Proposed capital structure

- h. AFUDC (allowance of funds used during construction) rates
- i. Upper limits on debt and equity
- j. Percent of outstanding short- and long-term debts to be retired
- k. Economic life of each expansion plant
- l. Investment tax credit rate
- m. Accelerated depreciation method chosen for calculating income taxes
- n. Preferred dividend rate
- o. Dividend payout ratio
- p. Percent of construction work in progress (CWIP) allowed in the rate base
- q. Allowed rate of return on rate base
- r. Beginning value of gross plant in service
- s. Beginning value of accumulated book and tax depreciation
- t. Beginning value of accumulated CWIP and AFUDC
- u. Beginning value of outstanding short- and long-term debt
- v. Beginning value of outstanding common and preferred stocks

Data from DYNO submodule are the following:

- a. The optimum and suboptimum solution specifying the plant mix
- b. Operating, maintenance and fuel costs of all units for each year for a given solution

Data from PLANT submodule are the following:

- a. Construction cost of each unit type in \$/MW
- b. Total rated capacity of each unit type in MW
- c. Economic life of each unit type in year

2.5.2 Calculations and Reports

The calculations performed by the financial module may be divided into the following four parts.

(1) Rate Base Calculations

These calculations combine the gross plant in service, the book depreciation, the CWIP, and the AFUDC.

(2) Income Calculations

Once the rate base and the allowed rate of return are found, the allowed net operating income is calculated. All the information necessary to develop the income statement is found from income taxes, interest charges, operating income and operating expenses and finally operating revenues. In these calculations, the flow-through accounting method is used (the savings realized through differences in tax and book depreciation are passed through to the customer).

(3) Uses and Sources of Funds

Simultaneously with income calculations, the sources and uses of funds are obtained. The internal funds include net earnings while the external funds are a user specified mix of common and preferred stocks, and short-term and long-term debts. The uses include gross plant added, the common and preferred dividends, debt retirements, and the net increase in working capital.

(4) Ratio Analysis

Key financial rates such as debt/assets, debt/equity, assets/equity, and the interest coverage ratio are calculated by the module. All the ratios are related to the capital structure and leverage. The capital structure is adjusted so that the user constraints on the maximum allowed

new debt and equity requirements for each year are not violated.

The FINAN module output consists of the following statements:

- i) Income statement
- ii) Balance sheet statement
- iii) Uses-and-sources-of-fund statement

The balance sheet statement contains the utility's assets and liabilities in each year during the study period.

More detail of the calculations is given in appendix E.

CONCLUSION

CERES, a highly efficient computer program to find optimum electric generation planning has been completed. CERES is applicable to much larger electric systems than is WASP-II, yet has a greater accuracy with a significantly lower computing cost. This has been made possible by adopting new mathematical techniques developed recently that include the cumulant method, fathoming, and automatic tunnel iterations. The user's work required to operate the program is also reduced substantially by an interactive procedure between the user and the computer, operated through a time-sharing terminal.

CERES should be of particular use to electric utilities and regulators alike.

APPENDIX A

PROBABILISTIC SIMULATION OF ELECTRIC GENERATING SYSTEM USING CUMULANTS

A-1 Load Probability and Load Frequency Functions

A load demand probability function $L(x)$ defined for a time interval $T(\text{hrs})$, where x is the demand in MW, has the meaning that the difference $L(a) - L(b)$ represents the probability that the load demand x MW takes a value in the interval, $a < x < b$, namely

$$P(a < x < b) = L(a) - L(b) \quad (1)$$

Assuming that $L(x)$ has the first derivative, the function $f(x) = -dL(x)/dx$ is called the "load frequency function." The probability that the demand x' takes a value in $x < x' < x + dx$ is written in the form

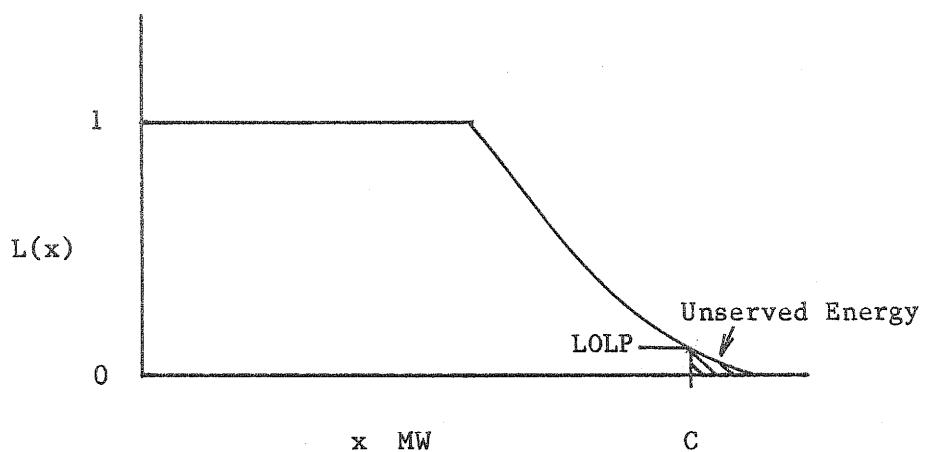
$$P(x < x' < x + dx) = f(x)dx \quad (2)$$

The load probability and load frequency functions are schematically illustrated in figure A-1. If the system capacity is denoted by C (MW) and we assume there is no generating unit outage, the value $L(C)$ is the loss-of-load probability (LOLP), that is, the probability that the load demand exceeds the capacity C . In figure A-1(a), LOLP is equal to the ordinate corresponding to $x=C$, which in figure A-1(b) is equal to the hatched area under the curve.

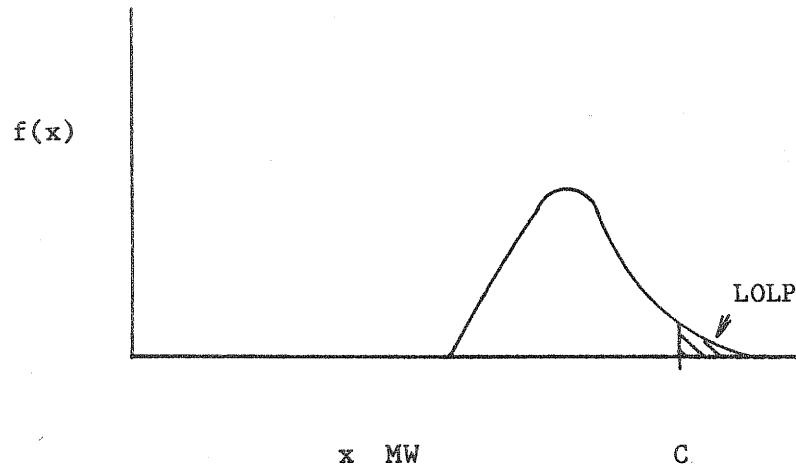
The hatched area under the curve of $L(x)$ in figure A-1(a) times T is the expected amount of demand that cannot be fulfilled by the system, and is called "unserved energy."

If an additional unit with capacity G is added to C , and if this unit is used only when the original capacity is exceeded by the load, then the energy generated by this additional unit is given by

$$T \int_C^{C+G} L(x) dx \quad (3)$$



(a) Load Probability Function



(b) Load Frequency Function

Figure A-1 Load probability and load frequency functions

where T is the length of time under consideration and the added unit is assumed to have no forced outage.

A-2 Equivalent Load and Equivalent Load Frequency Function

In the previous section we ignored the effect of outage. However, its effect may be taken into consideration by using an equivalent load function $L_e(x)$ or the corresponding equivalent load frequency function $f_e(x) = -dL_e(x)/dx$ as described below.

In evaluating the effect of outage of generating units, it is often more convenient to increase the load artificially rather than to decrease the total on-line capacity. It can be easily shown the two approaches are mathematically equivalent. To explain this further, suppose that the total capacity of a system on-line at a time is C MW, and the load function and the load frequency functions are given by $L(x)$ and $f(x)$, respectively.

Suppose also that an outage of z MW exists continuously all the time. Then, LOLP, unserved energy, and the amount of energy generated by an additional unit during the outage of z MW may be evaluated if $L(x)$ and $f(x)$ used in the previous section are all replaced respectively by

$$\begin{aligned} L_e(x) &= L(x-z) \\ f_e(x) &= f(x-z) \end{aligned} \tag{4}$$

The functions L_e and f_e are called "equivalent load" and "equivalent load frequency" functions, respectively. Note that in this equivalent approach, the capacity is unchanged in spite of the z MW outage, but rather the demand is increased by z MW all the time.

Extending this approach to more general cases, suppose that the outage of z MW happens with a probability q . Then, the equivalent load frequency function for the entire period of T becomes

$$f_e(x) = (1-q)f(x) + qf(x-z) \tag{5}$$

If partial outages of a unit can happen with different capacities, z_1 , z_2, \dots with probabilities, q_1, q_2, \dots , then the equivalent load frequency function becomes

$$f_e(x) = (1 - \sum_k q_k) f(x) + \sum_k q_k f(x-z_k) \quad (6)$$

A-3 Capacity Frequency Function

The capacity of an actual generating system fluctuates from time to time because of partial or total unscheduled outage of generating units. Although the change of the capacity on-line due to an outage is discrete in real systems, we assume the total capacity on-line is a continuous random variable. With this assumption, we define the on-line capacity function $g(x)$ which states that the probability that the capacity x' in $x < x' < x+dx$ is given by

$$P(x < x' < x+dx) = g(x)dx \quad (7)$$

The capacity frequency function $g(x)$ satisfies the normalization condition,

$$\int_0^C g(x)dx = 1 \quad (8)$$

where C is the total capacity of the system.

With the capacity frequency function $g(x)$ given, the equivalent load frequency function becomes

$$f_e(x) = \int_0^C f(x-y) g(C-y)dy \quad (9)$$

By defining the outage frequency function as

$$q(y) = g(C-y) \quad (10)$$

equation (9) may also be written as

$$f_e(x) = \int_0^C f(x-y)q(y)dy$$

A-4 Convolution of Units according to the Loading Order

The generating units in a system are loaded according to a "loading order." Suppose the system consists of N units with capacities $c_1, c_2, \dots, c_n, \dots, c_N$, and the loading order is in the increasing order of n . Suppose the function $L_{n-1}(x)$ represents the equivalent load function, in which the effect of outage of units $1, 2, \dots, n-1$ are already taken into consideration. Denoting the capacity frequency function for unit n by $g_n(x)$, the availability of the unit may be calculated by

$$p_n = \int_0^{c_n} yg_n(y)dy/c_n \quad (11)$$

Then the energy generated by the unit n during the time interval T becomes

$$E_n = T p_n \int_{A_{n-1}}^{A_n} L_{n-1}(x)dx \quad (12)$$

where A_n is the capacity total of units $1, 2, \dots$ up to n , or equivalently

$$A_n = \sum_{k=1}^n c_k$$

and E_n is the energy generated by unit n . The equivalent load curve for the next unit $n+1$ is obtained by a convolution integral

$$L_n(x) = \int_0^{c_n} L_{n-1}(x-y) g_n(c_n-y) dy \quad (13)$$

or equivalently,

$$L_n(x) = \int_0^{c_n} L_{n-1}(x-y) q_n(y) dy \quad (14)$$

Since $q(y) = 0$ for $y > c_n$, the above equation may be rewritten as

$$L_n(x) = \int_0^{\infty} L_{n-1}(x-y) q_n(y) dy \quad (15)$$

Notice that the equivalent load function for the first unit is equal to the load demand probability function,

$$L_0(x) = L(x)$$

Thus, once the actual load demand $L(x)$ is given, the equivalent load functions are successively generated by convolving the units in accordance with the loading order. Using the last equivalent load function $L_N(x)$ that includes the effect of outage of all the generating units, the LOLP and unserved energy are calculated as explained in sections A-1 and A-2.

The successive convolution of generating units may be performed in terms of the equivalent load frequency function as

$$f_n(x) = \int_0^{\infty} f_{n-1}(x-y) q_n(y) dy \quad (16)$$

Transformation of $f_n(x)$ to $L_n(x)$ may be performed by integrating $f_n(x) = -dL_n(x)/dx$.

A-5 Deconvolution

In the previous section, generating units are convolved in the loading order. However, this sequence of convolution is necessary only if the energy generated by each unit needs to be calculated by equation (12). In other words, if only $f_n(x)$ or $L_n(x)$ for a certain value of n is needed, the sequence of loading order is unimportant.

This principle may be used to withdraw a unit already convolved in $f_n(x)$ without repeating the sequence of convolution from the beginning. Suppose a unit $m < n$ is withdrawn from $f_n(x)$ without changing the unit numbering sequence. The new equivalent frequency function for unit n is denoted here by $f'(x)$, which could be obtained by convolving units 1, 2, .. $m-1$, $m+1, \dots, n$. Notice that $f_n(x)$ is related to $f'(x)$ by

$$f_n(x) = \int_0^{c_m} f'(x-y) g_m(y) dy \quad (17)$$

because both $f_n(x)$ and $f'(x)$ are not affected by the loading order. Solving the above equation for f' is called "deconvolution." The numerical scheme for deconvolution varies depending on the method used for convolution, so this subject is postponed until section A-8.

Most generating units are divided into two capacity blocks, namely, base capacity and peak capacity, each of which has a different loading order. Outage occurrences of the two capacity blocks are not independent, so they cannot be convolved independently. In order to take the simultaneous effect of the outage of both blocks, the base block is first deconvolved and then the total of the base and peak blocks is convolved.

A-6 Cumulant Method

The Fourier transform of a frequency function, $f(x)$,

$$F(t) = \int_{-\infty}^{\infty} e^{itx} f(x) dx \quad (18)$$

is called the "characteristic function" of $f(x)$. The frequency function $f(x)$ may be recovered by the inverse Fourier transform as

$$f(x) = (1/2\pi) \int_{-\infty}^{\infty} e^{-itx} F(t) dt \quad (19)$$

Let us denote the n -th derivative of $F(t)$ by $F^{(n)}(t)$. It can be shown easily that the n -th moment is defined by

$$a_n = \int_{-\infty}^{\infty} x^n f(x) dx \quad (20)$$

and related to $F(t)$ by the simple relation as

$$F^{(n)}(0) = i^n a_n \quad (21)$$

To derive equation (21), we take the n -th derivative of equation (18) and set $t=0$. In the neighborhood of $t=0$, $F(t)$ may be expanded in Maclaurin's series (Taylor expansion about $t=0$) as

$$F(t) = 1 + \sum_{m=1}^{\infty} t^m F^{(m)}(0)/m! \quad (22)$$

or using equation (21),

$$F(t) = 1 + \sum_{m=1}^{\infty} (it)^m a_m / m! \quad (23)$$

Thus, the characteristic function is expressed in terms of moments of the frequency function $f(x)$.

Referring to the Maclaurin expansion of $\log(1+z)$, we have

$$\log(1+z) = z/1 - z^2/2 + \dots z^k/k + O(z^k) \quad (24)$$

If z in equation (24) is set using equation (23) as

$$z = F(t) - 1 = \sum_{m=1}^{\infty} (it)^m a_m / m! \quad (25)$$

then the left side of equation (24) becomes $\log F(t)$. Because z is a polynomial of (it) , the right side of equation (24) may be written in the form

$$\log F(t) = \sum_{v=1}^k k_v (it)^v / v + O(t^n) \quad (26)$$

In the above equation, k_v are coefficients called "cumulants," and found by introducing equation (25) into equation (24) and comparing the coefficient of t^v on the right side of equation (26) with that on the right side of equation (24). The cumulant k_n is a polynomial in a_1, a_2, \dots, a_n , and conversely a_n is a polynomial in k_1, k_2, \dots, k_n as shown next:

$$k_1 = a_1 = M \quad (27)$$

$$k_2 = a_2 - a_1^2 = \sigma^2$$

$$k_3 = a_3 - 3a_1 a_2 + 2a_1^3$$

$$k_4 = a_4 - 3a_2^2 - 4a_1 a_3 + 12a_1^2 a_2 - 6a_1^4$$

$$a_1 = k_1 \quad (28)$$

$$a_2 = k_2 + k_1^2$$

$$a_3 = k_3 + 3k_1k_2 + k_1^3$$

$$a_4 = k_4 + 3k_2^2 + 4k_1k_3 + 6k_1^2k_2 + k_1^4$$

In the above equations, M is the mean value of x and k_2 is the variance.

A-7 The Central Moments

The moments about the mean of a frequency distribution $f(x)$ are called the "central moments" and are given by

$$u_v = \int_0^{\infty} (x - M)^v f(x) dx \quad (29)$$

In terms of the central moments, the expressions of the cumulants become

$$k_1 = M \quad (30)$$

$$k_2 = u_2 = \sigma^2 \quad (31)$$

$$k_3 = u_3 \quad (32)$$

$$k_4 = u_4 - 3u_2^2$$

$$k_5 = u_5 - 10u_2u_3$$

$$k_6 = u_6 - 15u_2u_4 - 10u_3^2 + 30u_2^3$$

A-8 Convolution and Deconvolution Using Cumulants

Suppose the moments of a frequency distribution $f(x)$ are denoted by a_v , and those of another frequency function $q(x)$ are denoted by b_v . The convolution of $f(x)$ is given by

$$j(x) = \int_{-\infty}^{\infty} f(x-y)q(y)dy \quad (33)$$

The characteristic function of $j(x)$ becomes

$$\begin{aligned} J(t) &= \int_{-\infty}^{\infty} e^{itx} j(x)dx \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{itx} f(x-y)q(y)dydx \\ &= \int_{-\infty}^{\infty} e^{itx'} f(x')dx' \int_{-\infty}^{\infty} e^{ity} q(y)dy \\ &= F(t)Q(t) \end{aligned} \quad (34)$$

Thus, referring to equation (26), we have

$$\begin{aligned} \log J(t) &= \log F(t) + \log Q(t) \\ &= \sum_{v=1}^k (a_v + b_v) (it)^v/v \end{aligned} \quad (35)$$

where a_v and b_v are cumulants of $f(x)$ and $q(x)$. Denoting cumulants
@@36

of $j(x)$ by r_v , we have

$$r_v = a_v + b_v \quad (36)$$

Therefore, the convolution of two frequency functions is expressed by summation of cumulants of the same order.

Deconvolution is just inverse of convolution. Knowing the functions $j(x)$ and $q(y)$, the solution for $f(x)$ of the equation

$$j(x) = \int_{-\infty}^{\infty} f(x-y)q(y)dy$$

is called "deconvolution." If the cumulants of $j(x)$ and $q(y)$ are given by r_v and b_v , those of $f(x)$ are simply

$$a_v = r_v - b_v \quad (37)$$

The above equation shows that a deconvolution needs only one subtraction operation for each cumulant.

A-9 Reconstruction of Load Probability Function from Cumulants

In the previous section, it is shown that convolutions and deconvolutions are easily done by adding and subtracting cumulants. However, we need a method to reconstruct equivalent probability function or equivalent load function from the cumulants. For this purpose, the Gram-Charlier expansion is used. The probability distribution, $f(z)$, which has the cumulants, g_i , may be expanded as

$$\begin{aligned} f(z) &= N(z) - g_1 N^{(3)}(z)/3! + g_2 N^{(4)}(z)/4! - g_3 N^{(5)}(z)/5! \\ &\quad + (g_4 + 10g_1^2)N^{(6)}(z)/6! - (g_6 + 10g_1g_2)N^{(7)}(z)/7! \end{aligned} \quad (38)$$

$$\begin{aligned}
& + (g_6 + 56g_1g_3 + 35g_2^2)N^{(8)}(z)/8! \\
& - (g_7 + 84g_1g_4 + 126g_2g_3 + 280g_1^3)N^{(9)}(z)/9! \\
& + (g_8 + 120g_1g_5 + 210g_2g_4 + 126g_3^2 \\
& \quad + 2100g_1^2g_2)N^{(10)}(z)/10! + \dots
\end{aligned}$$

where z is the standardized variable defined by

$$z = (x - M)/\sigma \quad (39)$$

$N(z)$ is the standard normal probability distribution

$$N(z) = 1/(\sqrt{2\pi}) \exp[-z^2/2] \quad (40)$$

and g_i is related to the cumulants by

$$g_i = k_{i+2}/\sigma^{i+2} \quad (41)$$

The derivatives of $N(z)$ are given by

$$N^{(n)}(z) = (-1)^n H_n(z)N(z), \quad n = 0, 1, 2, \dots \quad (42)$$

where $H_n(z)$ are the Hermite polynomials. Some of the members are

$$H_0(z) = 1, \quad H_1(z) = z, \quad H_2(z) = z^2 - 1 \quad (43)$$

$$H_3(z) = z^3 - 3z, \quad H_4(z) = z^4 - 6z^2 + 3$$

$$H_5(z) = z^5 - 10z^3 + 15z, \quad H_6(z) = z^6 - 15z^4 + 45z^2 - 15$$

The load probability function is obtained by integrating equation (38) in accordance with equation (3).

APPENDIX B

PIECEWISE LINEAR POLYNOMIAL EXPANSION OF LOAD FUNCTIONS

When expressed in terms of cumulants, an equivalent load probability function $L_m(x)$ for a large x can involve a significant amount of relative error, particularly where $L_m(x)$ is very close to zero. Such is the case when LOLP for a highly reliable system is calculated. The piecewise polynomial expansion has been shown to be significantly more accurate than the other expansions [7].

The piecewise polynomial expansion of $L_m(x)$ needs grid points, $x_i = x_{i-1} + \Delta x$, where i is the grid number and Δx is the interval between two consecutive grids. Given the values $L_m(x_i)$ on all the grid points, $L_m(x)$ for x in the interval between x_i and x_{i+1} is approximated by the linear interpolation:

$$L_m(x) = [(x_{i+1}-x)L_m(x_i) + (x-x_i)L_m(x_{i+1})]/\Delta x$$

Therefore, the piecewise polynomial $L_m(x)$ for the entire range of x is defined by specifying $L_m(x_i)$ for all the grids.

The convolution is straightforward if the equivalent load curve for the previous unit is given in the form of the above equation. The deconvolution with piecewise polynomial expansion is written as

$$L_{m-1}(x_i) = [L_m(x_i) - (1-P_m)L_{m-1}(x_i-C_m)]/P_m$$

The above calculation is done recursively in the increasing order of i . For $x < 0$, $L_{m-1}(x) = 1$ is used. Because C_m is not a multiple of the interval, a piecewise linear interpolation is used to find the value $L_{m-1}(x-C_m)$.

APPENDIX C

DYNAMIC PROGRAMMING OPTIMIZATION OF SYSTEM EXPANSION

C-1 Definition of the Optimization Problem

In applying the dynamic programming to optimization of a generating system expansion, we view the generating system expansion as a discrete control problem. Therefore, it is convenient to define the problem using the language of control theory.

We assume that the generating system is expanded at the beginning of each year, and only once a year. The year number denoted by n is used as the stage variable; the first year of expansion is referred to by $n = 1$.

The generating system configuration in the year n is expressed by a state vector

$$x_n = [x_{1,n}, x_{2,n}, \dots, x_{k,n}, \dots]^T \quad (1)$$

where $x_{k,n}$ is the number of generating units of type k in the year n .

The control vector is defined by

$$u_n = [u_{1,n}, u_{2,n}, \dots, u_{k,n}]^T \quad (2)$$

where $u_{k,n}$ denotes the number of units of type k added to the system at the beginning of the year n . Using x_n and u_n , the system equation is defined as

$$x_n = x_{n-1} + u_n \quad (3)$$

The state variable $x_{k,n}$ is subject to the constraints in the form

$$a_{k,n} \leq x_{k,n} \leq b_{k,n} \quad (4)$$

where $a_{k,n}$ and $b_{k,n}$ are the lower and upper bounds of the number of units of type k in the year n . There are two main reasons why these constraints are important. First, the permissible number of units of a particular type may be restricted by technical and financial reasons such as (a) diversity of unit types and sizes is desired, (b) manufacturing of a type is limited, (c) policy constraints, (d) limitation due to fuel availability, and (e) already committed without optimization. Second, the constraints in the form of equation (4) are used to reduce overall computing time as explained in more detail next.

The constrained range $a \leq x \leq b$ used for reducing computing time is called a "tunnel." It is easy to see that the computational time quickly increases as the tunnel width for every stage increases. Conversely, the computational time is reduced as the tunnel width for every stage decreases. In using this device, it is important that the constraints do not distort the solution. This is achieved by an automatic iteration scheme as follows. First, the dynamic programming is run with an appropriate tunnel for each candidate at each stage. If the solution of the dynamic programming optimization is on the upper bound ($x_{k,n} = b_{k,n}$), this indicates that the truly optimized $x_{k,n}$ may still be above $b_{k,n}$. Therefore, a and b are both increased. A similar procedure is applied when $x_{k,n}$ is on the lower bound except that a and b are decreased. The dynamic programming is run with the new constraints. (If the constraints are set for a legitimate reason other than reducing computing time, a and b are not altered.) The above procedure is repeated until $x_{k,n}$ does not fall on the lower or upper bound. There is a trade-off between the tunnel width and the number of iterations required. As the tunnel width is reduced, more iterations become necessary. On the other hand, in order to reduce the iteration times, the tunnel width must be increased; thus, computational time for iteration is increased. In CERES, the tunnel iteration is carried out automatically with optimum width to minimize the overall computational time.

The optimization requires the objective function to be minimized. The objective function in the case of traditional definition is given by

$$L_n = \sum_{j=1}^n (C_j - R_j + O_j) \quad (5)$$

where L_n is the objective function at stage n , C_j is the total construction cost for the units that entered the system in year j , R_j is the salvage value of all the units that exist at the end of year j , and O_j is the total operating cost in year j . More details on how to calculate C_j , R_j , and O_j are explained in the next section. The objective function in the case of fixed charge rate is calculated according to reference 11.

In summary, the optimization problem is stated as follows:
Minimize L_n , subjecting to the system equation

$$x_n = u_n + x_{n-1} \quad (6)$$

and the constraints

$$a_{n,k} \leq x_{n,k} \leq b_{n,k} \quad (7)$$

C-2 Cost Calculation Formulas

The cost incurred each year is calculated in terms of dollars of a reference (or base) year. This is done in order to take the effects of escalation and discount into consideration. The combined present-worth and escalation factor, which is used in the later equations, is given by

$$Q = (1 + m)^P(1 + i)^{-n} \quad (8)$$

where

m : escalation rate

i : present-worth discount rate

n : number of years from the present-worth base year

p : number of years from the escalation base year

The value Q is calculated separately for each generation unit candidate.

The construction cost for the generating units that come on-line in year j is calculated by

$$C_j = \sum_t QL_{t,j} \cdot I_{t,t} \cdot MWC_t \cdot N_{t,j} \quad (9)$$

where

C_j : the present-worth value of the capital expenditure for year j

t : unit type

$QL_{t,j}$: the combined present-worth and escalation factor in the year j

$I_{t,t}$: capital cost per MW capacity for type t

MWC_t : the capacity of expansion unit of type t, MW

$N_{t,j}$: the number of units of expansion unit of type t added in year j

A credit is given at the end of each year for the unused portion of the unit life. The total salvage value for the system at the end of year j is given by

$$R_j = \sum_t P_{t,nyr} S_{t,j} \quad (10)$$

where R_j is the salvage value, a credit to objective function for all the units added in year j; $P_{t,nyr}$ is the present-worth factor at the

study horizon; $S_{t,j}$ is the computed salvage value for all units of expansion candidate t added in year j (straight-line depreciation). The present-worth factor and the salvage value are respectively given by

$$P_{t,nyr} = (1 + i_t)^{-nyr} \quad (11)$$

$$S_{t,j} = (1 - y_t/L_t) C_{t,j} \quad (12)$$

where

nyr : number of years from present-worth base year to the end of year j

L_t : economic life of the units of type t in year

y_t : the number of years that the unit has been used by the end of year t in year

i_t : the discount rate for capital expenditures

$C_{t,j}$: escalated but undiscounted construction cost for type t in year j

The operating cost in year j is given by

$$O_j = \sum_t QL_{t,j} (CST_t + NFCST_t) \quad (13)$$

where

$QL_{t,j}$: the combined present-worth and escalation factor for operating expenditures for year j and type t

CST_t : the fuel cost for type t

$NFCST_t$: the total non-fuel-operating cost for type t

The factor $QL_{t,j}$ is given by

$$QL_{t,j} = (1 + m_t)^n / (1 + i_t)^{n+1/2} \quad (14)$$

where

m_t : the escalation rate for operating expenditures

i_t : the discount rate for operating expenditures

APPENDIX D

PROBABILISTIC SIMULATION WITH HYDROELECTRIC GENERATING UNITS

D-1 Classification of Hydroelectric Generating Units

Hydroelectric generating units may be classified into three categories: (1) normal hydro, (2) emergency hydro, and (3) pump storage.

The amount of energy generated by a hydro unit is limited by the supply of water as well as by various reservoir constraints. Hydro generation is distributed throughout the year to make optimum use of the hydroelectric resources by minimizing fuel costs of the thermal system. No energy cost is assigned to hydroelectric systems. The capacity of a normal hydro unit may be further divided into two portions, one called "run-of-river capacity," and another, "peak capacity." The former is the same as the base block of thermal units, while the latter is used only for peak-shaving duty in order to reduce the cost of operating peak thermal units with a high fuel cost.

The hydro units that cannot be used continuously because of insufficient water but used only for peak-shaving purposes are called "emergency hydro units." They are usually at a reservoir with a limited amount of water. In CERES, the emergency hydro units are treated identically with the normal hydro units except that they do not have a run-of-river capacity.

With pump storage hydro units, water is pumped to a higher elevation during the periods of low demand. When the system demand is high, the water is allowed to flow back through a turbine and generate electricity. This reduces the use of thermal peak units using high-cost fossil fuel and increases the system reliability. The usage of pump storage units for generating electricity is identical with the emergency hydro units. However, the difference is that they add load to the remainder of the system during the periods of low load in order to pump up water to higher reservoirs.

D-2 Aggregate Treatment of Hydro Generating Units

The number of hydro generating units as well as pump storage units can be many. In CERES, they are classified and aggregated into the following three capacities:

- (1) Scheduled normal hydro capacity
- (2) Expansion normal hydro capacity
- (3) Scheduled and expansion pump storage capacity

The scheduled hydro capacity includes all the normal hydro units that exist already or are firmly committed at the time of the expansion study. The expansion hydro capacity includes all the normal hydro units whose addition is subject to an optimum expansion decision. The scheduled and expansion storage capacities follow the same definition as for the normal hydro capacities. Each of these capacities may vary from year to year.

Each aggregate capacity is treated in probabilistic simulation as if it were a single generating unit. This aggregate treatment does not cause any serious error in probabilistic simulations because the forced outage rate of hydro units is normally very low.

D-3 Probabilistic Simulation with Normal Hydro Capacities

As mentioned earlier, there are two normal hydro capacities, one aggregating the scheduled units and another aggregating the expansion units, each of which is divided into base and peak blocks. The probabilistic simulation of a generating system with normal hydro capacities starts with loading the base block of the normal hydro capacities. The maximum hydro energy available less the expected base generation is the available energy for the peak block. The available capacity factor of each peak hydro capacity is defined as

$$CFA = \frac{\text{available peak hydro energy}}{(\text{available capacity})(\text{capacity availability})(T)} \quad (1)$$

where T is the number of hours in the period of simulation. CFA is then used to estimate the loading position of the peak hydro capacities. The procedure is as follows.

After the base hydro capacity block is loaded, thermal units and peak hydro capacities are given next loading priorities. However, before each thermal unit is loaded according to the loading order, the expected capacity factor of the peak hydro capacity (CF) is estimated as if it were loaded instead of the thermal unit. The estimated capacity factor CF is then compared with CFA defined in equation (1).

If $CF > CFA$, it means that the peak hydro capacity is not sufficient to fulfill the demand at this loading priority. Therefore, the thermal unit under consideration is loaded. If $CF < CFA$, then loading priority is given to the peak hydro capacity. After the peak hydro capacity is convolved, the energy generated is calculated. The energy generated at this loading position is, in general, less than the available peak hydro energy. In order to match the energy generated to the energy available, a small adjustment of the loading position of the peak hydro capacity is made as follows.

Let us assume that a block of a thermal unit is loaded just prior to the peak hydro capacity under consideration. It is obvious that the ideal loading position for the hydro capacity is in between the present one and the loading position of the thermal unit loaded just prior to it. So, the thermal block is partitioned into two subblocks, one of which is off-loaded, allowing the hydrocapacity to shift to a higher loading position by the capacity of the off-loaded subblock of the thermal unit. The capacity of the off-loaded subblock is determined in such a way that the energy generated by the peak hydro capacity becomes nearly equal to the peak energy available. The off-loaded thermal subblock is then reloaded next to the peak hydro capacity.

The remaining hydro capacity is treated in the same way except when it is loaded immediately after the first peak hydro capacity. In case the two peak hydro capacities are consecutively loaded, the loading position adjustment is performed as if the two peak hydro capacities are combined into one capacity block.

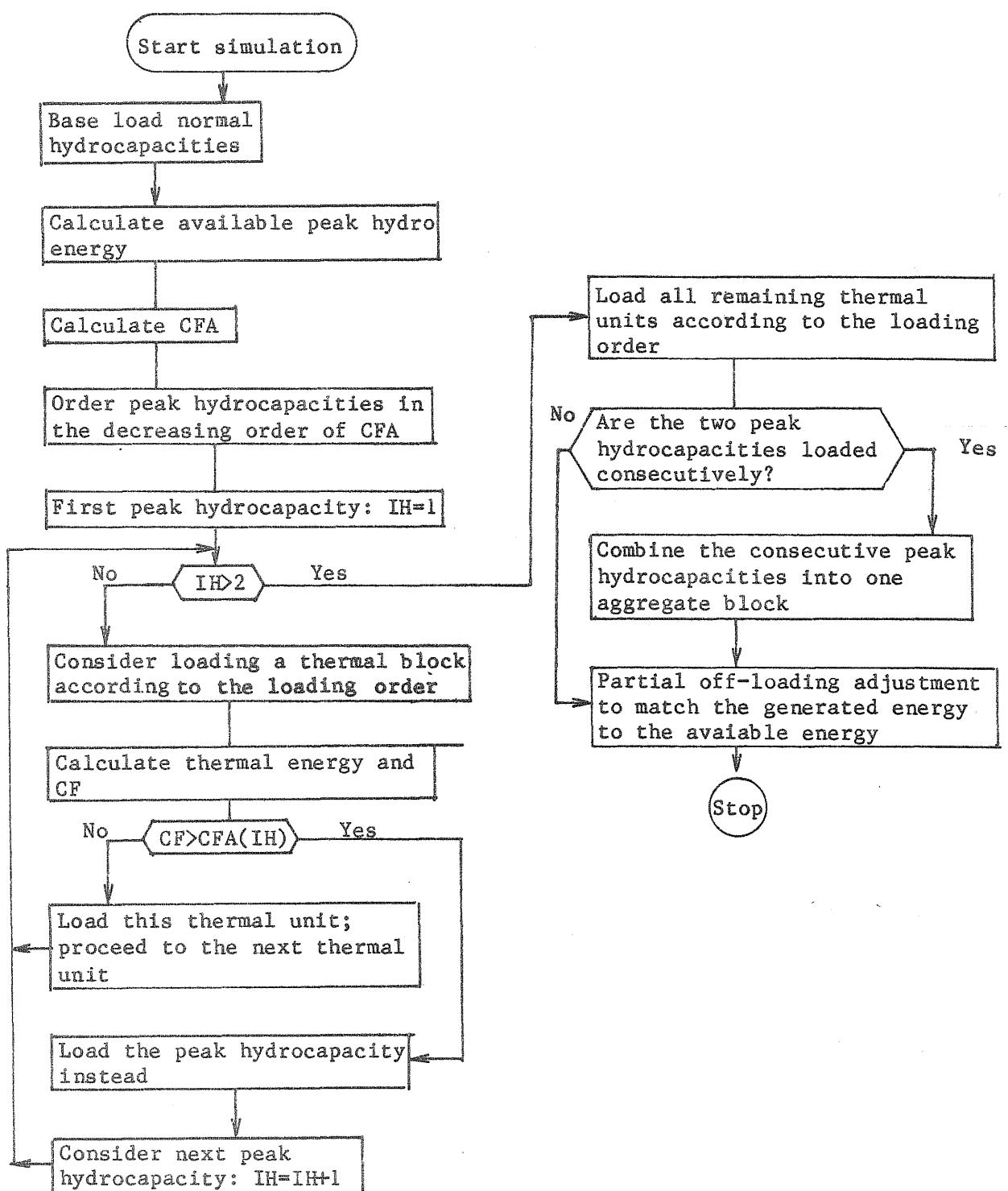


Figure D-1 Flowchart of probabilistic simulation with normal hydrocapacities

D-4 Probabilistic Simulation with Pump Storage Capacity

The electric generation with pump storage capacity work in the same way as the peak hydro capacities except that pump storage draw energy from the thermal capacity when the load demand is low. The available energy generated by the pump storage capacity is specified by input. So, the loading of the pump storage capacity for generation is treated in exactly the same way as the peak hydro capacities. If the pump storage capacity is loaded next to a peak hydro capacity, the loading position adjustment to match the energy generated to the available energy is done as if the peak hydro capacity and the pump storage capacity are combined into one capacity block.

The difference between the pump storage capacity and a peak hydro capacity arises from the fact that the former draws energy from thermal capacity during low demand periods. In order to simulate the additional load to the thermal capacity for pumping requirement, we consider an auxiliary load demand probability function as illustrated in figure D-1. The thin solid curve in figure D-1 is the load probability function without pumping requirement, while the thick curve is the auxiliary load curve including the effect of pumping. Notice that the curve $L_p(x)$ in $A < x < B$ is obtained by shifting $L(x)$ to the right by C_p , where C_p is the power required for pumping. At $x=B$, L_p is a vertical line. The hatched area is made equal to the total energy required for pumping.

The additional energy generated for pumping purpose by a block of a thermal unit at the i -th loading position is calculated by

$$PE_i = P_i P_{ps} T \int [L_{i-1}(x - C_p) - L_{i-1}(x)] dx \quad (2)$$

where PE_i is the additional energy generated, a and b are the lower and upper end of the loading position, T is the number of hours in the period under consideration, P_i is the availability of the thermal block, P_{ps} is the availability of the pumping capacity, and L_i is the i -th equivalent load probability function.

APPENDIX E

MATHEMATICAL FORMULAS IN THE FINAN MODULE

E-1 Rate Base Calculations

a. Gross Plant Values

The gross plant value of expansion units is found by summing the construction costs of individual units that are in service or will enter service in year n:

$$PLGSX_n = \sum_{i=1}^{imax} PLX_{i,n} \quad (1)$$

where

$PLGSX_n$: gross plant value of all expansion units
in year n

$PLX_{i,n}$: gross plant value of all the i-th
type in service in year n (supplied by DYN0)

$imax$: maximum number of expansion unit types con-
sidered (DYN0)

The gross plant value of all units (scheduled and expansion) is the sum of the above quantity and the gross plant value of the scheduled units:

$$PLGS1_n = PLGSX_n + PLGSF_n \quad (2)$$

where

$PLGS1_n$: value of gross plant in service for all units
in year n

$PLGSF_n$: gross plant value of scheduled units in
year n (user-supplied)

The gross plant added in year n is given by

$$PLGAI_n = PLGS1_n - PLGS1_{n-1} \quad (3)$$

Remark: n is an index of years in the utility planning horizon. Any value with the year subscript, $n=0$, is set to a beginning value specified by the user.

$L(x)$: load demand without pumping
 $L_p(x)$: load demand with pumping

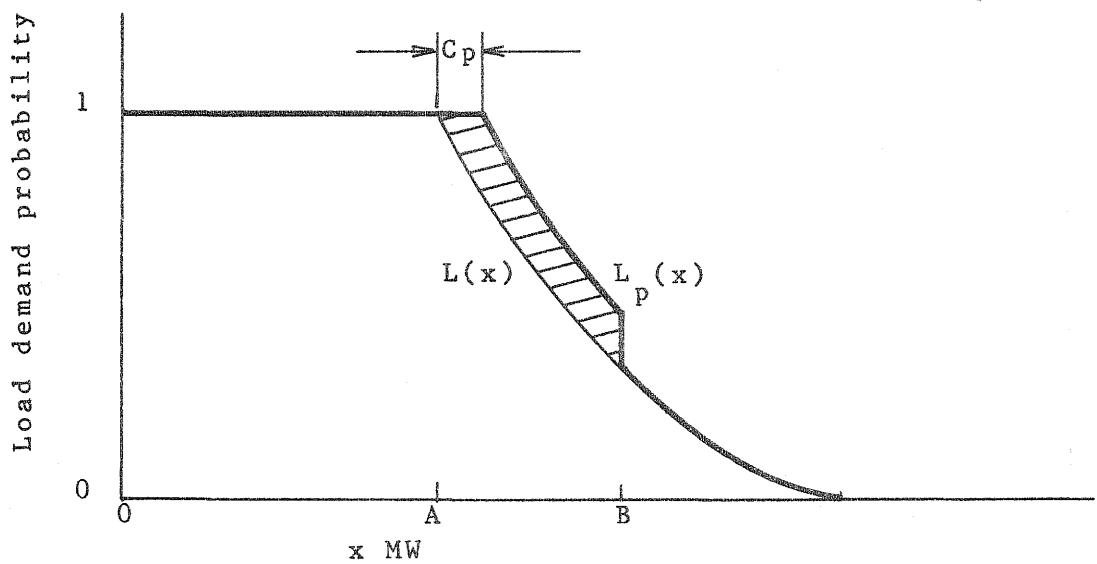


Figure D-2 Load demand probability functions with and without pumping requirement

b. Construction Expenditures

The construction expenditure for an expansion unit in a given year is identified by the unit type, i , and the number of years after the beginning of construction, j . Therefore, it is expressed as a function of i and j , denoted as $CEPLX_{i,j}$, and given as input by the user.

Obviously, $CEPLX_{i,j}=0$ if $j < 1$ or $j > t_i$, where t_i is the total number of years of construction. It is, however, more convenient to express j in terms of the year in which the expenditure is incurred, so the following relation will be substituted for j in the later equations:

$$j = t_i - m + n + 1$$

where

m : the year when the construction is completed

n : the year when the construction expenditure is incurred

t_i : the total number of years of construction

If the total number of units of the i -th type coming into service in year m is $IPLX_{i,m}$, the construction expenditure for such units in year n is $CEPLX_{i,j}$ times $IPLX_{i,m}$. The total construction expenditure for units of the i -th type in year n is obtained by summing the above quantity over all the years of completion, m :

$$CEPX_{i,n} = \sum_{m=1}^{n_{max}} (CEPLX_{i,j})(IPLX_{i,m}) \quad (4)$$

where

$CEPLX_{i,j}$: construction expenditure for a unit of type i during the j -th year of construction

$IPLX_{i,m}$: number of units of type i that come into service in year m (calculated from DYN0 output)

$CEPX_{i,n}$: total construction expenditure for expansion units of type i in year n

n_{max} : total number of study years (DYN0)

The beginning value of construction work in progress for expansion units is

$$CWIPXB = \sum_{i=1}^{imax} \sum_{m=1}^{nmax} t_{i-m+1} \sum_{k=1}^{t_{i-m+1}} (CEPLX_{i,k}) (IPLX_{i,m}) \quad (5)$$

where

$CWIPXB$: beginning value of cumulative construction work in progress for expansion units (user)

The total construction expenditure for all expansion units in year n , CEX_n , is simply the sum of $CEPX_{i,n}$ over all unit types:

$$CEX_n = \sum_{i=1}^{imax} CEPX_{i,n} \quad (6)$$

Finally, the total construction expenditure for all the units (scheduled and expansion) is given by

$$CE_n = CEF_n + CEX_n \quad (7)$$

where

CE_n : the total construction expenditure in year n

CEF_n : construction expenditure for all the scheduled units in year n

The cumulative construction expenditure up to year n is

$$CEC_n = \sum_{k=1}^n CE_k + PLGSB + CWIPB \quad (8)$$

$$CWIPB = CWIPFB + CWIPXB$$

where

$PLGSB$: beginning value of gross plant in service (user)

$CWIPB$: beginning value of cumulative construction work in progress for all scheduled and expansion units

$CWIPFB$: beginning value of cumulative construction work in progress for scheduled units (user)

c. Construction Work in Progress (CWIP)

The CWIP is the value of plant and equipment that are yet to come into service. The cumulative CWIP is given by the difference between the cumulative construction expenditure and the total plant in service (scheduled and expansion):

$$CWIPC_n = CEC_n - PLGS1_n \quad (9)$$

The CWIP increment is given by

$$CWIP_n = CWIPC_n - CWIPC_{n-1} \quad (10)$$

where

$CWIP_n$: increment of CWIP in year n

d. Allowance for Funds Used During Construction (AFUDC)

This is an allowance on construction expenditures that may be used as a part of the rate base when the plant goes into service, and it has two components. One belongs to the CWIP account and the other to the gross plant-in-service account. The CWIP component, AFDC1_n, is given by

$$AFDC1_n = (CWIPC_n) (AFDCR_n) (1 - FCWIP_n) \quad (11)$$

where

$AFDC1_n$: AFUDC in CWIP account in year n

$AFDCR_n$: AFUDC rate in year n

$FCWIP_n$: fraction of CWIP allowed in the rate base

The AFUDC in the plant-in-service account for expansion plant type i, coming into in year m is

$$AFCX2_{i,m} = \sum_{j=1}^{t_i} (CEPCX_{i,j}) (IPLX_{i,m}) (AFDCR_n) (1 - FCWIP_n) \quad (12)$$

$$CEPCX_{i,j} = \sum_{l=1}^i CEPXL_{i,l} \quad (13)$$

$$n = m - 1 - ITPL_i + j$$

where

- $CEPXC_{i,j}$: cumulative construction expenditure for unit type i in the j th year of construction
- $CEPLX_{i,j}$: annual construction expenditure for unit type i in the j th year of construction (user)
- $IPLX_{i,m}$: no. of units of type i entering service in year m (DYN)
- $AFDCR_n$: AFUDC rate in study year (user)
- $FCWIP_n$: fraction of CWIP allowed in the rate base in study year n (user)
- t_i : total construction time in years of a single unit of type i

If $n < 1$, $AFDCR_n$ and $FCWIP_n$ are set equal to $AFDCR_1$ and $FCWIP_1$, the respective values in the first year of the study period.

The total AFUDC for all expansion units entering service in year m is

$$AFCX2_m = \sum_{i=1}^{imax} AFDCX2_{i,m} \quad (14)$$

where

$$AFCX2_m : \text{total AFUDC in the plant-in-service account for expansion units}$$

The total AFUDC for all the scheduled and expansion units in year m is

$$AFDC2_m = AFDCF2_m + AFCX2_m \quad (15)$$

where

- $AFDC2_m$: AFUDC of all scheduled and expansion units entering service in year m
- $AFDCF2_m$: AFUDC of all scheduled units entering service in year m

To account for the two components of AFUDC properly, the AFUDC associated with the units coming into service is subtracted from the

cumulative AFUDC in the CWIP balance and added to the cumulative AFUDC in the plant-in-service balance:

$$AFDC1C_n = \sum_{k=1}^n (AFDC1_k - AFDC2_k) + AFDC1B \quad (16)$$

$$AFDC2C_n = \sum_{k=1}^n AFDC2_k + AFDC2B \quad (17)$$

where

$AFDC1C_n$: cumulative AFDC in CWIP balance

$AFDC2C_n$: cumulative AFDC in the plant-in-service balance

$AFDC1B$: beginning value of cumulative AFUDC in CWIP balance

$AFDC2B$: beginning value of cumulative AFUDC in the plant-in-service balance

e. Depreciation

(i) Straight-Line Method

Under the straight-line method, the depreciation factor for a unit of type i is given by

$$FDEP_i = 1/L_i \quad (18)$$

where

L_i : the expected economic life of the unit of type i

and $FDEP_i=0$ for all years beyond L_i in which the unit is useful if the unit's age exceeds the economic life.

The depreciation in year n for the units of the expansion type i is given by

$$DEPLX_{i,n} = \sum_{m=1}^n (PLX_{i,m}) (FDEP_i) \quad (19)$$

where

$PLX_{i,m}$: gross plant value of expansion units of type i entering service in year m

(ii) Sum of Year's Digits Method

The "sum of year's" for the unit type i including and up to the economic life, L_{sum_i} , is given by

$$L_{sum_i} = \sum_{k=1}^{L_i} k \quad (20)$$

where

L_i : life of a unit of type i

The depreciation factor, $FSUM_{i,j}$, is

$$\begin{aligned} FSUM_{i,j} &= (L_i - j) / L_{sum_i} \\ &= [L_i - (n - m)] / L_{sum_i} \end{aligned} \quad (21)$$

where

j: number of years that the unit has been in service
($j=n-m$)

Finally, the depreciation of units of the i-th type in year n is given by

$$DEPLX_{i,n} = \sum_{m=1}^n (PLX_{i,m}) (FSUM_{i,n-m}) \quad (22)$$

iii) Double Declining Method

In this method, a yearly depreciation factor is applied to the declining balance at the end of each year. This depreciation factor is given by

$$FDEP_i = 2/L_i \quad (23)$$

The declining balance in the beginning year of service is

$$BPL_{m,m} = (PLX_{i,m}) (1 - FDEP_i) \quad (24)$$

which is the gross plant value minus the depreciation in the beginning year, where m is the year of first service. For other years, the current year's depreciation is calculated by multiplying previous year's balance by the depreciation factor:

$$BPL_{m,n} = BPL_{m,n-1} (1 - FDEP_i) \quad (25)$$

The total depreciation for plants of the i -th type in year n is

$$DEPLX_{i,n} = \sum_{m=1}^n (BPL_{m,n-1} - BPL_{m,n}) \quad (26)$$

where

$DEPLX_{i,n}$: total depreciation of the units of type i in year n

The book depreciation and tax depreciation of plants are denoted by $DEPLX1_{i,n}$ and $DEPLX2_{i,n}$ respectively. Either of these quantities can be set to $DEPLX_{i,n}$ after the method of depreciation is selected.

The book and tax depreciation of expansion units are given respectively by

$$DEPX1_n = \sum_{i=1}^{imax} DEPLX1_{i,n} \quad (27)$$

$$DEPX2_n = \sum_{i=1}^{imax} DEPLX2_{i,n} \quad (28)$$

where

$DEPX1_n$: annual book depreciation expense in year n for expansion units

$DEPX2_n$: annual tax depreciation expense in year n for expansion units

i_{max} : the total number of expansion units

The accumulated book and tax depreciations are given by

$$DEP1C_n = \sum_{k=1}^n DEPX1_k + \sum_{k=1}^n DEPF1_k + DEP1B \quad (29)$$

$$DEP2C_n = \sum_{k=1}^n DEPX2_k + \sum_{k=1}^n DEPF2_k + DEP2B \quad (30)$$

where

$DEP1B$: beginning value of accumulated book appreciation

$DEP2C_n$: accumulated tax depreciation in year n

$DEP2B$: beginning value of accumulated tax depreciation

$DEPF1_n$: annual book depreciation expense in year n for scheduled units (user)

$DEPF2_n$: annual tax depreciation expense in year n for expansion units (user)

f. Net Plant Values

The book values of net plant in service are given by

$$PLNS1_n = PLGS1_n - DEP1C_n \quad (31)$$

The corresponding value of net plant added is

$$PLNA1_n = PLNS1_n - PLNS1_{n-1} \quad (32)$$

where

$PLNS1_n$: book value of net plant in service

$PLNA1_n$: book value of net plant added

g. Rate Base

The rate base for year n is given by

$$\begin{aligned} RATB_n &= PLNS1_n + AFDC2C_n \\ &\quad + FCWIPC_n (CWIPC_n + AFDC1C_n) \end{aligned} \quad (33)$$

where

FCWIPC_n : fraction of CWIP allowed in the rate base

E-2 Capital Requirements Calculations

The total capital expenditure required for year n is given by

$$\begin{aligned} CAPRQ_n &= CE_n + AFDC1_n - TXDEF_n + (GREV_{n-1}) (RWCAP_n) \\ &\quad - WKCAP_{n-1} + DEBLR_n + DEBSR_n \end{aligned} \quad (34)$$

where

CAPRQ_n: capital expenditure requirement in year n

CE_n : total construction expenditure in year n

AFDC1_n: AFUDC in CWIP account in year n

TXDEF_n: income taxes deferred in year n

GREV_n : gross operating revenue in year n

RWCAP_n: an estimate of the ratio of working capital
in year n to gross operating revenue in year n-1
(user)

WKCAP_n: working capital in year n

DEBLR_n: outstanding long term debt to be retired in year n

DEBSR_n: outstanding short term debt to be retired in year n

The long-term and short-term debts, and the preferred and the common stocks are given by multiplying CAPRQ_n by appropriate fractions:

$$DEBL_n = (CAPRQ_n) (FDEBL_n) \quad (35)$$

$$DEBS_n = (CAPRQ_n) (FDEBS_n) \quad (36)$$

$$STCOM_n = (CAPRQ_n) (FSTCOM_n) \quad (37)$$

$$STPR_n = (CAPRQ_n) (FSTPR_n) \quad (38)$$

where

$DEBL_n$: long-term debt borrowed in year n

$FDEBL_n$: fraction of total capital requirements
financed by long-term debts (user)

$DEBS_n$: short-term debt borrowed in year n

$FDEBS_n$: fraction of total capital requirement to
be financed by short-term debt in year n (user)

$STCOM_n$: common stock issued in year n

$FSTCOM_n$: fraction of capital requirement to be
financed by common stock in year n (user)

$STPR_n$: preferred stock issued in year n

$FSTPR_n$: fraction of capital requirement to be
financed by preferred stock (user)

If $DEBS_n$, $DEBL_n$, $STCOM_n$, or $STPR_n$ exceeds the user specified
limits, the user can either respecify these limits or stop the
calculations. The limits are denoted by

$DSLIM_n$: upper limit on short-term debt borrowed in year n (user)

$DLLIM_n$: upper limit on long-term debt borrowed in year n (user)

$COMLIM_n$: upper limit on common stocks issued in year n (user)

$PRLIM_n$: upper limit on preferred stocks issued in year n (user)

The total new debt borrowed in year n is given by

$$DEBN_n = DEBS_n + DEBL_n \quad (39)$$

where

$DEBN_n$: total new debt borrowed in year n

The total outstanding long-term debt and short-term debt are respectively given by

$$DEBLT_n = DEBLT_{n-1} + DEBL_n - DEBLR_n \quad (40)$$

$$DEBST_n = DEBST_{n-1} + DEBS_n - DEBSR_n \quad (41)$$

where

$DEBLT_n$: total outstanding long-term debt in year n

$DEBST_n$: total outstanding short-term debt in year n

The debt retirements in year n are given by

$$DEBLR_n = \sum_{m=1}^n (DEBL_m) (FDEBLR_{m,n}) \quad (42)$$

$$DEBSR_n = \sum_{m=1}^n (DEBS_m) (FDEBSR_{m,n}) \quad (43)$$

where

$FDEBLR_{m,n}$: fraction of long-term debt borrowed in year m to be retired in year n

$FDEBSR_{m,n}$: fraction of short-term debt borrowed in year m to be retired in year n

The total outstanding debt is

$$DEBT_n = DEBST_n + DEBLT_n \quad (44)$$

where

$DEBT_n$: total outstanding debt in year n

The total debt retirement in year n is

$$DEBR_n = DEBSR_n + DEBLR_n \quad (45)$$

E-3 Income and Expenses

a. Net Income

The allowed operating income is given by

$$OPINC_n = (RATB_n) (ARATE_n) \quad (46)$$

where

$OPINC_n$: allowed operating income in year n

$RATB_n$: rate base in year n

$ARATE_n$: allowed rate of return in year n

The total interest charge on debts in year n is the sum of the total interest charge on short-term debts and the total interest charge on long-term debts:

$$CDEBT_n = CDEBS_n + CDEBL_n \quad (47)$$

where

$CDEBS_n$: total interest on short-term debts in year n

$CDEBL_n$: total interest on outstanding long-term debts in year n

The total interest charges on short and long-term debts are calculated respectively by

$$CDEBS_n = \sum_{m=1}^n DEBS_m (1 - \sum_{k=m}^{n-1} FDEBSR_{m,k}) RDEBS_m \quad (48)$$

$$CDEBL_n = \sum_{m=1}^n DEBL_m (1 - \sum_{k=m}^{n-1} FDEBLR_{m,k}) RDEBL_m \quad (49)$$

where

$DEBS_n$: short-term debt borrowed in year n

$DEBL_n$: long-term debt borrowed in year n

$FDEBSR_{m,k}$: fraction of the short-term debt borrowed in year m that retires in year k (user)

$FDEBLR_{m,k}$: fraction of the long-term debt borrowed in year m that retires in year k (user)

RDEBS_n : interest rate on short-term debt in year n

RDEBL_n : interest rate on long-term debt in year n

The income before interest is given by

$$BINT_n = OPINC_n + (AFDC1_n) (IAFC) \quad (50)$$

where

IAFC = 1 if AFUDC is considered as income, 0 otherwise

BINT_n : income before interest

OPINC_n: allowed operating income

AFDC1_n: AFUDC component of CWIP in year n

The net income is calculated by subtracting interest for short- and long-term debts from the income before interest:

$$ERNET_n = BINT_n - CDEBS_n - CDEBL_n \quad (51)$$

where

ERNET_n: net income

b. Dividends

The total preferred dividend is given by

$$DIVPR_n = \sum_{k=1}^n (STPR_k) (RDIVP_k) \quad (52)$$

where

STPR_k : preferred stock issued in year k

RDIVP_k: preferred dividend rate in year k (user)

The earnings available to common is given by

$$ERNCOM_n = ERNET_n - DIVPR_n \quad (53)$$

Finally, the total common dividend and the total retained earning are given by

$$\text{DIVCOM}_n = (\text{ERNCOM}_n) (\text{FDIVC}_n) \quad (54)$$

$$\text{ERNRT}_n = (\text{ERNCOM}_n) (\text{FERN}_n) \quad (55)$$

where

FDIVC_n : fraction of common earning to be paid as common dividends in year n (dividend payout ratio)

FERN_n : fraction of common earnings retained in year n

c. Taxes and Tax Credits

The investment tax credit is given by

$$\text{TXCR}_n = (\text{CE}_n) (\text{FCE}_n) (\text{RTXCR}_n) \quad (56)$$

where

TXCR_n : investment tax credit in year n

CE_n : construction expenditure in year n

FCE_n : fraction of CE_n allowed for tax credit (user)

RTXCR_n : investment tax credit rate in year n (user)

The income before federal and state income taxes is given by

$$\text{BINC}_n = \text{ERNET}_n - (\text{AFDC1}_n) (\text{IAFC}) + \text{TXSPD}_n \quad (57)$$

where

BINC_n : income before federal and state income tax

TXSPD_n : federal and state income taxes paid in year n

The total of federal and state income taxes paid in year n is

$$TXSPD_n = (TXINC_n)(RTXST_n + RTXFD_n) - TXCR_n \quad (58)$$

where

$RTXST_n$: state income tax rate in year n (user)

$RTXFD_n$: federal income tax rate in year n (user)

$TXINC_n$: taxable income in year n

$TXINC_n$ is given by

$$TXINC_n = BINC_n - (DEP2_n - DEP1_n) \quad (59)$$

Simultaneous solution of equations (57) through (59) is necessary to obtain $BINC_n$, $TXINC_n$ and $TXSPD_n$.

The federal and state income taxes paid are respectively given by

$$TXFD_n = (RTXFD_n)(TXSPD_n) / (RTXFD_n + RTXST_n) \quad (60)$$

$$TXST_n = (RTXST_n)(TXSPD_n) / (RTXFD_n + RTXST_n) \quad (61)$$

where

$TXFD_n$: federal income tax in year n

$TXST_n$: state income tax in year n

The property tax is given by

$$TXP_n = (PLGS1_n + AFDC2C_n)(RTXP_n) \quad (62)$$

where

$RTXP_n$: property tax rate in year n

TXP_n : property tax in year n

$PLGS1_n$: gross plant in service in year n

$AFDC2C_n$: AFUDC (cumulative) in the plant in service account in year n

The deferred income tax is given by

$$TXDEF_n = (DEP2_n - DEP1_n)(RTXST_n + RTXFD_n)(IDEF) \quad (63)$$

where

$TXDEF_n$: deferred income tax in year n

$IDEF = 1$ if normalized accounting is used, 0 otherwise

The cumulative deferred income tax is given by

$$TXDEFC_n = TXDEFB + \sum_{m=1}^n TXDEF_m \quad (64)$$

where

$TXDEFC_n$: cumulative deferred income tax in year n

$TXDEFB$: beginning value of deferred income tax

d. Operating Expenses and Revenues

The total operating expense is given by

$$OPEXP_n = OMC_n + FUEL_n \quad (65)$$

where

$OPEXP_n$: total operating expense in year n

$DEP1C_n$: cumulative depreciation expense using the regular depreciation method

OMC_n : total operating and maintenance cost in year n

$FUEL_n$: total fuel cost in year n

The gross operating revenues, $GREV_n$, is set equal to

$$GREV_n = OPEXP_n + TXSPD_n + TXDEF_n + OPINC_n \quad (66)$$

where

GREV_n : gross operating revenue in year n

The operating revenue tax is given by

$$\text{TXREV}_n = (\text{GREV}_n) (\text{RTXREV}_n) \quad (67)$$

where

TXREV_n : operating revenue tax

RTXREV_n : revenue tax rate in year n

The total expense is the sum of operating expenses, taxes, tax deferred and operating revenue tax:

$$\begin{aligned} \text{TOTEXP}_n &= \text{OPEXP}_n + \text{TXSPD}_n + \text{TXDEF}_n + \text{TXCR}_n \\ &\quad + \text{TXREV}_n \end{aligned} \quad (68)$$

E-4 Sources and Uses of Funds

The internal source of funds is the sum of net income, depreciation, and deferred income tax:

$$\text{SRCIN}_n = \text{ERNET}_n + \text{DEPL}_n + \text{TXDEF}_n \quad (69)$$

where

SRCIN_n : internal source of funds

The external source of funds consists of common and preferred stocks, and short-term and long-term debts:

$$\text{SRCEX}_n = \text{STCOM}_n + \text{STPR}_n + \text{DEBS}_n + \text{DEBL}_n \quad (70)$$

where

SRCEX_n : external source of funds

The total source of funds is given by

$$\text{SRCTOT}_n = \text{SRCIN}_n + \text{SRCEX}_n \quad (71)$$

where

$SRCTOT_n$: total source of funds in year n

The uses of funds, USE_n , consists of annual construction expenditure, new AFUDC in the CWIP account, preferred and common dividends, and debt retirements:

$$USE_n = CE_n + AFDC1_n + DIVPR_n + DIVCOM_n + DEBR_n \quad (72)$$

The balance between sources and uses of funds is recorded as net increase in working capital:

$$CAPIN_n = SRCTOT_n - USE_n \quad (73)$$

where

$CAPIN_n$: balance between the total source of funds and the use of funds

E-5 Assets and Liabilities

The gross plant value (including AFUDC) is given by

$$PLGS2_n = PLGS1_n + AFDC2C_n \quad (74)$$

where

$PLGS2_n$: gross plant value including AFUDC

The net plant value is the gross plant value minus the accumulated depreciation:

$$PLNV_n = PLGS2_n - DEP1C_n \quad (75)$$

where

$PLNV_n$: net plant value

The net utility plant is equal to $PLNV_n$ plus the CWIP:

$$PLNUT_n = PLNV_n + CWIPC_n \quad (76)$$

where

$PLNUT_n$: net utility plant

The common equity, $EQCOM_n$, consists of common stocks and retained earnings:

$$EQCOM_n = STCOMC_n + ERNRTC_n \quad (77)$$

where

$$STCOMC_n = \sum_{k=1}^n STCOM_k + STCOMB \quad (78)$$

$$ERNRTC_n = \sum_{k=1}^n ERNRT_k + ERNRTB \quad (79)$$

and

$STCOMC_n$: total common stock held in year n

$ERNRTC_n$: accumulated retained earnings in year n

$STCOMB$: beginning value of common stocks

$ERNRTB$: beginning value of retained earnings

The total capital is the sum of common equity, outstanding preferred stocks, and long-term debt:

$$CAPTOT_n = EQCOM_n + STPRC_n + DEBLT_n \quad (80)$$

where

$$STPRC_n = \sum_{k=1}^n STPR_k + STPRB \quad (81)$$

and

$STPRC_n$: outstanding preferred stocks in year n

$STPRB$: beginning value of prepared stocks

$STPRC_n$: total preferred stock held in year n

$DEBLT_n$: outstanding long-term debt in year n

The total liability is the sum of total capital, outstanding short-term debt, and deferred income tax:

$$TLIAB_n = CAPTOT_n + DEBST_n + TXDEFC_n \quad (82)$$

where

$TLIAB_n$: total liabilities in year n

$DEBST_n$: outstanding short-term debt in year n

The net working capital, $WKCAP_n$, is given by

$$WKCAP_n = WKCAP_{n-1} + CAPIN_n \quad (83)$$

where

$WKCAP_n$: working capital in year n

$CAPIN_n$: net increase in working capital in year n

The total of assets is given by

$$ASSET_n = PLNUT_n + WKCAP_n \quad (84)$$

where

$ASSET_n$: total value of assets in year n

E-6 Ratio Analysis

The debt/asset ratio, debt/equity ratio, and the interest coverage ratio are respectively given by

$$DEBASS_n = DEBT_n / ASSET_n \quad (85)$$

$$DEBEQ_n = DEBT_n / EQTOT_n \quad (86)$$

$$COVRG_n = BINT_n / CDEBT_n \quad (87)$$

where EQTOT is the total equity given by

$$EQTOT_n = STCOM_n + STPR_n + ERNRT_n \quad (88)$$

and

DEBASS_n: debt/asset ratio of year n

DEBEQ_n : debt/equity ratio of year n

COVRG_n : interest coverage ratio of year n

BINT_n : income before interest in year n

Appendix F
PROGRAM LISTING

```

C      ALLOCATE F(FT11F001) DA(PARM·DATA)
C      ALLOCATE F(FT14F001) DA(KEY1·DATA)
C
C      PLANT DATA INPUT CODE
C
C      INTEGER YE,IP,IPNUM,IPMAX,JNUM,YEAR
C      INTEGER RESP, KEY1(42,18), YES
C      REAL PLANT(200,14)
C      COMMON NUF(200,20), IP,PLANT,INS,KEY1,IPNUM,IPMAX,JNUM
C      COMMON /CACAY/ NPEXP,IUP(30,20), NBYR, NTYR
C
C      DATA YES/3HYES/,YE/3HY /
C      DATA INP1,INP2/11,14/
C      JNUM=14
C      IPNUM=1
C      IPMAX=250
C      DO 2090 K=1,42
C      READ (INP2,115) (KEY1(K,J),J=1,18)
C 2090 CONTINUE
C 115 FORMAT (18A4)
C      WRITE (6,101)
C 101 FORMAT (//, *****, INPUT PROCEDURE FOR PLANT DATA*
C      *  STARTS ***** //, UNLESS OTHERWISE SPECIFIED */
C      * A PLANT CODE IDENTIFIES A THERMAL PLANT */
C      * DO YOU HAVE A FILE ALREADY FOR THE */
C      * PLANT DATA? /* ENTER YES OR NO. */,/2H ?)
C      READ (5,103) RESP
C 103 FORMAT (A3)
C      IF(RESP.EQ.'YES'.OR.RESP.EQ.'YE') GO TO 11
C
C      WRITE (6,311)
C 311 FORMAT (5X,5I1H SINCE A FILE DOES NOT EXIST, YOU CAN CREATE A FILE)
C      WRITE (6,411)
C 411 FORMAT (5X,5I1H FROM SCRATCH. A SET OF INSTRUCTIONS WILL APPEAR ON)
C      WRITE (6,511)
C 511 FORMAT (5X,5I1H THE TERMINAL TO WHICH YOU SHOULD RESPOND PROPERLY.)
C      PRINT 530
C      READ *,NBYR
C      PRINT 540
C      READ *, IEND
C      IF(IEND.LT.40) IEND=IEND+100
C 530 FORMAT (/10X, "ENTER THE FIRST YEAR OF EXPANSION (2 DIGIT ) .")
C 540 FORMAT (/10X, "ENTER THE LAST YEAR OF EXPANSION ")
C      NTYR=IEND+1-NBYR
C      IF(NTYR.GT.20) PRINT 550
C 550 FORMAT (/10X, "THE NUMBER OF YEARS OF EXPANSION EXCEEDS 20/")
C      * 10X, "REPEAT INPUT OF YEARS. ")
C      IF(NTYR.GT.20) GO TO 520
C      PRINT 610
C      PRINT 600
C 600 FORMAT(10X, "DECLARE THE MAXIMUM NUMBER OF EXPANSION PLANTS/")
C      * 10X, "(NU MORE THAN 20). /* ---")
C      READ 610, NPEXP
C 610 FORMAT(I2)
C 620 FORMAT(/10X, "PLANT CODE UP TO ",I2," ARE ASSIGNED/")
C      * 10X, "FOR EXPANSION PLANTS. ")
C      IF(NPEXP.GT.1) PRINT 620, NPEXP
C 645 FORMAT(I2)
C      CALL CREATE
C      GO TO 2020
C
C 11 READ (INP1,1400) IPNUM,NPEXP,NBYR,IEND
C 1400 FORMAT(4I10)
C      NTYR=IEND+1-NBYR
C      DO 1001 IP=1,IPNUM
C      READ (INP1,1422)(NUF(IP,N),N=1,20)

```

```

1422 FORMAT(40I2)          00000670
1001 READ (INP1,6420) (PLANT(IP,J),J=1,JNUM) 00000680
    DO 1003 IP=1,NPEXP 00000690
1003 READ (INP1,1422) (IUP(IP,N),N=1,20) 00000700
    REWIND INP1 00000710
C
C
2020 WRITE (6,105)          00000720
105 FORMAT (5X,30H ARE ANY CHANGES IN PLANT DATA, 00000730
+28H NECESSARY? ENTER YES OR NO.) 00000740
    READ (5,103) RESP 00000750
    IF (RESP.EQ.'YES.OR.RESP.EQ.YE') CALL CHANGE 00000760
    PRINT 107 00000770
107 FORMAT (5X,27H DO YOU WANT TO ADD PLANTS?, 00000780
+17H ENTER YES OR NO.) 00000790
    READ (5,103) RESP 00000800
    IF (RESP.EQ.'YES.OR.RESP.EQ.YE') CALL CREATE 00000810
    WRITE (6,185) 00000820
185 FORMAT (5X,'**** IF PRINT OF DATA IS NEEDED, ENTER YES.') 00000830
    READ (5,103) RESP 00000840
    IF (RESP.EQ.'YES.OR.RESP.EQ.YE') CALL LIST 00000850
    PRINT 8000 00000860
8000 FORMAT (5X,' IS YOUR PLANT DATA COMPLETE ? ENTER YES OR NO.') 00000870
    READ (5,103) RESP 00000880
    IF (RESP.EQ.'YES.OR.RESP.EQ.YE') GO TO 8300 00000890
    GO TO 2020 00000900
8300 REWIND INP1 00000910
    WRITE (INP1,1400) IPNUM,NPEXP,NBYR,IEND 00000920
    DO 8500 IP=1,IPNUM 00000930
    WRITE (INP1,1422) (NUF(IP,N),N=1,20) 00000940
    WRITE (INP1,6420) (PLANT(IP,J),J=1,JNUM) 00000950
8500 CONTINUE 00000960
    DO 8503 IP=1,NPEXP 00000970
8503 WRITE (INP1,1422) (IUP(IP,N),N=1,20) 00000980
    REWIND INP1 00000990
8420 FORMAT(5E15.5) 00001010
8590 FORMAT("// ATTNS THERMAL PLANT DATA FILE IS CREATED",/
# " OR REVISED ON UNIT 11") 00001020
    PRINT 8590 00001030
    PRINT 8600 00001040
8600 FORMAT(" **** END OF THERMAL PLANT INPUT PROCEDURE ****") 00001050
    PRINT 8700 00001060
8700 FORMAT("//, " DO YOU WANT TO INCLUDE HYDROELECTRIC",/
+ " AND PUMPED STOKAGE PLANTS",/ " IN YOUR INPUT DATA FILE?",/ 00001070
+ " ENTER YES OR NO."/2H ?) 00001075
    READ (5,103) RESP 00001076
    IF (RESP.EQ.'YES.OR.RESP.EQ.YE') CALL HYDPS 00001077
C    CALL LOAD1 00001078
    PRINT 8800 00001080
8800 FORMAT("//, "***** END OF PLANT DATA INPUT PROCEDURE *****") 00001081
    END 00001090
C
SUBROUTINE CREATE          00001100
-----THIS SUBPROGRAM CREATES A PLANT DATA 00001120
FILE FROM SCRATCH IF NONE EXISTED BEFORE. 00001130
USER CAN INPUT DATA FOR EACH PLANT 00001140
INTERACTIVELY 00001150
00001160
    INTEGER YE,IP,IPNUM,IPMAX,JNUM,YEAR 00001170
    INTEGER RESP, KEY1(42,18), YES 00001180
    REAL PLANT(200,14) 00001190
    COMMON NUF(200,20), IP,PLANT,INS,KEY1,IPNUM,IPMAX,JNUM 00001200
    COMMON /CACAC/ NPEXP,IUP(30,20), NBYR, NTYK 00001210
    DATA YES/3HYES/, YE/3HY / 00001220
    IEND=NBYR+NTYR-1 00001230
    10 WRITE (6,102) 00001240
                                         00001250
                                         00001260

```

```

102 FORMAT (//5X, " ENTER PLANT CODE NUMBER.")          00001270
103 FORMAT (4H      )          00001280
104 READ (5,104) IP          00001290
105 IF(IP.LT.0.OR.IP.GT.200) PRINT 2515          00001300
106   INPUT ERROR.      "
107 IF(IP.LT.0.OR.IP.GT.200) GO TO 10          00001310
108 FORMAT (I3)          00001320
109 FORMAT (1H 13)          00001330
110 C      .... ENTER ALL PLANT DATA          00001340
111 CALL NUNIT (1)          00001350
112 IF(IP.GT.1PNUM) IPNUM=IP          00001360
113 FORMAT (3H      )          00001370
114 DO 1002 K=2,JNUM          00001380
115 IF(K.EQ.9) GO TO 1002          00001390
116 WRITE (6,111) (KEY1(K+28,J),J=1,18)          00001400
117 FORMAT (1H 18A4)          00001410
118 WRITE (6,105)          00001420
119 FORMAT (11H      .      )          00001430
120 READ (5,106) PLANT(IP,K)          00001440
121 C      WRITE (6,206) PLANT(IP,K)          00001450
122 1002 CONTINUE          00001460
123 PLANT(IP,1)=FLOAT(IP)+0.00001          00001470
124 106 FORMAT (F10.5)          00001480
125 206 FORMAT (1H F10.5)          00001490
C      INP1=11          00001500
126 REWIND INP1          00001510
127 WRITE (INP1,1400) IPNUM,NPEXP,NBYR,IEND          00001520
128 1400 FORMAT(4I10)          00001530
129 DO 8500 IP=1,IPNUM          00001540
130 WRITE (INP1,1422) (NUF(IP,N),N=1,20)          00001550
131 1422 FORMAT(40I2)          00001560
132 WRITE (INP1,8420) (PLANT(IP,J),J=1,JNUM)          00001570
133 8420 FORMAT(5E15.5)          00001580
134 8500 CONTINUE          00001590
135 DO 8503 IP=1,NPEXP          00001600
136 8503 WRITE (INP1,1422) (IUP(IP,N),N=1,20)          00001610
137 REWIND INP1          00001620
C      ....PROMPT FOR ADDITION OF MORE PLANTS          00001630
138 WRITE (6,107)          00001640
139 107 FORMAT (5X,34H DO YOU WANT TO ADD ANOTHER PLANT?,          00001650
140 +17H ENTER YES OR NO.)          00001660
141 108 FORMAT (4H YYY)          00001670
142 READ (5,112) RESP          00001680
143 112 FORMAT (A3)          00001690
144 IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 10          00001700
145 RETURN          00001710
146 END          00001720
C      SUBROUTINE CHANGE          00001730
C      ....THIS SUBPROGRAM IS USED TO MODIFY EXISTING          00001740
C      PLANT DATA. ALL DATA MANIPULATIONS CAN          00001750
C      BE DONE INTERACTIVELY.          00001760
C
147 INTEGER YE,IP,IPNUM,IPMAX,JNUM,YEAR          00001770
148 INTEGER RESP, KEY1(42,18), YES          00001780
149 REAL PLANT(200,14)          00001790
150 COMMON NUF(200,20), IP,PLANT,INS,KEY1,IPNUM,IPMAX,JNUM          00001800
151 COMMON /CACA/ NPEXP,IUP(30,20), NBYR, NTYR          00001810
152 DATA YES/3HYES/          00001820
153 DATA YE/3HYE/          00001830

```

```

9 PRINT 400
READ(5,112) RESP
400 FORMAT("DO YOU WANT TO CHANGE ALL THE DATA OF A PLANT?"/
+ "ENTER YES OR NO.")
JREP=0
IF(RESP.EQ.YES.OR.RESP.EQ.YE) JREP=1
IF(JREP.EQ.1) CALL CREATE
IF(JREP.EQ.1) GO TO 3333
10 WRITE(6,101)
101 FORMAT(5X,"ENTER THE 3 DIGIT PLANT CODE FOR THE SINGLE DATA CHANGE")
102 FORMAT(14H ____)
READ(5,103) IP
IF(IP.LT.1.OR.IP.GT.IPNUM) GO TO 10
103 FORMAT(1I3)
11 WRITE(6,104)
104 FORMAT(5X,"ENTER DATA CODE IN 2 DIGIT, OR 99 FOR HELP.")
105 WRITE(6,105)
105 FORMAT(3W ____)
READ(5,106) JP
106 FORMAT(1I2)
IF(JP.EQ.99) GO TO 12
IF(JP.EQ.1) GO TO 15
GO TO 13
12 CALL HELP
GO TO 11
13 IF(JP.LT.2.OR.JP.GT.14) GO TO 11
WRITE(6,107) (KEY1(JP+28,KP),KP=1,18)
PRINT 180, PLANT(IP,JP)
180 FORMAT(5X,"THE VALUE BEFORE CHANGE=F10.5,")*
107 FORMAT(1H 18A4)
WRITE(6,108)
108 FORMAT(11M ____)
READ(5,109) PLANT(IP,JP)
109 FORMAT(F10.5)
GO TO 16
15 CALL NUNIT(3)
16 CONTINUE
WRITE(6,110) IP
110 FORMAT(5X,"ANY MORE CHANGE FOR PLANT CODE",I3,"?",/
+1H "ENTER YES OR NO.")
111 FORMAT(4H YYY)
READ(5,112) RESP
112 FORMAT(A3)
IF(RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 11
3333 WRITE(6,113)
113 FORMAT(5X,31H DO YOU WANT TO CHANGE DATA FOR,
+ "ANOTHER PLANT? ENTER YES OR NO.")
READ(5,112) RESP
IF(RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 9
RETURN
END

C
C
SUBROUTINE HELP
----THIS SUBPROGRAM PRINTS A LEGEND WHICH ASSIGNS
A 2 DIGIT INDEX FOR EACH PLANT PARAMETER
INTEGER YE,IP,IPNUM,IPMAX,JNUM,YEAR
INTEGER RESP, KEY1(42,18), YES
REAL PLANT(200,14)
COMMON NUF(200,20), IP,PLANT,INS,KEY1,IPNUM,IPMAX,JNUM
COMMON /CACAA/NPEXP,IUP(30,20), NBYK, NYK
DATA YES/3HYES/
DATA YE/3HYE/
DATA INP2/14/
WRITE(6,102)

```

```

102 FORMAT(" DATA CODE, CONTENT")
PRINT 200
200 FORMAT(7X,"01",1X,"NUMBER OF UNITS IN EACH YEAR")
DO 1002 K=2,JNUM
1002 IF(K.NE.4.NE.K.NE.9) WRITE (6,103) (KEY1(K+14,J),J=1,18)
103 FORMAT(7X,18A4)
WRITE (6,104)
104 FORMAT(/)
RETURN
END
C
SUBROUTINE LIST
DIMENSION JPR(200)
INTEGER YE,IP,IPNUM,IPMAX,JNUM,YEAR,UPT
INTEGER RESP, KEY1(42,18), YES
REAL PLANT(200,14)
COMMON NUF(200,20), IP,PLANT,INS,KEY1,IPNUM,IPMAX,JNUM
COMMON /CACAI/NPEXP,IUP(30,20), NBYR, NTYR
DATA YES/3HYES/
DATA ICV,ICF/"EXPN", "SCHD"/
DATA YE/3HY /
DATA INP1,INP2/11,14/
DO 500 I=1,IPNUM
JPR(I)=ICV
IF(I.GT.NPEXP) JPR(I)=ICF
500 CONTINUE
11 KTEST=KTEST+1
WRITE (6,101)
101 FORMAT (5X,32H ENTER 1,2,3,4 OR 5 ACCORDING TO,
+ PRINT OPTIONS)
IF (KTEST.GE.2) GO TO 12
WRITE (6,102)
102 FORMAT (5X,17H 1 PRINT ALL DATA)
WRITE (6,103)
103 FORMAT (5X,40H 2 PRINT ALL DATA FOR A PARTICULAR PLANT)
WRITE (6,104)
104 FORMAT (5X,37H 3 PRINT THE SAME DATA FOR ALL PLANTS)
WRITE (6,105)
105 FORMAT (5X,45H 4 PRINT SPECIFIC DATA FOR A PARTICULAR PLANT)
WRITE (6,106)
106 FORMAT (5X,41H 5 EARLIER REQUEST FOR PRINTING CANCELLED)
12 WRITE (6,107)
107 FORMAT (2H -)
READ (5,108) OPT
IF(OPT.GT.5) PRINT 200
IF(OPT.GT.5) GO TO 12
200 FORMAT(" INPUT ERROR. REENTER.")
108 FORMAT(1I1)
GO TO (20,30,40,50,60), OPT
C
20 CONTINUE
DO 4000 IP=1, IPNUM
CALL NUNIT(2)
4000 CONTINUE
IP1=1
IPPP=IPNUM/5+1
DO 2800 I1AX=1,IPPP
IP2=IP1++
IF(IPNUM.GT.IP2) GO TO 2813
IF(IP2-1PNUM.LE.4) IP2=IPNUM
2813 CONTINUE
PRINT 5400, (IP,IP=IP1,IP2)
PRINT 5500, (JPR(IP),IP=IP1, IP2)
5500 FORMAT(23X,5(A4,6X))
5400 FORMAT(3X," PLANT CODE ",5I10)
DO 1001 K=2,JNUM
IF( K.EQ.9) GO TO 1001

```

00002630
00002640
00002650
00002660
00002670
00002680
00002690
00002700
00002710
00002720
00002730
00002740
00002750
00002760
00002770
00002780
00002790
00002800
00002810
00002820
00002830
00002840
00002850
00002860
00002870
00002880
00002890
00002900
00002910
00002920
00002930
00002940
00002950
00002960
00002970
00002980
00002990
00003000
00003010
00003020
00003030
00003040
00003050
00003060
00003070
00003080
00003090
00003100
00003110
00003120
00003130
00003140
00003150
00003160
00003170
00003180
00003190
00003200
00003210
00003220
00003230
00003240
00003250
00003260
00003270
00003280
00003290
00003300

```

1001 PRINT 3011, (KEY1(K+14,J),J=1,5), (PLANT(IP,K),IP=IP1,IP2)      00003310
1001 CONTINUE
1001 PRINT 3011
1001 IF(IPNUM.EQ.IP2) GO TO 2830
1001 IP1=IP2+1
2800 CONTINUE
2830 CONTINUE
111 FORMAT (1H 18A4)
112 FORMAT (I3,1X,12,3(1X,F7.2),1X,F4.3,1X,F5.2,
+3(1X,F5.2),2(1X,F5.2),1X,F5.2,2(1X,F4.3))
112 GO TO 55
C
130 WRITE (6,113)
131 FORMAT (5X,30H ENTER PLANT CODE IN 3 DIGITS.)
131 WRITE (6,114)
134 FORMAT (4H ---)
134 READ (5,115) IP
135 FORMAT (I3)
135 CALL NUNIT(2)
135 PRINT 5400,IP
135 PRINT 5500,JPR(IP)
135 DO 1002 K=2,JNUM
135 IF(K.EQ.9) GO TO 1002
135 WRITE (6,3011) (KEY1(K+14,J),J=1,5),(PLANT(IP,K))
1002 CONTINUE
3011 FORMAT(1X,5A4,6F10.3)
3011 PRINT 3011
3011 PRINT 116
116 FORMAT (5X,36H DO YOU WANT TO PRINT DATA FOR ANOTHER,
+* PLANT? ENTER YES OR NO.*)
116 READ (5,136) RESP
116 IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 30
116 IF (RESP.EQ.YE) GO TO 30
116 GO TO 55
C
140 WRITE (6,119)
119 FORMAT (5X,36H WHICH DATA YOU WANT TO PRINT? ENTER,
+* A 2 DIGIT DATA CODE. FOR HELP, ENTER 99.*)
119 READ (5,120) J
120 FORMAT (I2)
120 IF (J.EQ.99) GO TO 44
120 IF (J.EQ.1) GO TO 42
41 DO 1004 IP=1,IPNUM
1004 WRITE (6,121) PLANT(IP,J)
121 FORMAT (10(1X,F8.3))
121 GO TO 45
42 DO 1005 IP=1,IPNUM
1005 WRITE (6,122) PLANT (IP,J)
122 FORMAT (20(1X,13))
122 GO TO 45
44 CALL HELP
44 GO TO 40
45 WRITE (6,123)
123 FORMAT (/5X,35H DO YOU WANT TO PRINT ANOTHER DATA?.
+* ENTEK YES CR NO.*)
123 READ (5,136) RESP
123 IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 40
123 IF (RESP.EQ.YE) GO TO 40
123 GO TO 55
C
50 WRITE (6,113)
50 WRITE (6,114)
50 READ (5,115) IP
51 WRITE (6,119)
51 READ (5,120) J
51 IF (J.EQ.99) GO TO 54
51 IF (J.EQ.1) GO TO 53
52 WRITE (6,126) PLANT(IP,J)

```

```

126 FORMAT (1H ,F8.3)          00003990
   GO TO 55                     00004000
53 WRITE (6,127) PLANT (IP,J)  00004010
127 FORMAT (1H ,12)             00004020
   GO TO 55                     00004030
54 CALL HELP                   00004040
   GO TO 51                     00004050
55 WRITE (6,124)               00004060
124 FORMAT (/5X,32H DO YOU WANT TO PRINT MORE DATA?, 00004070
+* ENTER YES OR NO.)          00004080
   READ (5,136) RESP            00004090
   IF (RESP.EQ.'YES.OR.RESP.EQ.'YE) GO TO 11 00004100
C
60 WRITE (6,125)               00004110
125 FORMAT(5X,'END OF PRINTING') 00004120
136 FORMAT (A3)                00004130
   RETURN                      00004140
   END                         00004150
00004160
C
SUBROUTINE NUNIT(IOPT)        00004170
  THIS SUB READS AND PRINTS THE NUMBER OF UNITS 00004180
  FOR EACH YEAR                      00004190
    IOPT=1      READ             00004200
    2           PRINT FOR TH ENTIRE PERIOD 00004210
    3           CHANGE            00004220
    4           PRINT ONLY SPECIFIED YEAR 00004230
00004240
DIMENSION JP(40)              00004250
COMMON NUF(200,20),IP,PLANT,INS,KEY1,IPNUM,IPMAX,JNUM 00004260
COMMON /CACCA/ NPEXP,IUP(30,20),NBYR,NTYR 00004270
DATA JP/ '80', '81', '82', '83', '84', '85', '86', '87', '88', '89',
+ '90', '91', '92', '93', '94', '95', '96', '97', '98', '99', '00',
+ '01', '02', '03', '04', '05', '06', '07', '08', '09', '10', '11',
+ '12', '13', '14', '15', '16', '17', '18', '19'/
DATA IDSH/* ____/ 00004280
K1=1+NBYR-80 00004290
L1=1 00004300
IF(IOPT.GT.2) GO TO 200 00004310
IF(IOPT.EQ.1.AND.IP.GT.NPEXP) PRINT 420,IP 00004320
IF(IOPT.EQ.1.AND.IP.LE.NPEXP) PRINT 520 00004330
520 FORMAT(5X,'ENTER MINIMUM NUMBER OF UNITS PERMITTED',// 00004340
+5X,'EACH YEAR (LOWER BOUND OF EXPANSION PLANTS)'// 00004350
+10X,'THEN ENTER THE MAX. NUMBER OF UNITS PERMITTED'// 00004360
+10X,'EACH YEAR (UPPER BOUND) ON THE SECOND LINE'// 00004370
420 FORMAT(5X,'ENTER NUMBER OF UNITS OF PLANT CODE ',I2, 00004380
+* WORKING IN EACH YEAR) 00004390
430 FORMAT(5X,'NUMBER OF UNITS OF PLANT CODE ',I2, 00004400
+* WORKING IN EACH YEAK') 00004410
  IF(IOPT.EQ.2.AND.IP.GT.NPEXP) PRINT 430,IP 00004420
  IF(IOPT.EQ.2.AND.IP.LE.NPEXP) PRINT 440,IP 00004430
440 FORMAT(5X,' LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)'// 00004440
+5X,' OF PLANT CODE ',I3,' (EXPANSION PLANT).') 00004450
DO 90 JJ=1,4 00004460
  IX=NTYR-L1 00004470
  L2=L1+19 00004480
  K2=K1+19 00004490
  IF(IX.GT.19) GO TO 40 00004500
  L2=L1+IX 00004510
  K2=K1+IX 00004520
40 PRINT 10, (JP(K),K=K1,K2) 00004530
  PRINT 20,(IDSH,K=K1,K2) 00004540
  IF(IOPT.EQ.1) READ 30, (NUF(IP,L),L=L1,L2) 00004550
  IF(IOPT.EQ.1.AND.IP.LE.NPEXP) READ 30,(IUP(IP,L),L=L1,L2) 00004560
  IF(IOPT.EQ.1) PRINT 10 00004570
  PRINT 31, (NUF(IP,L),L=L1,L2) 00004580
  IF(IP.LE.NPEXP) PRINT 31, (IUP(IP,L),L=L1,L2) 00004590
  PRINT 10 00004600
  IF(IX.LE.19) GO TO 50 00004610
  L1=L2+1 00004620
00004630
00004640
00004650
00004660

```

```

K1=K2+1          00004670
90 CONTINUE      00004680
10 FORMAT(1X,20(A2,1X)) 00004690
20 FORMAT(20A3)   00004700
30 FORMAT(12,1X,20(I2,1X)) 00004710
31 FORMAT(1X,20(I2,1X)) 00004720
50 CONTINUE      00004730
50 RETURN        00004740
00004750
200 PRINT 210    00004760
210 FORMAT(" WHAT IS THE YEAR ?/* ---") 00004770
220 FORMAT(2I2)   00004780
READ 220,JYR
LL=JYR          00004790
IF(LL.LT.NBYR) LL=LL+100 00004800
LL=LL+1-NBYR   00004810
IF(IP.LE.NPEXP) PRINT 250 00004820
250 FORMAT(" ENTER NIMUM AND MAXIMUM NUMBERS OF UNITS//"
+ " OF THE EXPANSION PLANTS./* ---,---") 00004830
IF(IP.GT.NPEXP) PRINT 240 00004840
240 FORMAT(" ENTEK THE NUMBER OF UNITS// ---") 00004850
IF(IOPT.EQ.3) READ 220, JNU,JUP 00004860
NUF(IP,LL)=JNU 00004870
IF(IP.LE.NPEXP) IUP(IP,LL)=JUP 00004880
IF(IP.GT.NPEXP) PRINT 300, IP,JP(LL), NUF(IP,LL) 00004890
IF(IP.LE.NPEXP) PRINT 327,IP,JP(LL),NUF(IP,LL),IUP(IP,LL) 00004900
327 FORMAT(" PLANT CODE=*,I2,* YR=*,A2,* MIN.=*,I2,* MAX=*,I2/*")
300 FORMAT(" PLANT CODE=*,I2,* YEAR=*,A2,* NUMBER OF UNITS=*" 00004910
+ *,I2//)
RETURN         00004920
END             00004930
00004940
00004950
00004960
00004970
00004980
00004990
00005000
00005010
00005020
00005030
00005040
00005050
00005062
00005067
00005068
00005070
00005075
00005080
00005085
00005100
00005110
00005120
00005130
00005140
00005150
00005160
00005170
00005180
00005190
00005200
00005210
00005220
00005230
00005240
00005250
00005260
00005270
00005280
00005290
00005300
00005310
00005320
00005330
00005340

C-----SUBROUTINE HYDPS-----C
C-----THIS SUBPROGRAM IS USED TO PROCESS AND STORE INPUT-----C
C-----DATA FOR HYDRELECTRIC AND PUMPED STORAG-----C
COMMON/TEN/HPLR(6,12),HPLI(6,12),HPLF(12,20),
+ MPLR1(6,12)
COMMON/ELEVEN/KEY2(36,18),KEY3(43,18),KEY4(11,18)
COMMON /TWELVE/ KHU,KHM,KHMX,KH1,JNUM,ITYP,IFLAG
COMMON /MULTPS/ PBM(4),EAM(4)
COMMON/UNITS/ IHUF(6,20),IHUL(6,20),IHUM(6,20),
+ NYR(20)

PROCEDURE FOR HYDROELECTRIC INPUT
IFLAG=1
CALL HYDA
PROCEDURE FOR PUMPED STORAGE INPUT
IFLAG=2
CALL PSDA

```

```

C          RETURN          00005350
C          END            00005260
C          00005370
C          00005380
C          00005390
C          00005400
C          00005410
C          00005420
C          00005430
C          00005440
C          00005450
C          00005460
C          00005470
C          00005480
C          00005490
C          00005500
C          00005510
C          00005520
C          00005530
C          00005540
C          00005550
C          00005560
C          00005570
C          00005580
C          00005590
C          00005600
C          00005610
C          00005620
C          00005630
C          00005640
C          00005650
C          00005660
C          00005670
C          00005680
C          00005690
C          00005700
C          00005710
C          00005720
C          00005730
C          00005740
C          00005750
C          00005760
C          00005770
C          00005780
C          00005790
C          00005800
C          00005810
C          00005820
C          00005830
C          00005840
C          00005850
C          00005860
C          00005870
C          00005880
C          00005890
C          00005900
C          00005910
C          00005920
C          00005930
C          00005940
C          00005950
C          00005960
C          00005970
C          00005980
C          00005990
C          00006000
C          00006010
C          00006020

C          SUBROUTINE HYDA

C THIS PROGRAM ACCEPTS INPUT DATA FOR OPERATING AND
C CUST PAKAMETERS OF HYDROELECTRIC PLANTS.
C THE ENTIRE PROCEDURE IS INTERACTIVE.

COMMON/TEN/HPLR(6,12),HPLI(6,12),HPLF(12,20),
+    HPLR1(6,12)
COMMON/ELEVEN/KEY2(30,18),KEY3(43,18),KEY4(11,18)
COMMON/TWELVE/ KHU,KHM,KHMX,KH1,JNUM,ITYP,IFLAG
COMMON/MULTPS/ PBM(4),EAM(4)
COMMON/UNITS/ IHUF(6,20),IHUL(6,20),IHUM(6,20),
+    NYR(20)
DOUBLE PRECISION ISSN(4)
INTEGER RESP,KEY2,KEY3,YES,YE,NO
INTEGER SEASON(4),BCAP,1ENA
DATA ISSN/8H WINTER ,8H SPR1NG ,
+8H SUMMER ,8H AUTUMN /
DATA YES,YE/3HYES,3HY /
DATA NO/3HNU /
IP=1
ITYP=1
KHU1=0
JNUM=8
KHMX=1
KHM=3
KHU=0
KH1=1
KX1=4
KXM=6
KMX=4
DO 100 K=1,34
READ (31,110) (KEY2(K,J),J=1,18)
100 CONTINUE
110 FORMAT (18A4)

C          BEGINNING OF INPUT PROCEDURE

C          WRITE (6,120)
120 FORMAT (//,2X,34H INPUT PROCEDURE FOR HYDROELECTRIC,
+21H PLANT SYSTEM BEGINS.)
WRITE (6,121)
123 FORMAT (//,32H DO YOU HAVE A FILE FOR EXISTING,
+22H HYDROELECTRIC PLANTS?/17H ENTER YES OR NO.)
READ (5,130) RESP
130 FORMAT (A3)
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 306
WRITE (6,153) KHM
153 FORMAT (//,44H BEGIN YOUR INPUT FOR EXISTING HYDROELECTRIC,
+8H PLANTS./,38H THE MAX. NO. OF SUCH PLANTS SHOULD BE,13)
152 CONTINUE
KHU=KHU+1
IF (KHU.LE.KHUI) KHU=KHUI+1
KHUI=KMU

```

```

170 FORMAT (I2) 00006030
  IF (KHU.EQ.KHM) KHMX=KHM 00006040
172 CONTINUE 00006050
  CALL HUNIT 00006060
175 CONTINUE 00006070
  CALL HCREAT 00006080
  CALL HKREG 00006090
C   CHANGES/CORRECTIONS OF INPUT DATA 00006100
C
217 CONTINUE 00006110
  WKITE (6,220) KHU 00006120
220 FORMAT (//,1H DO YOU WANT TO CHANGE ANY DATA FOR PLANT, 00006130
+13,1H?/17H ENTER YES OR NO.) 00006140
  READ (5,130) RESP 00006150
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 230 00006160
  IF (KHU.EQ.KH1.AND.KHMX.EQ.KH1) GO TO 250 00006170
  WRITE (6,257) 00006180
  READ (5,130) RESP 00006190
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 417 00006200
  READ (5,130) RESP 00006210
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 417 00006220
229 CONTINUE 00006230
  GO TO 250 00006240
230 WRITE (6,240) KHU 00006250
240 FORMAT (/,3H DO YOU WANT TO CHANGE ALL THE DATA FOR, 00006260
+8H PLANT,13,1H?/17H ENTER YES OR NO.) 00006270
  READ (5,130) RESP 00006280
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 172 00006290
  CALL HCHNG 00006300
  IF (KHU.EQ.KH1.AND.KHMX.EQ.KH1) GO TO 250 00006310
248 CONTINUE 00006320
  WRITE (6,257) 00006330
257 FORMAT (/,3H DO YOU WANT TO CHANGE DATA FOR, 00006340
+1H ANOTHER PLANT?/17H ENTER YES OR NO.) 00006350
  READ (5,130) RESP 00006360
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 417 00006370
250 CONTINUE 00006380
  IF (KHU.GE.KHM.OR.KHMX.GE.KHM) GO TO 440 00006390
C   ADDITIONS OF MORE PLANTS TO THE SYSTEM 00006400
C
260 WRITE (6,260) 00006410
260 FORMAT (//,3H DO YOU WANT TO ADD MORE PLANTS TO THE, 00006420
+8H SYSTEM?/17H ENTER YES OR NO.) 00006430
  READ (5,130) RESP 00006440
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 152 00006450
  KHMX=KHU 00006460
C   GO TO 440 00006470
306 CONTINUE 00006480
  READ (35,575,END=146) KHU 00006490
  READ (35,590) (IHUF(KHU,K),K=1,15) 00006500
154 FORMAT (//,313) 00006510
  READ (35,587) (HPLR(KHU,J),J=1,JNUM) 00006520
  GO TO 306 00006530
146 CONTINUE 00006540
  KHMX=KHU 00006550
  WRITE (6,158) KHMX 00006560
158 FORMAT (//,40H THE TOTAL NO. OF EXISTING HYDROELECTRIC, 00006570
+10H PLANTS IS,13) 00006580
C   DO 161 KHU=1,KHM 00006590
C   READ (37,575) KHU 00006600
C   READ (37,590) (IHUF(KHU,K),K=1,15) 00006610
C 161 CONTINUE 00006620
  WRITE (6,171) 00006630
171 FORMAT (//,31H ENTER YES IF YOU WANT TO PRINT, 00006640
+22H EXISTING SYSTEM DATA.) 00006650
  READ (5,130) RESP 00006660
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL HPRINT(2) 00006670
  00006680
  00006690
  00006700

```

```

148 CONTINUE
159 FORMAT(//,13)          00006710
265 WRITE (6,270)          00006720
270 FORMAT (//,44H DO YOU WANT TO CHANGE ANY DATA FOR EXISTING,
+/22H HYDROELECTRIC PLANTS?,17H ENTER YES OR NO.) 00006730
    READ (5,130) RESP      00006740
    IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 400 00006750
    GO TO 440              00006760
400 CONTINUE                00006770
    IF (KHM>KHM) GO TO 417 00006780
    WRITE (6,405)          00006790
405 FORMAT (//,39H ENTER DATA CHANGE OPTION ACCORDING TO,
+/15H THE FOLLOWING.) 00006800
    WRITE (6,406)          00006810
406 FORMAT (//,32H 1 ADD MORE PLANTS TO THE SYSTEM,
+/26H 2 CHANGE PLANT PARAMETERS. 00006820
    //,2H -)              00006830
    READ (5,407) IOPT      00006840
407 FORMAT (1I1)           00006850
    GO TO (411,417),IOPT  00006860
411 GO TO 152              00006870
417 WRITE (6,420)          00006880
420 FORMAT (//,34H ENTER PLANT CODE NUMBER(2 DIGITS),
+/35H FOR WHICH YOU WANT TO CHANGE DATA.,
//,3H --)              00006890
    READ (5,170) KHU        00006900
    GO TO 230              00006910
440 CONTINUE                00006920
    IF (KHU.GT.3) GO TO 750 00006930
    IF (ITYP.EQ.2) GO TO 750 00006940
    WRITE (6,450)          00006950
450 FORMAT (//,38H IS YOUR INPUT FILE FOR EXISTING HYDRO,
+/17H PLANTS COMPLETE?/17H ENTER YES OR NO.) 00006960
    READ (5,130) RESP      00006970
    IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 490 00006980
    GO TO 265              00006990
490 CONTINUE                00007000
    CALL HCUMP              00007010
575 FORMAT(1,3I3)          00007020
    REWIND 35              00007030
    DO 560 KHU=KH1,KHMX    00007040
    WRITE (35,575) KHU      00007050
    WRITE (35,590) (IHUF(KHU,K),K=1,15) 00007060
    WRITE (35,587) (HPLR(KHU,J),J=1,JNUM) 00007070
580 FORMAT (1H ,5F10.5)    00007080
560 CONTINUE                00007090
570 FORMAT (1H ,3I3)        00007100
585 FORMAT (1H ,5E13.5)    00007110
550 CONTINUE                00007120
590 FORMAT (20I3)          00007130
    REWIND 38              00007140
    DO 540 NY=1,15          00007150
    WRITE (38,575) NY      00007160
    WRITE (38,587) (HPLF(J,NY),J=1,JNUM) 00007170
587 FORMAT (5E12.5)        00007180
540 CONTINUE                00007190
600 CONTINUE                00007200
500 CONTINUE                00007210
602 ITYP=2                 00007220
    JNUM=11                 00007230
    KHU=KHM                 00007240
    KHM=KXM                 00007250
    KHI=KX1                 00007260
    KHMX=KX1                00007270
    KX=KXM-KX1+1            00007280
    WRITE (6,605)          00007290
605 FORMAT (//,33H DO YOU HAVE A FILE FOR EXPANSION, 00007300
                                            00007310
                                            00007320
                                            00007330
                                            00007340
                                            00007350
                                            00007360
                                            00007370
                                            00007380

```

```

+22H HYDROELECTRIC PLANTS?/17H ENTER YES OR NO.)          00007390
  READ (5,130) RESP                                         00007400
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 800                 00007410
  WRITE (6,610)                                              00007420
610 FORMAT (//,40H DO YOU WANT TO CONSIDER HYDRO PLANTS AS, 00007430
+//22H EXPANSION CANDIDATES?,17H ENTER YES OR NO.)        00007440
  READ (5,130) RESP                                         00007450
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 710                 00007460
  GO TO 900                                              00007470
710 WRITE (6,715) KX,KX1,KXM                                00007480
715 FORMAT (//,30H NO. OF PLANT TYPES CONSIDERED,           00007490
+17H FOR EXPANSION IS,13/18H THEY ARE ASSIGNED,          00007500
+19H PLANT CODE NUMBERS,13,8H THROUGH,13)                  00007510
  GO TO 152                                              00007520
750 CONTINUE                                              00007530
  WRITE (6,720)                                              00007540
720 FORMAT (//,33H IS YOUR INPUT FILE FOR EXPANSION,       00007550
+23H HYDRO PLANTS COMPLETE?/17H ENTER YES OR NO.)        00007560
  READ (5,130) RESP                                         00007570
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 850                 00007580
800 CONTINUE                                              00007590
  READ (39,575,END=830) KHU                                00007600
  READ (39,590) (IHUL(KHU,K),K=1,15)                      00007610
  READ (39,590) (IHUH(KHU,K),K=1,15)                      00007620
  READ (39,587) (HPLR(KHU,J),J=1,JNUM)                   00007630
  GO TO 830                                              00007640
830 CONTINUE                                              00007650
  KMMX=KHU                                              00007660
  KHU1=KHU                                              00007670
  KX=KMMX-KX1+1                                         00007680
  WRITE (6,715) KX,KX1,KXM                                00007690
  WRITE (6,832)                                              00007700
832 FORMAT (//,31H ENTER YES IF YOU WANT TO PRINT,         00007710
+23H EXPANSION SYSTEM DATA.)                            00007720
  READ (5,130) RESP                                         00007730
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL HPRINT(3)          00007740
  WRITE (6,835)                                              00007750
835 FORMAT (//,35H DO YOU WANT TO CHANGE ANY DATA FOR,    00007760
+24H EXPANSION HYDRO PLANTS?/17H ENTER YES OR NO.)       00007770
  READ (5,130) RESP                                         00007780
  IF (KESP.EQ.YES.OR.RESP.EQ.YE) GO TO 400                 00007790
  GO TO 750                                              00007800
850 CONTINUE                                              00007810
  KMMX=KHU1                                              00007820
  REWIND 39                                              00007830
  DO 870 KHU=KHU1,KMMX                                    00007840
  WRITE (39,575) KHU                                         00007850
  WRITE (39,590) (IHUL(KHU,K),K=1,15)                      00007860
  WRITE (39,590) (IHUH(KHU,K),K=1,15)                      00007870
  WRITE (39,587) (HPLR(KHU,J),J=1,JNUM)                   00007880
870 CONTINUE                                              00007890
900 CONTINUE                                              00007900
  WRITE (6,902)                                              00007910
902 FORMAT (//,31H DO YOU HAVE A FILE ALREADY FOR,         00007920
+21H SEASONAL MULTIPLIERS/20H OF BASE CAPACITY AND,      00007930
+21H TOTAL INFLOW ENERGY?/17H ENTER YES OR NO./2H ?)     00007940
  READ (5,130) RESP                                         00007950
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 941                 00007960
908 WRITE (6,910)                                              00007970
910 FORMAT (//,30H ENTER SEASONAL MULTIPLIERS OF,         00007980
+//39H BASE CAPACITY AND TOTAL INFLOW ENERGY./          00007990
+34H USE WINTER AS THE BASE SEASON AND/                 00008000
+40H SET MULTIPLIERS TO 1.0 FOR THIS SEASON.)            00008010
915 FORMAT (F10.5)                                              00008020
925 FORMAT (12H _____._____)                                 00008030
  WRITE (6,920)                                              00008040
920 FORMAT (//,29H BASE CAPACITY MULTIPLIER FOR)          00008050
  DO 924 I=1,4                                              00008060

```

```

930 WRITE (6,930) ISSN(I) 00008070
FORMAT (1H,A8,2H ?)
WRITE (6,925) 00008080
READ (5,915) PBM(I) 00008090
924 CONTINUE 00008100
WRITE (6,940) 00008110
940 FORMAT (/,29H INFLOW ENERGY MULTIPLIER FOR) 00008120
DU 934 I=1,4 00008130
WRITE (6,930) ISSN(I) 00008140
WRITE (6,925) 00008150
READ (5,915) EAM(I) 00008160
934 CONTINUE 00008170
GU TO 954 00008180
941 READ (40,587) (PBM(JS),JS=1,4) 00008190
KREAD (40,587) (EAM(JS),JS=1,4) 00008200
954 CONTINUE 00008210
WRITE (6,962) 00008220
962 FORMAT (//,25H ENTER YES IF YOU WANT TO, 00008230
+32H PRINT THE SEASONAL MULTIPLIERS./2H ?) 00008240
READ (5,130) RESP 00008250
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 970 00008260
GU TO 981 00008270
970 CONTINUE 00008280
WRITE (6,1980) 00008290
WRITE (6,1983) 00008300
WRITE (6,1982) (PBM(JS),JS=1,4) 00008310
WRITE (6,1985) 00008320
WRITE (6,1983) 00008330
WRITE (6,1982) (EAM(JS),JS=1,4) 00008340
981 WRITE (6,985) 00008350
985 FORMAT (//,25H ENTER YES IF YOU WANT TO, 00008360
+29H RENTER SEASONAL MULTIPLIERS./2H ?) 00008370
READ (5,130) RESP 00008380
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 908 00008390
REWIND 40 00008400
WRITE (40,587) (PBM(JS),JS=1,4) 00008410
WRITE (40,587) (EAM(JS),JS=1,4) 00008420
950 CONTINUE 00008430
00008440
C   WRITE (6,1965) 00008450
1965 FORMAT (//,37H PRINTING OF THE ENTIRE HYDROELECTRIC, 00008460
+21H SYSTEM DATA FOLLOWS./ 17H IF YOU WANT THIS, 00008470
+20H PRINTING,ENTER YES.) 00008480
READ (5,130) RESP 00008490
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 1975 00008500
GO TO 2000 00008510
1975 WRITE (6,1980) 00008520
1980 FORMAT (//,27H BASE CAPACITY MULTIPLIERS:) 00008530
WRITE (6,1983) 00008540
1983 FORMAT (/,4X,7H WINTER,2X,7H SPRING,3X, 00008550
+7H SUMMER,3X,7H AUTUMN) 00008560
WRITE (6,1982) (PBM(JS),JS=1,4) 00008570
WRITE (6,1985) 00008580
1985 FORMAT (//,27H INFLOW ENERGY MULTIPLIERS:) 00008590
WRITE (6,1983) 00008600
WRITE (6,1982) (EAM(JS),JS=1,4) 00008610
1982 FORMAT (4F10.5) 00008620
CALL HPRINT(1) 00008630
2000 CONTINUE 00008640
C   END OF INPUT PROCEDURE 00008650
00008660
C   WRITE (6,2654) 00008670
2654 FORMAT (//,*1H END OF INPUT PROCEDURE FOR HYDROELECTRIC, 00008680
+8H PLANTS./44H INPUT DATA STORED ON UNITS 35,38,39 AND 40.) 00008690
RETURN 00008700
END 00008710
00008720
C   00008730
00008740

```

```

SUBROUTINE HREG
-----  

THIS SUBROUTINE DETERMINES THE REGULATING AND  

AND PEAKING CHARACTERISTICS OF HYDRO PLANTS.  

COMMON/TEN/HPLR(6,12),HPL1(6,12),HPLF(12,20),  

+ HPLR1(6,12)  

COMMON/ELEVEN/KEY2(36,18),KEY3(43,18),KEY4(11,18)  

COMMON/TWELVE/ KHU,KHM,KHMX,KH1,JNUM,ITYP,IFLAG  

COMMON/MULTPS/ PBM(4),EAM(4)  

COMMON/UNITS/ IHUF(6,20),IHUL(6,20),IHUM(6,20),  

+ NYR(20)  

INTEGER RESP,KEY2,KEY3,YES,YE  

DATA YES,YE/3HYES,3HY /  

HR=2.16  

10 PB=HPLR(KHU,1)  

PC=HPLR(KHU,2)  

V=HPLR(KHU,3)  

EA=HPLR(KHU,4)  

EB=PB*HR  

PCH=PC*HR  

P1=EA/HR  

IF (P1.LE.PB) GO TO 220  

PA=PC-PB  

IF (PC.LE.PB) GO TO 220  

PMAX=PC-PA  

TV=1000.*V/PMAX  

VO=1000.*V/24.  

V1=(PMAX-PA)/12.  

V2=(PMAX-PA)*PA/PMAX  

V3=(3.*PMAX-7.*PA/5.)*PA/PMAX  

V4=3.*PA  

100 FORMAT (1H ,5F10.5)  

C FIND REGULATING CAPABILITY  

C  

IF (EA.GE.PCH) GO TO 101  

IF (EA.LE.EB) GO TO 104  

IF (TV.LT.2.0) GO TO 107  

IF (VO.GT.V1.AND.VU.LE.V2) GO TO 110  

IF (VO.GT.V2.AND.VU.LE.V3) GO TO 120  

IF (VO.GT.V4) GO TO 125  

101 CONTINUE  

WRITE (6,75)  

75 FORMAT (//,29H INFLOW ENERGY EXCEEDS PLANT ,  

+22H GENERATING CAPABILITY.)  

WRITE (6,99)  

99 FORMAT (35H THE PLANT HAS ONLY BASE BLOCK AND ,  

+18H NO PEAKING BLOCK.)  

HPLR(KHU,1)=HPLR(KHU,2)  

HPLR(KHU,2)=0  

GO TO 300  

104 CONTINUE  

WRITE (6,80)  

80 FORMAT (//,33H BASE ENERGY REQUIREMENT EXCEEDS ,  

+14H THE INFLOW ENERGY.)  

WRITE (6,99)  

HPLR(KHU,1)=HPLR(KHU,2)  

HPLR(KHU,2)=0  

GO TO 300  

107 CONTINUE  

WRITE (6,85)  

85 FORMAT (//,35H THIS PLANT HAS A RUN-OF-THE RIVER ,  

+10H RESERVOIR,/30M AND NO REGULATING CAPABILITY.)  

00008750  

00008760  

00008770  

00008780  

00008800  

00008810  

00008820  

00008830  

00008840  

00008850  

00008860  

00008870  

00008880  

00008890  

00008900  

00008910  

00008920  

00008930  

00008940  

00008950  

00008960  

00008970  

00008980  

00008990  

00009000  

00009010  

00009020  

00009030  

00009040  

00009050  

00009060  

00009070  

00009080  

00009090  

00009100  

00009110  

00009120  

00009130  

00009140  

00009150  

00009160  

00009170  

00009180  

00009190  

00009200  

00009210  

00009220  

00009230  

00009240  

00009250  

00009260  

00009270  

00009280  

00009290  

00009300  

00009310  

00009320  

00009330  

00009340  

00009350  

00009360  

00009370  

00009380  

00009390  

00009400  

00009410  

00009420

```

```

      WRITE (6,99)
      HPLR(KHU,1)=HPLR(KHU,2)
      HPLR(KHU,2)=0
      GO TO 300
110 CONTINUE
115 FORMAT (//,33H THIS IS A DAILY REGULATING PLANT)
      GO TO 300
120 CONTINUE
125 CONTINUE
126 FORMAT (//,34H THIS IS A WEEKLY REGULATING PLANT)
      GO TO 300
127 FORMAT (//,36H THIS IS A SEASONAL REGULATING PLANT)
      GO TO 300
220 CONTINUE
      P2=PC
      IF (P1.LE.PC) P2=P1
      WRITE (6,230) P2
230 FORMAT (//,31H ERROR! BASE CAPACITY SHOULD BE,
+10H LESS THAN,F10.5,4H MW./22H REENTER BASE CAPACITY,
+6H IN MW)
      WRITE (6,240)
240 FORMAT (//,11H -----)
      READ (5,250) HPLR(KHU,1)
250 FORMAT (F10.5)
      GO TO 10
300 CONTINUE
      RETURN
      END

C
C-----SUBROUTINE HCREAT
C-----THIS SUBROUTINE CREATES A FILE OF HYDRO PARAMETERS
C-----COMMON/TEN/HPLR(6,12),HPLI(6,12),HPLF(12,20),
C-----+ HPLR1(6,12)
C-----COMMON/ELEVEN/KEY2(36,18),KEY3(43,18),KEY4(11,18)
C-----COMMON /TWELVE/ KHU,KHM,KHMX,KH1,JNUM,ITYP,IFLAG
C-----COMMON /MULTPS/ PBM(4),EAM(4)
C-----COMMON/UNITS/ IHUF(6,20),IHUL(6,20),IHUH(6,20),
C-----+ NYR(20)
C----- INTEGER RESP,KEY2,KEY3,YES,YE
C----- DATA YES,YE/3HYES,3HV /
C----- GO TO (10,20),IFLAG
C----- 10 DO 180 K=1,JNUM
C-----     WRITE (6,112)
C----- 110 FORMAT (18A4)
C----- 112 FORMAT (/,1H)
C-----     WRITE (6,110) (KEY2(K+23,J),J=1,18)
C-----     WRITE (6,190)
C----- 190 FORMAT (/,1H -----)
C-----     READ (5,210) HPLR(KHU,K)
C----- 210 FORMAT (F10.5)
C----- 180 CONTINUE
C-----     GO TO 30
C----- 20 DO 280 K=1,JNUM
C-----     WRITE (6,112)
C-----     WRITE (6,110) (KEY3(K+23,J),J=1,18)

00009430
00009440
00009450
00009460
00009470
00009480
00009490
00009500
00009510
00009520
00009530
00009540
00009550
00009560
00009570
00009580
00009590
00009600
00009610
00009620
00009630
00009640
00009650
00009660
00009670
00009680
00009690
00009700
00009710
00009720
00009730
00009740
00009750
00009760
00009770
00009780
00009790
00009800
00009810
00009820
00009830
00009840
00009850
00009860
00009870
00009880
00009890
00009900
00009910
00009920
00009930
00009940
00009950
00009960
00009970
00009980
00009990
00010000
00010010
00010020
00010030
00010040
00010050
00010060
00010070
00010080
00010090
00010100

```

```

      WRITE (6,190)
      READ (5,210) HPLR(KHU,K)
180  CONTINUE
30   RETURN
END
SUBROUTINE HUNIT
COMMON/TEN/HPLK(6,12),HPLI(6,12),HPLF(12,20),
+    HPLR1(6,12)
COMMON/ELEVEN/KEY2(36,18),KEY3(43,18),KEY4(11,18)
COMMON/TWELVE/ KHU,KHM,KHMX,KH1,JNUM,ITYP,IFLAG
COMMON/UNITS/ IHUF(6,20),IHUL(6,20),IHUM(6,20),
+    NYR(20)
INTEGER RESP,KEY2,KEY3,YES,YE
DATA YES,YE/3HYES,3HY/
IC=1
NBYK=80
DO 110 K=1,15
110 NYR(K)=NBYR+K
GO TO 178,188,ITYP
178 CONTINUE
IF(IFLAG.GT.1) GO TO 10
WRITE (6,182) KHU
182 FORMAT (//,38H ENTER NUMBER OF SCHEDULED HYDRO UNITS,
+16H WITH PLANT CODE,I3,/14H WORKING EACH YEAR.)
GO TO 20
10  WRITE (6,183) KHU
183 FORMAT (//,40H ENTER NUMBER OF SCHEDULED STORAGE UNITS,
+16H WITH PLANT CODE,I3,/14H WORKING EACH YEAR.)
20  WRITE (6,184) (NYR(K),K=1,15)
WRITE (6,186)
READ (5,190) (IHUF(KHU,K),K=1,15)
GO TO 215
188 CONTINUE
WRITE (6,192) KHU
192 FORMAT (//,45H ENTER THE MIN. NO (LOWER BOUND) OF EXPANSION,
+6H UNITS/16H WITH PLANT CODE,I3,21H PERMITTED EACH YEAR.)
WRITE (6,184) (NYR(K),K=1,15)
WRITE (6,186)
READ (5,190) (IHUL(KHU,K),K=1,15)
WRITE (6,194) KHU
194 FORMAT (//,45H ENTER THE MAX. NO (UPPER BOUND) OF EXPANSION,
+6H UNITS/16H WITH PLANT CODE,I3,21H PERMITTED EACH YEAR.)
WRITE (6,184) (NYR(K),K=1,15)
184 FORMAT (/,1H ,2013)
WRITE (6,186)
186 FORMAT (/,1H ,15(3H --))
READ (5,190) (IHUM(KHU,K),K=1,15)
190 FORMAT (2013)
187 FORMAT (//,313)
215 CONTINUE
RETURN
END
SUBROUTINE HCHNG
COMMON/TEN/HPLK(6,12),HPLI(6,12),HPLF(12,20),
+    HPLR1(6,12)
COMMON/ELEVEN/KEY2(36,18),KEY3(43,18),KEY4(11,18)
COMMON/TWELVE/ KHU,KHM,KHMX,KH1,JNUM,ITYP,IFLAG
INTEGER RESP,KEY2,KEY3,YES,YE
DATA YES,YE/3HYES,3HY/
KCH=0
JNUM1=JNUM+1
105 CONTINUE
WRITE (6,110) KHU
110 FORMAT (//,34H WHICH DATA YOU WANT TO CHANGE FOR,
+6H PLANT,I3,2H ?/32H ENTER 2 DIGIT DATA CODE NUMBER.,
+//27H IF YOU NEED HELP,ENTER 99.,
+/3H --)
READ (5,120) K

```

```

120 FORMAT (I2)
  IF (K.GT.1.AND.K.LT.6) KCH=1
  IF (K.EQ.99) GO TO 127
  IF (K.EQ.1) GO TO 152
  GO TO 129
127 CONTINUE
  IF(IFLAG.GT.1) GO TO 10
  DO 141 K=1,JNUM1
    WRITE(6,135) (KEY2(K+11,J),J=1,18)
141 CONTINUE
  GO TO 105
10 DO 142 K=1,JNUM1
142 WRITE(6,135) (KEY3(K+11,J),J=1,18)
  GO TO 105
135 FORMAT (18A4)
129 CONTINUE
  IF(IFLAG.GT.1) GO TO 30
  WRITE(6,135) (KEY2(K+22,J),J=1,18)
  GO TO 40
30 WRITE(6,135) (KEY3(K+22,J),J=1,18)
40 WRITE(6,138) HPLR(KHU,K-1)
138 FORMAT (/,17H (PREVIOUS VALUE=,F10.5,1H),/)
  WRITE(6,140)
140 FORMAT (11H -----)
  READ(5,150) HPLR(KHU,K-1)
150 FORMAT (F10.5)
  GO TO 154
152 CALL HUNIT
154 CONTINUE
  WRITE(6,160) KHU
160 FORMAT (//36H DO YOU WANT TO CHANGE ANY MORE DATA,
  +10H FOR PLANT,13,2H ?,
  +/17H ENTER YES OR NO.)
  READ(5,130) RESP
130 FORMAT (A3)
  IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 105
  IF (KCH.EQ.1.AND.IFLAG.LE.1) CALL HREG
  RETURN
END

C
C
SUBROUTINE HCOMP
COMMON/TEN/HPLR(6,12),HPLI(6,12),HPLF(12,20),
+ HPLR1(6,12)
COMMON/ELEVEN/KEY2(36,18),KEY3(43,18),KEY4(11,18)
COMMON/TWELVE/ KHU,KHM,KHMX,KM1,JNUM,ITYP,IFLAG
COMMON/MULTPS/ PBM(4),EAM(4)
COMMON/UNITS/ IHUF(6,20),IHUL(6,20),IHUH(6,20),
+ NYR(20)
DIMENSION WT1(20),WT2(20),W1(5),W2(5)
INTEGER RESP,KEY2,KEY3,YES,YE
DATA YES,YE/3HYES,3HY /
C
DO 110 NY=1,15
DO 120 J=1,JNUM
HPLF(J,NY)=0
120 CONTINUE
WT1(NY)=0
WT2(NY)=0
110 CONTINUE
DO 385 KHU=KH1,KHMX
W1(KHU)=HPLR(KHU,2)
W2(KHU)=HPLR(KHU,4)
385 CONTINUE
DO 402 NY=1,15
DO 406 KHU=KH1,KHMX

```

00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880
00010890
00010900
00010910
00010920
00010930
00010940
00010950
00010960
00010970
00010980
00010990
00011000
00011010
00011020
00011030
00011040
00011050
00011060
00011070
00011080
00011090
00011100
00011110
00011120
00011130
00011140
00011150
00011160
00011170
00011180
00011190
00011200
00011210
00011220
00011230
00011240
00011250
00011260
00011270
00011280
00011290
00011300
00011310
00011320
00011330
00011340
00011350
00011360
00011370
00011380
00011390
00011400
00011410
00011420
00011430
00011440
00011450
00011460
00011470

```

WT1(NY)=WT1(NY)+W1(KHU)*IHUF(KHU,NY)
WT2(NY)=WT2(NY)+W2(KHU)*IHUF(KHU,NY)
401 CONTINUE
402 CONTINUE
IF(IFLAG.GT.1) GO TO 10
JB1=1
JE1=4
JB2=5
JE2=7
GO TO 20
10 JB1=1
JE1=2
JB2=3
JE2=7
20 DO 430 NY=1,15
DO 440 KHU=KH1,KHMX
WM1=W1(KHU)*IHUF(KHU,NY)/WT1(NY)
WM2=W2(KHU)*IHUF(KHU,NY)/WT2(NY)
DO 445 J=JB1,JE1
HPLF(J,NY)=HPLF(J,NY)+HPLR(KHU,J)*IHUF(KHU,NY)
445 CONTINUE
DO 450 J=JB2,JE2
HPLF(J,NY)=HPLF(J,NY)+HPLR(KHU,J)*WM1
450 CONTINUE
HPLF(JNUM,NY)=HPLF(JNUM,NY)+HPLR(KHU,JNUM)*WM2
440 CONTINUE
430 CONTINUE
RETURN
END
C
SUBROUTINE HPRINT(IPRT)
COMMON/TEN/HPLR(6,12),HPLI(6,12),HPLF(12,20),
+ HPLR1(6,12)
COMMON/ELEVEN/KEY2(36,18),KEY3(43,18),KEY4(11,18)
COMMON /TWELVE/ KHU,KHM,KHMX,KH1,JNUM,ITYP,IFLAG
COMMON /MULTPS/ PBM(4),EAM(4)
COMMON/UNITS/ IHUF(6,20),IHUL(6,20),IHUH(6,20),
+ NYS(20)
INTEGER RESP,KEY2,KEY3,YES,YE
DIMENSION KH(6)
DATA YES,YE/3HYES,3HY /
DO 95 J=1,6
95 KH(J)=J
C
GO TO (200,200,400),IPRT
200 CONTINUE
WRITE(6,230)
230 FORMAT(//,21H NO. OF UNITS OF EACH,
+ 30H PLANT CODE WORKING EACH YEAR:)
WRITE(6,235)
235 FORMAT(18H (EXISTING PLANTS))
WRITE(6,250) (KH(I),I=1,3)
250 FORMAT(/,13H HYD PLT CODE,3I5)
DO 260 NY=1,15
KY=NY+1980
WRITE(6,270) KY,(IHUF(JH,NY),JH=1,3)
260 CONTINUE
270 FORMAT(/,8X,4I5)
WRITE(6,280)
280 FORMAT(//,23H INDIVIDUAL PLANT DATA:)
WRITE(6,290) (KH(I),I=1,3)
290 FORMAT(//,13H HYD PLT CODE,9X,3(1X,I10),/)
DO 300 J=1,8
WRITE(6,310) (KEY2(J,K),K=1,6),(HPLR(JH,J),JH=1,3)
300 CONTINUE
00011490
00011500
00011510
00011520
00011530
00011540
00011550
00011560
00011570
00011580
00011590
00011600
00011610
00011620
00011630
00011640
00011650
00011660
00011670
00011680
00011690
00011700
00011710
00011720
00011730
00011740
00011750
00011760
00011770
00011780
00011790
00011800
00011810
00011820
00011830
00011840
00011850
00011860
00011870
00011880
00011890
00011900
00011910
00011920
00011930
00011940
00011950
00011960
00011970
00011980
00011990
00012000
00012010
00012020
00012030
00012040
00012050
00012060
00012070
00012080
00012090
00012100
00012110
00012120
00012130
00012140
00012150
00012160

```

```

IF (IPRT.EQ.2) GO TO 500          00012170
310 FORMAT (3X,6A4,3(1X,F10.5))   00012180
400 CONTINUE                       00012190
        WRITE (6,430)                 00012200
430 FORMAT (//,30H LOWER AND UPPER BOUNDS ON NO.,
+12H UF UNITS OF/27H EACH PLANT CODE EACH YEAR:) 00012210
        WRITE (6,435)                 00012220
435 FORMAT (19H (EXPANSION PLANTS)) 00012230
        WRITE (6,450) (KH(I),I=4,6)    00012240
450 FORMAT (/,13H HYD PLT CODE,3I10) 00012250
        WRITE (6,455)                 00012260
455 FORMAT (/,15X,3(4X,1HL,4X,1HU)) 00012270
DO 460 NY=1,15                     00012280
        KY=NY+1980                  00012290
        WRITE (6,470) KY,(IHUL(JH,NY),IHUH(JH,NY),JH=4,6) 00012300
460 CONTINUE                         00012310
470 FORMAT (/,5X,15,5X,6I5)          00012320
        WRITE (6,280)                 00012330
        WRITE (6,435)                 00012340
        WRITE (6,290) (KH(I),I=4,6)  00012350
DO 495 J=1,11                      00012360
        WRITE (6,310) (KEY2(J,K),K=1,6),(HPLR(JH,J),JH=4,6) 00012370
495 CONTINUE                         00012380
500 CONTINUE                         00012390
        WRITE (6,520)                 00012400
520 FORMAT (//,34H * K$ MEANS DOLLARS IN THOUSANDS) 00012410
        RETURN                         00012420
        END                           00012430
00012440
00012450
00012460
00012470
00012480
00012490
00012500
00012510
00012520
00012530
00012540
00012550
00012560
00012570
00012580
00012590
00012600
00012610
00012620
00012630
00012640
00012650
00012660
00012670
00012680
00012690
00012700
00012710
00012720
00012730
00012740
00012750
00012760
00012770
100 CONTINUE                         00012780
110 FORMAT (18A4)                   00012790
BEGINNING OF INPUT PROCEDURE       00012800
WRITE (6,120)                      00012810
00012820
00012830
00012840

```

```

120 FORMAT (//,5X,35H INPUT PROCEDURE FOR PUMPED STORAGE,  

+21H PLANT SYSTEM BEGINS.) 00012850  

121 WRITE (6,123) 00012860  

122 FORMAT (//,32H DO YOU HAVE A FILE FOR EXISTING,  

+23H PUMPED STORAGE PLANTS?/17H ENTER YES OR NO.) 00012880  

123 READ (5,130) RESP 00012890  

130 FORMAT (A3) 00012900  

131 IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 306 00012910  

132 WRITE (6,153) KHM 00012920  

133 FORMAT (//,38H BEGIN YOUR INPUT FOR EXISTING STORAGE,  

+8H PLANTS./,38H THE MAX. NO. OF SUCH PLANTS SHOULD BE,I3) 00012930  

134 CONTINUE 00012940  

135 WRITE (6,160) 00012950  

136 FORMAT (//,36H ENTER PLANT CODE NUMBER (2 DIGITS)., 00012960  

+//3H --) 00012970  

137 READ(5,170) KHU 00012980  

138 FORMAT (I2) 00012990  

139 IF (KHU.EQ.KHM) KHMX=KHM 00013000  

140 CONTINUE 00013010  

141 CALL HUNIT 00013020  

142 CONTINUE 00013030  

143 CALL MCREAT 00013040  

C   CHANGES/CORRECTIONS OF INPUT DATA 00013050  

C   144 CONTINUE 00013060  

145 WRITE (6,220) KHU 00013070  

146 FORMAT (//,41H DO YOU WANT TO CHANGE ANY DATA FOR PLANT,  

+I3,1H?/17H ENTER YES OR NO.) 00013080  

147 READ (5,130) RESP 00013090  

148 IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 230 00013100  

149 IF (ITYP.EQ.2) GO TO 750 00013110  

150 IF (KHU.LT.2.AND.KHMX.LT.2) GO TO 250 00013120  

151 WRITE (6,257) 00013130  

152 READ (5,130) RESP 00013140  

153 IF (RESP.EQ.YES.OK.RESP.EQ.YE) GO TO 417 00013150  

154 CONTINUE 00013160  

155 GO TO 250 00013170  

156 WRITE (6,240) KHU 00013180  

157 FORMAT (//,39H DO YOU WANT TO CHANGE ALL THE DATA FOR,  

+6H PLANT,I3,1H?/17H ENTER YES OR NO.) 00013190  

158 READ (5,130) RESP 00013200  

159 IF (RESP.EQ.YES.OK.RESP.EQ.YE) GO TO 172 00013210  

160 CALL HCHNG 00013220  

161 IF (ITYP.EQ.2) GO TO 750 00013230  

162 IF (KHU.LT.2.AND.KHMX.LT.2) GO TO 250 00013240  

163 CONTINUE 00013250  

164 WRITE (6,257) 00013260  

165 FORMAT (//,31H DO YOU WANT TO CHANGE DATA FOR,  

+15H ANOTHER PLANT?/17H ENTER YES OR NO.) 00013270  

166 READ (5,130) RESP 00013280  

167 IF (RESP.EQ.YES.OK.RESP.EQ.YE) GO TO 417 00013290  

168 CONTINUE 00013300  

169 IF (KHU.GE.KHM.OR.KHMX.GE.KHM) GO TO 440 00013310  

C   ADDITIONS OF MORE PLANTS TO THE SYSTEM 00013320  

C   170 WRITE (6,260) 00013330  

171 FORMAT (//,38H DO YOU WANT TO ADD MORE PLANTS TO THE,  

+8H SYSTEM?/17H ENTER YES OR NO.) 00013340  

172 READ (5,130) RESP 00013350  

173 IF (RESP.EQ.YES.OK.RESP.EQ.YE) GO TO 152 00013360  

174 KHMX=KHU 00013370  

C   GO TO 440 00013380  

175 CONTINUE 00013390  

176 READ (45,575,END=146) KHU 00013400  

177 FORMAT (313) 00013410  

178 00013420  

179 00013430  

180 00013440  

181 00013450  

182 00013460  

183 00013470  

184 00013480  

185 00013490  

186 00013500  

187 00013510  

188 00013520

```

```

READ (45,590) (IHUF(KHU,K),K=1,15)
READ (45,587) (HPLR(KHU,J),J=1,JNUM)
GO TO 306
146 CONTINUE
KHMX=KHU
WRITE (6,158) KHMX
158 FORMAT (//,34H THE TOTAL NO. OF EXISTING STORAGE,
+10H PLANTS IS,13)
C DO 161 KHU=1,KHMX
C READ (47,575) KHU
C READ (47,590) (IHUF(KHU,K),K=1,15)
C 161 CONTINUE
C WRITE (6,171)
171 FORMAT (//,31H ENTER YES IF YOU WANT TO PRINT,
+22H EXISTING SYSTEM DATA.)
READ (5,130) RESP
IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL SPRINT(2)
IF (IP.EQ.1) GO TO 265
DO 149 NY=1,15
READ (48,575) NY
READ (48,587) (HPLF(J,NY),J=1,JNUM)
149 CONTINUE
148 CONTINUE
159 FORMAT(//,13)
C 265 WRITE (6,270)
270 FORMAT (//,35H DO YOU WANT TO CHANGE ANY DATA FOR,
+25H EXISTING STORAGE PLANTS?/17H ENTER YES OR NO.)
READ (5,130) RESP
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 400
GO TO 440
400 CONTINUE
IF (KHMX.GE.KHM) GO TO 417
WRITE (6,405)
405 FORMAT (//,39H ENTER DATA CHANGE OPTION ACCORDING TO,
+15H THE FOLLOWING.)
WRITE (6,406)
406 FORMAT (//,32H 1 ADD MORE PLANTS TO THE SYSTEM,
+/,26H 2 CHANGE PLANT PARAMETERS,
+//,2H --)
READ (5,407) IOPT
407 FORMAT (1I1)
GO TO (411,417),IOPT
411 GO TO 152
417 WRITE (6,420)
420 FORMAT (//,34H ENTER PLANT CODE NUMBER(2 DIGITS),
+//35H FOR WHICH YOU WANT TO CHANGE DATA.,
+//3H --)
READ (5,170) KHU
GO TO 230
440 CONTINUE
WRITE (6,450)
450 FORMAT (//,40H IS YOUR INPUT FILE FOR EXISTING STORAGE,
+17H PLANTS COMPLETE?/17H ENTER YES OR NO.)
READ (5,130) RESP
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 490
GO TO 265
490 CONTINUE
CALL HCOMP
575 FORMAT(1,3I3)
REWIND 45
DO 560 KHU=1,KHMX
WRITE (45,575) KHU
WRITE (45,590) (IHUF(KHU,K),K=1,15)
WRITE (45,587) (HPLR(KHU,J),J=1,JNUM)
560 FORMAT (1H ,5F10.5)
560 CONTINUE
570 FORMAT (1,1H ,3I3)

```

```

585 FORMAT (1H ,5E13.5) 00014210
550 CONTINUE 00014220
      REWIND 47 00014230
C      DO 554 KHU=1,KHMX 00014240
C      WRITE (47,575) KHU 00014250
C      WRITE (47,590) (IHUF(KHU,K),K=1,15) 00014260
554 CONTINUE 00014270
C      FORMAT (20I3) 00014280
590 FORMAT (20I3) 00014290
      DO 540 NY=1,15 00014300
      WRITE (48,575) NY 00014310
      WRITE (48,587) (HPLF(J,NY),J=1,JNUM) 00014320
587 FORMAT (5E12.5) 00014330
540 CONTINUE 00014340
600 CONTINUE 00014350
500 CONTINUE 00014360
602 ITYP=2 00014370
      JNUM=11 00014380
      KHU=KHMX+KH1 00014390
      KH1=KHU 00014400
      WRITE (6,605) 00014410
605 FORMAT (//,33H DO YOU HAVE A FILE FOR EXPANSION? 00014420
      +23H PUMPED STORAGE PLANTS?/17H ENTER YES OR NO.) 00014430
      READ (5,130) RESP 00014440
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 800 00014450
      WRITE (6,610) 00014460
610 FORMAT (//,42H DO YOU WANT TO CONSIDER STORAGE PLANTS AS? 00014470
      +22H EXPANSION CANDIDATES?,/17H ENTER YES OR NO.) 00014480
      READ (5,130) RESP 00014490
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 710 00014500
      GO TO 900 00014510
710 WRITE (6,715) KH1,KHU 00014520
715 FORMAT (//,5H ONLY,12,14H PLANT TYPE IS, 00014530
      +26H CONSIDERED FOR EXPANSION./15H IT IS ASSIGNED, 00014540
      +15H PLANT CODE NO.,13) 00014550
      GO TO 172 00014560
750 CONTINUE 00014570
      WRITE (6,720) 00014580
720 FORMAT (//,33H IS YOUR INPUT FILE FOR EXPANSION? 00014590
      +25H STORAGE PLANTS COMPLETE?/17H ENTER YES OR NO.) 00014600
      READ (5,130) RESP 00014610
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 850 00014620
      GO TO 217 00014630
800 CONTINUE 00014640
      WRITE (6,715) KH1,KHU 00014650
      READ (49,590) (IHUL(KHU,K),K=1,15) 00014660
      READ (49,590) (IHUM(KHU,K),K=1,15) 00014670
      READ (49,587) (HPLR(KHU,J),J=1,JNUM) 00014680
      WRITE (6,832) 00014690
832 FORMAT (//,31H ENTER YES IF YOU WANT TO PRINT, 00014700
      +23H EXPANSION SYSTEM DATA.) 00014710
      READ (5,130) RESP 00014720
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL SPRINT(3) 00014730
      GO TO 217 00014740
850 CONTINUE 00014750
      REWIND 49 00014760
      WRITE (49,590) (IHUL(KHU,K),K=1,15) 00014770
      WRITE (49,590) (IHUM(KHU,K),K=1,15) 00014780
      WRITE (49,587) (HPLR(KHU,J),J=1,JNUM) 00014790
900 CONTINUE 00014800
      WRITE (6,965) 00014810
965 FORMAT (//,38H PRINTING OF THE ENTIRE PUMPED STORAGE, 00014820
      +21M SYSTEM DATA FOLLOWS./ 17H IF YOU WANT THIS, 00014830
      +20H PRINTING,ENTER YES.) 00014840
      READ (5,130) RESP 00014850
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL SPRINT(1) 00014860
1000 CONTINUE 00014870
C      END OF INPUT PROCEDURE 00014880

```

```

C      WRITE (6,654)
654 FORMAT (//,42H END OF INPUT PROCEDURE FOR PUMPED STORAGE,
+8M PLANTS.,/,41H INPUT DATA STORED ON UNITS 45,48 AND 49.)
      RETURN
      END

C      SUBROUTINE SPRINT(IPRT)
COMMON/TEN/HPLK(6,12),HPLI(6,12),HPLF(12,20),
+      HPLR(6,12)
COMMON/ELEVEN/KEY2(36,18),KEY3(43,18),KEY4(11,18)
COMMON /TWELVE/ KHU,KHM,KHMX,KH1,JNUM,ITYP,IFLAG
COMMON /MULTPS/ PBIN(4),EAM(4)
COMMON/UNITS/ IHUF(6,20),IHUL(6,20),IHUM(6,20),
+      NYR(20)
      INTEGER RESP,KEY2,KEY3,YES,YE
      DIMENSION KH(6)
      DATA YES,YE/3HYES,3HY/
      DO 95 J=1,6
95  KH(J)=J

C      GO TO (200,200,400),IPRT
200 CONTINUE
      WRITE (6,230)
230 FORMAT (//,21H NO. OF UNITS OF EACH,
+30H PLANT CODE WORKING EACH YEAR:)
      WRITE (6,235)
235 FORMAT (18H (EXISTING PLANTS))
      WRITE (6,250) (KH(I),I=1,3)
250 FORMAT (/,12H PS PLT CODE,3I5)
      DO 260 NY=1,15
      KY=NY+1980
      WRITE (6,270) KY,(IHUF(JH,NY),JH=1,3)
260 CONTINUE
270 FORMAT (/,7X,4I5)
      WRITE (6,280)
      WRITE (6,285)
280 FORMAT (//,23H INDIVIDUAL PLANT DATA:)
      WRITE (6,290) (KH(I),I=1,3)
290 FORMAT (//,12H PS PLT CODE,9X,3(1X,I10),/)
      DO 300 J=1,8
      WRITE (6,310) (KEY3(J,K),K=1,6),(HPLR(JH,J),JH=1,3)
300 CONTINUE
      WRITE (6,320)
      IF (IPRT.EQ.2) GO TO 500
310 FORMAT (2X,0A4,3(1X,F10.5))
400 CONTINUE
      WRITE (6,430)
430 FORMAT (//,30H LOWER AND UPPER BOUNDS ON NO.,
+12H OF UNITS OF/31H THE EXPANSION PLANT EACH YEAR:)
      WRITE (6,450) KH(4)
450 FORMAT (/,12H PS PLT CODE,3I10)
      WRITE (6,455)
455 FORMAT (/,19X,1HL,4X,1HU)
      DO 460 NY=1,15
      KY=NY+1980
      WRITE (6,470) KY,IHUL(4,NY),IHUM(4,NY)
460 CONTINUE
470 FORMAT (/,5X,15,5X,6I5)
      WRITE (6,280)
      WRITE (6,435)
435 FORMAT (18H (EXPANSION PLANT))
      WRITE (6,490) KH(4)
490 FORMAT (//,12H PS PLT CODE,10X,I10,/)
      DO 495 J=1,11
      WRITE (6,310) (KEY3(J,K),K=1,6),HPLR(4,J)
495 CONTINUE

```

500 WRITE (6,52G)
500 CONTINUE
520 FORMAT (//,34H * K\$ MEANS DOLLARS IN THOUSANDS)
 RETURN
 END

00015570
00015580
00015590
00015600
00015610

```

***** **** * **** * **** * **** * **** * **** * **** * **** * **** * **** *
      SUBROUTINE LUADI
      THIS PROGRAM READ LOAD DATA AND CALCULATES LOAD DURATION
      CURVES AND CUMULANTS FOR EACH SEASON EVERY YEAR.
      F(FT20F001) DA(LPRZ0.DATA)
      F(FT21F001) DA(LLF21.DATA)
      F(FT25F001) DA(CUR25.DATA)
      F(FT26F001) DA(CUM26.DATA)

      COMMON /YEAR/ NBYR,NBY1,IEND,ICY,NY
      INTEGER YES,Y,IRESP
      DATA IYES,IY/"YES","Y"/
      REWIND 11
      READ (11,10) NDUM1,NDUM2,NBYR,IEND
      NBY1=NBYR-1
      REWIND 11
      10 FORMAT(4I10)
      99 FORMAT(//'* **** * **** START OF LOAD INPUT PROCEDURE **** * ****')
      PRINT 99

      CCCCCCCCCCCCCCCCCC READ LOAD CHARACTERISTICS
      CALL MULT
      CCCCCCCCCCCCCCCCCC READ REFERENCE LOAD MULTIPLIER
      NPROD=91
      NPER=4
      IPNT=500
      101 FORMAT(4A4)
      WRITE (6,102)
      102 FORMAT (/10X,"DOES A FILE OF REFERENCE LOAD DATA",
      +" EXIST? ENTER YES OR NO." )
      READ (5,103) IRESP
      103 FORMAT (A3)
      135 FORMAT (1H,A3)
      IF (IRESP.NE.IYES.AND.IRESP.NE.IY) GO TO 50
      CALL PANDA

      CCCCCCCCCCCCCCCCCC CALCULATE LOAD DURATION CURVE
      AND CUMULANT
      DO 200 I=NBYR,IEND
      NY=I-NBY1
      ICY=I
      CALL DURAT
      CALL CUMUL
      200 CONTINUE
      50 WRITE (6,100)
      106 FORMAT (1H,*' **** * **** END OF INPUT PROCEDURE FOR LOAD DATA****')
      +' LOAD MULTIPLIERS STORED IN UNIT 21./'
      +' LOAD DURATION CURVES STORED IN UNIT 25./'
      +' CUMULANTS STORED IN UNIT 26 //')
      STOP
      END

***** **** * **** * **** * **** * **** * **** * **** * **** * **** *

      SUBROUTINE MULT
      THIS SUBPROGRAM READ THE ENERGY MULTIPLIER AND LOAD FACTOR
      COMMON /YEAR/ NBYR,NBY1,IEND,ICY,NY
      COMMON /CHRLD/ EMLPY(4,30),FLD(4,30)
      +           ,AVKB(4,5),PKAB(4,2),HL(24),PK(4),BSL(4)
      +           ,SUM(4),AVL(4),ELD(500),ELF(4)

```

```

INTEGER Y,YES
DATA Y,YES/ 'Y   ', 'YES '/
30 FORMAT( /"          ENTER: " )
PRINT 3
3 FORMAT( /"          DOES A FILE OF ENERGY MULTIPLIER AND LOAD FACTOR? "
+ " EXISTS?      ENTER YES OR NO." )
READ 99, IANS
IFI IANS.EQ.YES.OR.IANS.EQ.Y) GO TO 201
C C C C C C C C C C C C          READ INPUT DATA
PRINT 30
PRINT 200
DO 70 I=NBYR, IEND
ICY=I
NY=I-NBY1
PRINT 40,I
40 FORMAT( /" YEAR",I5)
PRINT 50
50 FORMAT( "-----")
READ 80, (EMLPY(J,NY),J=1,4)
80 FORMAT( 4F1.4)
70 CONTINUE
100 PRINT#
100 PRINT#
PRINT 130
130 FORMAT( /"          LOAD FACTOR FOR EACH SEASON EACH YEAR" )
DO 170 I=NBYR, IEND
NY=I-NBY1
PRINT 40,I
PRINT 50
READ 80,(FLD(J,NY),J=1,4)
170 CONTINUE
GO TO 202
C C C C C C C C C C C C          PRINT TO VERIFY
201 CONTINUE
REWIND 21
READ (21,510) EMLPY,FLD
REWIND 21
202 CONTINUE
PRINT 200
200 FORMAT( /"          ENERGY MULTIPLIER FOR EACH SEASON EACH YEAR" )
DO 270 I=NBYR, IEND
NY=I-NBY1
DU 231 J=1,4
IF(EMLPY(J,NY).EQ.0) EMLPY(J,NY)=1.0
IF(FLD(J,NY).EQ.0) FLD(J,NY)=0.0
231 CONTINUE
PRINT 300 ,I, (EMLPY(J,NY),J=1,4)
270 CONTINUE
PRINT #
PRINT 130
DO 280 I=NBYR, IEND
NY=I-NBY1
PRINT 300,I,(FLD(J,NY),J=1,4)
280 CONTINUE
PRINT 80
C C C C C C C C C C C C          CHANGE DATA
PRINT 290
290 FORMAT( /"          IS YOUR INPUT COMPLETE ?      ENTER YES OR NO. " )
READ 99,IANS
IFI IANS.EQ.YES.OR.IANS.EQ.Y) GO TO 500
300 FORMAT( /"          YEAR",I5," : ",4F10.5)
300 FORMAT( /"          YEAR",I5," : ",4F10.5)

```

```

315 PRINT 330
330 FORMAT(1X," DO YOU WANT TO CHANGE ENERGY MULTIPLIER?"/
+,"3X," ENTER YES OR NO.") 00001370
READ 99,IANS 00001380
IF(IANS.NE.YES.AND.IANS.NE.Y) GO TO 415 00001390
PRINT 340 00001400
340 FORMAT(" 00001410
ENTER WHAT YEAR.") 00001420
READ *, I 00001430
NY=I-NBY1 00001440
PRINT 40, I 00001450
PRINT 50 00001460
READ 80,(EMLPY(J,NY),J=1,4) 00001470
GO TO 315 00001480
00001490
415 PRINT 430
430 FORMAT(" 00001500
ENTER YES OR NO.") 00001510
READ 99,IANS 00001520
IF(IANS.NE.YES.AND.IANS.NE.Y) GO TO 202 00001530
99 FORMAT(A4) 00001540
PRINT 340 00001550
READ *, I 00001560
NY=I-NBY1 00001570
PRINT 40, I 00001580
PRINT 50 00001590
READ 80,(FLD(J,NY),J=1,4) 00001600
GO TO 415 00001610
500 WRITE (21,510) EMLPY,FLD 00001620
510 FORMAT(1PDE12.5) 00001630
RETURN 00001640
END 00001650
00001660
00001670
***** ****
00001680
SUBROUTINE PANDA
COMMON /YEAR/ NBYR,NBY1,IEND,ILY,NY 00001690
COMMON /CHRLD/ EMLPY(4,30),FLD(4,30) 00001700
+ , AVRL(4,5),PKAB(4,5),HL(24),PK(4),BSL(4) 00001710
+ , SUM(4),AVL(4),ELD(500),ELF(4) 00001720
+ , 00001730
DIMENSION IHRL(24) 00001740
INTEGER SSN(4) 00001750
DATA SSN/"W1","SP","SU","AU"/ 00001760
PKMAX=0.0 00001770
INT=20 00001780
NPER=4 00001790
00001800
DO 10 K=1,4 00001810
BSL(K)=1.0E10 00001820
ND=91 00001830
IF (K.EQ.4) ND=92 00001840
DO 20 ID=1,ND 00001850
READ (INT,899,END=55) (IHRL(I),I=1,24) 00001860
55 CONTINUE 00001870
DO 30 I=1,24 00001880
HL(I)=IHRL(I) 00001890
SUM(K)=SUM(K)+HL(1) 00001900
899 FORMAT (20X,12I5) 00001910
IF (HL(1).GT.PK(K)) PK(K)=HL(1). 00001920
IF (HL(1).NE.0.AND.HL(1).LT.BSL(K)) BSL(K)=HL(1) 00001930
30 CONTINUE 00001940
20 CONTINUE 00001950
AVL(K)=SUM(K)/ND/24 00001960
ELF(K)=AVL(K)/PK(K) 00001970
10 CONTINUE 00001980
00001990
REWIND INT 00002000
DO 40 K=1,NPER 00002010
IF (PK(K).GT.PKMAX) PKMAX=PK(K) 00002020
40 CONTINUE 00002030
00002040

```

```

107 FORMAT (9X, "THE ANNUAL PEAK OF THE REFERENCE LOAD DATA IS ", 00002050
  +1PE13.4, " MW")
  WRITE (6,108) 00002060
108 FORMAT (/," SSN      PEAK LD(MW)  MIN LD(MW)  AVE LD(MW)", 00002070
  + " TOT ENER(MWH)  LOAD FACTOR") 00002080
  DO 70 K=1,NPER 00002090
  WRITE (6,109) SSN(K),PK(K),BSL(K),AVL(K),SUM(K),ELF(K) 00002100
109 FORMAT (3X,A4,1X,1P5E13.5) 00002110
 70 CONTINUE 00002120
  WRITE (6,107) PKMAX 00002130
  PRINT 109 00002140
  PRINT 109 00002150
  PRINT 109 00002160
  REWIND INT 00002170
  DO 140 K=1,4 00002180
  PTS=PK(K)-BSL(K) 00002190
  DO 166 NN=1,4 00002200
  AVRBL(K,NN)=0.0 00002210
166 CONTINUE 00002220
  NPROD=91 00002230
  IF(K.EQ.4) NPROD=92 00002240
  DO 180 ND=1,NPROD 00002250
  READ (INT,809,END=155) (IHRL(I),I=1,24) 00002260
155 CONTINUE 00002270
  DO 160 I=1,24 00002280
  HL(I)=IHRL(I) 00002290
  AVRBL(K,1)=AVRBL(K,1)+HL(I)*(1.0-(HL(I)-BSL(K))/PTS*0.4) 00002300
  AVRBL(K,2)=AVRBL(K,2)+HL(I)*(1.0-(HL(I)-BSL(K))/PTS*0.2) 00002310
  AVRBL(K,3)=AVRBL(K,3)+HL(I) 00002320
  AVRBL(K,4)=AVRBL(K,4)+HL(I)*(1.0+(HL(I)-BSL(K))/PTS*0.2) 00002330
  AVRBL(K,5)=AVRBL(K,5)+HL(I)*(1.0+(HL(I)-BSL(K))/PTS*0.4) 00002340
160 CONTINUE 00002350
180 CONTINUE 00002360
  YY=-0.6 00002370
  DO 151 NN=1,5 00002380
  YY=YY+0.2 00002390
  BX=NPROD*24 00002400
  AVRBL(K,NN)=AVRBL(K,NN)/BX 00002410
  PKAB(K,NN)=PK(K)*(1.0+YY)/AVRBL(K,NN) 00002420
151 CONTINUE 00002430
140 CONTINUE 00002440
  RETURN 00002450
  END 00002460
  00002470
  ****
  SUBROUTINE DURAT 00002480
  COMMON /AVRAGE/ AVT(4) 00002490
  DOUBLE PRECISION AVT 00002500
  COMMON /YEAR/ NBYR,NBY1,IEND,ICY,NY 00002510
  COMMON /CHKLD/ EMLPY(4,30),FLD(4,30) 00002520
  + , AVRD(4,5),PKAB(4,5),HL(24),PK(4),BSL(4) 00002530
  + , SUM(4),AVL(+),ELD(500),ELF(4) 00002540
  DIMENSION IHRL(24) 00002550
  DOUBLE PRECISION ETOT,AVLD 00002560
108 FORMAT (" SSN      PEAK LD(MW)  MIN LD(MW)  AVE LD(MW)", 00002570
  + " TOT ENER(MWH)  LOAD FACTOR") 00002580
  PRINT 100, ICY 00002590
100 FORMAT (/," YEAR UF ",14) 00002600
  PRINT 108 00002610
  PKMAX=0.0 00002620
  PTS=500 00002630
  INT=20 00002640
  REWIND INT 00002650
  IPNT=500 00002660
  DO 214 K=1,4 00002670
  ETOT = 0.0 00002680
  CALL KONST(K,C1,C2,PKF) 00002690
  PRINT 214 00002700
  ETOT = 0.0 00002710
  PRINT 214 00002720

```

```

209 DO 209 J=1,IPNT          00002730
      ELD(J)=0.0          00002740
      NPRD=91          00002750
      IF (K.EQ.4) NPRD=92          00002760
      DX=(PK(K)*PKF-BSL(K)*C1)/FLOAT(IPNT-1)
      PRINT *,DX,C1,C2,PKF          00002770
      DO 218 ND=1,NPRD          00002780
      READ (INT,899,LND=255) (IHRL(I),I=1,24)          00002790
      FORMAT(20X,1215)          00002800
      899 FORMAT(20X,1215)          00002810
      255 CONTINUE          00002820
      00002830
      DO 216 I=1,24          00002840
      HL(I)=IHRL(I)          00002850
      EKK= PK(K)-BSL(K)          00002860
      HL(I)=HL(I)*( C1+(HL(I)-BSL(K))/EKK+C2)          00002870
      IF( HL(I).EQ.0) GO TO 216          00002880
      LA=(HL(I)-BSL(K)*C1)/DX+1.0          00002890
      IF(LA.LT.1.0R-LA.GT.1.0) GO TO 400          00002900
      ELD(LA)=ELD(LA)+1.0          00002910
      ETOT=ETOT+HL(I)          00002920
      216 CONTINUE          00002930
      300 FORMAT(10F7.0)          00002940
      218 CONTINUE          00002950
      DO 340 IX=2,IPNT          00002960
      I=IPNT+1-IX          00002970
      ELD(I)=ELD(I)+ELD(I+1)          00002980
      340 CONTINUE          00002990
      C C C C C C C C           MAKE A FILE OF LOAD DURATION 00003000
      WRITE (25) 1CY,K          00003010
      355 FORMAT(1",YEAR=",15," SEASLN=",15)          00003020
      BSL0=BSL(K)*C1          00003030
      PKC=PK(K)*PKF          00003040
      AVLD=ETOT/(24*NPRD)          00003050
      WRITE (25) DX,BSLD,PKC,ETOT,AVLD          00003060
      WRITE (25) ELD          00003070
      AVT(K)=AVLD          00003080
      ELB=AVLD/PKC          00003090
      PRINT 109,K, PKC,BSLD,AVLD,ETOT,ELB          00003100
      109 FORMAT (3X,14,1X,1P5E13.5)          00003110
      350 FORMAT(1P5E12.5)          00003120
      214 CONTINUE          00003130
      RETURN          00003140
      00003150
      400 PRINT 410,LA,I,ND          00003160
      410 FORMAT(" ERROR WITH UNIT 20 : SEE SUBROUTINE DURAT")          00003170
      REWIND INT          00003180
      RETURN          00003190
      END          00003200
      00003210
***** **** * ***** **** * ***** **** * ***** **** * ***** **** * ****
      SUBROUTINE CUMUL          00003220
      COMMON /AVRG/ AVT(4)          00003230
      DOUBLE PRECISION AVT          00003240
      COMMON /YEAR/ NBYK,NBY1,IEND,1CY,NY          00003250
      COMMON /CHRLU/ EMLPY(4,30),FLU(4,30)          00003260
      + , AVRB(4,5),PKAB(4,5),HL(24),PK(4),BSL(4)          00003270
      + , SUM(4),AVL(4),ELU(500),ELF(4)          00003280
      + DIMENSION AVG(4),HOURC(24),HRL(24),EM(8),IHRL(24)          00003290
      DOUBLE PRECISION HOURC,EM          00003300
      INT=20          00003310
      REWIND INT          00003320
      DO 1008 J=1,8          00003330
      1008 EM(J)=0.0          00003340
      DC 1004 K=1,4          00003350
      PRINT 110          00003360
      CALL KUNST(K,C1,C2,PKF)          00003370
      NPRD=91          00003380
      00003390
      00003400

```

```

IF (K.EQ.4) NPROD=42          00003410
DO 1020 ND=1,NPROD           00003420
READ (INT,659+END=66) (IMRL(1),I=1,24) 00003430
66 CONTINUE                   00003440
DO 1025 I=1,24               00003450
HL(I)=IMRL(I)                00003460
EKK= PK(K)-ESL(K)            00003470
HL(I)=HL(I)+(C1+(HL(I)-BSL(K))/EKK+C2) 00003480
HRL(I)=HL(I)-AVT(K)          00003490
HOURL=1.0                     00003500
DO 1030 J=1,8                 00003510
HOURL=HOUKL+HRL(I)           00003520
1030 EM(J)=EM(J)+HOUKL      00003530
1025 CONTINUE                  00003540
1020 CONTINUE                  00003550
                                         00003560
                                         00003570
DO 1035 J=1,8                 00003580
1035 EM(J)=EM(J)/(NPROD*24)   00003590
HOURC(1)=EM(1)                00003600
HOURC(2)=EM(2)                00003610
HOURC(3)=EM(3)                00003620
HOURC(4)=EM(4)-3.*EM(2)*EM(2) 00003630
HOURC(5)=EM(5)-10.*EM(3)*EM(2) 00003640
HOURC(6)=EM(6)-15.*EM(4)*EM(2)-10.*EM(5)*EM(3)+ 00003650
+ 30.*EM(2)*EM(2)             00003660
HOURC(7)=EM(7)-21.*EM(5)*EM(2)-55.*EM(4)*EM(3)+ 00003670
+ 210.*EM(3)*EM(2)*EM(2)       00003680
HOURC(8)=EM(8)-28.*EM(6)+EM(2)-20.*EM(5)*EM(3)- 00003690
+ 35.*EM(4)*EM(4)+20.*EM(4)*EM(2)*EM(2)+ 00003700
+ 300.*EM(3)*EM(3)*EM(2)-50.*EM(2)*EM(2)*EM(2)*EM(2) 00003710
355 FORMAT(7X,I1, YEAR=,I5,I1, SEASON=,I1,I4X,I1,AVERAGE=,1PE20.10) 00003720
HOURC(1)=AVT(K)               00003730
PRINT 110                     00003740
PRINT 134                     00003750
WRITE (6,355) ICY,K,AVT(K)    00003760
134 FORMAT (1H ,THE LOCAL CUMULANTS FOR *,15,IX,14,* ARE*) 00003770
WRITE (6,110) (HOURC(J),J=1,8) 00003780
WRITE (20,355) ICY,K,AVT(K)   00003790
WRITE (20,110) (HOURC(J),J=1,8) 00003800
1004 CONTINUE                  00003810
                                         00003820
110 FORMAT(1P3E23.14)           00003830
899 FORMAT(20X,I2I5)           00003840
REWIND INT                     00003850
RETURN                         00003860
END                            00003870
***** ****
SUBROUTINE KUNS(K,C1,C2,PKF) 00003880
COMMON /YEAR/ NYR,NBY1,IEND,ICY,NY 00003890
COMMON /CHRLU/ EMLPY(4,30),FLU(4,30) 00003900
+ , AVKB(4,5),PKAB(4,5),HL(24),PK(4),BSL(4) 00003910
+ , SUM(4),AVL(4),ELU(500),ELF(4) 00003920
PTAA=1.0/FLD(K,NY)             00003930
XA=0.0                         00003940
DO 10 NN=2,5                   00003950
XA=X(A+0.2                      00003960
XB=X(A+0.2                      00003970
XB=X(A+0.2                      00003980
XB=X(A+0.2                      00003990
IF(PTAA.GT.PKAB(K,NN)) GO TO 20 00004000
10 PTAB=PKAB(K,NN)              00004010
PTAK=PKAB(K,NN-1)               00004020
X=0.2*(PTAA-PTAK)/(PTAB-PTAK)+XA 00004030
AVA=((XB-X)*AVKB(K,NN-1)+(X-XA)*AVRB(K,NN))/0.2 00004040
C1=AVL(K)*EMLPY(K,NY)/AVA      00004050
C2=C1*X                         00004060
PKF=C1+C2                        00004070
                                         00004080

```

RETURN
30 IF(NN.EQ.5) GO TO 10
40 CONTINUE
END

00004090
00004100
00004110
00004120

**** PREP SUBMODULE ****

THIS SUBMODULE DETERMINES, FOR EACH YEAR IN THE STUDY PERIOD

- (1) MINIMUM AND MAXIMUM RESERVES
- (2) ECONOMIC LOADING ORDER
- (3) LOWER BOUND OF OPERATION COSTS

ON INPUT :

- (1) PLANT DATA (UNIT 11)
- (2) NORMAL DISTRIBUTION TABLE (UNIT 15)

ON OUTPUT :

- (1) FINAL LDC AFTER THE CONVOLUTION OF SCHEDULED SYSTEM FOR EACH YEAR(UNIT 22)
- (2) ECONOMIC LOADING ORDER AND SCHEDULED SYSTEM MAINTENANCE ALLOCATION FOR EACH YEAR (UNIT 24)
- (3) HOURLY LOAD CUMULANTS AND SCHEDULED SYSTEM OUTAGE CUMULANTS FOR EACH YEAR(UNIT 27)
- (4) PLANT DATA (UNIT 23)
- (5) OTHER INFORMATION REQUIRED BY DYNO-SUBMODULE(UNIT 10).

COMMON VARIABLE DESCRIPTION

NAME	TYPE	SIZE	DEFINITION	
AVSPE	R*4	20,4	AVSPE(I,J)—AVAILABLE SPACE FOR MAINTENANCE ALLOCATION I : YEAR INDEX J : SEASON INDEX	00000010 00000020 00000030 00000040 00000050 00000060 00000070 00000080 00000090 00000100 00000110 00000120 00000130 00000140 00000150 00000160 00000170 00000180 00000190 00000200 00000210 00000220 00000230 00000240 00000250 00000260 00000270 00000280 00000290 00000300 00000310 00000320 00000330 00000340 00000350 00000360 00000370 00000380 00000390 00000400 00000410 00000420 00000430 00000440 00000450 00000460 00000470 00000480 00000490 00000500 00000510 00000520 00000530 00000540 00000550 00000560 00000570 00000580 00000590 00000600 00000610 00000620 00000630 00000640 00000650 00000660 00000670 00000680
CSTOP	R*4	20	LOWER BOUND OF OPERATING COSTS FOR EACH YEAR	
CMULT	R*4	4	SEASNCAL CAPACITY MULTIPLIER	
EAVAL	R*4	3	EAVAL(I)—MAX AVAILABLE HYDRO ENERGY IN MWH I=1 SCHLD HYDRO PLANT I=2 EXPN HYDRO PLANT I=3 PUMPED STORAGE	
EC	R*4	20	SCHEDULED SYSTEM CAPACITY FOR EACH YEAR	
EL	R*4	4,1250	EL(I,J)—NORMALIZED LOAD DURATION CURVE I : SEASON INDEX J : ABSCISSA INDEX	
EMOR	R*4	220,4	EMOR(I,J)—THERMAL PLANT MAINTENANCE OUTAGE I : PLANT INDEX (EXPNS&SCHLD) J : INDEX OF SEASON IN WHICH MAINTENANCE IS SCHEDULED	
EMORH	R*4	3,4	EMORH(I,J)--HYDRO PLANT MAINTENANCE OUTAGE I=1 SCHEDULED HYDRO I=2 EXPANSION HYDRO I=3 PUMPED STORAGE J : SAME AS IN EMOR	
EMULT	R*4	4	SEASONAL ENERGY MULTIPLIER	
EXPLCU	R*4	4,8	EXPLCU(I,J)—EXPN PLANT OUTAGE CUMULANTS I : FIRST FOUR CUMULANTS J : PLANT TYPE	
HRCUM	R*8	4,8	HRCUM(I,J)—LOAD CUMULANTS FOR EACH SEASON I : SEASON INDEX J : CUMULANT ORDER	

C	HYDR	R*4	3,11	HYDR(I,J) — HYDRO PLANT CHARACTERISTICS I=1 SCHEDULED HYDRO I=2 EXPANSION HYDRO I=3 PUMPED STORAGE J=1 I=1,2 : BASE CAPACITY MW I=3 : STORAGE CAPACITY MW J=2 I=1,2 : MAX CAPACITY MW I=3 : ENERGY LIMIT GWH J=3 I=1,2 : STR ENERGY LIMIT GWH I=3 : PUMPING EFFICIENCY J=4 I=1,2 : ENERGY LIMIT GWH I=3 : GENERATING EFFICIENCY J=5 : FORCED OUTAGE RATE J=6 : MAINTENANCE OUTAGE D/YR J=7 : FIXED OEM COST K\$/MW/YR J=8 : VARIABLE OEM COST \$/MWH J=9 : CAPITAL COST \$/KW J=10 : CAPITAL COST ESCALATION RATE J=11 : SALVAGE VALUE K\$	00000690 00000700 00000710 00000720 00000730 00000740 00000750 00000760 00000770 00000780 00000790 00000800 00000810 00000820 00000830 00000840 00000850 00000860 00000870 00000880 00000890 00000900 00000910 00000920 00000930 00000940 00000950 00000960 00000970 00000980 00000990 00001000 00001010 00001020 00001030 00001040 00001050 00001060 00001070 00001080 00001090 00001100 00001110 00001120 00001130 00001140 00001150 00001160 00001170 00001180 00001190 00001200 00001210 00001220 00001230 00001240 00001250 00001260 00001270 00001280 00001290 00001300 00001310 00001320 00001330 00001340 00001350 00001360
	HYEXPN	R*4	11	HYEXPN(I) — HYDRO EXPN CHARACTERISTICS I : SAME AS IN J OF HYDR ARRAY	00000880
	IBK	I	500,3	IBK(I,J) — ECONOMIC LOADING ORDER I : LOADING POSITION INDEX J=1 PLANT CODE NUMBER J=2 CAPACITY BLOCK IDENTIFIER J=3 NO. OF UNITS (-1 DENOTES AN EXPN TYPE) FLAG USED IN MIN RESERVE CALCULATION	00000900 00000910 00000920 00000930 00000940 00000950
	ID	I	-	ID=1 SCHEDULED SYSTEM CONVOLUTION ID=2 EXPANSION ADDITION ID=3 RESIMULATION WITH MIXED SYSTEM	00000960
	IPMAX	I	-	TOTAL NU. OF THERMAL PLANT TYPES CONSIDERED	00000980
	JPT	I	-	NO. OF POINTS DESIRED IN PW-LOLP CALCULATION	00000990
	ITK	I	500,3	ITK(I,J) — EFFECTIVE LOADING ORDER FOR OPERATING COST CALCULATION I : SAME AS IN IBK J : SAME AS IN IBK	00001000 00001010
	IUP	I	20,20	IUP(I,J) — ALLOWED UPPER BOUND OF THERMAL EXPN CANDIDATES I : PLANT INDEX J : YEAR INDEX	00001020 00001030 00001040 00001050
	IX	I	20	THERMAL EXPN PLANT TYPES INDEX ARRAY	00001060
	IYRB	I	-	BEGINNING YEAR OF STUDY PERIOD	00001070
	JYR	I	-	STUDY YEAR COUNTER	00001080
	KMAX	I	-	MAX NO. ON LOADING POSITIONS	00001090
	MAINS	I	200	SCHEDULED THERMAL SYSTEM MAINTENANCE	00001100
	MAXI	I	-	ALLOCATION SPECIFICATION	00001110
	MX	I	-	EXPECTED MAX NO. OF POINTS IN X-ARRAY FOR LOLP CALCULATION	00001120
	MXELDC	I	20,4	MAX NO. OF POINTS ALLOWED IN LOLP CALCULATION MXELDC(I,J) — MAX NO. OF POINTS IN THE FINAL EL CURVE AFTER THE CONVOLUTION OF SCHLD THERMAL SYSTEM	00001130 00001140 00001150 00001160 00001170 00001180 00001190
	NHYNM	I	20	I : YEAR INDEX J : SEASON INDEX MIN NO. OF HYDRO EXPN UNITS SPECIFIED FOR EACH	00001200 00001210 00001220 00001230
	NHYMX	I	20	STUDY YEAR MAX NO. OF HYDRO EXPN UNITS ALLOWED FOR EACH	00001240 00001250
	NP	I	-	STUDY YEAR NO. OF SIMULATION SEASONS(PERIODS)	00001260 00001270
	NPEXP	I	-	NO. OF THERMAL EXPN PLANT TYPES	00001280
	NPSMN	I	20	MIN NO. OF PUMPED STORAGE EXPN UNITS SPECIFIED FOR EACH STUDY YEAR	00001290
	NPSMX	I	20	MAX NO. OF PUMPED STORAGE EXPN UNITS ALLOWED FOR EACH YEAR	00001300 00001310
	NSTPRE	I	20,10	NSTPRE(I,J) — EXPN ADDITION CONFIGURATION FOR EACH YEAR I : YEAR INDEX J : PLANT TYPE	00001320 00001330 00001340 00001350 00001360

NT	I	20	CUMULATIVE NO OF UNITS OF THERMAL EXPANSION	00001370
NTHY	I	20	ADDITION FOR EACH PLANT TYPE	00001380
NTPS	I	20	CUMULATIVE NO OF UNITS OF HYDRO EXPANSION	00001390
NTYR	I	20	FOR EACH YEAR	00001400
NUF	I	220,20	CUMULATIVE NO OF UNITS OF PUMPED STORAGE	00001410
			EXPANSION FOR EACH YEAR	00001420
			LENGTH OF STUDY PERIOD	00001430
			NUF(I,J)--FOR THERMAL SCHLD SYSTEM : MAX NO OF UNITS ALLOWED FOR EACH TYPE	00001440
			FOR THERMAL EXPN SYSTEM : LOWER BOUND FOR EACH TYPE	00001450
			I : PLANT INDEX	00001460
			J : YEAR INDEX	00001470
NX	I	20	NO OF THERMAL EXPN ADDITION FOR EACH TYPE	00001480
NXHY	I	20	NO FU HYDRO EXPN ADDITION FOR EACH YEAR	00001490
NXPS	I	20	NO OF PUMPED STORAGE EXPN ADDITION FOR EACH YEAR	00001500
PEAK	R*4	20,4	PEAK(I,J)--SEASONAL PEAK LOAD DEMAND FOR EACH YEAR IN MW	00001510
			I : YEAR INDEX	00001520
			J : SEASON INDEX	00001530
PERENE	R*4	20,4	PERENE(I,J)--TOTAL ENERGY DEMAND IN MWH FOR EACH SEASON EACH YEAR	00001540
			I : YEAR INDEX	00001550
			J : SEASON INDEX	00001560
PLANT	R*4	200,14	PLANT(I,J)--THERMAL PLANT CHARACTERISTICS	00001570
			I : PLANT CODE NUMBER	00001580
			J=2 BASE CAPACITY MW	00001590
			J=3 MAX CAPACITY MW	00001600
			J=4 MAINT OUTAGE DAYS/YR	00001610
			J=5 FORCED OUTAGE RATE	00001620
			J=6 CAPITAL COST \$/KW	00001630
			J=7 BASE FUEL COST \$/MWH	00001640
			J=8 MAX FUEL COST \$/MWH	00001650
			J=9 PLANT ECONOMIC LIFE YR	00001660
			J=10 FIXED OEM COST \$/MW/YR	00001670
			J=11 VARIABLE OEM COST \$/MWH	00001680
			J=12 SALVAGE VALUE K\$	00001690
			J=13 CAPITAL ESCALATION RATE	00001700
			J=14 FUEL ESCALATION RATE	00001710
PLACA	R*4	20,14	PLACA(I,J)--THERMAL EXPN CHARACTERISTICS	00001720
			I : PLANT INDEX	00001730
PPMAX	R*4	-	J : SAME AS IN PLANT ARRAY	00001740
PPMIN	R*4	-	LOLP VALUE ASSIGNED FOR MAX RESERVE	00001750
			CALCULATION	00001760
			LOLP VALUE ASSIGNED FOR MIN RESERVE	00001770
PSEXP	R*4	11	PSEXP(I)--PUMPED STORAGE EXPN CHARACTERISTICS	00001780
PTCUM	R*8	211,16	PTCUM(I,J)--THERMAL PLANT OUTAGE CUMULANTS	00001790
			I : PLANT INDEX	00001800
			J=1 8 FOR BASE BLOCK	00001810
			J=9 16 FOR PEAK BLOCK	00001820
PTCUMH	R*8	3,16	PTCUMH(I,J)--HYDRO PLANT OUTAGE CUMULANTS	00001830
			I=1 SCHLD HYDRO PLANT	00001840
			I=2 EXPN HYDRO PLANT	00001850
			I=3 PUMPED STORAGE	00001860
			J : SAME AS IN PTCUM ARRAY	00001870
RSMAX	R*4	20	MAX RESERVE FOR EACH YEAR IN MW	00001880
RSMIN	R*4	20	MIN RESERVE FOR EACH YEAR IN MW	00001890
SCHCUM	R*4	4,20	SCHCUM(I,J)--SUM OF LOAD&SCHLD PLANT CUMULANTS FOR PEAK PERIOD FOR EACH YEAR	00001900
			I : FIRST FOUR CUMULANTS	00001910
			J : YEAR INDEX	00001920
TINT	R*8	120	CUMULATIVE NORMAL DISTRIBUTION	00001930
TY	R*8	120	CORRESPONDING TO TY	00001940
			NORMALIZED INDEP VARIABLE IN NORMAL DISTR. TABLE	00001950
				00001960
				00001970
				00001980
				00001990
				00002000
				00002010
				00002020
				00002030
				00002040

X	R*4 1250 NORMALIZED MW ARRAY	00002050 00002060 00002070 00002080 00002090 00002100 00002110 00002120 00002130 00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
<u>MAIN PROGRAM</u>		
+ COMMON/ONE/	EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20), PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250)	00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
+ COMMON/TWO/	EMOR(220,4),EMORH(3,4),IX(20),MAINS(200), NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20), RSMAX(20)	00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
+ COMMON/THREE/	AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3), ITK(500,3),MXELOC(20,4),NSTPRE(20,10), PERENE(20,4)	00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
+ COMMON/FOUR/	ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP, NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP	00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
+ COMMON/FIVE/	HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8) DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM	00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
+ COMMON/SIX/	TINT(120),TY(120) DOUBLE PRECISION TINT,TY	00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
+ COMMON/SEVEN/	CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8), HYEXP(11),NHYN(20),NPSMN(20),NTHY(20), NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20)	00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
DATA K11,K15/11,15/		00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
UERS SPECIFICATION OF PPMIN & PPMAX		
960	WRITE(6,960) FORMAT(1H ,*SPECIFY THE CRITICAL LOLP VALUE TO BE USED IN THE*/ * MINIMUM RESERVE CALCULATION IN DAYS PER TEN YEARS*) READ(5,*) PPMIN WRITE(6,970)	00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
970	FORMAT(1H ,*SPECIFY THE LOLP VALUE TO BE USED IN THE MAXIMUM*/ * RESERVE CALCULATION IN DAYS PER TEN YEARS*) READ(5,*) PPMAX PPMAX=PPMAX/3650.0 PPMIN = PPMIN/3650.	00002140 00002150 00002160 00002170 00002180 00002190 00002200 00002210 00002220 00002230 00002240 00002250 00002260 00002270 00002280 00002290 00002300 00002310 00002320 00002330 00002340 00002350 00002360 00002370 00002380 00002390 00002400 00002410 00002420 00002430 00002440 00002450 00002460 00002470 00002480 00002490 00002500 00002510 00002520 00002530 00002540 00002550 00002560 00002570 00002580 00002590 00002600 00002610 00002620 00002630 00002640 00002650 00002660 00002670 00002680 00002690 00002700 00002710 00002720
*** SECTION 1 ***		
INITIALIZE PARAMETERS AND READ YEAR INDEPENDENT INPUT		
NP=4		

NCUM=4	00002730
IPT=500	00002740
JNUM=14	00002750
MX=1250	00002760
DXNORM = 1.0/FLOAT(IPT)	00002770
DO 25 I = 1,1250	00002780
25 X(I) = FLOAT(I) * DXNORM	00002790
 C CCCCC	00002800
READ PLANT DATA	00002810
 C CCCCC	00002820
READ(K11,1030) IPMAX,NPEXP,IYRB,IYRE	00002830
NTYR=IYRE-IYRB+1	00002840
DO 10 IP=1,IPMAX	00002850
READ(K11,1040) (NUF(IP,N),N=1,20)	00002860
READ(K11,1050) (PLANT(IP,J),J=1,JNUM)	00002870
PLANT(IP,9)=30.	00002880
 C CCCCC	00002890
WRITE PLANT INFORMATION ON UNIT 23	00002900
 C CCCCC	00002910
10 WRITE(23,904) (PLANT(IP,J),J=1,JNUM)	00002920
CONTINUE	00002930
I1=NPEXP+1	00002940
DO 12 I=I1,IPMAX	00002950
WRITE(23,1040) (NUF(I,N),N=1,NTYR)	00002960
12 CONTINUE	00002970
 C CCCCC	00002980
STORE INFORMATION FOR EXP. SYSTEM	00002990
 C CCCCC	00003000
DO 15 IP=1,NPEXP	00003010
DO 15 J=1,JNUM	00003020
15 PLACA(IP,J)=PLANT(IP,J)	00003030
 C CCCCC	00003040
DO 20 IP=1,NPEXP	00003050
READ(K11,1040) (IUP(IP,N),N=1,20)	00003060
20 CONTINUE	00003070
 C CCCCC	00003080
READ NORMAL DISTRIBUTION TABLE	00003090
 C CCCCC	00003100
READ(K15,1060) (TY(I),TINT(I),I=1,120)	00003110
 C CCCCC	00003120
HYDRO MULTIPLIERS	00003130
 C CCCCC	00003140
READ(30,1080) (CMULT(I),I=1,4)	00003150
READ(30,1080) (EMULT(I),I=1,4)	00003160
 C CCCCC	00003170
HYDRO EXPN SYSTEM	00003180
 C CCCCC	00003190
READ(32,1090) NDUM	00003200
READ(32,1100) (NHYMN(I),I=1,20)	00003210
READ(32,1100) (NHYMX(I),I=1,20)	00003220
READ(32,1080) (HYEXPN(J),J=1,11)	00003230
 C CCCCC	00003240
PUMPED STORAGE EXPN SYSTEM	00003250
 C CCCCC	00003260
	00003270
	00003280
	00003290
	00003300
	00003310
	00003320
	00003330
	00003340
	00003350
	00003360
	00003370
	00003380
	00003390
	00003400

```

READ(34,1100) (NPSMN(I),I=1,20)          00003410
READ(34,1100) (NPSMX(I),I=1,20)          00003420
READ(34,1080) (PSEXP(J),J=1,11)          00003430
                                              00003440
                                              00003450
                                              00003460
                                              00003470
*** S E C T I O N  2 ***
                                              00003480
START SIMULATION FOR EACH STUDY YEARS      00003490
                                              00003500
DO 30 J=1,NTYR                            00003510
JYR=J                                     00003520
IYEAR=IYRB+JYR-1+1900                     00003530
WRITE(17,1070) IYEAR                      00003540
                                              00003550
                                              00003560
                                              00003570
READ YEAR DEPENDENT INPUTS                00003580
                                              00003590
CALL RINPUT                                00003600
                                              00003610
DETERMINE MIN. & MAX. RESERVES            00003620
                                              00003630
CALL RESERV                                00003640
                                              00003650
                                              00003660
                                              00003670
DETERMINE LOADING ORDER ACCORDING TO FUEL COST 00003680
                                              00003690
                                              00003700
CALL LOADER                                00003710
                                              00003720
DETERMINE HYDRO COMPOSITE CHARACTERISTICS   00003730
                                              00003740
CALL COMPST                                 00003750
                                              00003760
                                              00003770
                                              00003780
                                              00003790
COMPUTE LOWER BOUND OF OPERATION COSTS     00003800
                                              00003810
                                              00003820
                                              00003830
CALL OPCOST(INCUM)                         00003840
                                              00003850
30 CONTINUE                               00003860
                                              00003870
CREATE FILES FOR DP-MODULE                00003880
                                              00003890
CALL INOUT                                 00003900
                                              00003910
                                              00003920
                                              00003930
                                              00003940
                                              00003950
                                              00003960
                                              00003970
904 FORMAT(F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2) 00003980
1030 FORMAT(4I10)                           00003990
1040 FORMAT(4O12)                           00004000
1050 FORMAT(5E15.5)                          00004010
1060 FORMAT(F10.4,F12.8)                   00004020
1070 FORMAT(/1H1,19(=:))// STUDY YEAR = *,15/1X,19(=)// 00004030
1080 FORMAT(5E12.5)                          00004040
1090 FORMAT(/,3I3)                           00004050
1100 FORMAT(20I3)                           00004060
1120 FORMAT(//1H1,*** END OF PREP SUBMODULE SIMULATION ***//) 00004070
STOP                                     00004080

```

```

END
00004090
00004100
00004110
00004120
00004130
00004140
00004150
00004160
00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

SUBROUTINE RESERV

COMMON/ONE/ EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
+ PEAK(20,4),PLACA(20,14),X(1250) 00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

COMMON/TWO/ EMOR(220,4),EMORM(3,4),IX(20),MAINS(200),
+ NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
+ RSMAX(20) 00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
+ ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
+ PERENE(20,4) 00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

COMMON/FOUR/ ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
+ NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP 00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

COMMON/FIVE/ HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM 00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

COMMON/SIX/ TINT(120),TY(120)
DOUBLE PRECISION TINT,TY 00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),
+ HYEXP(11),NHYN(20),NPSMN(20),NTHY(20),
+ NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

-----  

THIS ROUTINE COMPUTES THE MINIMUM RESERVE MARGIN FOR EACH  

YEAR. IT IS OBTAINED BY REQUIRING THAT THE CRITICAL  

SEASONAL LOLP IS LESS THAN OR EQUAL TO THE CRITICAL  

VALUE ASSIGNED BY THE USER AT THE BEGINNING. 00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

-----  

OBTAINED INFORMATION ARE EXPRESSED IN TERMS OF : 00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760

-----  

RSV(YEAR) : AMOUNT IN MW 00004490
: IN PERCENT 00004500
NX(I) : NUMBER OF UNIT OF TYPE IX(I) 00004510
: (EXPANSION SYSTEM ONLY) 00004520
NT(I) : TOTAL NO OF ADDITION OF 00004530
: TYPE IX(I) 00004540
: (EXPANSION SYSTEM ONLY) 00004550
IX(I) : TYPE OF UNIT 00004560
: (EXPANSION SYSTEM ONLY) 00004570
SUBROUTINE CALLED : 00004580
LPD 00004590
-----  

DIMENSION IUT(20),SCAP(4) 00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760
-----  


```

```

C FIND YEARLY PEAK 00004770
C PMAX(JYR)=0.0 00004780
DO 10 I=1,4 00004790
IF(PMAX(JYR,I).GT.PMAX(JYR)) PMAX(JYR)=PEAK(JYR,I) 00004800
10 CONTINUE 00004810
00004820
00004830
00004840
00004850
00004860
00004870
00004880
00004890
00004900
00004910
00004920
00004930
00004940
00004950
00004960
00004970
00004980
00004990
00005000
00005010
00005020
00005030
00005040
00005050
00005060
00005070
00005080
00005090
00005100
00005110
00005120
00005130
00005140
00005150
00005160
00005170
00005180
00005190
00005200
00005210
00005220
00005230
00005240
00005250
00005260
00005270
00005280
00005290
00005300
00005310
00005320
00005330
00005340
00005350
00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440

*** SECTION 1 ***
SET EXPN CANDIDATE INDEX ARRAY 00004850
DETERMINE UPPER BOUND OF EXPN CANDIDATE 00004860
INITIALIZE NX-ARRAY 00004870
00004880
00004890
00004900
00004910
00004920
00004930
00004940
00004950
00004960
00004970
00004980
00004990
00005000
00005010
00005020
00005030
00005040
00005050
00005060
00005070
00005080
00005090
00005100
00005110
00005120
00005130
00005140
00005150
00005160
00005170
00005180
00005190
00005200
00005210
00005220
00005230
00005240
00005250
00005260
00005270
00005280
00005290
00005300
00005310
00005320
00005330
00005340
00005350
00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440

IF(JYR.GT.1) GO TO 30
IX(1)=NPEXP 00004930
IUT(1)=IUP(1,JYR)
NX(1)=0 00004940
NT(1)=0 00004950
DO 20 I=2,NPEXP 00004960
IX(I)=IX(I-1)-1 00004970
NX(I)=0 00004980
NT(I)=0 00004990
20 IUT(I)=IUP(I,JYR)
NHAVAL=NHYMX(JYR) 00005000
NPAVAL=NPSMX(JYR) 00005010
NXPS(JYR)=0 00005020
NTPS(JYR)=0 00005030
NXHY(JYR)=0 00005040
NTHY(JYR)=0 00005050
GO TO 50 00005060
00005070
00005080
00005090
00005100
00005110
00005120
00005130
00005140
00005150
00005160
00005170
00005180
00005190
00005200
00005210
00005220
00005230
00005240
00005250
00005260
00005270
00005280
00005290
00005300
00005310
00005320
00005330
00005340
00005350
00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440

30 CONTINUE 00005110
00005120
00005130
00005140
00005150
00005160
00005170
00005180
00005190
00005200
00005210
00005220
00005230
00005240
00005250
00005260
00005270
00005280
00005290
00005300
00005310
00005320
00005330
00005340
00005350
00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440

DETERMINE NO OF UNITS AVAILABLE FOR THE PRESENT STUDY 00005140
YEAR AND INITIALIZE NX(I) ARRAY 00005150
00005160
00005170
00005180
00005190
00005200
00005210
00005220
00005230
00005240
00005250
00005260
00005270
00005280
00005290
00005300
00005310
00005320
00005330
00005340
00005350
00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440

DO 40 I=1,NPEXP 00005180
IUT(IX(I))=IUP(IX(I),JYR)-NT(I)
40 NX(I)=0 00005190
JPAST=JYR-1 00005200
NHAVAL=NHYMX(JYR)-NTHY(JPAST) 00005210
NPAVAL=NPSMX(JYR)-NTPS(JPAST) 00005220
NXPS(JYR)=0 00005230
NXHY(JYR)=0 00005240
NTHY(JYR)=0 00005250
GO TO 50 00005260
00005270
00005280
00005290
00005300
00005310
00005320
00005330
00005340
00005350
00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440

*** SECTION 2 ***
MINIMUM RESERVE CALCULATION 00005290
THIS IS OBTAINED BY REQUIRING THAT CRITICAL LOLP 00005300
IS LESS THAN OR EQUAL TO THE ASSIGNED PPMIN. 00005310
00005320
00005330
00005340
00005350
00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440

SCHEDULED SYSTEM CONVOLUTION 00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440

50 ID=1 00005380
IDL=-1 00005390
ADD = 0.0 00005400
CALL SYSLP(CLP,SCAP,ECP,IDL,NCRT,NHAVAL,NPAVAL) 00005410
IF(CLP.LE.PPMIN) GO TO 1001 00005420
00005430
00005440

```

```

C OTHERWISE PUT HYDRO EXPN UNITS ON LINE
C ONE AT A TIME
C
C ID=2
C IDL=0
60 IF(NHAVAL.LE.0) GO TO 65
NXHY(JYR)=NXHY(JYR)+1
ADD=ADD+HYEXP(2)
CALL SYSLP(CLP,SCAP,ECP,IDL,NCRT,NHAVAL,NPAVAL)
IF(CLP.LE.PPMIN) GO TO 1001
NHAVAL=NHAVAL-1
GO TO 60
C
C PUMP STORAGE EXPN ADDITION
C
65 IF(NPAVAL.LE.0) GO TO 70
NXPS(JYR)=NXPS(JYR)+1
ADD=ADD+PSEXP(1)
CALL SYSLP(CLP,SCAP,ECP,IDL,NCRT,NHAVAL,NPAVAL)
IF(CLP.LE.PPMIN) GO TO 1001
NPAVAL=NPAVAL-1
GO TO 65
C
C OTHERWISE PUT THE SMALLEST THERMAL CANDIDATE ON LINE
C ONE AT A TIME
C
70 CONTINUE
ID=2
IDL=1
NX(1)=1
80 ADD=ADD+PLANT(IX(1),3)
CALL SYSLP(CLP,SCAP,ECP,IDL,NCRT,NHAVAL,NPAVAL)
IF(CLP.LE.PPMIN) GO TO 100
90 NX(1)=NX(1)+1
GO TO 80
100 IF(NX(1).LE.IUT(IX(1))) GO TO 1001
C
C DISTRIBUTE THIS AMOUNT BETWEEN OTHER AVAILABLE
C CANDIDATES
C
IDL=2
RV=ADD
DO 110 J=2,NPEXP
NX(J)=1
IF(NX(J).LE.IUT(IX(J))) GO TO 115
GO TO 110
115 IE=J-1
170 ADD=0.0
DO 120 I=1,IE
120 NX(I)=0
ADD=ADD+FLOAT(NX(J))*PLANT(IX(J),3)
IF(ADD.GE.RV) GO TO 155
DO 130 I=1,IE
NX(I)=1
160 IF(NX(I).LE.IUT(IX(I))) GO TO 140
IF(I.EQ.IE) GO TO 150
NX(I)=NX(I)-1
GO TO 130
140 ADD=ADD+PLANT(IX(I),3)
IF(ADD.GE.RV) GO TO 155

```

```

      NX(I)=NX(I)+1          00006130
      GO TO 160              00006140
130  CONTINUE              00006150
170  NX(J)=NX(J)+1          00006160
      IF(NX(J).GT.IUT(IX(J))) GO TO 110  00006170
      GO TO 170              00006180
155  CALL SYSLP(CLP,SCAP,ECP,IDL,NCRT,NHAVAL,NPAVAL) 00006190
      IF(CLP.LE.PPMIN) GO TO 1001  00006200
C
C   START ADDING THE SMALLEST CANDIDATE AGAIN          00006210
C
C   GO TO 90              00006220
110  CONTINUE              00006230
C
C   PRINT 1020              00006240
1020 FORMAT(1X," NO CONVERGED SOLUTION ",/,
           *1X," CHECK YOUR INPUT FOR EXPANSION SYSTEM//") 00006250
      GO TO 110              00006260
C
C   1001 RSMIN(JYR)=ECP-PMAX(JYR)          00006280
      IF(RSMIN(JYR).LE.0.0) RSMIN(JYR)=0.0  00006290
      RMIN=RSMIN(JYR)/PMAX(JYR)*100.0  00006300
C
C   STORE CURRENT EXPN ADDITION IN NSTPRE-ARRAY          00006310
C
C   DO 900 I=1,NPEXP          00006320
      II=NPEXP-1+1          00006330
900  NSTPRE(JYR,I)=NX(II)  00006340
C
C   DETERMINED PSEUDO-PLANT PARAMETERS          00006350
C
C   II=IPMAX+1          00006360
      PLANT(II,2)=ADD          00006370
      PLANT(II,3)=ADD          00006380
      PLANT(II,4)=10.0          00006390
      PLANT(II,5)=0.02          00006400
      IP=IBK(1,1)          00006410
      DO 505 J=6,14          00006420
      505 PLANT(II,J)=PLANT(IP,J)  00006430
      DO 510 I=1,NPEXP          00006440
      J=NPEXP-1+1          00006450
      IF(INT(J).LE.0) GO TO 510  00006460
      IP=IX(J)          00006470
      PLANT(IP,2)=PLANT(IP,3)  00006480
      DO 520 K=4,14          00006490
      520 PLANT(IP,K)=PLANT(II,K)  00006500
      510 CONTINUE          00006510
C
C   REALLOCATE MAINTENANCE          00006520
C
C   ID=3          00006530
      CALL MAINT1(ECP,IDL)  00006540
C
C   FIND TOTAL NO OF UNITS ADDED UP TO THE PRESENT          00006550
      STUDY YEAR          00006560
C
C   DO 500 I=1,NPEXP          00006570

```

```

500 NT(I)=NT(I)+NX(I) 00006810
IF(JYR.LE.1) GO TO 530 00006820
NTHY(JYR)=NTHY(JPAST)+NXHY(JYR) 00006830
NTPS(JYR)=NTPS(JPAST)+NXPS(JYR) 00006840
GO TO 540 00006850
530 NTHY(JYR)=NXHY(JYR) 00006860
NTPS(JYR)=NXPS(JYR) 00006870
00006880
00006890
00006900
00006910
00006920
00006930
00006940
00006950
00006960
00006970
00006980
00006990
00007000
00007010
00007020
00007030
00007040
00007050
00007060
00007070
00007080
00007090
00007100
00007110
00007120
00007130
00007140
00007150
00007160
00007170
00007180
00007190
00007200
00007210
00007220
00007230
00007240
00007250
00007260
00007270
00007280
00007290
00007300
00007310
00007320
00007330
00007340
00007350
00007360
00007370
00007380
00007390
00007400
00007410
00007420
00007430
00007440
00007450
00007460
00007470
00007480

*** SECTION 3 ***
MAX RESERVE CALCULATION
THIS IS OBTAINED BY REQUIRING THAT THE SMALLEST LOLP
IS LESS THAN OR EQUAL TO THE ASSIGNED PPMAX.

540 IF(CLP.LE.PPMAX) GO TO 600 00007000
CALL RESMX(ECP,SCAP,NCRL) 00007010
RSMAX(JYR)=ECP-PMAX(JYR) 00007020
RMAX=RSMAX(JYR)/PMAX(JYR)*100.0 00007030
GO TO 610 00007040
600 RS MAX(JYR)=RSMIN(JYR)*1.1 00007050
RMAX=RMIN*1.1 00007060
00007070
00007080
00007090
00007100
00007110
00007120
00007130
00007140
00007150
00007160
00007170
00007180
00007190
00007200
00007210
00007220
00007230
00007240
00007250
00007260
00007270
00007280
00007290
00007300
00007310
00007320
00007330
00007340
00007350
00007360
00007370
00007380
00007390
00007400
00007410
00007420
00007430
00007440
00007450
00007460
00007470
00007480

PRINT RESULTS
610 CONTINUE
WRITE(17,3) RSMIN(JYR),RMIN 00007120
WRITE(17,6) RS MAX(JYR),RMAX 00007130
WRITE(17,8) CLP 00007140
00007150
00007160
00007170
00007180
00007190
00007200
00007210
00007220
00007230
00007240
00007250
00007260
00007270
00007280
00007290
00007300
00007310
00007320
00007330
00007340
00007350
00007360
00007370
00007380
00007390
00007400
00007410
00007420
00007430
00007440
00007450
00007460
00007470
00007480

C 1100 RETURN
END

SUBROUTINE SYSLP(CLP,SCAP,ECP,IDL,NCRT,NHAVAL,NPAVAL)
COMMON/ONE/ EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
+ PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250) 00007400
00007410
00007420
00007430
00007440
00007450
00007460
00007470
00007480

COMMON/TWO/ EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
+ NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
+ RS MAX(20) 00007440
00007450
00007460
00007470
00007480

```

```

COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
+ ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
+ PERENE(20,4) 00007490
00007500
00007510
00007520
00007530
00007540
00007550
00007560
00007570
00007580
00007590
00007600
00007610
00007620
00007630
00007640
00007650
00007660
00007670
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

COMMON/FOUR/ ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
+ NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP 00007530
00007540
00007550
00007560
00007570
00007580
00007590
00007600
00007610
00007620
00007630
00007640
00007650
00007660
00007670
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

COMMON/FIVE/ HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM 00007580
00007590
00007600
00007610
00007620
00007630
00007640
00007650
00007660
00007670
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

COMMON/SIX/ TINT(120),TY(120)
DOUBLE PRECISION TINT,TY 00007610
00007620
00007630
00007640
00007650
00007660
00007670
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

COMMON/SEVEN/ CMULT(4),EVAL(3),EMULT(4),EXPLCU(4,8),
+ HYEXP(11),NHYN(20),NPSMN(20),NTHY(20),
+ NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00007650
00007660
00007670
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

COMPUTES SEASONAL LOLP USING PIECE-WISE LINEAR
METHOD. MAINTENANCE OUTAGE IS CONSIDERED USING
EQUIVALENT CAPACITY ALGORITHM. 00007700
00007710
00007720
00007730
00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

SUBROUTINE CALLED :
MAINT1,PHADD,PLOLP 00007740
00007750
00007760
00007770
00007780
00007790
00007800
00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

DIMENSION PLP(4),ELDC(4,1250),SUMY(4),SCAP(4)
EQUIVALENCE (PTCUM(1),ELDC(1)) 00007810
00007820
00007830
00007840
00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

K22=22
ISTAR=(JYR-1)*4+1 00007850
00007860
00007870
00007880
00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

CALL MAINT1(ECP,IDL) 00007890
00007900
00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

PP=0.0
CLP=0.0 00007910
00007920
00007930
00007940
00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

LOOP FOR EACH SEASON 00007950
00007960
00007970
00007980
00007990
00008000
00008010
00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

DO 20 NS=1,4 00007980
NS=NS 00007990
PK=PEAK(JYR,N) 00008000
DX=PK/FLOAT(IPT) 00008010
MAXI=(ECP/DX)+1+IPT 00008020
00008030
00008040
00008050
00008060
00008070
00008080
00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

GO TO 100,IDL 00008050
100 CONTINUE 00008060
*** SECTION 1 *** 00008070
SCHEDULED SYSTEM CONVOLUTION 00008080
ID=1 00008090
00008100
00008110
00008120
00008130
00008140
00008150
00008160

SUMY(N)=0.0 00008150
00008160

```

```

SCAP(N)=0.0          00008170
HYDRO SYSTEM CONVOLUTION      00008180
CONVOLVE THE SCHEDULED SYSTEM FIRST 00008190
THEN CONVOLVE THE EXPN SYSTEM DETERMINED 00008200
FROM PREVIOUS STUDY YEARS 00008210
00008220
00008230
00008240
00008250
00008260
00008270
00008280
00008290
00008300
00008310
00008320
00008330
00008340
00008350
00008360
00008370
00008380
00008390
00008400
00008410
00008420
00008430
00008440
00008450
00008460
00008470
00008480
00008490
00008500
00008510
00008520
00008530
00008540
00008550
00008560
00008570
00008580
00008590
00008600
00008610
00008620
00008630
00008640
00008650
00008660
00008670
00008680
00008690
00008700
00008710
00008720
00008730
00008740
00008750
00008760
00008770
00008780
00008790
00008800
00008810
00008820
00008830
00008840

SCHEDULED HYDRO & PUMP STORAGE SYSTEM

Y=HYDR(1,2)*(1.0-EMORH(1,N))/PK
P=1.0-HYDR(1,5)
SUMY(N)=SUMY(N)+Y
SCAP(N)=SCAP(N)+Y
CALL PWADD(Y,P,N,NMAXI)
Y=HYDR(3,1)*(1.0-EMORH(3,N))/PK
P=1.0-HYDR(3,5)
SUMY(N)=SUMY(N)+Y
SCAP(N)=SCAP(N)+Y
CALL PWADD(Y,P,N,NMAXI)

DO 110 I=1,IPMAX
IL=IPMAX-I+1
IF(IL.GT.NPEXP) GO TO 120
IF(INT(IL).LE.0) GO TO 110
JE=NT(IL)
IP=IX(IL)
IN=2*IP
GO TO 130
120 IF(NUF(IL,JYR).LE.0) GO TO 110
JE=NUF(IL,JYR)
IP=IL
IN=IP+NPEXP
C
130 DO 140 J=1,JE
Y=PLANT(IP,3)*(1.0-EMOR(IN,N))/PK
SUMY(N)=SUMY(N)+Y
SCAP(N)=SCAP(N)+Y
P=1.0-PLANT(IP,5)
140 CALL PWADD(Y,P,N,NMAXI)
IF(IL-1.EQ.NPEXP) GO TO 150
GO TO 110
150 CONTINUE
C
STORE INFORMATION AFTER THE CONVOLUTION OF
THE ORIGINAL SCHEDULED SYSTEM
MXELDC(JYR,N)=NMAXI
WRITE(K22) (EL(N,K),K=1,MX)
C
110 CONTINUE
C
EXPANSION HYDRO SYSTEM FROM PREVIOUS YEARS

IF(JYR.LE.1) GO TO 90
JPAST=JYR-1
IF(NTHY(JPAST).LE.0) GO TO 80
HMULT=FLOAT(NTHY(JPAST))
Y=HMULT*HYEXP(2)*(1.0-EMORH(2,N))/PK
P=1.0-HYEXP(5)
SUMY(N)=SUMY(N)+Y
SCAP(N)=SCAP(N)+Y
CALL PWADD(Y,P,N,NMAXI)

```

```

C
PUMPED STORAGE SYSTEM FROM PREVIOUS YEARS
C
80 IF(NTPS(JPAST).LE.0) GO TO 90          00008850
HMULT=FLOAT(NTPS(JPAST))                  00008860
Y=HMULT*PSEXP(1)*(1.0-EMORH(3,N))/PK      00008870
P=1.0-PSEXP(5)                            00008880
SUMY(N)=SUMY(N)+Y                          00008890
SCAP(N)=SCAP(N)+Y                          00008900
CALL PWADD(Y,P,N,NMAXI)                   00008910
90 S=SUMY(N)                                00008920
A=S                                         00008930
IF(NHAVAL.LE.0.AND.NPAVAL.LE.0) GO TO 160  00008940
GO TO 500                                  00008950
160 DO 170 I=1,1250                         00008960
170 ELDC(N,I)=EL(N,I)                      00008970
GO TO 500                                  00008980
C
200 CONTINUE
C
*** SECTION 2 ***
START ADDING EXPANSION SYSTEM
IDL=2                                     00009000
IDL=0 -- HYDRO EXPN ADDITION             00009010
IDL=1 -- SMALLEST THERMAL ADDITION       00009020
IDL=2 -- RESIMULATION                     00009030
C
IF(IDL.LE.0) GO TO 300                    00009040
IF(IDL.EQ.1) GO TO 205                    00009050
C
RETRIVE FINAL LDC FROM ELDC ARRAY
C
STH=SUMY(N)                                00009060
SCAP(N)=SUMY(N)                            00009070
DO 210 I=1,1250                           00009080
210 EL(N,I)=ELDC(N,I)                      00009090
GO TO 206                                  00009100
C
300 IF(NHAVAL.LE.0) GO TO 310              00009110
Y=HYEXP(2)*(1.0-EMORH(2,N))/PK            00009120
P=1.0-HYEXP(5)                            00009130
SUMY(N)=SUMY(N)+Y                          00009140
SCAP(N)=SCAP(N)+Y                          00009150
CALL PWADD(Y,P,N,NMAXI)                   00009160
A=SUMY(N)                                   00009170
IF(NHAVAL.GT.1) GO TO 500                 00009180
IF(NPAVAL.LE.0) GO TO 320                 00009190
GO TO 500                                  00009200
C
310 Y=PSEXP(1)*(1.0-EMORH(3,N))/PK        00009210
P=1.0-PSEXP(5)                            00009220
SUMY(N)=SUMY(N)+Y                          00009230
SCAP(N)=SCAP(N)+Y                          00009240
CALL PWADD(Y,P,N,NMAXI)                   00009250
A=SUMY(N)                                   00009260
IF(NPAVAL.GT.1) GO TO 500                 00009270
320 DO 330 I=1,1250                         00009280
330 ELDC(N,I)=EL(N,I)                      00009290
GO TO 500                                  00009300
C

```

```

205 IF(NX(1).GT.1) GO TO 206          00009530
      STH=SUMY(N)                      00009540
      SCAP(N)=SUMY(N)                  00009550
206 DO 220 I=1,NPEXP                 00009560
      IF(NX(I).LE.0) GO TO 220         00009570
      JE=NX(I)
      IP=IX(I)
      IN=2*IP                         00009580
      IN=2*IP                         00009590
      JB=1                            00009600
      IF(IDL.EQ.1) JE=1                00009610
      DO 230 J=JB,JE                  00009620
      Y=PLANT(IP,3)*(1.0-EMOR(IN,N))/PK
      STH=STH+Y                      00009630
      SCAP(N)=SCAP(N)+Y               00009640
      P=1.0-PLANT(IP,5)              00009650
230 CALL PWADD(Y,P,N,NMAXI)        00009660
220 CONTINUE                         00009670
      A=STH                           00009680
                                         00009690
                                         00009700
                                         00009710
                                         00009720
                                         00009730
                                         00009740
                                         00009750
                                         00009760
                                         00009770
                                         00009780
                                         00009790
                                         00009800
                                         00009810
                                         00009820
                                         00009830
                                         00009840
                                         00009850
                                         00009860
                                         00009870
                                         00009880
                                         00009890
                                         00009900
                                         00009910
                                         00009920
                                         00009930
                                         00009940
                                         00009950
                                         00009960
                                         00009970
                                         00009980
                                         00009990
00010000
00010010
00010020
00010030
00010040
00010050
00010060
00010070
00010080
00010090
00010100
00010110
00010120
00010130
00010140
00010150
00010160
00010170
00010180
00010190
00010200

CCCCC
      FIND LOLP

500 CONTINUE
      PLP(N)=PLOLP(A,N)
      IF(PLP(N).GE.CLP) GO TO 510
      GO TO 520
510 CLP=PLP(N)
      NCRT=N
520 PP=PP+PLP(N)
20 CONTINUE

CCCCC
      FIND SYSTEM LOLP
      PLOP=PP/FLOAT(NP)

C      RETURN
      END

CCCCC
SUBROUTINE MAINT1(ECP,IDL)

CCCCC
      COMMON/ONE/
      +     EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
      +     PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250)
      +     00009980
      +     00009990
      +     00010000
      +     00010010
      +     00010020
      +     00010030
      +     00010040
      +     00010050
      +     00010060
      +     00010070
      +     00010080
      +     00010090
      +     00010100
      +     00010110
      +     00010120
      +     00010130
      +     00010140
      +     00010150
      +     00010160
      +     00010170
      +     00010180
      +     00010190
      +     00010200

CCCCC
      COMMON/TWO/
      +     EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
      +     NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
      +     RSMAX(20)
      +     00010020
      +     00010030
      +     00010040
      +     00010050
      +     00010060
      +     00010070
      +     00010080
      +     00010090
      +     00010100
      +     00010110
      +     00010120
      +     00010130
      +     00010140
      +     00010150
      +     00010160
      +     00010170
      +     00010180
      +     00010190
      +     00010200

CCCCC
      COMMON/THREE/
      +     AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
      +     ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
      +     PERENE(20,4)
      +     00010020
      +     00010030
      +     00010040
      +     00010050
      +     00010060
      +     00010070
      +     00010080
      +     00010090
      +     00010100
      +     00010110
      +     00010120
      +     00010130
      +     00010140
      +     00010150
      +     00010160
      +     00010170
      +     00010180
      +     00010190
      +     00010200

CCCCC
      COMMON/FOUR/
      +     ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
      +     NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
      +     00010020
      +     00010030
      +     00010040
      +     00010050
      +     00010060
      +     00010070
      +     00010080
      +     00010090
      +     00010100
      +     00010110
      +     00010120
      +     00010130
      +     00010140
      +     00010150
      +     00010160
      +     00010170
      +     00010180
      +     00010190
      +     00010200

CCCCC
      COMMON/FIVE/
      HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
      +     00010020
      +     00010030
      +     00010040
      +     00010050
      +     00010060
      +     00010070
      +     00010080
      +     00010090
      +     00010100
      +     00010110
      +     00010120
      +     00010130
      +     00010140
      +     00010150
      +     00010160
      +     00010170
      +     00010180
      +     00010190
      +     00010200

```

```

COMMON/SIX/ TINT(120),TY(120) 00010210
DOUBLE PRECISION TINT,TY 00010220
C
COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8), 00010230
+ HYEXP(11),NHYMN(20),NPSMN(20),NTHY(20), 00010240
+ NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00010250
C----- 00010260
C----- 00010270
C----- 00010280
C----- 00010290
C----- ALLOCATES MAINTENANCE FOR SCHEDULED SYSTEM AND 00010300
C----- EXPANSION SYSTEM ACCORDING TO FLAG CONTROL ID. 00010310
C----- MAINTENANCE OUTAGE IS STORED IN EMOR(IP,NP), 00010320
C----- WHERE IP=PLANT INDEX, NP=SEASON INDEX. 00010330
C----- 00010340
C----- DEFINITION : 00010350
C----- ID=1, FOR "SCHEDULED" SYSTEM (ORIGINAL SCHEDULED 00010360
C----- SYSTEM+EXP. CANDIDATES DETERMINED FROM 00010370
C----- PREVIOUS STUDY YEARS). 00010380
C----- ID= 2, FOR EXPANSION SYSTEM WITH ONE-BLOCK 00010390
C----- REPRESENTATION. 00010400
C----- ID=3, RESCHEDULE MAINTENANCE (ALLOCATE ORIGINAL 00010410
C----- SCHEDULED SYSTEM FIRST THEN NEW EXP. 00010420
C----- CANDIDATES). 00010430
C----- 00010440
C----- 00010450
C----- 00010460
C----- 00010470
C----- 00010480
C----- DIMENSION AVSP(4),APT(4),APH(4),EP(20) 00010490
C----- 00010500
C----- 00010510
C----- 00010520
C----- RLT=365./FLOAT(NP) 00010530
C----- GO TO 1100,200,400,1D 00010540
C----- 00010550
C----- *** S E C T I O N 1 ***
C----- ALLOCATE MAINTENANCE FOR SCHEDULED SYSTEM 00010560
C----- ID=1 00010570
C----- 00010580
C----- 100 EP(JYR)=0.0 00010590
C----- EC(JYR)=0.0 00010600
C----- DO 105 J=1,4 00010610
C----- 00010620
C----- 00010630
C----- 00010640
C----- 105 EMOR(I,J)=0.0 00010650
C----- DO 106 I=1,3 00010660
C----- 00010670
C----- 106 EMORH(I,J)=0.0 00010680
C----- 00010690
C----- 00010700
C----- 00010710
C----- 00010720
C----- 00010730
C----- 00010740
C----- DETERMINE INSTALLED SYSTEM CAPACITY 00010750
C----- 00010760
C----- 00010770
C----- IB=1 00010780
C----- IE=IPMAX 00010790
C----- DO 110 I=IB,IE 00010800
C----- IF(I.GT.NPEXP) GO TO 111 00010810
C----- EP(JYR)=EP(JYR)+FLOAT(NT(I))*PLANT(IX(I),3) 00010820
C----- GO TO 110 00010830
C----- 111 EP(JYR)=EP(JYR)+FLOAT(NUF(I,JYR))*PLANT(I,3) 00010840
C----- EC(JYR)=EC(JYR)+FLOAT(NUF(I,JYR))*PLANT(I,3) 00010850
C----- 110 CONTINUE 00010860
C----- EC(JYR)=EC(JYR)+HYDR(1,2)+HYDR(3,1) 00010870
C----- EP(JYR)=EP(JYR)+HYDR(1,2)+HYDR(3,1) 00010880

```

```

IF(JYR.LE.1) GO TO 112
JPAST=JYR-1
HMULT=FLOAT(NTHY(JPAST))
EP(JYR)=EP(JYR)+HMULT*HYEXP(2)
HMULT=FLOAT(NTPS(JPAST))
EP(JYR)=EP(JYR)+HMULT*PSEXP(1)
112 ECP = EP(JYR)

C
C          HYDRO MAINTENANCE ALLOCATION
C
DO 120 N=1,NP
120 AVSP(N)=ECP-PEAK(JYR,N)
DO 121 I=1,3
IF(I.GT.1) GO TO 122
D=HYDR(1,6)
Y=HYDR(1,2)
HMULT=1.0
GO TO 124
122 IF(I.GT.2) GO TO 123
IF(JYR.LE.1) GO TO 127
IF(NTHY(JYR).LE.0) GO TO 127
D=HYEXP(6)
Y=HYEXP(2)
HMULT=FLOAT(NTHY(JPAST))
GO TO 124
127 DO 128 N=1,NP
128 EMORH(2,N)=EMORH(1,N)
GO TO 121
123 D=HYDR(3,6)
Y=HYDR(3,1)
HMULT=1.0
124 EMXAS=0.0
DO 125 L=1,NP
IF(AVSP(L).GT.EMXAS) GO TO 126
GO TO 125
126 EMXAS=AVSP(L)
LL=L
125 CONTINUE
EMORH(I,LL)=D/RLT
AVSP(LL)=AVSP(LL)-Y*HMULT*EMORH(I,LL)
IF(I.LT.3) GO TO 121
IF(JYR.LE.1) GO TO 121
IF(NTPS(JYR).LE.0) GO TO 121
AVSP(LL)=AVSP(LL)-PSEXP(1)*FLOAT(NTPS(JPAST))*EMORH(I,LL)
121 CONTINUE

C
C          DO 129 N=1,NP
129 APH(N)=AVSP(N)

C          THERMAL MAINTENANCE ALLOCATION
C
DO 130 I=IB,IE
IF(I.GT.NPEXP) GO TO 131
IF(INT(I).EQ.0) GO TO 130
IP=IX(I)
IN=2*IP
GO TO 132
131 IF(NUF(I,JYR).EQ.0) GO TO 130
IP=I
IN=IP+NPEXP
132 EMXAS=0.0
DO 140 L=1,NP
IF(AVSP(L).GT.EMXAS) GO TO 150
      00010890
      00010900
      00010910
      00010920
      00010930
      00010940
      00010950
      00010960
      00010970
      00010980
      00010990
      00011000
      00011010
      00011020
      00011030
      00011040
      00011050
      00011060
      00011070
      00011080
      00011090
      00011100
      00011110
      00011120
      00011130
      00011140
      00011150
      00011160
      00011170
      00011180
      00011190
      00011200
      00011210
      00011220
      00011230
      00011240
      00011250
      00011260
      00011270
      00011280
      00011290
      00011300
      00011310
      00011320
      00011330
      00011340
      00011350
      00011360
      00011370
      00011380
      00011390
      00011400
      00011410
      00011420
      00011430
      00011440
      00011450
      00011460
      00011470
      00011480
      00011490
      00011500
      00011510
      00011520
      00011530
      00011540
      00011550
      00011560

```

```

      GO TO 140
150 EMXAS=AVSP(L)
      LL=L
140 CONTINUE
      EMOR(IN,LL)=PLANT(IP,4)/RLT
      IF(I.GT.NPEXP) GO TO 133
      AVSP(LL)=AVSP(LL)-PLANT(IP,3)*FLOAT(NT(I))*EMOR(IN,LL)
      GO TO 130
133 AVSP(LL)=AVSP(LL)-PLANT(IP,3)*FLOAT(NUF(I,JYR))*EMOR(IN,LL)
130 CONTINUE
      DO 160 N=1,NP
160 APT(N)=AVSP(N)
      GO TO 1000
C
C 200 CONTINUE
C
*** SECTION 2 ***
      ALLOCATE MAINTENANCE FOR EXPANSION SYSTEM
      USING 1-BLOCK REPRESENTATION
      ID=2
      ECP=EP(JYR)
      HMULT=FLOAT(NXHY(JYR))
      ECP=ECP+HMULT*HYEXP(2)
      DO 265 N=1,NP
      AVSP(N)=APT(N)
265 AVSP(N)=AVSP(N)+HMULT*HYEXP(2)*(1.0-EMORH(2,N))
      IF(NXPS(JYR).LE.0) GO TO 266
      HMULT=FLOAT(NXPS(JYR))
      ECP=ECP+HMULT*PSEXP(1)
      DO 270 N=1,NP
270 AVSP(N)=AVSP(N)+HMULT*PSEXP(1)*(1.0-EMORH(3,N))
266 IF(IDL.EQ.0) GO TO 1000
      DO 210 I=1,NPEXP
      IF(NX(I).EQ.0) GO TO 210
      ECP=ECP+FLOAT(NX(I))*PLANT(IX(I),3)
210 CONTINUE
      DO 220 N=1,NP
      DO 230 I=1,NPEXP
      IF(NX(I).EQ.0) GO TO 230
      AVSP(N)=AVSP(N)+FLOAT(NX(I))*PLANT(IX(I),3)
230 CONTINUE
220 CONTINUE
      DO 240 I=1,NPEXP
      IF(NX(I).EQ.0) GO TO 240
      IP=IX(I)
      IN=2*IP
      EMXAS=0.0
      DO 250 N=1,NP
      IF(AVSP(N).GT.EMXAS) GO TO 260
      GO TO 250
260 EMXAS=AVSP(N)
      NN=N
250 CONTINUE
      EMOR(IN,NN)=PLANT(IP,4)/RLT
      AVSP(NN)=AVSP(NN)-PLANT(IP,3)*FLOAT(NX(I))*EMOR(IN,NN)
240 CONTINUE
      GO TO 1000
C
C 400 CONTINUE
C

```

```

*** SECTION 2 ***
REALLOCATE MAINTENANCE WITH A SINGLE
PSEUDO PLANT REPRESENTING TOTAL CURRENT
THERMAL EXPANSION ADDITION

INITIALIZATION
DO 405 J=1,4
DO 405 I=1,220
405 EMOR(I,J)=0.0
HMULT=FLOAT(NXHY(JYR))
DO 700 N=1,NP
AVSPE(JYR,N)=EC(JYR)-PEAK(JYR,N)
700 AVSP(N)=APH(N)+HMULT*HYEXP(2)*(1.0-EMORH(1,N))
HMULT=FLOAT(NXPS(JYR))
DO 710 N=1,NP
710 AVSP(N)=AVSP(N)+HMULT*PSEXP(1)*(1.0-EMORH(3,N))
IP = IPMAX + 1
IN = IP + NPEXP
IF ( PLANT(IP,3).LE.0.0 ) GO TO 435
EMXAS=0.0
DO 420 N=1,NP
420 IF(AVSP(N).GT.EMXAS) GO TO 430
GO TO 420
430 EMXAS=AVSP(N)
NN=N
420 CONTINUE
EMOR(IN,NN)=PLANT(IP,4)/RLT
C
AVSP(NN)=AVSP(NN)-PLANT(IP,3)*EMOR(IN,NN)
435 CONTINUE
C
IB=NPEXP+1
IE=IPMAX
495 CONTINUE
C
DO 440 I=IB,IE
IF(I.GT.NPEXP) GO TO 450
I1=NPEXP-I+1
IF(INT(I1).EQ.0) GO TO 440
IP=IX(I1)
IN=2*IP
GO TO 460
450 IP=I
IN=I+NPEXP
C
460 IF( PLANT(IP,4).LE.0.0 ) GO TO 440
EMXAS=0.0
DO 470 N=1,NP
470 IF(AVSP(N).GT.EMXAS) GO TO 480
GO TO 470
480 EMXAS=AVSP(N)
NN=N
470 CONTINUE
IF(I.LE.NPEXP) GO TO 475
K=I-NPEXP
MAINS(K)=NN
475 EMOR(IN,NN)=PLANT(IP,4)/RLT
IF(I.GT.NPEXP) GO TO 490
AVSP(NN)=AVSP(NN)-PLANT(IP,3)*FLOAT(NT(I1))*EMOR(IN,NN)
GO TO 440
490 AVSP(NN)=AVSP(NN)-PLANT(IP,3)*FLOAT(NUF(IP,JYR))*EMOR(IN,NN)

```

```

AVSPE(JYR,NN)=AVSPE(JYR,NN)-PLANT(IP,3)*FLOAT(NUF(IP,JYR))      00012930
1*EMOR(IN,NN)      00012940
440 CONTINUE      00012950
C      00012960
C      IF(IB.EQ.1) GO TO 600      00012970
DO 510 N=1,4      00012980
AVSPE(JYR,N)=AVSPE(JYR,N)-HYDR(1,2)*EMORH(1,N)      00012990
AVSPE(JYR,N)=AVSPE(JYR,N)-HYDR(3,1)*EMORH(3,N)      00013000
510 CONTINUE      00013010
C      00013020
C      00013030
C      00013040
C      00013050
C      IB=1      00013060
C      IE=NPEXP      00013070
C      GO TO 495      00013080
600 CONTINUE      00013090
EMXAS=0.0      00013100
DO 610 L=1,NP      00013110
IF(AVSP(L).GT.EMXAS) GO TO 620      00013120
GO TO 610      00013130
620 EMXAS=AVSP(L)      00013140
LL=L      00013150
610 CONTINUE      00013160
C      00013170
C      EMOR(2,LL)=PLANT(1,4)/RLT      00013180
C      00013190
1000 RETURN      00013200
END      00013210
C      00013220
C      00013230
C      00013240
C      00013250
C      00013260
C      00013270
C      00013280
C      00013290
C      SUBROUTINE PWADD(Y,P,N,NEWMAX)
C      00013300
C      COMMON/ONE/    EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
+ PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250)      00013310
C      00013320
C      COMMON/TWO/    EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
+ NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
+ RSMAX(20)      00013340
C      00013350
C      COMMON/THREE/   AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
+ ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
+ PERENE(20,4)      00013410
C      00013420
C      00013430
C      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
+ NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP      00013440
C      00013450
C      00013460
C      COMMON/FIVE/    HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM      00013480
C      00013490
C      00013500
C      00013510
C      COMMON/SIX/     TINT(120),TY(120)
DOUBLE PRECISION TINT,TY      00013520
C      00013530
C      00013540
C      00013550
C      00013560
C      COMMON/SEVEN/   CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),
+ HYEXP(11),NHYMN(20),NPSMN(20),NTHY(20),
+ NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20)      00013570
C      00013580
C      00013590
C      00013600

```

```

C DIMENSION ELF(1250) 00013610
C C C DATA PLPHAX/0.999/, ELMIN/0.000001/, ONE/1.0/ 00013620
C IF(P.GE.PLPHAX) GO TO 100 00013630
C Q=ONE-P 00013640
C J=1 00013650
C DO 50 I=1,MAXI 00013660
C XY=X(I)-Y 00013670
C IF(XY.LE.X(1)) GO TO 30 00013680
C J=J+1 00013690
C 25 IF(ABS(XY-X(J)).LE.ELMIN) GO TO 40 00013700
C ELY=(X(J-1)-XY)/(X(J-1)-X(J))*EL(N,J)+ 00013710
C * (XY-X(J))/(X(J-1)-X(J))*EL(N,J-1) 00013720
C GO TO 45 00013730
C 30 ELY=ONE 00013740
C GO TO 45 00013750
C 40 ELY=EL(N,J) 00013760
C 45 CONTINUE 00013770
C ELF(I)=P*EL(N,I)+Q*ELY 00013780
C MAXIMU=I 00013790
C IF(ELF(I).LE.ELMIN) GO TO 55 00013800
C 50 CONTINUE 00013810
C 55 NEWMAX=MAXIMU 00013820
C IF (NEWMAX.GT.MAXI) NEWMAX=MAXI 00013830
C DO 60 I=1,NEWMAX 00013840
C EL(N,I)=ELF(I) 00013850
C 60 CONTINUE 00013860
C 100 CONTINUE 00013870
C RETURN 00013880
C END 00013890
C C C C FUNCTION PLOLP(Y,N) 00013890
C C C C
C COMMON/ONE/ EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20), 00013900
C + PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250) 00013910
C COMMON/TWO/ EMOR(220,4),EMORH(3,4),IX(20),MAINS(200), 00013920
C + NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20), 00013930
C + RSMAX(20) 00013940
C COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3), 00013950
C + ITK(500,3),MXELDC(20,4),NSTPRE(20,10), 00013960
C + PERENE(20,4) 00013970
C COMMON/FOUR/ ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP, 00013980
C + NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP 00013990
C COMMON/FIVE/ HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8) 00014000
C DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM 00014010
C COMMON/SIX/ TINT(120),TY(120) 00014020
C DOUBLE PRECISION TINT,TY 00014030
C COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8), 00014040
C + 00014050
C + 00014060
C + 00014070
C + 00014080
C + 00014090
C + 00014100
C + 00014110
C + 00014120
C + 00014130
C + 00014140
C + 00014150
C + 00014160
C + 00014170
C + 00014180
C + 00014190
C + 00014200
C + 00014210
C + 00014220
C + 00014230
C + 00014240
C + 00014250
C + 00014260
C + 00014270
C + 00014280

```

```

+          HYEXP(11),NHYMN(20),NPSMN(20),NTHY(20),
+          NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20)      00014290
+          YY=Y*IPT                                         00014300
+          IY=YY                                         00014310
+          MAXIM=MAXI-1                                     00014320
+          IF(IY .GT. MAXIM) GO TO 100                      00014330
+          DY=IY+1-YY                                      00014340
+          PLOLP=(EL(N,IY)-EL(N,IY+1))*DY + EL(N,IY+1)    00014350
+          RETURN                                         00014360
100     PLOLP=0.0                                       00014370
+          RETURN                                         00014380
+          END                                           00014390
+          00014400
+          00014410
+          00014420
+          00014430
+          00014440
+          00014450
+          00014460
+          00014470
+          00014480
+          00014490
+          00014500
+          00014510
+          COMMON/ONE/   EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),      00014520
+          PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250)        00014530
+          00014540
+          COMMON/TWO/   EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),      00014550
+          NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),      00014560
+          RSMAX(20)                                         00014570
+          00014580
+          00014590
+          COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),      00014600
+          ITK(500,3),MXELDC(20,4),NSTPRE(20,10),      00014610
+          PERENE(20,4)                                         00014620
+          00014630
+          00014640
+          00014650
+          COMMON/FOUR/  ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,      00014660
+          NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP                  00014670
+          00014680
+          00014690
+          COMMON/FIVE/  HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)  00014700
+          DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
+          00014710
+          00014720
+          00014730
+          COMMON/SIX/   TINT(120),TY(120)                         00014740
+          DOUBLE PRECISION TINT,TY                               00014750
+          00014760
+          COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),      00014770
+          HYEXP(11),NHYMN(20),NPSMN(20),NTHY(20),      00014780
+          NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20)    00014790
+          00014800
+          00014810
+          00014820
+          00014830
+          00014840
+          00014850
+          00014860
+          00014870
+          00014880
+          00014890
+          DIMENSION M(3),B(500)                                 00014900
+          00014910
+          00014920
+          00014930
+          00014940
+          00014950
+          00014960
+          DATA K24/24/
+          II=1
DO 30  I=1,IPMAX
B(II)=PLANT(I,7)
IBK(II,1)=I
IBK(II,2)=I
IBK(II,3)=-1
IF(I .GT. NPEXP) IBK(II,3)=NUF(I,JYR)
II=II+1

```

```

C=PLANT(I,3)-PLANT(I,2)          00014970
IF(C.EQ.0.0) GO TO 30            00014980
B(II)=PLANT(I,8)                 00014990
IBK(II,1)=1                      00015000
IBK(II,2)=2                      00015010
IBK(II,3)=-1                    00015020
IF(I.GT.NPEXP) IBK(II,3)=NUF(I,JYR)
II=II+1                          00015030
30 CONTINUE                       00015040
KMAX=II-1                        00015050
00015060
00015070
00015080
00015090
00015100
00015110
00015120
00015130
00015140
00015150
00015160
00015170
00015180
00015190
00015200
00015210
00015220
00015230
00015240
00015250
00015260
00015270
00015280
00015290
00015300
00015310
00015320
00015330
00015340
00015350
00015360
00015370
00015380
00015390
00015400
00015410
00015420
00015430
00015440
00015450
00015460
00015470
00015480
00015490
00015500
00015510
00015520
00015530
00015540
00015550
00015560
00015570
00015580
00015590
00015600
00015610
00015620
00015630
00015640

ARRANGE "IBK" ARRAY ACCORDING TO FUEL COST

N=KMAX
NE=N-1
DO 40 I=1,NE
N=N-1
DO 50 K=1,N
IF(B(K).LE.B(K+1)) GO TO 50
S=B(K)
B(K)=B(K+1)
B(K+1)=S
DO 60 L=1,3
M(L)=IBK(K,L)
IBK(K,L)=IBK(K+1,L)
60 IBK(K+1,L)=M(L)
50 CONTINUE
40 CONTINUE
WRITE(K24,901) ((IBK(I,J),J=1,3),I=1,420)
WRITE(K24,902) (MAINS(J),J=1,200)
902 FORMAT(200I2)
901 FORMAT(125(I3,I2,I3))
RETURN
END

SUBROUTINE OPCOST(NCUM)
COMMON/ONE/ EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
+ PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250)
COMMON/TWO/ EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
+ NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
+ RSMAX(20)
COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
+ ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
+ PERENE(20,4)
COMMON/FOUR/ ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
+ NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
COMMON/FIVE/ HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
COMMON/SIX/ TINT(120),TY(120)
DOUBLE PRECISION TINT,TY

```

```

C COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),
+           HYEXPN(11),NHYMN(20),NPSMN(20),NTHY(20),
+           NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00015650
C           00015660
C           00015670
C           00015680
C           00015690
C           00015700
C           00015710
C           00015720
C           00015730
C           00015740
C           00015750
C           00015760
C           00015770
C           00015780
C           00015790
C           00015800
C           00015810
C           00015820
C           00015830
C           00015840
C           00015850
C           00015860
C           00015870
C           00015880
C           00015890
C           00015900
C           00015910
C           00015920
C           00015930
C           00015940
C           00015950
C           00015960
C           00015970
C           00015980
C           00015990
C           00016000
C           00016010
C           00016020
C           00016030
C           00016040
C           00016050
C           00016060
C           00016070
C           00016080
C           00016090
C           00016100
C           00016110
C           00016120
C           00016130
C           00016140
C           00016150
C           00016160
C           00016170
C           00016180
C           00016190
C           00016200
C           00016210
C           00016220
C           00016230
C           00016240
C           00016250
C           00016260
C           00016270
C           00016280
C           00016290
C           00016300
C           00016310
C           00016320
C
C DIMENSION ENRT(220,2),PCUM(201,8,1),ENRH(3,2),CFH(3),
*ENY(2,30),ENP(2,30),KLWI(3),KLWZ(3),NORDER(4)
DOUBLE PRECISION SUCUM(8) 00015710
C           00015720
C           00015730
C           00015740
C           00015750
C           00015760
C           00015770
C           00015780
C           00015790
C           00015800
C           00015810
C           00015820
C           00015830
C           00015840
C           00015850
C           00015860
C           00015870
C           00015880
C           00015890
C           00015900
C           00015910
C           00015920
C           00015930
C           00015940
C           00015950
C           00015960
C           00015970
C           00015980
C           00015990
C           00016000
C           00016010
C           00016020
C           00016030
C           00016040
C           00016050
C           00016060
C           00016070
C           00016080
C           00016090
C           00016100
C           00016110
C           00016120
C           00016130
C           00016140
C           00016150
C           00016160
C           00016170
C           00016180
C           00016190
C           00016200
C           00016210
C           00016220
C           00016230
C           00016240
C           00016250
C           00016260
C           00016270
C           00016280
C           00016290
C           00016300
C           00016310
C           00016320
C
C T=8760./NP
C IYR=IYR+JYR-1+1900
C ISTAR=(JYR-1)*4+1
C
C INITIALIZATION
C
C DO 5 J=1,16
C DO 5 I=1,211
C      5 PTCUM(I,J)=0.0
C
C      DO 50 J=1,2
C      DO 50 I=1,220
C      50 ENRT(I,J)=0.0
C      DO 805 I=1,4
C      805 NORDER(I)=0
C
C      COSTOP(JYR)=0.0
C      DTOT=0.0
C      GENTOT=0.0
C      HYTOT=0.0
C
C DETERMINE EFFECTIVE LOADING ORDER
C
C KK=1
C IE=IPMAX+1
C IF(PLANT(IE,2).LE.0.0) GO TO 19
C ITK(KK,1)=IE
C ITK(KK,2)=1
C ITK(KK,3)=1
C KK=KK+1
C
C 19 DO 10 I=1,NPEXP
C     J=NPEXP-I+1
C     IF((NT(J)-NX(J)).EQ.0) GO TO 10
C     IP=IX(J)
C     ITK(KK,1)=IP
C     ITK(KK,2)=1
C     ITK(KK,3)=NT(J)-NX(J)
C     KK=KK+1
C
C 10 CONTINUE
C
C DO 15 K=1,KMAX
C     IF(IBK(K,3).LE.0) GO TO 15
C     DO 16 J=1,3
C     16 ITK(KK,J)=IBK(K,J)
C     KK=KK+1
C
C 15 CONTINUE
C     NMAX=KK-1
C

```

```

DO 2000 N=1,4
IF(PEEK(JYR,N).EQ.PMAX(JYR)) GO TO 2010
2000 CONTINUE
2010 NCRT=N
00016330
00016340
00016350
00016360
00016370
00016380
00016390
00016400
00016410
00016420
00016430
00016440
00016450
00016460
00016470
00016480
00016490
00016500
00016510
00016520
00016530
00016540
00016550
00016560
00016570
00016580
00016590
00016600
00016610
00016620
00016630
00016640
00016650
00016660
00016670
00016680
00016690
00016700
00016710
00016720
00016730
00016740
00016750
00016760
00016770
00016780
00016790
00016800
00016810
00016820
00016830
00016840
00016850
00016860
00016870
00016880
00016890
00016900
00016910
00016920
00016930
00016940
00016950
00016960
00016970
00016980
00016990
00017000

*** SECTION 1 ***

START ENERGY & COST CALCULATION FOR EACH SEASON
LOOP 60

DO 60 NSA=1,NP
N=NSA
00016470
00016480
00016490
00016500
00016510
00016520
00016530
00016540
00016550
00016560
00016570
00016580
00016590
00016600
00016610
00016620
00016630
00016640
00016650
00016660
00016670
00016680
00016690
00016700
00016710
00016720
00016730
00016740
00016750
00016760
00016770
00016780
00016790
00016800
00016810
00016820
00016830
00016840
00016850
00016860
00016870
00016880
00016890
00016900
00016910
00016920
00016930
00016940
00016950
00016960
00016970
00016980
00016990
00017000

INITIALIZATION

ESHORT=0.0
DO 17 K=1,3
17 CFH(K)=0.0
DTOT=DTOT+PERENE(JYR,N)
DO 18 K=1,8
18 SUCUM(K)=0.0D00
DO 55 J=1,2
55 ENRH(I,J)=0.0
00016540
00016550
00016560
00016570
00016580
00016590
00016600
00016610
00016620
00016630
00016640
00016650
00016660
00016670
00016680
00016690
00016700
00016710
00016720
00016730
00016740
00016750
00016760
00016770
00016780
00016790
00016800
00016810
00016820
00016830
00016840
00016850
00016860
00016870
00016880
00016890
00016900
00016910
00016920
00016930
00016940
00016950
00016960
00016970
00016980
00016990
00017000

CUMULANTS OF SCHEDULED THERMAL PLANTS

IFLAG=1
CALL CUCAL(IFLAG,N)
00016690
00016700
00016710
00016720
00016730
00016740
00016750
00016760
00016770
00016780
00016790
00016800
00016810
00016820
00016830
00016840
00016850
00016860
00016870
00016880
00016890
00016900
00016910
00016920
00016930
00016940
00016950
00016960
00016970
00016980
00016990
00017000

CALCULATE CUMULANTS OF HYDRO PLANTS

IFLAG=4
CALL CUCAL(IFLAG,N)
DO 3000 K=1,4
3000 SUCUM(K)=HRCUM(N,K)+PTCUMH(3,K)
IFLAG=3
CALL CUCAL(IFLAG,N)
00016760
00016770
00016780
00016790
00016800
00016810
00016820
00016830
00016840
00016850
00016860
00016870
00016880
00016890
00016900
00016910
00016920
00016930
00016940
00016950
00016960
00016970
00016980
00016990
00017000

STORE & WRITE PTCUM,HRCUM

DO 64 I=1,201
DO 64 J=1,8
64 PCUM(I,J,1)=0.0
IF(N.NE.NCRT) GO TO 2020
DO 56 K=1,4
56 KK=K+8
SUCUM(K)=SUCUM(K)+PTCUMH(1,KK)
IE=IPMAX
IB=NPEXP+1
DO 51 I=IB, IE
C=PLANT(I,3)-PLANT(I,2)
00016870
00016880
00016890
00016900
00016910
00016920
00016930
00016940
00016950
00016960
00016970
00016980
00016990
00017000
00016930
00016940
00016950
00016960
00016970
00016980
00016990
00017000

```

```

JN=NUF(I,JYR)
DO 52 K=1,4
IF(C.LE.0.0) GO TO 53
KK=K+8
SUCUM(K)=SUCUM(K)+DFLOAT(JN)*PTCUM(I,KK)
GO TO 52
53 SUCUM(K)=SUCUM(K)+DFLOAT(JN)*PTCUM(I,K)
52 CONTINUE
51 CONTINUE
DO 54 K=1,4
54 SCHCUM(K,JYR)=SNGL(SUCUM(K))
      00017010
      00017020
      00017030
      00017040
      00017050
      00017060
      00017070
      00017080
      00017090
      00017100
      00017110
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

C
2020 DO 65 K=1,8
PCUM(I,K,1)=SNGL(HRCUM(N,K))
65 CONTINUE
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

CCC
IE=IPMAX-NPEXP+1
DO 61 I=2,IE
IP=I+NPEXP-1
DO 62 K=1,8
IFIK.GT.4) GO TO 63
PCUM(I,K,1)=SNGL(PTCUM(IP,K))
GO TO 62
63 KK=K+4
PCUM(I,K,1)=SNGL(PTCUM(IP,KK))
62 CONTINUE
61 CONTINUE
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

CCCC
CALCULATE PLANT CUMULANTS FOR SINGLE "PSEUDO"-PLANT
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

CCCCC
IFLAG=2
CALL CUCAL(IFLAG,N)
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

CCCCC
INITIALIZE SYSTEM CUMULANTS
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

CCCCC
DO 30 I=1,8
30 SYSCUM(I)=HRCUM(N,I)
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

CCCCC
SUMP=0.0
EPMP=0.0
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

CCCCC
*** SECTION 2 ***
BASE LOAD RUN-OF-RIVER CAPACITY
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

CCCCC
IH=4
DO 20 K=1,2
P=1.0-HYDR(K,5)
Y=HYDR(K,1)*CMULT(N)*(1.0-EMORM(K,N))
Y1=SUMP
SUMP=SUMP+Y
Y2=SUMP
IF(K.EQ.1) GO TO 25
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

CCCCC
CONVOLVE THE FIRST HYDRO
      00017120
      00017130
      00017140
      00017150
      00017160
      00017170
      00017180
      00017190
      00017200
      00017210
      00017220
      00017230
      00017240
      00017250
      00017260
      00017270
      00017280
      00017290
      00017300
      00017310
      00017320
      00017330
      00017340
      00017350
      00017360
      00017370
      00017380
      00017390
      00017400
      00017410
      00017420
      00017430
      00017440
      00017450
      00017460
      00017470
      00017480
      00017490
      00017500
      00017510
      00017520
      00017530
      00017540
      00017550
      00017560
      00017570
      00017580
      00017590
      00017600
      00017610
      00017620
      00017630
      00017640
      00017650
      00017660
      00017670
      00017680

```

```

C DO 24 J=1,NCUM          00017690
C   24 SYSCUM(J)=SYSCUM(J)+PTCUMH(1,J) 00017700
C
C   EVALUATE AREA UNDER EL-CURVE           00017710
C
C   25 CALL AREA(Y1,Y2,P,EH,CF,CFP,IH,N) 00017720
C     ENRH(K,1)=EH                         00017730
C
C   MAX. HYDRO ENERGY AVAILABLE           00017740
C
C   EAVAL(K)=HYDR(K,4)*EMULT(N)*1000.0-EH 00017750
C     C=HYDR(K,2)-HYDR(K,1)*CMULT(N)        00017760
C     IF(C.LE.0.0) GO TO 20                 00017770
C     CFH(K)=EAVAL(K)/(C*P*T)              00017780
C
C   20 CONTINUE                           00017790
C     EPS=HYDK(3,2)*1000.0                  00017800
C
C   CONVOLVE THE SECOND HYDRO            00017810
C
C   DO 27 J=1,NCUM          00017820
C   27 SYSCUM(J)=SYSCUM(J)+PTCUMH(2,J) 00017830
C
C   ORDER OF HYDRO PLANT IN DECREASING CAPACITY FACTOR 00017840
C
C   NUMBER=0                            00017850
C   IF(CFH(1).GE.CFH(2)) GO TO 800      00017860
C   NORDER(1)=2                          00017870
C   IF(CFH(1).LE.0.0) GO TO 802         00017880
C   NORDER(2)=1                          00017890
C   NUMBER=2                            00017900
C   GO TO 801                           00017910
C   800 IF(CFH(1).LE.0.0) GO TO 801      00017920
C   NORDER(1)=1                          00017930
C   IF(CFH(2).LE.0.0) GO TO 802         00017940
C   NORDER(2)=2                          00017950
C   NUMBER=2                            00017960
C   GO TO 801                           00017970
C
C   801 IF(EPS.LE.0.0) GO TO 803         00017980
C   NUMBER=NUMBER+1                     00017990
C   NORDER(NUMBER)=3                   00018000
C
C   802 NUMBER=1                         00018010
C   801 IF(EPS.LE.0.0) GO TO 803         00018020
C   NUMBER=NUMBER+1                     00018030
C   NORDER(NUMBER)=3                   00018040
C
C   803 NUMBER=NUMBER+1                     00018050
C   NORDER(NUMBER)=4                   00018060
C   GO TO 801                           00018070
C
C   800 IF(CFH(1).LE.0.0) GO TO 801      00018080
C   NORDER(1)=1                          00018090
C   IF(CFH(2).LE.0.0) GO TO 802         00018100
C   NORDER(2)=2                          00018110
C   NUMBER=2                            00018120
C   GO TO 801                           00018130
C
C   801 NUMBER=1                         00018140
C   801 IF(EPS.LE.0.0) GO TO 803         00018150
C   NUMBER=NUMBER+1                     00018160
C   NORDER(NUMBER)=3                   00018170
C
C   803 NUMBER=NUMBER+1                     00018180
C   NORDER(NUMBER)=4                   00018190
C   GO TO 801                           00018200
C
C   INITIALIZE INDEX FOR HYDRO LOADING ORDER 00018210
C
C   DO 28 K=1,3                         00018220
C     KLW1(K)=0                          00018230
C   28 KLW2(K)=0                          00018240
C     NOHY=1                            00018250
C     IH=NORDER(NOHY)                   00018260
C     IPUMP=-1                          00018270
C     IF(EPS.LE.0.0) IPUMP=1            00018280
C     KLOW1=0                           00018290
C     KLOW2=0                           00018300
C     SUEN2=0.0                         00018310
C
C   C
C   28 KLW1(K)=0                          00018320
C     KLOW1=0                           00018330
C     SUEN2=0.0                         00018340
C
C   C
C   28 KLW2(K)=0                          00018350
C     SUEN2=0.0                         00018360

```

```

*** SECTION 3 ***
START LOADING THE THERMAL PLANTS
LOOP 70
DO 70 KC=1,NMAX
K=KC
ID1=ITK(K,2)
IP=ITK(K,1)
DO 630 J=1,30
ENP(2,J)=0.0
630 ENY(2,J)=0.0
INDEX FOR EMOR ARRAY
IF( IP.GT.NPEXP) GO TO 71
IMANT=2*IP-ID1+1
GO TO 72
71 IMANT=IP+NPEXP
72 IUNITB=1
IUNITE=ITK(K,3)
PT=1.0-PLANT(IP,5)
IF(ID1.GT.1) GO TO 73
YT=PLANT(IP,2)*(1.0-EMOR(IMANT,N))
GO TO 74
73 YT=(PLANT(IP,3)-PLANT(IP,2))*(1.0-EMOR(IMANT,N))

LOOP FOR IDENTICAL UNITS
LOOP 75
DO 75 ISTAR=IUNITB,IUNITE
I=ISTAR
Y1=SUMP
SUMP=SUMP+YT
Y2=SUMP
IF(I.EQ.1) GO TO 76
IQ=IP
ID2=ID1
GO TO 79
76 IF(K.EQ.1) GO TO 150
KIX=K-1
IQ=ITK(KIX,1)
ID2=ITK(KIX,2)

PLANT CONVOLUTION
DO 90 J=1,NCUM
J1=J
IF(ID2.GT.1) J1=J+8
90 SYSCUM(J)=SYSCUM(J)+PTCUM(IQ,J1)
IF(ID1.LE.1) GO TO 150

PLANT DECONVOLUTION

```

00018370
00018380
00018390
00018400
00018410
00018420
00018430
00018440
00018450
00018460
00018470
00018480
00018490
00018500
00018510
00018520
00018530
00018540
00018550
00018560
00018570
00018580
00018590
00018600
00018610
00018620
00018630
00018640
00018650
00018660
00018670
00018680
00018690
00018700
00018710
00018720
00018730
00018740
00018750
00018760
00018770
00018780
00018790
00018800
00018810
00018820
00018830
00018840
00018850
00018860
00018870
00018880
00018890
00018900
00018910
00018920
00018930
00018940
00018950
00018960
00018970
00018980
00018990
00019000
00019010
00019020
00019030
00019040

```

C          DO 100 J=1,NCUM      00019050
C          100 SYSCUM(J)=SYSCUM(J)-PTCUM(IP,J) 00019060
C          150 CONTINUE      00019070
C          EVALUATE AREA UNDER THE EL-CURVE 00019080
C          152 CALL AREA(Y1,Y2,PT,ET,CF,CFP,IH,N) 00019090
C          ENY(2,I)=ET      00019100
C          ANY MORE HYDRO LEFT      00019110
C          IF(IH.GE.4) GO TO 75      00019120
C          CHECK HYDRO GENERATION FEASIBILITY 00019130
C          IF(CF.LE.CFH(IH)) GO TO 200 00019140
C          CHECK PUMPING DUTY ASSIGNMENT 00019150
C          IF(IPUMP.GT.0.OR.CFP.GE.1.0) GO TO 75 00019160
C          PUMPING ASSIGNMENT      00019170
C          PP=1.0-HYDR(3,5)      00019180
C          YP1=Y1-HYDR(3,1)*(1.0-EMORH(3,N)) 00019190
C          YP2=Y2-HYDR(3,1)*(1.0-EMORH(3,N)) 00019200
C          CALL AREA(YP1,YP2,PT,EP,CF,CFP,IH,N) 00019210
C          ENP(2,I)=PP*(EP-ET) 00019220
C          EPMP=EPMP+ENP(2,I)*HYDR(3,3) 00019230
C          IF(EPMP.LT.EPS) GO TO 570 00019240
C          STOP PUMPING      00019250
C          IPUMP=1      00019260
C          ENP(2,I)=ENP(2,I)-(EPMP-EPS)/HYDR(3,3) 00019270
C          EPMP=EPS      00019280
C          EAVAL(3)=EPS*HYDR(3,4) 00019290
C          CFH(3)=EAVAL(3)/(T*HYDR(3,1)*(1.0-HYDR(3,5))) 00019300
C          GO TO 75      00019310
C          570 EAVAL(3)=EPMP*HYDR(3,4) 00019320
C          CFH(3)=EAVAL(3)/(T*HYDR(3,1)*(1.0-HYDR(3,5))) 00019330
C          GO TO 75      00019340
C          HYDRO GENERATION      00019350
C          200 CONTINUE      00019360
C          ICNTL=-1      00019370
C

```

```

C   REPLACE THE THERMAL BY HYDRO          00019730
C   SUMP=SUMP-YT                          00019740
C   ENY(2,I)=0.0                           00019750
C   ENP(2,I)=0.0                           00019760
C   IF(ID1.LE.1) GO TO 215                00019770
C
C   RETRIVE EL-CURVE INFO                 00019780
C
C   DO 210 J=1,NCUM                      00019790
C   210 SYSCUM(J)=SYSCUM(J)+PTCUM(IP,J)  00019800
C   215 CONTINUE                           00019810
C
C   DECONVE BASE LOADED HYDRO            00019820
C
C   216 DO 220 J=1,NCUM                  00019830
C   220 SYSCUM(J)=SYSCUM(J)-PTCUMH(IH,J) 00019840
C
C   PH=1.0-HYDR(IH,5)                    00019850
C   YH=(HYDR(IH,2)-HYDR(IH,1)*CMULT(N))*(1.0-EMORH(IH,N)) 00019860
C   IF(IH.EQ.3) YH=HYDR(IH,1)*(1.0-EMORH(IH,N))           00019870
C   Y1=SUMP                                00019880
C   SUMP=SUMP+YH                           00019890
C   Y2=SUMP                                00019900
C   CALL AREA(Y1,Y2,PH,EH,CF,CFP,IH,N)    00019910
C   ENRH(IH,2)=EH                          00019920
C   IF(ICNTL.GT.0) GO TO 240              00019930
C   KLOW1=K                                00019940
C   KLOW2=I-1                             00019950
C   KLW1(IH)=K                            00019960
C   KLW2(IH)=I-1                           00019970
C
C   CONVOLVE THIS HYDRO                  00019980
C
C   DO 230 J=1,NCUM                      00019990
C   J1=J+8                                 00020000
C   230 SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J1) 00020010
C
C   NEXT HYDRO                            00020020
C
C   IF(IH.EQ.3) GO TO 620                00020030
C
C   NOHY=NOHY+1                           00020040
C   IH=NORDER(NOHY)                      00020050
C   ICNTL=1                               00020060
C   GO TO 216                            00020070
C   240 IF(EH.GT.EAVAL(IH)) GO TO 250    00020080
C   ENRH(IH,2)=EH                         00020090
C   KLOW1=K                                00020300
C   KLOW2=I-1                             00020310
C   KLW1(IH)=K                            00020320
C   KLW2(IH)=I-1                           00020330
C   GO TO 300                            00020340
C   250 CONTINUE                           00020350
C
C   REPLACE THIS HYDRO BY THERMAL        00020360
C                                         00020370
C                                         00020380
C                                         00020390
C                                         00020400

```

```

C      SUMP=SUMP-YH          00020410
      DO 270 J=1,NCUM        00020420
270   SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J) 00020430
      NOHY=NOHY-1           00020440
      IH=NORDER(NOHY)       00020450
C      620 IBLK=-1          00020460
      CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP, 00020470
      *NORDER)               00020480
      IPUMP=1                00020490
      GO TO 455             00020500
C      300 CONTINUE          00020510
C      FORM CLUSTER AND TEST SWAP LOADING POSITION 00020520
C      NPAST=NOHY-1          00020530
      IHPT=NORDER(NPAST)     00020540
      IBLK=1                 00020550
      IF(KLOW2.LE.0) GO TO 301 00020560
      KIX=KLOW1              00020570
      ISWP=2                 00020580
      ITRM=KLOW2              00020590
      GO TO 302              00020600
301   KIX=KLOW1-1          00020610
      ISWP=1                 00020620
      ITRM=ITK(KIX,3)         00020630
302   III=ITK(KIX,1)        00020640
      II2=ITK(KIX,2)         00020650
      DO 310 J=1,NCUM        00020660
      J1=J                   00020670
      IF(II2.GT.1) J1=J+8    00020680
310   SYSCUM(J)=SYSCUM(J)-PTCUM(III,J1) 00020690
      DO 320 J=1,NCUM        00020700
      J1=J+8                00020710
320   SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J1) 00020720
      P=1.0-PLANT(III,5)     00020730
      IF(III.GT.NPEXP) GO TO 321 00020740
      IMANT=2*III-II2+1      00020750
      GO TO 322              00020760
321   IMANT=III+NPEXP      00020770
322   R=1.0-EMOR(IMANT,N)  00020780
      Y=PLANT(III,2)*R      00020790
      IF(II2.GT.1) Y=(PLANT(III,3)-PLANT(III,2))*R 00020800
      Y2=SUMP                00020810
      Y1=SUMP-Y              00020820
      CALL AREA(Y1,Y2,P,ET,CF,CFP,IH,N) 00020830
      SUEN1=ENY(ISWP,ITRM)+ENRH(IH,2)+ENRH(IHPT,2) 00020840
      SUEN2=SUEN1-ET          00020850
      ETOT=EAVAL(IH)+EAVAL(IHPT) 00020860
      IF(SUEN2.GT.ETOT) GO TO 400 00020870
C      ACCEPT SWAP TEST      00020880
C      ENY(ISWP,ITRM)=ET      00020890
C      KLOW1=K                00020900
      KLOW2=I-2               00020910
      KLW1(IH)=K              00020920
      KLW2(IH)=I-2            00020930
      KLW1(IHPT)=K            00020940
C      143

```

```

C KLW2(IHPT)=I-2 00021090
C COST=CST1*ENP(ISWP,ITRM)
C EPMP=EPMP-ENP(ISWP,ITRM) 00021100
C ENP(ISWP,ITRM)=0.0 00021110
C CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP,
+NRDER) 00021120
C IPUMP=1 00021130
C GO TO 450 00021140
C 400 CONTINUE 00021150
C REJECT SWAP TEST 00021160
C SUEN2=ENRH(IH,2)+ENRH(IHPT,2) 00021170
C CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP,
+NRDER) 00021180
C IPUMP=1 00021190
C 450 CONTINUE 00021200
C LOAD THE CURRENT OFF-LOADED THERMAL PLANT 00021210
C DO 460 J=1,NCUM 00021220
J1=J 00021230
IF(II2.GT.1) J1=J+8 00021240
460 SYSCUM(J)=SYSCUM(J)+PTCUM(III,J1) 00021250
C 455 NOHY=NOHY+1 00021260
IH=NRDER(NOHY) 00021270
C Y1=SUMP 00021280
SUMP=SUMP+YT 00021290
Y2=SUMP 00021300
IF(ID1.NE.2) GO TO 470 00021310
DO 468 J=1,NCUM 00021320
468 SYSCUM(J)=SYSCUM(J)-PTCUM(IP,J) 00021330
470 CALL AREAL(Y1,Y2,PT,ET,CF,CFP,IH,N) 00021340
ENY(2,I)=ET 00021350
C END OF IDENTICAL UNITS 00021360
C 75 CONTINUE 00021370
IF(K.LE.1) GO TO 80 00021380
KIX=K-1 00021390
L1=ITK(KIX,1) 00021400
L2=ITK(KIX,2) 00021410
L3=ITK(KIX,3) 00021420
DO 77 L=1,L3 00021430
GENTOT=GENTOT+ENY(1,L) 00021440
77 ENRT(L1,L2)=ENRT(L1,L2)+ENY(1,L)+ENP(1,L) 00021450
80 DO 78 I=IUNITB,IUNITE 00021460
ENP(1,I)=ENP(2,I) 00021470
78 ENY(1,I)=ENY(2,I) 00021480
C IF(K.LT.NMAX) GO TO 70 00021490
DO 81 I=IUNITB,IUNITE 00021500
GENTOT=GENTOT+ENY(2,I) 00021510
81 ENRT(IP,IDL)=ENRT(IP,IDL)+ENY(2,I)+ENP(2,I) 00021520
00021530
00021540
00021550
00021560
00021570
00021580
00021590
00021600
00021610
00021620
00021630
00021640
00021650
00021660
00021670
00021680
00021690
00021700
00021710
00021720
00021730
00021740
00021750
00021760

```

```

C
      END OF SECTION 3
C
      70 CONTINUE
C
      IF(IH.GE.4) GO TO 66
      KLW1(3)=NMAX
      IP=ITK(NMAX,1)
      ID=ITK(NMAX,2)
      DO 600 J=1,NCUM
      J1=J
      IF(ID.GT.1) J1=J+8
      600 SYSCUM(J)=SYSCUM(J)+PTCUM(IP,J1)
      YH=HYDR(3,1)*(1.0-EMORH(3,N))
      PH=1.0-HYDR(3,5)
      Y1=SUMP
      Y2=SUMP+YH
      CALL AREA(Y1,Y2,PH,EH,CF,CFP,IH,N)
      IF(EH.GT.EAVAL(3)) GO TO 610
      ENRH(3,2)=EAVAL(3)
      ENRT(IP,ID)=ENRT(IP,ID)-(EAVAL(3)-EH)
      GO TO 66
      610 ENRH(3,2)=EAVAL(3)
      ESHORT=EH-EAVAL(3)
C
      66 DO 67 I=1,2
      DO 67 J=1,2
      67 HYTOT=HYTOT+ENRH(I,J)
      GENTOT=GENTOT+ENRH(3,2)
      WRITE(17,905)
      C 905 FORMAT(//, " ++++++ SEASONAL HYDRO OPERATIONAL SUMMARY ++++++ //")
      C 906 FORMAT(/3X,"PLANT",2X,"LOAD",2X,"BASE MAX.",5X,"BASE ENERGY",5X,
      + "PEAK ENERGY")
      C 910 FORMAT(4X,"NO POS.",2X,"CAP. CAP.",9X,"(MWH)",11X,"(MWH")/)
      350 CONTINUE
      DO 351 I=1,3
      FIX=HYDR(I,7)
      VAB=HYDR(I,8)
      FAC=(1.0/1.2)**(FLOAT(JYR)+0.5)
      DO 352 J=1,2
      IF(I.GE.3) GO TO 353
      C=HYDR(I,1)*CMULT(N)
      IF(J.GT.1) C=HYDR(I,2)-C
      GO TO 354
      353 IF(J.LE.1) GO TO 352
      C=HYDR(3,1)
      354 COST=(ENRH(I,J)*VAB+C*FIX)*FAC/1.0E+06
      COSTOP(JYR)=COSTOP(JYR)+COST
      352 CONTINUE
      351 CONTINUE
C
      END OF SECTION 2
C
      60 CONTINUE
C
      *** SECTION 4 ***

```

```

C COST CALCULATION
C
C   IE=IPMAX+1          00022450
C   DO 700 I=1,IE        00022460
C   IF(I.GT.NPEXP) GO TO 711 00022470
C   J=NPEXP-1+1          00022480
C   MM=NT(J)-NX(J)      00022490
C   IF(MM.LE.0) GO TO 700 00022500
C   711 FIX=PLANT(I,10) 00022510
C   VAB=PLANT(I,11)      00022520
C   DO 710 J=1,2         00022530
C   IF(IJ.GT.1) GO TO 715 00022540
C   CFL=PLANT(I,7)       00022550
C   C=PLANT(I,2)          00022560
C   GO TO 720            00022570
C   715 CFL=PLANT(I,8)   00022580
C   C=PLANT(I,3)-PLANT(I,2) 00022590
C   720 FAC=(1.0+PLANT(I,14))/1.2**FLOAT(JYR)+0.5 00022600
C   COSTOP(JYR)=COSTOP(JYR)+(ENRT(I,J)*(CFL+VAB)+C*FIX)*FAC/1.0E+06 00022610
C   710 CONTINUE          00022620
C   700 CONTINUE          00022630
C
C REPORT RESULT
C
C   WRITE(17,902)          00022640
C   902 FORMAT(//, '++++++ ANNUAL THERMAL OPERATIONAL SUMMARY +++++//') 00022650
C   C WRITE(17,903)          00022660
C   903 FORMAT(/3X, 'PLANT NO', 3X, 'BASE CAP', 3X, 'MAX. CAP', 3X,
C   +' BASE ENERGY(MWH)', 3X, 'PEAK ENERGY(MWH)',/) 00022670
C   1230 FORMAT(//, '++++++ ANNUAL SYSTEM OPERATIONAL SUMMARY +++//') 00022680
C   GENTOT=GENTOT+HYTOT 00022690
C   C WRITE(17,1240) DTOT, GENTOT 00022700
C   1240 FORMAT(/2X, 'ENERGY DEMAND = ', E14.7, ' MWH//',
C   +'2X, 'TOTAL GENERATION = ', E14.7, ' MWH//') 00022710
C   C WRITE(17,1990) COSTOP(JYR) 00022720
C   1990 FORMAT(/1H, ' LOWER BOUND OF OP. COST = ', E14.7,
C   +' MILLION DOLLARS//') 00022730
C
C   DO 810 J=1,14          00022740
C   DO 810 I=1,NPEXP        00022750
C   810 PLANT(I,J)=PLACA(I,J) 00022760
C   RETURN                  00022770
C   END                      00022780
C
C   SUBROUTINE TRIM(ENY,ENP,ENRH,KLOW1,KLOW2, SUEN2,IBLK,NOHY,CFH,
C   +IPUMP,NORDER)          00022790
C
C   COMMON/ONE/ EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
C   + PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250) 00022800
C
C   COMMON/TWO/ EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
C   + NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
C   + RSMAX(20) 00022810
C
C   COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
C   + ITK(500,3),MXELDC(20,4),NSTPRE(20,10), 00022820
C

```

```

+ PERENE(20,4) 00023130
+ COMMON/FOUR/ ID, IPMAX, IPT, IYRB, JYR, KMAX, MAXI, MX, NP, NPEXP, 00023140
+ NTYR, PPMAX, PPMIN, PMAX(20), EPMP 00023150
+ COMMON/FIVE/ HRCUM(4,8), PTCUM(211,16), PTCUMH(3,16), SYSCUM(8) 00023160
+ DOUBLE PRECISION HRCUM, PTCUM, PTCUMH, SYSCUM 00023170
+ COMMON/SIX/ TINT(120), TY(120) 00023180
+ DOUBLE PRECISION TINT, TY 00023190
+ COMMON/SEVEN/ CMULT(4), EAVAL(3), EMULT(4), EXPLCU(4,8), 00023200
+ HYEXP(11), NHYMN(20), NPSMN(20), NTHY(20), 00023210
+ NTPS(20), NXHY(20), NXPS(20), PSEXP(11), SCHCUM(4,20) 00023220
+ DIMENSION CFH(3), ENY(2,30), ENP(2,30), ENRH(3,2), NORDER(4) 00023230
+ IF(KLOW2.LE.0) GO TO 10 00023240
10 ISWP=2 00023250
ITRM=KLOW2 00023260
L=KLOW1 00023270
GO TO 15 00023280
10 L=KLOW1-1 00023290
ISWP=1 00023300
ITRM=ITK(L,3)+KLOW2 00023310
15 IP=ITK(L,1) 00023320
ID=ITK(L,2) 00023330
IF(IBLK.GT.0) GO TO 30 00023340
+ TRIM HYDRO BLOCK 00023350
+ IH=NORDER(NOHY) 00023360
+ DIFF=EAVAL(IH)-ENRH(IH,2) 00023370
+ ENRH(IH,2)=EAVAL(IH) 00023380
+ IF(NOHY.LE.1) GO TO 40 00023390
+ ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF 00023400
+ GO TO 60 00023410
+ TRIM HYDRO CLUSTER 00023420
30 IH=NORDER(NOHY) 00023430
NPAST=NOHY-1 00023440
IHPT=NORDER(NPAST) 00023450
DIFF=EAVAL(IH)+EAVAL(IHPT)-SUEN2 00023460
ENRH(IH,2)=EAVAL(IH) 00023470
ENRH(IHPT,2)=EAVAL(IHPT) 00023480
IF(NPAST.LE.1) GO TO 40 00023490
ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF 00023500
GO TO 60 00023510
+ PUMPING ENERGY ADJUSTMENT 00023520
40 IF(IPUMP.GT.0) GO TO 50 00023530
EDF=ENP(ISWP,ITRM)*(DIFF/ENY(ISWP,ITRM)) 00023540
ENP(ISWP,ITRM)=ENP(ISWP,ITRM)-EDF 00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

```

```

ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF          00023810
EPMP=EPMP-EDF*HYDR(3,3)                      00023820
EAVAL(3)=EPMP*HYDR(3,4)                      00023830
P=1.0-HYDR(3,5)                            00023840
CP=HYDR(3,1)                                00023850
IF(CP.LE.0.0) GO TO 60                      00023860
CFH(3)=EAVAL(3)/(2190.0*P*CP)              00023870
GO TO 60                                     00023880
50 ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF      00023890
60 RETURN                                     00023900
END                                         00023910
00023920
00023930
00023940
00023950
00023960
00023970
00023980
00023990
00024000
00024010
00024020
00024030
00024040
00024050
00024060
00024070
00024080
00024090
00024100
00024110
00024120
00024130
00024140
00024150
00024160
00024170
00024180
00024190
00024200
00024210
00024220
00024230
00024240
00024250
00024260
00024270
00024280
00024290
00024300
00024310
00024320
00024330
00024340
00024350
00024360
00024370
00024380
00024390
00024400
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480

SUBROUTINE AREA(IY1,Y2,P,E,CF,CFP,IH,N)

COMMON/ONE/   EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
+             PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250) 00024000
00024010
00024020
00024030
00024040
00024050
00024060
00024070
00024080
00024090
00024100
00024110
00024120
00024130
00024140
00024150
00024160
00024170
00024180
00024190
00024200
00024210
00024220
00024230
00024240
00024250
00024260
00024270
00024280
00024290
00024300
00024310
00024320
00024330
00024340
00024350
00024360
00024370
00024380
00024390
00024400
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480

COMMON/TWO/   EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
+             NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
+             RSMAX(20) 00024000
00024010
00024020
00024030
00024040
00024050
00024060
00024070
00024080
00024090
00024100
00024110
00024120
00024130
00024140
00024150
00024160
00024170
00024180
00024190
00024200
00024210
00024220
00024230
00024240
00024250
00024260
00024270
00024280
00024290
00024300
00024310
00024320
00024330
00024340
00024350
00024360
00024370
00024380
00024390
00024400
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480

COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
+               ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
+               PERENE(20,4) 00024000
00024010
00024020
00024030
00024040
00024050
00024060
00024070
00024080
00024090
00024100
00024110
00024120
00024130
00024140
00024150
00024160
00024170
00024180
00024190
00024200
00024210
00024220
00024230
00024240
00024250
00024260
00024270
00024280
00024290
00024300
00024310
00024320
00024330
00024340
00024350
00024360
00024370
00024380
00024390
00024400
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480

COMMON/FOUR/   ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
+               NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP 00024000
00024010
00024020
00024030
00024040
00024050
00024060
00024070
00024080
00024090
00024100
00024110
00024120
00024130
00024140
00024150
00024160
00024170
00024180
00024190
00024200
00024210
00024220
00024230
00024240
00024250
00024260
00024270
00024280
00024290
00024300
00024310
00024320
00024330
00024340
00024350
00024360
00024370
00024380
00024390
00024400
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480

COMMON/FIVE/   HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM 00024000
00024010
00024020
00024030
00024040
00024050
00024060
00024070
00024080
00024090
00024100
00024110
00024120
00024130
00024140
00024150
00024160
00024170
00024180
00024190
00024200
00024210
00024220
00024230
00024240
00024250
00024260
00024270
00024280
00024290
00024300
00024310
00024320
00024330
00024340
00024350
00024360
00024370
00024380
00024390
00024400
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480

COMMON/SIX/    TINT(120),TY(120)
DOUBLE PRECISION TINT,TY 00024000
00024010
00024020
00024030
00024040
00024050
00024060
00024070
00024080
00024090
00024100
00024110
00024120
00024130
00024140
00024150
00024160
00024170
00024180
00024190
00024200
00024210
00024220
00024230
00024240
00024250
00024260
00024270
00024280
00024290
00024300
00024310
00024320
00024330
00024340
00024350
00024360
00024370
00024380
00024390
00024400
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480

COMMON/SEVEN/  CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),
+               HYEXP(11),NHYMN(20),NPSMN(20),NTHY(20),
+               NTPS(20),NXHY(20),NXP(20),PSEXP(11),SCHCUM(4,20) 00024000
00024010
00024020
00024030
00024040
00024050
00024060
00024070
00024080
00024090
00024100
00024110
00024120
00024130
00024140
00024150
00024160
00024170
00024180
00024190
00024200
00024210
00024220
00024230
00024240
00024250
00024260
00024270
00024280
00024290
00024300
00024310
00024320
00024330
00024340
00024350
00024360
00024370
00024380
00024390
00024400
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480

10 COEFF(J)=SYSCUM(J)/SIGS
ZL1=(DBLE(Y1)-COEFF(1))/SIGMA
ZL2=(DBLE(Y2)-COEFF(1))/SIGMA
CALL VALUE(ZL1,COEFF,NCUM,ZV1)
CALL VALUE(ZL2,COEFF,NCUM,ZV2)
C1=(ZV1+ZV2)/2.0
CFP=C1
X1=Y2-Y1
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480

```

```

E=X1*2190.0*P*C1          00024490
IF(IH.GE.4) GO TO 60        00024500
X2=HYDR(IH,2)-HYDR(IH,1)*CMULT(N)
IF(IH.EQ.3) X2=HYDR(3,1)
IF(IH.NE.3) GO TO 20
T1=ZV1
T2=ZV2
GO TO 50
20 Q=HYDR(IH,5)
X3=HYDR(IH,1)
R=(X1+X3)/X3
IF(X3.GT.X1) GO TO 30
S=X1/(X1-X3)
GO TO 40
30 S=1.0-(X3/(X3-X1))
D1=1.0-Q-R*Q/(1.0-R)
D2=1.0-Q+Q/S
D3=(1.0-S)*Q*Q
D4=(1.0-R)*S
DE=D1*D2+D3/D4
T1=(ZV1*D2-ZV2*Q/(1.0-R))/DE
T2=(ZV2*D1+ZV1*Q*(1.0-S)/S)/DE
50 DFAC=T1-(T1-T2)*(X2/X1)
CF=(T1+DFAC)/2.0
GO TO 70
60 CF=C1
70 CONTINUE
RETURN
END



---


SUBROUTINE CUCAL(IFLAG,N)


---


COMMON/ONE/   EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
+ PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250)      00024800
00024810
00024820
00024830
00024840
00024850
00024860
00024870
00024880
00024890
00024900
00024910
00024920
00024930
00024940
00024950
00024960
00024970
00024980
00024990
00025000
00025010
00025020
00025030
00025040
00025050
00025060
00025070
00025080
00025090
00025100
00025110
00025120
00025130
00025140
00025150
00025160
COMMON/TWO/   EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
+ NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
+ RSMAX(20)      00024900
00024910
00024920
00024930
00024940
00024950
00024960
00024970
00024980
00024990
00025000
00025010
00025020
00025030
00025040
00025050
00025060
00025070
00025080
00025090
00025100
00025110
00025120
00025130
00025140
00025150
00025160
COMMON/THREE/  AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
+ ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
+ PERENE(20,4)      00024950
00024960
00024970
00024980
00024990
00025000
00025010
00025020
00025030
00025040
00025050
00025060
00025070
00025080
00025090
00025100
00025110
00025120
00025130
00025140
00025150
00025160
COMMON/FOUR/   ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
+ NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP      00025000
00025010
00025020
00025030
00025040
00025050
00025060
00025070
00025080
00025090
00025100
00025110
00025120
00025130
00025140
00025150
00025160
COMMON/FIVE/   HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM      00025040
00025050
00025060
00025070
00025080
00025090
00025100
00025110
00025120
00025130
00025140
00025150
00025160
COMMON/SIX/    TINT(120),TY(120)
DOUBLE PRECISION TINT,TY      00025080
00025090
00025100
00025110
00025120
00025130
00025140
00025150
00025160
COMMON/SEVEN/   CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),
+ HYEXP(11),NHYMN(20),NPSMN(20),NTHY(20),
+ NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00025110
00025120
00025130
00025140
00025150
00025160

```

```

C      DOUBLE PRECISION R(8),C,Q
C      GO TO (100,200,600,700),IFLAG
C
C CUMULANTS OF THERMAL PLANTS
C
100 LB=1
LE=IPMAX
GO TO 500
200 LB=1
LE=1
C
500 CONTINUE
DO 510 L=LB,LE
IF(L.GT.NPEXP) GO TO 515
IF(IFLAG.EQ.2) GO TO 517
IF((NT(L)-NX(L)).LE.0) GO TO 510
I=IX(L)
GO TO 516
517 I=IPMAX+1
GO TO 516
515 I=L
516 Q=PLANT(I,5)
DO 520 J=1,2
IF(L.GT.NPEXP) GO TO 530
IFI(IFLAG.EQ.2) GO TO 530
C=PLANT(I,2)*(1.0-EMOR(2*I,N))
IF(J.GT.1) C=PLANT(I,3)*(1.-EMOR(2*I,N))
GO TO 540
530 C=PLANT(I,2)*(1.-EMOR(NPEXP+I,N))
IFI(J.GT.1) C=PLANT(I,3)*(1.-EMOR(NPEXP+I,N))
540 CONTINUE
R(1)=C*Q
R(2)=C*C*Q*(1.0-Q)
R(3)=C*C*C*Q*(1.0-(3.0-2.0*Q)*Q)
R(4)=C*C*C*C*Q*(1.0-(7.0-(12.0-6.0*Q)*Q)*Q)
R(5)=C*C*C*C*C*Q*(1.0+(-15.0+(50.0+(-60.0+24.0*Q)*Q)*Q)*Q)
R(6)=C*C*C*C*C*C*Q*(1.0+(-31.0+(180.0+(-390.0+(*360.0-120.0*Q)*Q)*Q)*Q)*Q)
R(7)=C*C*C*C*C*C*C*Q*(1.0+(-63.0+(602.0+(-2100.0+(3360.0+(*(-2520.0+720.0*Q)*Q)*Q)*Q)*Q)*Q)
R(8)=C*C*C*C*C*C*C*Q*(1.0+(-127.0+(1932.0+(-10206.0+(25200.0+(*(-31920.0+(20160.0-5040.0*Q)*Q)*Q)*Q)*Q)*Q)*Q)
DO 550 K=1,8
KK=K
IFI(J.GT.1) KK=K+8
550 PTCUM(I,KK)=R(K)
520 CONTINUE
510 CONTINUE
C
GO TO 1000
C
C CUMULANTS OF HYDRO PLANTS
C
600 CONTINUE
DO 610 I=1,3
Q=HYDR(I,5)
DO 620 J=1,2
IFI(J.GT.1) GO TO 630
C=HYDR(I,1)*CMULT(N)*(1.0-EMORM(I,N))

```

```

IF(I.GE.3) C=0.0D00          00025850
GO TO 640                      00025860
630 C=HYDR(1,2)*(1.0-EMORH(I,N)) 00025870
IF(I.GE.3) C=HYDR(3,1)*(1.0-EMORH(3,N)) 00025880
640 CONTINUE                     00025890
R(1)=C*Q                        00025900
R(2)=C*C*Q*(1.0-Q)             00025910
R(3)=C*C*C*Q*(1.0-(3.0-2.0*Q)*Q) 00025920
R(4)=C*C*C*C*Q*(1.0-(7.0-(12.0-6.0*Q)*Q)*Q) 00025930
R(5)=C*C*C*C*C*Q*(1.0+(-15.0+(50.0+(-60.0+24.0*Q)*Q)*Q)*Q) 00025940
R(6)=C*C*C*C*C*Q*(1.0+(-31.0+(180.0+(-390.0+ 00025950
*(360.0-120.0*Q)*Q)*Q)*Q)*Q) 00025960
R(7)=C*C*C*C*C*C*Q*(1.0+(-63.0+(602.0+(-2100.0+(3360.0+ 00025970
*-2520.0+720.0*Q)*Q)*Q)*Q)*Q) 00025980
R(8)=C*C*C*C*C*C*C*Q*(1.0+(-127.0+(1932.0+(-10206.0+(25200.0+ 00025990
*(-31920.0+(20160.0-5040.0*Q)*Q)*Q)*Q)*Q)*Q) 00026000
DO 650 K=1,8                   00026010
KK=K                          00026020
IF(J.GT.1) KK=K+8              00026030
650 PTCUMH(I,KK)=R(K)         00026040
620 CONTINUE                     00026050
610 CONTINUE                     00026060
C
C GO TO 1000                    00026070
C
C 700 CONTINUE                   00026080
Q=PSEXP(5)                     00026090
C=PSEXP(1)*(1.0-EMORH(3,N))   00026100
PTCUMH(3,1)=C*Q               00026110
PTCUMH(3,2)=C*C*Q*(1.0-Q)    00026120
PTCUMH(3,3)=C*C*C*Q*(1.0-(3.0-2.0*Q)*Q) 00026130
PTCUMH(3,4)=C*C*C*C*Q*(1.0-(7.0-(12.0-6.0*Q)*Q)*Q) 00026140
C
1000 RETURN                     00026150
END                           00026160
C
C
C SUBROUTINE VALUE(Z,C,NCUM,FRE) 00026170
C
C
C COMMON/ONE/      EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20), 00026180
C + PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250) 00026190
C
C COMMON/TWO/      EMOR(220,4),EMORH(3,4),IX(20),MAINS(200), 00026200
C + NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20), 00026210
C + RSMAX(20)       00026220
C
C COMMON/THREE/    AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3), 00026230
C + ITK(500,3),MXELDC(20,4),NSTPRE(20,10), 00026240
C + PERENE(20,4)    00026250
C
C COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP, 00026260
C + NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP 00026270
C
C COMMON/FIVE/    HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8) 00026280
C DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM 00026290
C
C

```

```

COMMON/SIX/ TINT(120),TY(120)          00026530
      DOUBLE PRECISION TINT,TY          00026540
C
COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),
+           HYEXPN(11),NHYMN(20),NPSMN(20),NTHY(20),
+           NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00026550
C
      DOUBLE PRECISION Z,ZN(8),FRE,FAE,F3,F4,F5,F6,F7,F8,F9,E,
*ZINTEG,ZSQ,C(8)                      00026560
      IF ( Z.GT.5.90D00 ) GO TO 40        00026570
      ZSQ = Z*Z/2.0D00                     00026580
      E = DEXP(-ZSQ)/2.506628275D00     00026590
      ZN(1) = -Z * E                      00026600
      ZN(2) = E * (Z*Z - 1.0D00)          00026610
      ZN(3) = -Z*E*(Z*Z - 3.0D00)         00026620
      ZN(4) = E*(Z*Z*(Z*Z - 6.0D00) + 3.0D00) 00026630
      ZN(5) = -Z*E*(Z*Z*(Z*Z - 10.0D00) + 15.0D00) 00026640
      ZN(6) = E*(Z*Z*(Z*Z*(Z*Z - 15.0D00) + 45.0D00) - 15.0D00) 00026650
      ZN(7) = -Z*E*(Z*Z*(Z*Z*(Z*Z-21.0D00)+105.0D00)-105.0D00) 00026660
      ZN(8) = E*(Z*Z*(Z*Z*(Z*Z*(Z*Z-28.0D00)+210.0D00)-420.0D00)+ 00026670
      +105.0D00)                           00026680
      F3 = 6.0D00                         00026690
      F4 = 24.0D00                        00026700
      F5 = 120.0D00                       00026710
      F6 = 720.0D00                       00026720
      F7 = 5040.0D00                      00026730
      F8 = 40320.0D00                     00026740
      F9 = 362880.0D00                    00026750
      NUMCU=NCUM-5                      00026760
C
      CALL TALOOK(Z,ZINTEG)             00026770
C
      FRE = 1.0D00 - ZINTEG            00026780
      FRE = FRE + C(3)*ZN(2)/F3       00026790
      +           - C(4)*ZN(3)/F4       00026800
      +           - C(3)*C(3)*ZN(5)*10.0D00/F6 00026810
C
      IF(NUMCU) 5,10,15                00026820
C
      FOUR CUMULANT EXPANSION        00026830
C
      5 IF(FRE.LT.0.0D00) FRE=0.0D00   00026840
      IF(FRE.GT.1.0D00) FRE=1.0D00    00026850
      RETURN                            00026860
C
      FIVE CUMULANT EXPANSION       00026870
C
      10 FRE = FRE + C(5)*ZN(4)/F5    00026880
      +           + C(3)*C(4)*ZN(6)*35.0D00/F7 00026890
      +           + C(3)*C(3)*ZN(8)*280.0D00/F9 00026900
      IF(FRE.LT.0.0D00) FRE=0.0D00    00026910
      IF(FRE.GT.1.0D00) FRE=1.0D00    00026920
      RETURN                            00026930
C
      EIGHT CUMULANT EXPANSION      00026940
C
      15 FRE = FRE + C(5)*ZN(4)/F5    00026950
      +           - C(6)*ZN(5)/F6       00026960
      +           + (C(7)+10.0D00*C(3)*C(4))*ZN(6)/F7 00026970
      +           - (C(8)+56.0D00*C(3)*C(5)+35.0D00*C(4)*C(4))*ZN(7)/F8 00026980
      IF(FRE.LT.0.0D00) FRE=0.0D00    00026990
      IF(FRE.GT.1.0D00) FRE=1.0D00    00027000
      RETURN                            00027010
C

```

```

40 FRE = 0.0D00          00027210
RETURN                      00027220
END                         00027230
                            00027240
                            00027250
                            00027260
                            00027270
                            00027280
                            00027290
                            00027300
                            00027310
                            00027320
                            00027330
                            00027340
                            00027350
                            00027360
                            00027370
                            00027380
                            00027390
                            00027400
                            00027410
                            00027420
                            00027430
                            00027440
                            00027450
                            00027460
                            00027470
                            00027480
                            00027490
                            00027500
                            00027510
                            00027520
                            00027530
                            00027540
                            00027550
                            00027560
                            00027570
                            00027580
                            00027590
                            00027600
                            00027610
                            00027620
                            00027630
                            00027640
                            00027650
                            00027660
                            00027670
                            00027680
                            00027690
                            00027700
                            00027710
                            00027720
                            00027730
                            00027740
                            00027750
                            00027760
                            00027770
                            00027780
                            00027790
                            00027800
                            00027810
                            00027820
                            00027830
                            00027840
                            00027850
                            00027860
                            00027870
                            00027880

SUBROUTINE TALOOK(Z,ZINTEG)                                00027300
COMMON/ONE/      EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
+                 PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250)    00027310
COMMON/TWO/      EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
+                 NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
+                 RSMAX(20)                                         00027320
COMMON/THREE/    AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
+                 ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
+                 PERENE(20,4)                                       00027330
COMMON/FOUR/     ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
+                 NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP                00027340
COMMON/FIVE/     HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM                  00027350
COMMON/SIX/      YINT(120),Y(120)                           00027360
DOUBLE PRECISION YINT,Y                                     00027370
COMMON/SEVEN/    CMULT(4),EVAL(3),EMULT(4),EXPLCU(4,8),
+                 HYEXP(11),NHYNM(20),NPSMN(20),NTHY(20),
+                 NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20)
DOUBLE PRECISION AVZ,Z,ZINTEG,DY,PK,ZINT,FACTOR,
+                 R1,R2,R3,R4,Q12,Q13,Q14,Q23,Q24,Q34,Q21,Q31,Q41,Q32,Q42,Q43 00027380
LOOKUP IN NORMAL DISTRIBUTION FUNCTION THE VALUE OF        00027390
THE INTEGRAL OF EXP(-Z2/2) FROM MINUS INFINITY TO Z.       00027400
INTERPOLATE WITH CUBIC FIT BETWEEN THE 120 DATA POINTS     00027410
COPIED FROM TABLES. THE REFERENCE USED IS: TABLES OF      00027420
NORMAL PROBABILITY FUNCTIONS BY THE U.S. DEPARTMENT        00027430
OF COMMERCE.                                              00027440
DEFINITION OF KEY VARIABLES:                                00027450
NAME   TYPE   SIZE   MEANING
Z      REAL   -      NORMALIZED INDEPENDENT VARIABLE 00027460
ZINTEG REAL   -      INTEGRAL VALUE                   00027470
Y      REAL   120    NORMALIZE INDEPENDENT VARIABLE 00027480
YINT   REAL   120    INPUT DATA                      00027490
Y     INT.   -      GRID POINT CORRESPONDING TO Z 00027500
DY     REAL   -      Y INCREMENT                     00027510
IF(Z.GE.-5.9D00) GO TO 5                                 00027520
ZINTEG = 0.0D00                                           00027530
RETURN                                                    00027540
5 CONTINUE                                                 00027550
IF ( DABS(Z).GT.1.0D-10 ) GO TO 10                      00027560
ZINTEG = 0.5D00                                           00027570

```

```

      RETURN
10  CONTINUE
      DY = Y(2) - Y(1)
      PK = DABS(Z)/DY + 1.0D00
      K = PK
      AVZ = DABS(Z)
      R1 = 0.0D00
      R2 = 0.0D00
      R3 = 0.0D00
      R4 = 0.0D00
      IF ( AVZ.NE.Y(K))R1 = AVZ - Y(K)
      IF ( AVZ.NE.Y(K+1))R2 = AVZ - Y(K+1)
      IF ( AVZ.NE.Y(K+2))R3 = AVZ - Y(K+2)
      IF ( AVZ.NE.Y(K+3))R4 = AVZ - Y(K+3)
C
      Q12= Y(K) - Y(K+1)
      Q13= Y(K) - Y(K+2)
      Q14= Y(K) - Y(K+3)
      Q23= Y(K+1) - Y(K+2)
      Q24= Y(K+1) - Y(K+3)
      Q34= Y(K+2) - Y(K+3)
      Q21= -Q12
      Q31= -Q13
      Q41= -Q14
      Q32= -Q23
      Q42= -Q24
      Q43= -Q34
      IF ( DABS(R1).GT.1.0D-10)GO TO 20
      ZINT = YINT(K)*R2*R3*R4/(Q12*Q13*Q14)
      GO TO 30
20  IF ( DABS(R2).GT.1.0D-10)GO TO 22
      ZINT = YINT(K+1)*R1*R3*R4/(Q21*Q23*Q24)
      GO TO 30
22  IF ( DABS(R3).GT.1.0D-10)GO TO 24
      ZINT = YINT(K+2)*R1*R2*R4/(Q31*Q32*Q34)
      GO TO 30
24  IF ( DABS(R4).GT.1.0D-10)GO TO 26
      ZINT = YINT(K+3)*R1*R2*R3/(Q41*Q42*Q43)
      GO TO 30
C
26  ZINT = YINT(K)*R2*R3*R4/(Q12*Q13*Q14) +
      + YINT(K+1)*R1*R3*R4/(Q21*Q23*Q24) +
      + YINT(K+2)*R1*R2*R4/(Q31*Q32*Q34) +
      + YINT(K+3)*R1*R2*R3/(Q41*Q42*Q43)
C
30  FACTOR = 1.0D00
      IF (Z.LT.0.0D00)FACTOR = -1.0D00
      ZINTEG = 0.5D00 * ( 1.0D00 + FACTOR * ZINT )
      IF ( ZINTEG.GT.1.0D00 ) ZINTEG = 1.0D00
      RETURN
      END

SUBROUTINE RINPUT

COMMON/ONE/   EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
      + PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250)          00028480
      + 00028490
      + 00028500
      + 00028510
      + 00028520
      + 00028530
      + 00028540
      + 00028550
      + 00028560

COMMON/TWO/   EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
      + NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
      + RSMAX(20)          00028440
      + 00028460
      + 00028470
      + 00028490
      + 00028500
      + 00028510
      + 00028520
      + 00028530
      + 00028540
      + 00028550
      + 00028560

```

```

C COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
+ ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
+ PERENE(20,4) 00028570
C COMMON/FOUR/ ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
+ NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP 00028580
C COMMON/FIVE/ HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
+ DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM 00028590
C COMMON/SIX/ TINT(120),TY(120)
+ DOUBLE PRECISION TINT,TY 00028600
C COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),
+ HYEXPN(11),HYHMN(20),NPSMN(20),NTHY(20),
+ NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00028610
C DATA K25,K26/25,26/ 00028620
C READ LOAD CUMULANTS 00028630
C DO 10 I=1,4 00028640
C READ(K26,1000) (HRCUM(I,J),J=1,8) 00028650
1000 FORMAT(1H /(3E23.14)) 00028660
10 CONTINUE 00028670
C READ PEAK AND ORIGINAL LOC 00028680
C INITIALIZATION 00028690
C DO 20 I=1,4 00028700
C DO 20 J=1,MX 00028710
20 ELL(I,J)=0.0 00028720
C DO 30 I=1,4 00028730
C READ(K25) ICY,KS 00028740
C READ(K25) DUM,DUM,PEAK(JYR,I),PERENE(JYR,I),DUM 00028750
C READ(K25) (EL(I,J),J=1,500) 00028760
30 CONTINUE 00028770
C NORMALIZATION 00028780
C DO 40 I=1,4 00028790
C FNOR=EL(I,1) 00028800
C DO 45 J=1,500 00028810
45 ELL(I,J)=EL(I,J)/FNOR 00028820
40 CONTINUE 00028830
C HYDRO SCHEDULED SYSTEM 00028840
C READ(31,1010) NNUM 00028850
C READ(31,1020) (HYDR(I,J),J=1,8) 00028860
C PUMPED STORAGE SYSTEM 00028870
C READ(33,1010) NNUM 00028880

```

```

      READ(33,1020) (HYDR(3,J),J=1,8)
1010 FORMAT(/,3I3)                                     00029250
1020 FORMAT(5E12.5)                                   00029260
      RETURN                                         00029270
      END                                           00029280
                                               00029290
                                               00029300
                                               00029310
                                               00029320
                                               00029330
                                               00029340
                                               00029350
                                               00029360
                                               00029370
                                               00029380
                                               00029390
                                               00029400
                                               00029410
                                               00029420
                                               00029430
                                               00029440
                                               00029450
                                               00029460
                                               00029470
                                               00029480
                                               00029490
                                               00029500
                                               00029510
                                               00029520
                                               00029530
                                               00029540
                                               00029550
                                               00029560
                                               00029570
                                               00029580
                                               00029590
                                               00029600
                                               00029610
                                               00029620
                                               00029630
                                               00029640
                                               00029650
                                               00029660
                                               00029670
                                               00029680
                                               00029690
                                               00029700
                                               00029710
                                               00029720
                                               00029730
1000 FORMAT(//18X,7HMINIMUM,5X,7HMAXIMUM,5X, 00029740
  +12HLOWER BOUND,5X,6HANNUAL)                      00029750
      WRITE(6,1001)                                     00029760
1001 FORMAT(18X,7HRESERVE,5X,7HRESERVE,5X,12HOF OPERATING, 00029770
  +5X,6H PEAK )                                     00029780
      WRITE(6,1002)                                     00029790
1002 FORMAT(9X,4HYEAR,6X,5H(MW)*,7X,4H(MW),7X,7HCOST **, 00029800
  +1IX,4H(MW))                                     00029810
      WRITE(6,1003)                                     00029820
1003 FORMAT(9X,4(1H-),2(5X,7(1H-)),5X,12(1H-),5X,6(1H-)) 00029830
      DO 100 NY=1,NTYR                                00029840
      IY=IYRB+NY-1+1900                               00029850
      WRITE(6,1004) IY,RSMIN(NY),RSMAX(NY),COSTOP(NY),PMAX(NY) 00029860
100 CONTINUE                                         00029870
1004 FORMAT(9X,I4,2(5X,F7.2),6X,F9.3,6X,F7.1)        00029880
      WRITE(6,1005) PPMIN                            00029890
1005 FORMAT(//9X,29H* CRITICAL LOLP ASSIGNED IS ,F9.6) 00029900
      IY=IYRB-1+1900                               00029910
      WRITE(6,1006) IY                                00029920

SUBROUTINE INOUT
COMMON/ONE/ EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20), 00029380
  + PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250)          00029390
COMMON/TWO/ EMOR(220,4),EMORH(3,4),IX(20),MAINS(200), 00029400
  + NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20), 00029410
  + RSMAX(20)                                         00029420
COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3), 00029430
  + ITK(500,3),MXELDC(20,4),NSTPRE(20,10), 00029440
  + PERENE(20,4)                                         00029450
COMMON/FOUR/ ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP, 00029460
  + NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP                00029470
COMMON/FIVE/ HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8) 00029480
  DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM           00029490
COMMON/SIX/ TINT(120),TY(120)                         00029500
  DOUBLE PRECISION TINT,TY                           00029510
COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8), 00029520
  + HYEXP(11),NHYMN(20),NP SMN(20),NTHY(20), 00029530
  + NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00029540
DATA L10/10/
      WRITE(6,1000)                                     00029550
1000 FORMAT(//18X,7HMINIMUM,5X,7HMAXIMUM,5X, 00029560
  +12HLOWER BOUND,5X,6HANNUAL)                      00029570
      WRITE(6,1001)                                     00029580
1001 FORMAT(18X,7HRESERVE,5X,7HRESERVE,5X,12HOF OPERATING, 00029590
  +5X,6H PEAK )                                     00029600
      WRITE(6,1002)                                     00029610
1002 FORMAT(9X,4HYEAR,6X,5H(MW)*,7X,4H(MW),7X,7HCOST **, 00029620
  +1IX,4H(MW))                                     00029630
      WRITE(6,1003)                                     00029640
1003 FORMAT(9X,4(1H-),2(5X,7(1H-)),5X,12(1H-),5X,6(1H-)) 00029650
      DO 100 NY=1,NTYR                                00029660
      IY=IYRB+NY-1+1900                               00029670
      WRITE(6,1004) IY,RSMIN(NY),RSMAX(NY),COSTOP(NY),PMAX(NY) 00029680
100 CONTINUE                                         00029690
1004 FORMAT(9X,I4,2(5X,F7.2),6X,F9.3,6X,F7.1)        00029700
      WRITE(6,1005) PPMIN                            00029710
1005 FORMAT(//9X,29H* CRITICAL LOLP ASSIGNED IS ,F9.6) 00029720
      IY=IYRB-1+1900                               00029730
      WRITE(6,1006) IY                                00029740
      WRITE(6,1001)                                     00029750
1001 FORMAT(18X,7HRESERVE,5X,7HRESERVE,5X,12HOF OPERATING, 00029760
  +5X,6H PEAK )                                     00029770
      WRITE(6,1002)                                     00029780
1002 FORMAT(9X,4HYEAR,6X,5H(MW)*,7X,4H(MW),7X,7HCOST **, 00029790
  +1IX,4H(MW))                                     00029800
      WRITE(6,1003)                                     00029810
1003 FORMAT(9X,4(1H-),2(5X,7(1H-)),5X,12(1H-),5X,6(1H-)) 00029820
      DO 100 NY=1,NTYR                                00029830
      IY=IYRB+NY-1+1900                               00029840
      WRITE(6,1004) IY,RSMIN(NY),RSMAX(NY),COSTOP(NY),PMAX(NY) 00029850
100 CONTINUE                                         00029860
1004 FORMAT(9X,I4,2(5X,F7.2),6X,F9.3,6X,F7.1)        00029870
      WRITE(6,1005) PPMIN                            00029880
1005 FORMAT(//9X,29H* CRITICAL LOLP ASSIGNED IS ,F9.6) 00029890
      IY=IYRB-1+1900                               00029900
      WRITE(6,1006) IY                                00029910
      WRITE(6,1001)                                     00029920

```

```

1006 FORMAT(1X,15H** MILLIONS OF ,I4,8H DOLLARS//) 00029930
      WRITE(L10,902) PPMIN,T 00029940
902 FORMAT(5E16.8) 00029950
      IBASYR=IYRB-1+1900 00029960
      MAXPLA=IPMAX-NPEXP 00029970
      WRITE(L10,900) IBASYR,MX,NPEXP,MAXPLA,IPT,NTYR,KMAX 00029980
900 FORMAT(20I4) 00029990
      DO 10 J=1,NTYR 00030000
10   WRITE(L10,900) (MXELDC(J,I),I=1,4) 00030010
      DO 20 J=1,NTYR 00030020
20   WRITE(L10,902) EC(J),RSMIN(J),RSMAX(J),COSTOP(J) 00030030
      DO 30 J=1,NTYR 00030040
30   WRITE(L10,902) (PEAK(J,I),I=1,4) 00030050
      DO 35 J=1,NTYR 00030060
35   WRITE(L10,902) (PERENE(J,I),I=1,4) 00030070
      WRITE(L10,911) (HYEXPN(J),J=1,11) 00030080
      WRITE(L10,911) (PSEXP(J),J=1,11) 00030090
911 FORMAT(5E12.5) 00030100
      DO 40 IP=1,NPEXP 00030110
40   WRITE(L10,904) (PLANT(IP,J),J=1,14) 00030120
904 FORMAT(F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2) 00030130
      DO 60 J=1,NTYR 00030140
60   WRITE(L10,900) NHYMN(J),NPSTMN(J),(NUF(IP,J),IP=1,NPEXP) 00030150
      DO 70 J=1,NTYR 00030160
70   WRITE(L10,900) NHYMX(J),NPSPMX(J),(IUP(IP,J),IP=1,NPEXP) 00030170
      DO 80 J=1,NTYR 00030180
80   WRITE(L10,902) (AVSPE(J,N),N=1,4) 00030190
      DO 90 NY=1,NTYR 00030200
90   WRITE(L10,909) (SCHCUM(J,NY),J=1,4) 00030210
909 FORMAT(4E16.8) 00030220
      WRITE(L10,911) (CMULT(J),J=1,4) 00030230
      WRITE(L10,911) (EMULT(J),J=1,4) 00030240
      00030250
C      RETURN 00030260
C      END 00030270
C
C      SUBROUTINE RESMX(ECP,SCAP,NCRL) 00030280
C
C      COMMON/ONE/ EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20), 00030290
C      PEAK(20,4),PLANT(200,14),PLACAT(20,14),X(1250) 00030300
C
C      COMMON/TWO/ EMOR(220,4),EMORH(3,4),IX(20),MAINS(200), 00030310
C      NHYMX(20),NPSPMX(20),NT(20),NX(20),RSMIN(20), 00030320
C      RSMAX(20) 00030330
C
C      COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3), 00030340
C      ITK(500,3),MXELDC(20,4),NSTPRE(20,10), 00030350
C      PERENE(20,4) 00030360
C
C      COMMON/FOUR/ ID,IPMAX,IPT,IYRB,NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP 00030370
C
C      COMMON/FIVE/ HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8) 00030380
C      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM 00030390
C
C      COMMON/SIX/ TINT(120),TY(120) 00030400
C      DOUBLE PRECISION TINT,TY 00030410
C
C      00030420
C      00030430
C      00030440
C      00030450
C      00030460
C      00030470
C      00030480
C      00030490
C      00030500
C      00030510
C      00030520
C      00030530
C      00030540
C      00030550
C      00030560
C      00030570
C      00030580
C      00030590
C      00030600

```

```

C COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),
+ HYEXP(11),NHYMN(20),NPSMN(20),NTHY(20),
+ NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00030610
00030620
00030630
00030640
00030650
00030660
00030670
00030680
00030690
00030700
00030710
00030720
00030730
00030740
00030750
00030760
00030770
00030780
00030790
00030800
00030810
00030820
00030830
00030840
00030850
00030860
00030870
00030880
00030890
00030900
00030910
00030920
00030930
00030940
00030950
00030960
00030970
00030980
00030990
00031000
00031010
00031020
00031030
00031040
00031050
00031060
00031070
00031080
00031090
00031100
00031110
00031120
00031130
00031140
00031150
00031160
00031170
00031180
00031190
00031200
00031210
00031220
00031230
00031240
00031250
00031260
00031270
00031280

C DIMENSION PLP(4),SCAP(4)

C ICNTL=-1
10 ECP=ECP+PLANT(1,3)
CLPMX=10.0
DO 20 N=1,NP
PK=PEAR(JYR,N)
IF(ICNTL.LT.0) S=SCAP(N)
Y=PLANT(1,3)*(1.0-EMOR(2,N))/PK
P=1.0-PLANT(1,5)
S=S+Y
CALL PWADD(Y,P,N,NMAXI)
A=S
PLP(N)=PLOLP(A,N)
IF(PLP(N).LE.CLPMX) GO TO 15
GO TO 20
15 CLPMX=PLP(N)
NCRL=N
20 CONTINUE
IF(CLPMX.LE.PPMAX) GO TO 30
ICNTL=1
GO TO 10
30 RETURN
END

C -----
SUBROUTINE COMPST
-----

C COMMON/ONE/ EL(4,1250),HYDR(3,11),IUP(20,20),NUF(220,20),
+ PEAK(20,4),PLANT(200,14),PLACA(20,14),X(1250) 00031000
00031010
00031020
00031030
00031040
00031050
00031060
00031070
00031080
00031090
00031100
00031110
00031120
00031130
00031140
00031150
00031160
00031170
00031180
00031190
00031200
00031210
00031220
00031230
00031240
00031250
00031260
00031270
00031280

C COMMON/TWO/ EMOR(220,4),EMORH(3,4),IX(20),MAINS(200),
+ NHYMX(20),NPSMX(20),NT(20),NX(20),RSMIN(20),
+ RSMAX(20) 00031040
00031050
00031060
00031070
00031080
00031090
00031100
00031110
00031120
00031130
00031140
00031150
00031160
00031170
00031180
00031190
00031200
00031210
00031220
00031230
00031240
00031250
00031260
00031270
00031280

C COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
+ ITK(500,3),MXELDC(20,4),NSTPRE(20,10),
+ PERENE(20,4) 00031100
00031110
00031120
00031130
00031140
00031150
00031160
00031170
00031180
00031190
00031200
00031210
00031220
00031230
00031240
00031250
00031260
00031270
00031280

C COMMON/FOUR/ ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
+ NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP 00031150
00031160
00031170
00031180
00031190
00031200
00031210
00031220
00031230
00031240
00031250
00031260
00031270
00031280

C COMMON/FIVE/ HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM 00031180
00031190
00031200
00031210
00031220
00031230
00031240
00031250
00031260
00031270
00031280

C COMMON/SIX/ TINT(120),TY(120)
DOUBLE PRECISION TINT,TY 00031220
00031230
00031240
00031250
00031260
00031270
00031280

C COMMON/SEVEN/ CMULT(4),EAVAL(3),EMULT(4),EXPLCU(4,8),
+ HYEXP(11),NHYMN(20),NPSMN(20),NTHY(20), 00031220
00031230
00031240
00031250
00031260
00031270
00031280

```

```

+      NTPS(20),NXHY(20),NXPS(20),PSEXP(11),SCHCUM(4,20) 00031290
+      00031300
+      00031310
+      00031320
+      00031330
+      00031340
+      00031350
+      00031360
+      00031370
+      00031380
+      00031390
+      00031400
+      00031410
+      00031420
+      00031430
+      00031440
+      00031450
+      00031460
+      00031470
+      00031480
+      00031490
+      00031500
+      00031510
+      00031520
+      00031530
+      00031540
+      00031550
+      00031560
+      00031570
+      00031580
+      00031590
+      00031600
+      00031610
+      00031620
+      00031630
+      00031640
+      00031650
+      00031660
+      00031670
+      00031680
+      00031690

      HYDRO EXPN PLANT

      IF(NXHY(JYR).LE.0) GO TO 10
      HMULT=FLOAT(NXHY(JYR))
      GO TO 20
10     HMULT=FLOAT(NTHY(JYR))
20     DO 30 J=1,4
      HYDR(2,J)=HMULT*HYEXPN(J)
30     CONTINUE
      DO 35 J=5,11
      HYDR(2,J)=HYEXPN(J)

      PUMPED STORAGE PLANT

      IF(NXPS(JYR).LE.0) GO TO 40
      HMULT=FLOAT(NXPS(JYR))
      GO TO 50
40     HMULT=FLOAT(NTPS(JYR))
50     W1=HYDR(3,1)
      W2=HYDR(3,2)
      W3=HMULT*PSEXP(1)
      W4=HMULT*PSEXP(2)

      HYDR(3,1)=W1+W3
      HYDR(3,2)=W2+W4
      DO 60 J=3,7
60     HYDR(3,J)=(W1*HYDR(3,J)+W3*PSEXP(J))/HYDR(3,1)
      HYDR(3,8)=(W2*HYDR(3,8)+W4*PSEXP(8))/HYDR(3,2)
      DO 70 J=9,11
70     HYDR(3,J)=PSEXP(J)

      RETURN
      END

```

*** DYN02.FORT ***			
DYNO IS THE OPTIMIZATION SUBMODULE OF THE OPTIM MODULE IN THE CERES CODE.			
NAME	TYPE	SIZE	DEFINITION
CAPABS	REAL	20	SPACE AVAILABLE FOR SCHEDULED MAINTENANCE SCHEDULED SYSTEM CAPACITY FOR EACH YEAR OF THE STUDY PERIOD.
CLOLP	REAL	=	CRITICAL LOSS-OF-LOAD-PROBABILITY(LOLP)
CPLOLP	REAL	=	CRITICAL LOSS-OF-LOAD-PROBABILITY(LOLP) AS DEFINED IN PREP SUBMODULE
DISRAT	REAL	-	DISCOUNT RATE
DX	REAL	-	NORMALIZED MW INCREMENT (DX=1./MAXPO)
EL	REAL	1250	AUXILIARY ELD C FOR LOLP CALCULATION
ELDC	REAL	4,1250	FINAL LOAD DURATION CURVE AFTER ALL THE UNITS OF THE SCHEDULED SYSTEM HAVE BEEN CONVOLVED (USED FOR LOLP CALCULATIONS). ELDC IS READ FROM THE DIRECT ACCESS FILE 22
ELF	REAL	1260	AUXILIARY ELD C FOR LOLP CALCULATION
ENEDEM	REAL	20,4	ENERGY DEMAND FOR EACH SEASON
EXPLCU	REAL	4,8	{JCUM,I} EXPANSION PLANT CUMULANTS JCUM=1,4 IS THE JCUM-TH CUMULANT I=1,8 IS THE I-TH PLANT TYPE IN THE RESTRICTED EX.PLANT LIST.
FATOPC	REAL	20	LOWER BOUND FOR ENERGY COST FOR EACH YEAR (IN MILLION DOLLARS. USED FOR FATHOMING)
FCR	REAL	-	FIXED CHARGE RATE
HOURS	REAL	-	NUMBER OF HOURS PER SIMULATION PERIOD
IBASYR	INT.	-	BASE YEAR CALENDAR YEAR NEXT TO WHICH THE STUDY PERIOD BEGINS.
IDEXP	INT.	8	IDEXP(I)=IP MEANS THAT THE I-TH PLANT PLANT IS THE IP-TH INPUT MODULE PLANT.
IPCH	INT,	8	NUMBER OF UNITS BY WHICH THE ARTIFICIAL
ISOL	INT.	20,8	IISOL(N,J): OPTIMUM OF SUBOPTIMUM SOLUTION N: YEAR OF STUDY PERIOD
ITIN	INT.	-	I: NUMBER OF UNITS OF I-TH PLANT TYPE TUNNEL CHANGE FOR EACH EXP. CANDIDATE
ITMAX	INT.	-	NUMBER OF TUNNEL ITERATIONS
ITOUT	INT.	-	MAXIMUM ITIN
LBOLD	INT.	8	NUMBER OF DYNO SENSITIVITY ANALYSES (I) ARTIFICIAL LOWER TUNNEL BOUNDARY BEFORE THE CURRENT CALL TO THE ADJUST ROUTINE.

C LIST	INT.	8,1000	I=1,8 I-TH PLANT IN RESTRICTED EXP.PLANT LIST LIST OF STATES GENERATED WITHIN TUNNELS FROM ORIGIN STATE	00000690 00000700 00000710
C LOWB	INT.	20,8	ARTIFICIAL YEARLY PLANTS TUNNEL LOWER BOUND	00000720
C MAINS	INT.	200	MAINS(I) DEFINES THE SEASON IN WHICH SCHEDULED PLANT I IS SET DOWN FOR MAINTENANCE.	00000730 00000740
C MAXADD	INT.	8	MAXIMUM NUMBER OF UNITS THAT CAN BE ADDED EACH YEAR.	00000750 00000760
C MAXALL	INT.	-	MAXPLA + MXPL	00000770
C MAXINP	INT.	-	MAXIMUM NUMBER OF EXPANSION CANDIDATES DEFINED BY INPUT MODULE	00000780 00000790
C MAXI	INT.	-	MAXIMUM NUMBER OF POINTS IN THE ELD C ARRAYS	00000800
C MAXPLA	INT.	-	MAXIMUM NUMBER OF PLANTS IN THE SCHEDULED SYSTEM.	00000810 00000820
C MAXPO	INT.	-	MAXIMUM NUMBER OF POINTS IN THE NORMALIZED ORIGINAL ELD C.	00000830 00000840
C MAXOR	INT.	-	NUMBER OF PLANT BLOCKS IN THE LOADING ORDER	00000850
C MAXTUN	INT.	8	(I) ARTIFICIAL UPPER TUNNEL BOUNDARY FOR CURRENT ORIGIN STATE.	00000860 00000870
C MAXUN	INT.	20,8	I=1,8 I-TH PLANT IN RESTRICTED EXP.PLANT LIST	00000880
C MINTUN	INT.	8	MAXIMUM ALLOWED NUMBER OF UNITS FOR EACH YEAR AND EXP. CANDIDATE	00000890 00000900
C MINUN	INT.	20,8	(I) ARTIFICIAL LOWER TUNNEL BOUNDARY FOR CURRENT ORIGIN STATE.	00000910 00000920
C MXELDC	INT.	20,4	I=1,8 I-TH PLANT IN RESTRICTED EXP.PLANT LIST	00000930
C MXPL	INT.	-	MINIMUM ALLOWED NUMBER OF UNITS FOR EACH YEAR AND EXP. CANDIDATE	00000940 00000950
C MXYEAR	INT.	-	MAXIMUM NUMBER OF POINTS IN EACH OF THE SCHEDULED SYSTEM ELD C'S FOR THE STUDY PERIOD	00000960 00000970
C NEXPID	INT.	8	MAXIMUM NUMBER OF NEW CANDIDATES.	00000980
C NORDER	INT.	420,3	MAXIMUM NUMBER OF YEARS IN STUDY PERIOD	00000990
			NEXPID(IP)=I MEANS THAT THE IP-TH INPUT MODULE PLANT IS THE I-TH PLACA PLANT.	00001000
			IF I<0 THE PLANT IS NOT USED IN THE CURRENT SENSITIVITY ANALYSIS	00001010
			NORDER(I,J): PLANT LOADING ORDER.	00001020
			I=1,420: PLANT BLOCK LOADING ORDER	00001030
			J=1: INDICATES THE POSITION OF THE PLANT IN THE SCHEDULED OR NEW CANDIDATE FILES (PLANTS & PLACA RESPECTIVELY)	00001040
			J=2: PLANT BLOCK	00001050
			J=3: IF >=0, IT INDICATES THE NUMBER OF UNITS OF PLANT (I,J=1) IN THE SCHEDULED SYSTEM.	00001060
			IF < 0 (USUALLY -1), INDICATES THAT THE CORRESPONDING PLANT (I,J=1) IS A NEW CANDIDATE.	00001070
			THE NUMBER OF UNITS FOR THIS PLANT IS FOUND FROM THE STATE UNDER EXAMINATION.	00001080
C NUBOLD	INT.	8	(I) ARTIFICIAL UPPER TUNNEL BOUNDARY BEFORE THE CURRENT CALL TO THE ADJUST ROUTINE.	00001190 00001200
C NSTPRE	INT.	20,8	I=1,8 I-TH PLANT IN RESTRICTED EXP.PLANT LIST	00001210
C NUPB	INT.	20,8	YEARLY STATES USED FOR THE MINIMUM RESERVE	00001220
C PEAKS	REAL	20	MARGIN CALCULATION IN PREP SUBMODULE. THEY USED HERE FOR THE FIRST ESTIMATE OF UBOUND.	00001230 00001240
C PLACA	REAL	8,14	ARTIFICIAL YEARLY PLANTS TUNNEL UPPER BOUND	00001250
			YEARLY PEAKS FOR THE 20 YEARS OF THE STUDY PERIOD	00001260
			PLACA(I,J): PLANT CANDIDATES.	00001270
			I: PLANT NUMBER IN THE SAME ORDER AS IN THE NEW CANDIDATE PLANTS FILE.	00001280
			J=1 NUMBER OF UNITS BELONGING TO THIS PLANT CODE	00001290 00001300
			J=2 BASE CAPACITY IN MW	00001310 00001320
			J=3 MAX CAPACITY IN MW	00001330
			J=4 MAINTENANCE REQUIREMENT IN DAYS PER YEAR	00001340
			J=5 FORCED OUTAGE RATE	00001350
			J=6 CAPITAL COST IN \$/KW	00001360

			J=7	BASE FUEL COST IN \$/MWH	00001370
			J=8	MAX OPERATING FUEL COST IN \$/MWH	00001380
			J=9	ECONOMIC PLANT LIFE IN YEARS	00001390
			J=10	FIXED OPERATION AND MAINTENANCE COSTS IN \$/MW.YEAR	00001400
			J=11	VARIABLE OPERATION AND MAINTENANCE COSTS IN \$/MWH	00001410
			J=12	SALVAGE VALUE IN THOUSAND DOLLARS	00001430
			J=13	ANNUAL CAPITAL COST ESCALATION RATE	00001450
			J=14	FUEL COST ESCALATION RATE	00001460
PLANTS	REAL	200,14	PLANTS(I,J):	SCHEDULED SYSTEM PLANTS	00001470
			I :	PLANT NUMBER IN THE SAME ORDER AS IN THE SCHEDULED SYSTEM FILE.	00001480
			J=1	NUMBER OF UNITS BELONGING TO THIS PLANT CODE	00001490
			J=2	BASE CAPACITY IN MW	00001500
			J=3	MAX CAPACITY IN MW	00001510
			J=4	MAINTENANCE REQUIREMENT IN DAYS PER YEAR	00001520
			J=5	FORCED OUTAGE RATE	00001530
			J=6	CAPITAL COST IN \$/KW	00001540
			J=7	BASE FUEL COST IN \$/MWH	00001550
			J=8	MAX OPERATING FUEL COST IN \$/MWH	00001560
			J=9	ECONOMIC PLANT LIFE IN YEARS	00001570
			J=10	FIXED OPERATION AND MAINTENANCE COSTS IN \$/MW.YEAR	00001580
			J=11	VARIABLE OPERATION AND MAINTENANCE COSTS IN \$/MWH	00001590
			J=12	SALVAGE VALUE IN THOUSAND DOLLARS	00001600
			J=13	ANNUAL CAPITAL COST ESCALATION RATE	00001610
			J=14	FUEL COST ESCALATION RATE	00001620
POPEC	REAL	225,20	POPEC(IP,NY)	OPERATING COST	00001630
			IP:	PLANT ID	00001640
			NY:	YEAR OF STUDY PERIOD	00001650
PTCUM	REAL	200,8,4	PTCUM(IP,I,J)	SCHEDULED SYSTEM PLANT CUMULANTS	00001660
			IP:	PLANT ID AS DEFINED IN INPUT MODULE	00001670
			I=1,4	BASE BLOCK CUMULANTS	00001680
			I=5,8	PLANT CUMULANTS	00001690
			J=1,4	YEARLY SEASON EXP. CANDIDATE CUMULANTS DEFINED AS IN PTCUM	00001700
PTCUMX	REAL	8,8,4	ARRAY MOR(I,J)	ARRAY MOR(I,J)	00001710
PESMAR	REAL	-		MAXIMUM RESERVE MARGIN (DEFINED IN % AND USED AS DECIMAL)	00001720
RESMAX	REAL	20		MAXIMUM RESERVE MARGIN EACH YEAR	00001730
RESMIN	REAL	20		MINIMUM RESERVE MARGIN EACH YEAR	00001740
ROM	REAL	20,4		EXP. CANDIDATE MAINTENANCE OUTAGE RATE	00001750
			I:	EXP. CANDIDATE NUMBER	00001760
SOL	REAL	20,2	J=1,4	SEASON NUMBER	00001770
			SOL(N,J):	OPTIMUM OR SUBOPTIMUM SOLUTION CHARACTERISTICS.	00001780
			N:	YEAR OF STUDY PERIOD	00001790
			J=1	LOLP	00001800
SSCUM	REAL	4,20	J=2	OBJECTIVE FUNCTION	00001810
			(JCUM,NY)	SCHEDULED SYSTEM CUMULANTS FOR THE PEAK SEASON IN EACH YEAR	00001820
			JCUM=1,4	IS THE JCUM-TH CUMULANT	00001830
			NY=1,20	IS THE YEAR IN THE STUDY DERIOD.	00001840
UBOUND	REAL	-		UPPEK BOUND OF OBJECTIVE FUNCTION. IN MILLION DOLLARS. (USED FOR FATHOMING)	00001850
UNE	REAL	20		UNSERVED ENERGY FOR EACH YEAR	00001860
X	REAL	1250		NORMALIZED MW ARRAY	00001870
					00001880
					00001890
					00001900
					00001910
					00001920
					00001930
					00001940
					00001950
					00001960
					00001970
					00001980
					00001990
					00002000
					00002010
					00002020
					00002030
					00002040
				FILE DESCRIPTION	
NAME	UNIT		TYPE	DEFINITION	
C INOUT	8		SEQUEN.	INPUT DATE READ BY READIN SUBROUTINE	

```

C DPOUT    11      SEQUEN.   DEBUG OUTPUT          00002050
C STATES   14      DIR.AC.   YEARLY ACCEPTED STATES 00002060
C LDFIN    22      SEQUEN.   SCHEDULED SYSTEM FINAL ELDCS 00002070
C PLSCH    23      DIR.AC.   SCHEDULED PLANT DATA     00002080
C LORD     24      SEQUEN.   ECONOMIC LOADING ORDER    00002090
C PTCUM    27      SEQUEN.   SCHEDULED PLANT CUMULANTS 00002100
C SOL      28      DIR.AC.   FINAL SOLUTIONS AFTER EACH TUNNEL ITERATION 00002110
C
C
C      DEFINE FILE 14(20,8004,L,1DAF),15(20,16004,L,1DAF),
C      + 28(400,60,L,128)                                     00002120
C
C
C      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
C      1 FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
C      2 ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
C      3 MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
C      4 RESMAR,UBOUND                                     00002130
C
C      COMMON /ARS/ AVSP(20,4),CAPAB(20),ENEDEM(20,4),EXPLCU(4,8),
C      1 IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
C      2 IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
C      3 MAXUN(20,20),MAXADD(8),MINUN(20,20),
C      4 NEXPID(20),NSTPRE(20,20),NUPB(20,8),
C      5 PEAKS(20,4),PLACA(20,14),PLANT(210,14),
C      6 RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
C      7 FATOP(20),SOL(20,2),AUPEAK(20)                 00002140
C
C      COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
C      + MXELOC(20,4),X(1250)                           00002150
C
C      COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8) 00002160
C
C      COMMON /TNS/ LBQLD(8),LIST(8,1000),MAXTUN(8),
C      + MINTUN(8),NUBOLD(8)                            00002170
C
C      COMMON /TBL/ TINT(120),TY(120)                  00002180
C
C      DOUBLE PRECISION TINT,TY                         00002190
C
C      COMMON /HYS/ CMULT(4),EVAL(3),EMULT(4),HYDR(3,11),
C      + HYEXP(11),HYSCHD(20,8),PSEXPN(11),
C      + PSSCHD(20,8),SYSCUM(4)                      00002200
C
C
C      DIMENSION CUMLOL(1000), MINSNU(20), NSTCUR(20), NSTATE(20),
C      + NOPTIM(20,2), NUMACC(20), OPTIMA(20,4), RESCP(20,20),
C      + NST2(2,1000), NST1(1000), STN2(4,1000), STN1(1000) 00002210
C
C      ***** SECTION 1 *****                           00002220
C
C
C      DATA MINSNU/20*5/
C      READ INPUT DATA                                00002230
C      CALL READIN                                     00002240
C      INITIALIZE PROGRAM FLAGS                     00002250
C      IRES = 0                                         ?
C      ITOUT = 0                                       ?
C      IFLAG = 0                                         ?
C      INITIALIZE OBJECTIVE FUNCTION UPPER BOUND    00002260
C      CALL UFIRST                                     00002270
C      RESET=UBOUND                                    00002280

```

```

C   DEFINE INPUT PARAMETERS FOR CURRENT OUTER ITERATION      00002730
10 CALL CHOICE(ISTOP)                                         00002740
    IF (ISTOP .GT. 0) GO TO 600                               00002750
    UBOUND=RESET                                              00002760
C
C   CALCULATE PLANT CUMULANTS OF EXPANSION CANDIDATES      00002770
    WITHOUT MAINTENANCE CONSIDERATION                         00002780
C
C   N=1                                                       00002840
C   ICUM=-1                                                 00002850
C   DO 16 I=1,MXPL                                         00002860
16 NSTCUR(I)=1                                              00002870
    CALL CUCAL1(NSTCUR,N,ICUM)                                00002880
    DO 17 I=1,MXPL                                         00002890
    IP=IDEXP(I)+2                                           00002900
    DO 18 K=1,4                                             00002910
18 EXPLCU(K,I)=PTCUMX(IP,K)                                00002920
17 CONTINUE                                                 00002930
C
C   DEFINE THE PACKING BASE                                 00002940
    CALL BASE                                                 00002950
C
C   INITIALIZE INNER (TUNNEL) ITERATION COUNTER AND ORIGINAL 00002970
C   ARTIFICIAL TUNNEL BOUNDARIES                           00002980
C   ITIN = 1                                                 00002990
    CALL CHANNEL(MAXSTA, ICFLAG)                            00003000
    IF (ICFLAG .GT. 0) GO TO 10                            00003010
C
C   START INNER ITERATION CALCULATIONS                      00003030
20 CONTINUE                                                 00003040
C   INITIALIZE AUXILIARY ARRAYS                            00003050
    DO 30 NYF = 1,MXYEAR                                  00003060
    UNE(NYF) = 0.0                                         00003070
    NUMACC(NYF) = 0                                         00003080
    DO 30 I=1,8                                           00003090
    RESCAP(NYF,I) = 0.0                                    00003100
30 ISOL(NYF,I) = 0                                         00003120
C
C   WRITE(6,926)                                            00003130
926 FORMAT(1H1//)
C
C   ***** S E C T I O N 2 *****

C
C   SET BASE YEAR ORIGIN STATE                            00003220
    DO 40 I=1,MXPL                                         00003230
    NSTATE(I) = 0                                           00003240
40 CONTINUE                                                 00003250
    OBJORI = 0.0                                           00003260
    NORI = 1                                               00003270
C
C   SET BEGINNING YEAR                                     00003280
    NYCR = 1                                               00003290
    NYOR = 1                                               00003300
    KYEAR = IBASR + NYCR                                 00003310
    WRITE(6,926)                                            00003320
    WRITE(6,900) KYEAR                                    00003330
C
C   READ SEQUENTIAL FILES FOR THIS YEAR                  00003340
    REWIND 22                                             00003350
    REWIND 24                                             00003360
    REWIND 27                                             00003370
    DO 50 NS = 1,4                                         00003380
50

```

```

MAXK = MXELDC(NYCR,NS) 00003410
READ(22) (ELDC(NS,KPOINT),KPOINT=1,1250) 00003420
READ(27) ((PTCUM(IP,JCUM,NS),IP=1,201),JCUM=1,8) 00003430
CONTINUE 00003440
50 READ(24,902) ((LORDER(JBLOCK,KATTR),KATTR=1,3),JBLOCK=1,420) 00003450
READ(24,921) (MAINS(IP),IP=1,200) 00003460
- DEFINE THE FIRST YEAR ORIGINAL TUNNEL. SET TUNNEL FLAG 00003470
    TO INDICATE THAT THIS IS THE ORIGINAL TUNNEL DEFINITION. 00003480
IFLTUN=-1 00003490
SET ACCEPTED STATE COUNTER 00003500
NACCEP = 0 00003510
DEFINE TUNNEL 00003520
60 CALL TUNNEL(NSTATE,NYCR,IFLTUN) 00003530
    GENERATE THE STATES FOR THE FIRST YEAR. STORE THEM IN THE 00003540
    "LIST" ARRAY IN UNPACKED FORM.
    CALL STAGEN(NSTCNT,CUMLOL,NYCR) 00003550
    INITIALIZE AUXILIARY VARIABLES 00003560
NSTOLD = 0 00003570
DOLLP = 1.0 00003580
    CHECK THE STATES IN "LIST" ARRAY FOR PW-LOLP (IF NEEDED) 00003590
    AND FATHOMING. CALCULATE THE OBJECTIVE FUNCTION FOR THE 00003600
    ACCEPTED STATES. 00003610
DO 100 NL=1,NSTCNT 00003620
WRITE(11,903) NYCR,NL 00003630
    COPY CURRENT STATE FROM THE "LIST" ARRAY INTO "NSTCUR" 00003640
    DO 70 I=1,MXPL 00003650
70 NSTCUR(I) = LIST(I,NL) 00003660
    RETRIEVE THE CUMULANT LOLP FROM THE CUMLOL ARRAY. 00003670
    CULOLP = CUMLOL(NL) 00003680
        IF CRITICAL LOLP IS LESS THAN 0.001, CALCULATE LOLP 00003690
        WITH THE PW-LINEAR METHOD. REJECT UNACCEPTABLE STATES. 00003700
        IF(CULOLP.GT.0.001) GO TO 80 00003710
        CALL FLOLP(NSTCUR, CULOLP, NSTOLD, DOLLP, NYCR) 00003720
        IF (CULOLP .GT. CLOLP) GO TO 100 00003730
        CURRENT STATE IS ACCEPTED FOR LOLP. CALCULATE ITS 00003740
        OPERATING COST. 00003750
80 CALL OPERCO(NSTCUR, NYCR, OPEC,IRES) 00003760
    CALCULATE THE OBJECTIVE FUNCTION 00003770
    CALL OBJFUN(NSTCUR, NYCR, OPEC, NSTATE, CAPCO, OBJORI, OBJ, 00003780
    + SALV,RESCAP,IRES) 00003790
    WRITE(11,*) (NSTCUR(I),I=1,MXPL) 00003800
    WRITE(11,*) NYCR,OPEC,CAPCO,OBJORI,OBJ 00003810
        EXERCISE FATHOMING 00003820
        CALL FATHOM(NSTCUR, NYCR, OBJ, IFATH) 00003830
        IF (IFATH .EQ. 0) GO TO 100 00003840
        REACHING THIS POINT MEANS THAT THE STATE CONSIDERED 00003850
        IS ACCEPTABLE. INCREMENT THE ACCEPTED STATES COUNTER. 00003860
NACCEP = NACCEP+1 00003870
        PACK ACCEPTED STATE. 00003880
CALL PACK(NSTCUR, NST) 00003890
        STORE ACCEPTED STATE ATTRIBUTES IN CORE. 00003900
WRITE(11,*) NST,NORI,CULOLP,OPEC,CAPCO,OBJ 00003910
NST2(1,NACCEP) = NST 00003920
NST1(NACCEP) = NST 00003930
NST2(2,NACCEP) = NORI 00003940
STN2(1,NACCEP) = CULOLP 00003950
00003960
00003970
00003980
00003990
00004000
00004010
00004020
00004030
00004040
00004050
00004060
00004070
00004080

```

```

STN2(2,NACCEP) = OPEC          00004090
STN2(3,NACCEP) = CAPCO         00004100
STN2(4,NACCEP) = OBJ           00004110
STN1(NACCEP) = OBJ             00004120
00004130
C      STORE THE CURRENT STATE LOLP FOR OLD LOLP USAGE        00004140
NSTOLD = NST                  00004150
OLOLP = CULOLP                00004160
00004170
C 100 CONTINUE                 00004180
C      CHECK IF THE NUMBER OF ACCEPTED STATES IS ADEQUATE.    00004190
IF(NACCEP.GT.MINNSNU(NYCR)) GO TO 150
C      THE NUMBER OF STATES IS TOO SMALL. INCREASE THE ARTIFITAL
TUNNEL WIDTH                   00004200
CALL ADJUST(NYCR,IFLTUN)       00004210
00004220
IF(IFLTUN.GT.0) GO TO 60
C      REACHING THIS POINT MEANS THAT THE ARTIFITAL TUNNELS
DO NOT CONSTRAIN STATE GENERATION. 00004230
00004240
WRITE(6,914) KYEAR            00004250
C      IF NO STATES WERE ACCEPTED IN THE FIRST YEAR, REDIFINE
THE INPUT DATA FOR THIS ITERATION. 00004260
00004270
IF(NACCEP.LT.1) GO TO 480
00004280
00004290
00004300
00004310
00004320
00004330
C      STORE THE NUMBER OF ACCEPTED STATES IN NUMACC ARRAY FOR
CURRENT YEAR                  00004340
150 NUMACC(NYCR) = NACCEP      00004350
C      STORE ACCEPTED STATES ATTRIBUTES ON DISK.               00004360
IDAF = 1                        00004370
WRITE(14*IDAF) NST2            00004380
IDAF=1                         00004390
WRITE(15*IDAF) STN2            00004400
00004410
WRITE(6,909) NACCEP,NYCR      00004420
WRITE(11,904) NYCR,NACCEP     00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
C 200 NYCR = NYOR+1            00004530
KYEAR = IBASYSR + NYCR        00004540
WRITE(6,900) KYEAR             00004550
C      READ SEQUENTIAL FILES FOR YEAR NYCR.
DO 210 NS = 1,4                00004560
MAXK = MXELDC(NYCR,NS)         00004570
READ(22) ((ELDC(NS,KPOINT),KPOINT=1,1250) 00004580
READ(27) ((PTCUM(IP,JCUM,NS),IP=1,201),JCUM=1,8) 00004590
210 CONTINUE                   00004600
READ(24,902) ((LORDER(JBLOCK,KATTR),KATTR=1,3),JBLOCK=1,420) 00004610
READ(24,921) (MAINS(IP),IP=1,200) 00004620
C      SET TUNNEL FLAG FOR ORIGINAL TUNNEL DEFFINITION
IFLTUN=-1                      00004630
00004640
00004650
C 220 MAXNST = NUMACC(NYOR)    00004660
C      SET STATE COUNTER          00004670
NACCEP = 0                      00004680
00004690
C DO 300 N=1,MAXNST            00004700
C      IDENTIFY THE CURRENT ORIGIN STATE AND ITS OBJECTIVE
FUNCTION.                      00004710
NSTORI = NST1(N)                00004720
OBJORI = STN1(N)                00004730
C      INITIALIZE ORIGIN LOLP FOR OLD LOLP USAGE.           00004740
OLOLP = 1.0                      00004750
00004760

```

```

NSTOLD = NSTORI
C CALL UNPACK(NSTORI,NSTATE)
C WRITE(11,*)(NSTATE(I),I=1,MXPL)
C     SET TUNNELS
C CALL TUNNEL(NSTATE, NYCR,IFLTUN)
C
C     GENERATE STATES IN STAGE NYCR, ORIGINATING FROM NSTATE
C     AND WITHIN THE DEFINED TUNNELS.
C
C CALL STAGEN(NSTCNT,CUMLOL,NYCR)
C IF(NSTCNT.LT.1) GO TO 300
C
C     NOT BRANCHING TO 300 MEANS THAT AT LEAST ONE STATE
C     WAS GENERATED IN THE CURRENT STAGE ((YEAR "NYCR")) FROM
C     THE ORIGIN STATE. CHECK WHETHER ANY OF THE CURRENT
C     STATES HAS ALREADY BEEN GENERATED THROUGH A DIFFERENT
C     ORIGIN STATE. FOR ANY STATE THAT HAS BEEN GENERATED
C     BEFORE CHECK THE STATE'S LOLP AND IF ACCEPTABLE
C     THAN UPDATE "NCR1" (THE ORIGIN STATE SEQUENCE
C     NUMBER) TO POINT TO THE ORIGIN STATE THAT RESULTS IN
C     THE SMALLEST CURRENT STATE OBJECTIVE FUNCTION (I.E.
C     EXERCISE THE BELMAN'S PRINCIPLE OF OPTIMALITY). IF THE
C     CURRENT STATE HAS NEVER BEFORE BEEN GENERATED CALCULATE
C     PW-LOLP (IF NEEDED) AND THE STATE'S OBJECTIVE FUNCTION.
C
C     STATES WITH UNACCEPTABLE LOLP ARE NOT DISCARDED BUT
C     INSTEAD THEY ARE ASSIGNED LARGE OBJECTIVE FUNCTIONS.
C     THIS IS DONE SO THAT WHEN THESE STATES ARE GENERATED
C     AGAIN FROM A DIFFERENT ORIGIN STATE, THEIR LOLP NEED NOT
C     BE RECALCULATED. THESE STATES WILL BE FATHOMED WHEN
C     NO MORE STATES CAN BE GENERATED IN THE CURRENT YEAR.
C
C DO 280 NL=1,NSTCNT
C     DEFINE THE CURRENT STATE.
C DO 230 I=1,MXPL
230 NSTCUR(I) = LIST(I,NL)
C     PACK CURRENT STATE.
C CALL PACK(NSTCUR,NST)
C IF (NL.EQ.1) WRITE(11,906) NL,(NSTCUR(I),I=1,MXPL)
C     IF THIS IS NOT THE FIRST TIME STATES IN STAGE NYCR ARE
C     SIMULATED, CHECK IF THE SAME STATE WAS SIMULATED BEFORE.
C IF (N.EQ.1) GO TO 240
C     SEARCH PREVIOUSLY GENERATED STATES IN NYCR STAGE
C CALL SEARCH(NST,NST2,NUMACC,NYCR,NACCEP,NTH)
C IF(NTH.EQ.0) GO TO 240
C     STATE WAS PREVIOUSLY CONSIDERED. RECALCULATE OBJ.
C     CHECK FOR LOLP
C IF (STN2(1,NTH) .GT. CLOLP) GO TO 280
C     THIS STATE'S LOLP IS ACCEPTABLE
C     CALCULATE OBJECTIVE FUNCTION FROM THE NEW ORIGIN STATE
C OPEC = STN2(2,NTH)
C CALL OBJFUN(NSTCUR, NYCR, OPEC, NSTATE, CAPCO, OBJRI,
+OBJ, SALV, RESCAP, IRES)
C
C IF (OBJ .GT. STN2(4,NTH)) GO TO 280
C     NEW OBJ IS SMALLER. REPLACE THE CURRENT STATE' OLD
C     ATTRIBUTES WITH THE NEW ONES.
C NST2(2,NTH) = N
C STN2(3,NTH) = CAPCO
C STN2(4,NTH) = OBJ
C GO TO 280
C
C     ASSIGN THE CUMULANT LOLP TO CULOLP
240 CULOLP = CUMLOL(NL)
C
C     CHECK LOLP CALCULATED THROUGH THE PW-LINEAR METHOD, ONLY

```

```

C      IF THE CRITICAL LOLP IS LESS THAN 0.001.          00005450
      IF(CLOLP.GT.0.001) GO TO 250                      00005460
      CALL FLOLP(NSTCUR, CULOLP, NSTOLD, OLOLP, NYCR)   00005470
      IF (CULOLP .GT. CLOLP) GO TO 260                 00005480
C      CALCULATE OBJECTIVE FUNCTION                   00005490
250  CALL OPERCO(NSTCUR, NYCR, OPEC,IRES )           00005500
      CALL OBJFUN(NSTCUR, NYCR, OPEC, NSTATE, CAPCO, OBJORI,
      +OBJ, SALV, RESCAP,IRES)                         00005510
      GO TO 270                                         00005520
260  OPEC = UBOUND*2.0                                00005530
      CAPCO = OPEC                                     00005540
      OBJ = OPEC                                      00005550
270  CONTINUE                                         00005560
C      REACHING THIS POINT MEANS THAT THE STATE CONSIDERED 00005570
      IS ACCEPTABLE EXCEPT FOR FATHOMING. STATES WITH BAD 00005580
      RELIABILITY WILL BE FATHOMED.                     00005590
CCCCC
C      NACCEP = NACCEP+1                               00005600
      NTH = NUMACC(NYCR)+NACCEP                        00005610
      NST2(1,NTH) = NST                                00005620
      NST2(2,NTH) = N                                   00005630
      STN2(1,NTH) = CULOLP                           00005640
      STN2(2,NTH) = OPEC                             00005650
      STN2(3,NTH) = CAPCO                           00005660
      STN2(4,NTH) = OBJ                            00005670
C      STORE LOLP FOR OLD LOLP USAGE.                00005680
      NSTOLD = NST                                     00005690
      OLOLP = CULOLP                                 00005700
C      280 CONTINUE                                     00005710
300  CONTINUE                                         00005720
C      EXERCISE FATHOMING FOR THE STATES IN YEAR NYCR 00005730
      WRITE(11,912) NYCR,NACCEP                        00005740
      IF (NACCEP.LT.1) GO TO 355                      00005750
C      SET THE RANGE OF STATES TO BE CHECKED FOR FATHOMING. 00005760
      NTH1 = NUMACC(NYCR) + 1                         00005770
      NTH2 = NUMACC(NYCR) + NACCEP                    00005780
      NACCEP = 0                                       00005790
      NPOSTN=NUMACC(NYCR)                            00005800
      DO 350 NTH = NTH1,NTH2                         00005810
      NST = NST2(1,NTH)                            00005820
      OBJ = STN2(4,NTH)                           00005830
      CALL UNPACK(NST, NSTCUR)
      CALL FATHOM(NSTCUR, NYCR, OBJ, IFATH)          00005840
      IF (IFATH.EQ. 0) GU TO 350                     00005850
CCCCC
C      STORE THE ACCEPTED STATES IN THE NST2 AND STN2 ARRAYS. 00005860
      NOTE THAT THE INDEX FOR THE ACCEPTED STATES IS LESS OR 00005870
      EQUAL TO THE INDEX OF ALL STATES,(I.E.NACCEP.LE.NTH). 00005880
      NACCEP = NACCEP + 1                           00005890
      NPOSTN=NPOSTN+1                               00005900
      NST2(1,NPOSTN) = NST                         00005910
      NST2(2,NPOSTN) = NST2(2,NTH)                  00005920
      STN2(1,NPOSTN) = STN2(1,NTH)                  00005930
      STN2(2,NPOSTN) = STN2(2,NTH)                  00005940
      STN2(3,NPOSTN) = STN2(3,NTH)                  00005950
      STN2(4,NPOSTN) = OBJ                         00005960
350  CONTINUE                                         00005970
C      RECORD THE NUMBER OF STATES ACCEPTED AFTER FATHOMING 00005980
355  NUMACC(NYCR) = NUMACC(NYCR) + NACCEP          00005990
      NACCEP = NUMACC(NYCR)                         00006000
      WRITE(6,909) NACCEP,NYCR                      00006010
      WRITE(11,909) NACCEP,NYCR                      00006020
C      CHECK IF THE NUMBER OF STATES ACCEPTED IN YEAR NYCR 00006030
      IS ADEQUATE.                                  00006040
C      IF(NACCEP.GT.MINSNU(NYCR)) GO TO 360          00006050
C      THE NUMBER OF STATES IS TOO SMALL.ADJUST ARTIFICIAL TUNNEL 00006060
                                          00006070
                                          00006080
                                          00006090
                                          00006100
                                          00006110
                                          00006120

```

```

CALL ADJUST(NYCR,IFLTUN)
C   IF(IFLTUN.GT.0) GO TO 220
      REACHING THIS POINT MEANS THAT THE ARTIFITIAL TUNNELS
      DO NOT CONSTRAIN STATE GENERATION.
      WRITE(6,914) KYEAR
      IF(NACCEP.LT.1) GO TO 480
360 IDAF = NYCR
C     STORE THE ACCEPTED STATES ON DISK.
      WRITE(14>IDAF) NST2
      IDAF=NYCR
      WRITE(15>IDAF) STN2

C     STORE THE STATES FOR YEAR NYCR IN THE NST1 AND STN1
C     ARRAYS, SO THAT THEY CAN BE USED AS ORIGIN STATES IN
C     NEXT YEAR'S CALCULATIONS.
      DO 365 N=1,NACCEP
      NST1(N) = NST2(1,N)
365 STN1(N) = STN2(4,N)

C     GO TO NEXT STAGE
      NYOR = NYOR+1
      IF (NYOR .LT. MXYEAR) GO TO 200

C     ***** S E C T I O N 4 *****


---


C     REACHING THIS POINT MEANS ALL STAGES(YEARS) WERE EXAMINED.
C     FIND THE MINIMUM OBJECTIVE FUNCTION AND TRACE BACK THE
C     OPTIMUM SOLUTION
      OBJMIN = UBOUND
      DO 370 N=1,NACCEP
      IF (OBJMIN .LT. STN1(N)) GO TO 370
      OBJMIN = STN1(N)
      NFIND = N
370 CONTINUE
C     DO 380 IATTR=1,4
      OPTIMA(MXYEAR,IATTR) = STN2(IATTR,NFIND)
      IF (IATTR.GT.2) GO TO 380
      NOPTIM(MXYEAR,IATTR) = NST2(IATTR,NFIND)
380 CONTINUE
      WRITE(6,911) STN1(NFIND)

C     WRITE(11,910) (NST2(I,NFIND),I=1,2), (STN2(I,NFIND),I=1,4)
C     TRACE BACK THE OPTIMUM SOLUTION
      NSTOP = MXYEAR-1
      DO 390 NYF=1,NSTOP
      NYB = NSTOP - NYF + 1
      IDAF = NYB
      READ(14>IDAF) NST2
      IDAF=NYB
      READ(15>IDAF) STN2

C     NORI = NOPTIM(NYB+1,2)
      DO 385 IATTR=1,4
      OPTIMA(NYB,IATTR) = STN2(IATTR,NORI)
      IF(IATTR.GT.2) GO TO 385
      NOPTIM(NYB,IATTR) = NST2(IATTR,NORI)
385 CONTINUE
390 CONTINUE

```

00006130
00006140
00006150
00006160
00006170
00006180
00006190
00006200
00006210
00006220
00006230
00006240
00006250
00006260
00006270
00006280
00006290
00006300
00006310
00006320
00006330
00006340
00006350
00006360
00006370
00006380
00006390
00006400
00006410
00006420
00006430
00006440
00006450
00006460
00006470
00006480
00006490
00006500
00006510
00006520
00006530
00006540
00006550
00006560
00006570
00006580
00006590
00006600
00006610
00006620
00006630
00006640
00006650
00006660
00006670
00006680
00006690
00006700
00006710
00006720
00006730
00006740
00006750
00006760
00006770
00006780
00006790
00006800

```

C      PRINT ITERATION NUMBER                               00006810
      WRITE(6,917) ITIN                                00006820
      WRITE(11,917) ITIN                                00006830
C      WRITE THE OPTIMUM SOLUTION                         00006840
      I28 = (ITIN-1) * 20 + 1                            00006850
      WRITE(6,898)
898 FORMAT(/5X,4HLOLP,5X,8HOP. COST,3X,9HCAP. COST,2X,
      112HOBJ FUNCTION,3X,10HPLANT MIX/)                00006860
      DO 400 NYF=1,MXYEAR                               00006880
      NST = NOPTIM(NYF,1)                                00006890
      CALL UNPACK(NST,NSTCUR)
      DO 395 I = 1,MXPL                                00006900
      ISOL(NYF,I) = NSTCUR(I)                           00006910
      SOL(NYF,1) = OPTIMA(NYF,1)                         00006920
      SOL(NYF,2) = OPTIMA(NYF,4)                         00006930
      WRITE(6,899) (OPTIMA(NYF,IATTR),IATTR=1,4),(NSTCUR(I),I=1,MXPL) 00006940
899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00006950
C      WRITE(28*I28,913)(ISOL(NYF,I),I=1,8),SOL(NYF,1),SOL(NYF,2) 00006960
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00006970
C      WRITE(28*I28,913)(ISOL(NYF,I),I=1,8),SOL(NYF,1),SOL(NYF,2) 00006980
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00006990
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007000
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007010
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007020
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007030
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007040
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007050
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007060
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007070
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007080
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007090
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007100
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007110
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007120
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007130
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007140
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007150
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007160
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007170
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007180
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007190
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007200
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007210
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007220
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007230
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007240
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007250
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007260
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007270
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007280
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007290
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007300
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007310
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007320
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007330
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007340
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007350
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007360
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007370
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007380
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007390
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007400
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007410
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007420
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007430
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007440
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007450
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007460
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007470
C      899 FORMAT(5X,F7.5,2X,FB.3,3X,F8.3,3X,F9.3,5X,8I3) 00007480

***** S E C T I O N S *****

C      CHECK WHETHER THE OPTIMUM SOLUTION IS CONSTRAINED BY THE
      ARTIFICIAL LOWER OR UPPER TUNNEL BOUNDARIES FOR EACH
      YEAR IN THE STUDY PERIOD. ADJUST BOUNDARIES AS NEEDED.
      INITIALIZE THE TUNNEL VIOLATION FLAG.
      NYFLAG = 21
      THE YEAR LOOP GOES BACKWARDS (I.E. FROM MXYEAR TO 1)
      SO THAT THE LOWEST YEAR OF TUNNEL VIOLATIONS CAN BE
      IDENTIFIED.

      DO 430 NYF=1,MXYEAR
      NYB = MXYEAR-NYF +1
      LOFLAG = 0
      NUFLAG = 0
      NST = NOPTIM(NYB,1)
      CALL UNPACK(NST,NSTCUR)
      C      CHECK AND IF NEEDED ADJUST THE ARTIFICIAL LOWER BOUNDARY
      DO 420 I=1,MXPL
      IDP = INDEXP(I)+2
      IF ( LOWB(NYB,I).LE.MINUN(NYB,IDP)) GO TO 410
      IF ( NSTCUR(I).LE.MINUN(NYB,IDP)) GO TO 410
      IF ( NSTCUR(I).NE.LOWB(NYB,I)) GO TO 410
      REACHING HERE MEANS THE SOLUTION IS CONSTRAINED.
      LOWER THE LOWER BOUNDARY.
      LOWB(NYB,I) = LOWB(NYB,I) - IPCH(I)
      ADJUST THE ARTIFICIAL LOWER BOUNDARY TO COMPLY WITH THE
      REAL CONSTRAINTS.
      IF (LOWB(NYB,I).LT.MINUN(NYB,IDP)) LOWB(NYB,I) = MINUN(NYB,IDP)
      ACTIVATE THE BOUNDARY VIOLATION CONSTRAINT.
      LOFLAG = 1
      ADJUST THE ARTIFICIAL UPPER BOUNDARY TO COMPLY WITH THE

```

```

C      MAXIMUM WIDTH CONSTRAINT.          00007490
C      NUPB(NYB,I) = LOWB(NYB,I) + IWIDTH(I) -1 00007500
C      IF (NUPB(NYB,I).GT.MAXUN(NYB, IDP)) NUPB(NYB,I) = MAXUN(NYB, IDP) 00007510
C      GO TO 420 00007520
C
C 410 CONTINUE 00007530
C      CHECK AND IF NEEDED ADJUST THE ARTIFICIAL UPPER BOUNDARY 00007540
C      IF (NUPB(NYB,I).GE.MAXUN(NYB, IDP)) GO TO 420 00007550
C      IF (NSTCUR(I).GE.MAXUN(NYB, IDP)) GO TO 420 00007560
C      IF (NSTCUR(I).NE.NUPB(NYB,I)) GO TO 420 00007570
C      REACHING HERE MEANS THE SOLUTION IS CONSTRAINED. 00007580
C      INCREASE THE UPPER BOUNDARY. 00007590
C      NUPB(NYB,I) = NUPB(NYB,I) + IPCH(I) 00007600
C      ADJUST THE ARTIFICIAL UPPER BOUNDARY TO COMPLY WITH THE 00007610
C      REAL CONSTRAINTS. 00007620
C      IF(NUPB(NYB,I).GT.MAXUN(NYB, IDP)) NUPB(NYB,I) = MAXUN(NYB, IDP) 00007630
C      ACTIVATE THE BOUNDARY VIOLATION CONSTRAINT. 00007640
C      NUFLAG = I 00007650
C      ADJUST THE ARTIFICIAL LOWER BOUNDARY TO COMPLY WITH THE 00007660
C      MAXIMUM WIDTH CONSTRAINT. 00007670
C      LOWB(NYB,I) = NUPB(NYB,I) - IWIDTH(I)+1 00007680
C      IF (LOWB(NYB,I).LT.MINUN(NYB, IDP)) LOWB(NYB,I) = MINUN(NYB, IDP) 00007690
C
C 420 CONTINUE 00007700
C      IF (LOFLAG.GT.0) WRITE(6,915) NYB 00007710
C      IF (NUFLAG.GT.0) WRITE(6,916) NYB 00007720
C
C      RECORD THE FIRST YEAR THAT ARTIFICIAL BOUNDS CONSTRAINED 00007730
C      THE SOLUTION 00007740
C      IF (LOFLAG.LT.1.AND.NUFLAG.LT.1 ) GO TO 430 00007750
C      NYFLAG = NYB 00007760
C
C 430 CONTINUE 00007770
C      ADJUST THE ARTIFICIAL TUNNEL BOUNDARIES SO THAT THEY ARE 00007780
C      AN INCREASING FUNCTION OF THE YEARS. 00007790
C      DO 450 I = 1,MXPL 00007800
C      IDP = IDEXP(I)+2 00007810
C      DO 440 NYF=2,MXYEAR 00007820
C      NYB = MXYEAR - NYF + 1 00007830
C      IF(LOWB(NYB+1,I).GE.LOWB(NYB,I)) GO TO 440 00007840
C      LOWB(NYB,I) = LOWB(NYB+1,I) 00007850
C      NUPPER = LOWB(NYB,I) + IWIDTH(I) -1 00007860
C      NUPB(NYB,I) = MINO(NUPPER,MAXUN(NYB, IDP)) 00007870
C      NUPB(NYB,I)=NUPB(NYB+1,I) 00007880
C
C 440 CONTINUE 00007890
C
C 450 CONTINUE 00007900
C      ITERATE THROUGH DP AT MOST ITMAX TIMES 00007910
C      IF (NYFLAG .GT. MXYEAR) GO TO 500 00007920
C      ITIN = ITIN + 1 00007930
C      IF (ITIN.GE.ITMAX) GO TO 470 00007940
C
C      WRITE(6,923) 00007950
C      DO 455 NYF=1,MXYEAR 00007960
C      KYEAR = IBASYR + NYF 00007970
C      WRITE(6,924) KYEAR,(LOWB(NYF,I),I=1,MXPL) 00007980
C      455 WRITE(6,925) (NUPB(NYF,I),I=1,MXPL) 00007990
C      WRITE(6,926) 00008000
C
C      IF(NYFLAG.EQ.1) GO TO 20 00008010
C      SET THE ORIGIN YEAR FOR NEW INNER ITERATION. 00008020
C      NYOR = NYFLAG - 1 00008030
C      REWIND AND RE-READ THE FIRST NYOR RECORDS OF THE 00008040
C      SEQUENTIAL FILES. 00008050
C      REWIND 22 00008060
C      REWIND 24 00008070
C      REWIND 27 00008080
C      DO 465 NYF=1,NYOR 00008090
C
C

```

```

DO 460 NS = 1,4          00008170
MAXK = MXELDC(NYF,NS)    00008180
READ(22) ((ELDC(NS,KPOINT),KPOINT=1,1250) 00008190
READ(27) ((PTCUM(IP,JCUM,NS),IP=1,201),JCUM=1,8) 00008200
460 CONTINUE               00008210
READ(24,902) ((LORDER(JBLOCK,KATTR),KATTR=1,3),JBLOCK=1,420) 00008220
READ(24,921) (MAINS(IP),IP=1,200) 00008230
465 CONTINUE               00008240
IDAF=NYOR                 00008250
READ(14*IDAF) NST2        00008260
IDAF=NYOR                 00008270
READ(15*IDAF) STN2        00008280
NACCEP=NUMACC(NYOR)       00008290
DO 475 N=1,NACCEP         00008300
NST1(N)=NST2(1,N)         00008310
475 STN1(N)=STN2(4,N)     00008320
NYCR=NYOR+1                00008330
DO 476 NYF=NYCR,MYEAR    00008340
476 NUMACC(NYF)=0          00008350
C   GO TO 200              00008360
C
470 WRITE(6,918) ITMAX    00008370
GO TO 500                 00008380
480 WRITE(6,922) KYEAR,NACCEP 00008390
ITOUT = 0                  00008400
GO TO 10                  00008410
500 CONTINUE               00008420
C   IF (ISTOP.LE.0 ) GO TO 10 00008430
C
600 STOP                  00008440
900 FORMAT(6H YEAR ,I4)    00008450
902 FORMAT(125(I3,I2,I3))  00008460
903 FORMAT(5H YEAR ,I4,7H STATE ,I4) 00008470
904 FORMAT(5H YEAR ,I4,27H NUMBER OF ACCEPTED STATES ,I4) 00008480
906 FORMAT(4H NL=,I4,9H NSTCUR=,10I2) 00008490
908 FORMAT(6H NYCR=,I4,17H FINAL NACCEP IS=, I4) 00008500
909 FORMAT(1H ,I4,37H STATES WERE ACCEPTED AFTER FATHOMING , 00008510
  + 8H IN YEAR ,I4)        00008520
910 FORMAT(1H ,2I10, F11.8, 3F13.2) 00008530
911 FORMAT(1H //40H THE FINAL YEAR OBJECTIVE FUNCTION IS: 00008540
  + ,E13.6)                00008550
912 FORMAT( * THE UNFATHOMED STATES FOR YEAR ,I4, * ARE*,I4) 00008560
913 FORMAT(8I4,2E14.7)      00008570
914 FORMAT(42H ARTIFICIAL BOUNDARIES ELIMINATED IN YEAR ,I5) 00008580
915 FORMAT(40H **** SOLUTION AGAINST LOWER BOUNDARY , 00008590
  + 9H IN YEAR ,I4,7H ****) 00008600
916 FORMAT(40H +++++ SOLUTION AGAINST UPPER BOUNDARY , 00008610
  + 9H IN YEAR ,I4,7H +++) 00008620
917 FORMAT(29H THE NUMBER OF ITERATION IS:,I4) 00008630
918 FORMAT(10H MORE THAN ,I4, 00008640
  + 41H ITERATIONS HAVE BEEN RUN. PROGRAM STOPS ) 00008650
919 FORMAT(29H THE NEW BOUNDARIES FOR YEAR,14,7H ARE : ,20I3) 00008660
920 FORMAT(35H NEW UPPER BOUND FOR FATHOMING IS ,F13.2/) 00008670
921 FORMAT(20I2)            00008680
922 FORMAT(38H THE NUMBER OF STATES DEFINED IN YEAR ,I5, 00008690
  +4H IS ,13//19H PROBABLE REASONS: / 00008700
  +34H A. SMALL MAXIMUM RESERVE MARGIN. / 00008710
  +55H B. THE SIZE OF THE EXPANSION PLANT TYPES SELECTED IS / 00008720
  +47H TOO SMALL TO PROVIDE ADEQUATE RELIABILITY. / 00008730
  +41H **** RERUN WITH PROPER ADJUSTMENTS ****) 00008740
923 FORMAT(1H //36H THE NEW LOWER AND UPPER BOUNDS ARE: //) 00008750
924 FORMAT(7H YEAR ,I4,2X,20I4) 00008760
925 FORMAT(7H ,6X,20I4) 00008770
END                         00008780
C

```

SUBROUTINE ADJUST(NYCR, IFLTUN)

		00008850
		00008860
		00008870
		00008880
		00008890
		00008900
		00008910
		00008920
		00008930
		00008940
		00008950
		00008960
		00008970
		00008980
		00008990
		00009000
		00009010
		00009020
		00009030
		00009040
		00009050
		00009060
		00009070
		00009080
		00009090
		00009100
		00009110
		00009120
		00009130
		00009140
		00009150
		00009160
		00009170
		00009180
		00009190
		00009200
		00009210
		00009220
		00009230
		00009240
		00009250
		00009260
		00009270
		00009280
		00009290
		00009300
		00009310
		00009320
		00009330
		00009340
		00009350
		00009360
		00009370
		00009380
		00009390
		00009400
		00009410
		00009420
		00009430
		00009440
		00009450
		00009460
		00009470
		00009480
		00009490
		00009500
		00009510
		00009520

```

C COMMON /TBL/ TINT(120),TY(120)          00009530
C DOUBLE PRECISION TINT,TY               00009540
C
C COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),    00009550
C   * HYEXPN(11),HYSCHD(20,8),PSEXPN(11),    00009560
C   * PSSCHD(20,8),SYSCUM(4)                 00009570
C
C IDENTIFY THE CURRENT CALENDAR YEAR      00009580
C KYEAR = IBASYR + NYCR                00009590
C CHECK IF LOWER BOUND HAS REACHED THE MINIMUM 00009600
C IFLTUN=0                                00009610
C DO 15 I=1,MXPL                         00009620
C LBCR=LOWB(NYCR,I)                      00009630
C LBOLD(I)=LBCR                           00009640
C IDP=IDEXP(I)+2                         00009650
C MN=MINUN(NYCR, IDP)                    00009660
C IF(NYCR.NE.1) GO TO 10                  00009670
C LB=MN                                  00009680
C GO TO 11                                00009690
C 10 LB=LOWB(NYCR-1,I)                   00009700
C 11 IF(LBCR.GT.LB) GO TO 12            00009710
C   GO TO 15                               00009720
C 12 ITEMP=LBCR-1                        00009730
C   IF(ITEMP.GE.MN) GO TO 13            00009740
C   GO TO 15                               00009750
C 13 IFLTUN=1                            00009760
C   LOWB(NYCR,I)=ITEMP                  00009770
C 15 CONTINUE                            00009780
C
C IF(IFLTUN.LE.0) GO TO 20              00009790
C WRITE(6,900) KYEAR, (LOWB(NYCR,I), I=1,MXPL) 00009800
C 900 FORMAT(33H, NEW TUNNEL LOWER BOUND FOR YEAR,IS,
C   *4H IS,8I3/)                         00009810
C RETURN                                 00009820
C
C 20 DO 25 I=1,MXPL                      00009830
C   NBCR=NUPB(NYCR,I)                   00009840
C   IDP=IDEXP(I)+2                     00009850
C   MX=MAXUN(NYCR, IDP)                00009860
C   NUBOLD(I)=NBCR                     00009870
C   IF(NBCR.GT.MX) NUBOLD(I)=MX       00009880
C   IF(NYCR.NE.MXYEAR) GO TO 21      00009890
C   NB=MX                                00009900
C   GO TO 22                               00009910
C 21 NB=NUPB(NYCR+1,I)                  00009920
C 22 IF(NB.GT.NBCR) GO TO 23           00009930
C   GO TO 25                               00009940
C 23 ITEMP=NBCR+1                       00009950
C   IF(ITEMP.LE.MX) GO TO 24           00009960
C   GO TO 25                               00009970
C 24 IFLTUN=2                            00009980
C   NUPB(NYCR,I)=ITEMP                 00009990
C 25 CONTINUE                            00010000
C   IF(IFLTUN.LE.0) GO TO 30           00010010
C
C WRITE(6,901) KYEAR, (NUPB(NYCR,I), I=1,MXPL) 00010020
C 901 FORMAT(33H, NEW TUNNEL UPPER BOUND FOR YEAR,IS,
C   *4H IS,8I3/)                         00010030
C
C 30 RETURN                             00010040

```

```

END
00010210
00010220
00010230
00010240
00010250
00010260
00010270
00010280
00010290
00010300
00010310
00010320
00010330
00010340
00010350
00010360
00010370
00010380
00010390
00010400
00010410
00010420
00010430
00010440
00010450
00010460
00010470
00010480
00010490
00010500
00010510
00010520
00010530
00010540
00010550
00010560
00010570
00010580
00010590
00010600
00010610
00010620
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

SUBROUTINE AREA(Y1,Y2,P,E,CF,CFP,IH,N)

COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1 FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
2 ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3 MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
4 RESMAR,UBOUND
00010330
00010340
00010350
00010360
00010370
00010380
00010390
00010400
00010410
00010420
00010430
00010440
00010450
00010460
00010470
00010480
00010490
00010500
00010510
00010520
00010530
00010540
00010550
00010560
00010570
00010580
00010590
00010600
00010610
00010620
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
1 IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
2 IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
3 MAXUN(20,20),MAXADD(8),MINUN(20,20),
4 NEXPID(20),NSTPRE(20,20),NUPB(20,8),
5 PEAKS(20,4),PLACAL(20,14),PLANT(210,14),
6 RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
7 FATOP(20),SOL(20,2),AUPEAK(20)
00010330
00010340
00010350
00010360
00010370
00010380
00010390
00010400
00010410
00010420
00010430
00010440
00010450
00010460
00010470
00010480
00010490
00010500
00010510
00010520
00010530
00010540
00010550
00010560
00010570
00010580
00010590
00010600
00010610
00010620
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+ MXELDC(20,4),XL(1250)
00010480
00010490
00010500
00010510
00010520
00010530
00010540
00010550
00010560
00010570
00010580
00010590
00010600
00010610
00010620
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

COMMON /CMS/ PTCUM(20,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)
00010510
00010520
00010530
00010540
00010550
00010560
00010570
00010580
00010590
00010600
00010610
00010620
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+ MINTUN(8),NUBOLD(8)
00010520
00010530
00010540
00010550
00010560
00010570
00010580
00010590
00010600
00010610
00010620
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

COMMON /TBL/ TINT(120),TY(120)
00010550
00010560
00010570
00010580
00010590
00010600
00010610
00010620
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

DOUBLE PRECISION TINT,TY
00010590
00010600
00010610
00010620
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
+ HYEXP(11),HYSCHD(20,8),PSEXP(11),
+ PSSCHD(20,8),SYSCUM(4)
00010600
00010610
00010620
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

DOUBLE PRECISION ZLL1,ZLL2
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

DIMENSION COEFF(4)
00010630
00010640
00010650
00010660
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

NCUM=4
00010670
00010680
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

COEFF(1)=SYSCUM(1)
00010690
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

COEFF(2)=SYSCUM(2)
00010700
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

SIGMA=SQRT(SYSCUM(2))
00010710
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

SIGS=SIGMA*SIGMA
00010720
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

DO 10 J=3,NCUM
00010730
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

SIGS=SIGS*SIGMA
00010740
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

10 COEFF(J)=SYSCUM(J)/SIGS
00010750
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

ZLL1=(Y1-COEFF(1))/SIGMA
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

ZLL2=(Y2-COEFF(1))/SIGMA
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

ZLL1=DBLE(ZLL1)
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

ZLL2=DBLE(ZLL2)
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

CF3=COEFF(3)
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

CF4=COEFF(4)
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880

CALL VALUE(ZLL1,CF3,CF4,ZV1)
00010820
00010830
00010840
00010850
00010860
00010870
00010880

CALL VALUE(ZLL2,CF3,CF4,ZV2)
00010830
00010840
00010850
00010860
00010870
00010880

C1=(ZV1+ZV2)/2.0
00010840
00010850
00010860
00010870
00010880

CFP=C1
00010850
00010860
00010870
00010880

X1=Y2-Y1
00010860
00010870
00010880

E=X1*2190.0*P*C1
00010870
00010880

IF(IH.GE.4) GO TO 60
00010880

```

```

X2=HYDR(IH,2)-HYDR(IH,1)*CMULT(N)
IF(IH.EQ.3) X2=HYDR(3,1)
IF(IH.NE.3) GO TO 20
T1=ZV1
T2=ZV2
GO TO 50
20 Q=HYDR(IH,5)
X3=HYDR(IH,1)
R=(X1+X3)/X3
IF(X3.GT.X1) GO TO 30
S=X1/(X1-X3)
GO TO 40
30 S=1.0-(X3/(X3-X1))
40 D1=1.0-Q-R*Q/(1.0-R)
D2=1.0-Q/S
D3=(1.0-S)*Q*Q
D4=(1.0-R)*S
DE=D1*D2*D3/D4
T1=(ZV1*D2-ZV2*Q/(1.0-R))/DE
T2=(ZV2*D1+ZV1*Q*(1.0-S)/S)/DE
50 DFAC=T1-(T1-T2)*(X2/X1)
CF=(T1+DFAC)/2.0
GO TO 70
60 CF=C1
70 CONTINUE
RETURN
END

```

SUBROUTINE BASE

TITLE: PACKING BASE

FUNCTION: THIS ROUTINE DETERMINES THE PACKING BASE VECTOR (DENOTED BY IPABA) USED FOR PACKING THE STATES IN ONEWORD. IPABA(I), I=1,MXPL, IS SET TO 10**(I-1), IF THE MAXIMUM NUMBER OF UNITS FOR ALL EXPANSION PLANT TYPES (MAXUN) IS LESS THAN 10. OTHERWISE, IPABA(I) IS IDENTICAL TO THE PRODUCT OF MAXUN(I) II=1-I. IF IN THIS CASE THE HIGHEST IPABA ENTRY IS GREATER THAN THE MAXIMUM INTEGER SUPPORTED BY THE HOST COMPUTER (HERE TAKEN AS 2.1 * 1000000000), THE USER IS PROMPTED TO REDUCE THE ENTRIES OF MAXUN(I). THE PACKING BASE VECTOR AND THE "PACK" SUBROUTINE ARE CONSTRUCTED IN SUCH A MANNER SO THAT, WHEN THE NEW STATES GENERATED BY THE "STAGEN" ROUTINE ARE PACKED THEY WILL BE IN ASCENDING ORDER.

CALLED BY: MAIN
CALLS TO: NONE

COMMON VARIABLES USED: IBASYR, IDEXP, IPABA, MAXUN,
MXPL, MXYEAR

LOCAL VARIABLES

NAME	DESCRIPTION
I	PLANT TYPE INDEX
IDP	PLANT TYPE INDEX IN ORIGINAL LIST
ITEMP	TEMPORARY ARRAY WITH MAXIMUM UNIT NUMBERS FOR EACH PLANT TYPE IN CURRENT YEAR
KYEAR	CURRENT CALENDAR YEAR

00010890
00010900
00010910
00010920
00010930
00010940
00010950
00010960
00010970
00010980
00010990
00011000
00011010
00011020
00011030
00011040
00011050
00011060
00011070
00011080
00011090
00011100
00011110
00011120
00011130
00011140
00011150
00011160
00011170
00011180
00011190
00011200
00011210
00011220
00011230
00011240
00011250
00011260
00011270
00011280
00011290
00011300
00011310
00011320
00011330
00011340
00011350
00011360
00011370
00011380
00011390
00011400
00011410
00011420
00011430
00011440
00011450
00011460
00011470
00011480
00011490
00011500
00011510
00011520
00011530
00011540
00011550
00011560

LIMIT	MAXIMUM INTEGER SIZE ALLOWED FOR CURRENT VALUE OF BASE VECTOR ENTRY.	00011570
NYF	YEAR INDEX	00011580
NY1	FIRST YEAR FOR WHICH MAXIMUM UNIT LIMITS SHOULD	00011590
COMMON /VRS/	CLDLPL, CLDLPU, CPLOLP, DISRAT, DX, FCR, HOURS, IBASYSR, IMXPL, ITIN, ITMAX, ITOUT, MAXALL, MAXI, MAXINP, MAXOR, MAXPLA, MAXPD, MXALL, MXPL, MXYEAR, RESMAR, UBOUND	00011600 00011610 00011620 00011630 00011640 00011650 00011660 00011670
C	COMMON /ARS/ AVSP(20,4), CAPABS(20), ENEDEM(20,4), EXPLCU(4,8), IDEP(20), IPABA(8), IPCH(8), ISOL(20,8), IWIDTH(8), LORDER(420,3), LOWB(20,8), MAINS(200), MAXUN(20,20), MAXADD(8), MINUN(20,20), NEXPID(20), NSTPRE(20,20), NUPB(20,8), PEAKS(20,4), PLACA(20,14), PLANT(210,14), RESMAX(20), RESMIN(20), SSCUM(4,20), UNE(20), FATOPE(20), SOL(20,2), AUPEAK(20)	00011680 00011690 00011700 00011710 00011720 00011730 00011740 00011750 00011760 00011770 00011780 00011790 00011800 00011810 00011820 00011830 00011840 00011850 00011860 00011870 00011880 00011890 00011900 00011910 00011920 00011930 00011940 00011950 00011960 00011970 00011980 00011990 00012000 00012010 00012020 00012030 00012040 00012050 00012060 00012070 00012080 00012090 00012100 00012110 00012120 00012130 00012140 00012150 00012160 00012170 00012180 00012190 00012200 00012210 00012220 00012230 00012240
DIMENSION ITEMP(8)		
	THE MAXIMUM NUMBER OF UNITS ALLOWED FOR EACH EXPANSION CANDIDATE IS STORED IN THE MAXUN ARRAY ENTRIES FOR THE LAST YEAR (MXYEAR) OF THE PLANNING HORIZON.	00011970 00011980 00011990 00012000
	CHECK IF ANY OF THE VALUES OF MAXUN EXCEED 9.	00012010
DO 10 I = 1, MXPL	IDP=IDEXP(1)+2	00012020
10 IF (MAXUN(MXYEAR, IDP) .GE. 10) GO TO 30		00012030
CONTINUE		00012040
	DROPPING THROUGH THE LOOP MEANS THAT BASE 10 PACKING IS ACCEPTABLE	00012050
	SET PACKING VECTOR IPABA WITH BASE 10	00012060
IPABA(1) = 10		00012070
DO 20 I=2, MXPL		00012080
20 IPABA(I) = 10 * IPABA(I-1) + 1		00012090
RETURN		00012100
	REACHING 30 MEANS PACKING BASE SHOULD BE IDENTICAL TO MAXIMUM UNIT NUMBER.	00012110
	SET DEFAULT BASE TO 1 SO THAT IF MXPL<8 THE IPABA VECTOR IS VALID	00012120
30 DO 40 I=1,8		00012130
40 IPABA(I) = 1		00012140
	SET BASE VECTOR EQUAL TO THE PRODUCT OF MAXIMUM UNIT NUMBER FOR SUCCESSIVE PLANT TYPES	00012150
	IDP = IDEXP(1)+2	00012160
		00012170
		00012180
		00012190
		00012200
		00012210
		00012220
		00012230
		00012240

```

C   IPABA(1) = MAXUN(MXYEAR, IDP)          00012250
DO 50 I=2,MXPL                           00012260
IDP=IDEXP(I-1)+2                         00012270
C   CHECK BASE VECTOR SIZE                00012280
LIMIT = 2000000000/MAXUN(MXYEAR, IDP)    00012290
IF (IPABA(I-1) .GT. LIMIT) GO TO 60     00012300
50 IPABA(I) = IPABA(I-1) * (MAXUN(MXYEAR, IDP) + 1) 00012310
WRITE(6,900) (IPABA(I), I=1,MXPL)        00012320
900 FORMAT (17H PACKING BASE IS/8I10/)   00012330
      RETURN                                00012340
CCCCCC
      REACHING 60 MEANS BASE IS TOO LARGE. THE USER SHOULD 00012350
      REDUCE THE MAXIMUM UNIT NUMBER FOR SOME EXPANSION 00012360
      CANDIDATES SO THAT THEIR PRODUCT IS <2.0*10***9. 00012370
      00012380
      00012390
      00012400
      00012410
      00012420
      00012430
      00012440
      00012450
      00012460
      00012470
      00012480
      00012490
      00012500
      00012510
      00012520
      00012530
      00012540
      00012550
      00012560
      00012570
      00012580
      00012590
      00012600
      00012610
      00012620
      00012630
      00012640
      00012650
      00012660
      00012670
      00012680
      00012690
      00012700
      00012710
      00012720
      00012730
      00012740
      00012750
      00012760
      00012770
      00012780
      00012790
      00012800
      00012810
      00012820
      00012830
      00012840
      00012850
      00012860
      00012870
      00012880
      00012890
      00012900
      00012910
      00012920
CCCC
      SET DEFAULT BEGINNING YEAR OF CHANGES. 00012530
      NY1=1
C   BEGIN LISTINGS AND CHANGES. 00012540
      70 DO 100 NYF=NY1,MXYEAR               00012550
C   DEFINE CURRENT CALENDAR YEAR. 00012560
      KYEAR=IBASYR + NYF                  00012570
C   LOOP FOR ALL PLANT TYPES IN CURRENT YEAR. 00012580
      DO 75 I=1,MXPL                      00012590
      IDP=IDEXP(I)+2                     00012600
      75 ITEMP(I) = MAXUN(NYF, IDP)       00012610
      WRITE(6,902) KYEAR,(ITEMP(I),I=1,MXPL) 00012620
      902 FORMAT (6H YEAR,15/16H OLD LIMITS ARE /1H ,8I3, 00012630
      + 18H ENTER NEW LIMITS )           00012640
      80 READ(5,903) (ITEMP(I),I=1,MXPL)   00012650
      903 FORMAT (8I3)                      00012660
CCCC
      CREATE AND CHECK NEW PACKING BASE VECTOR. 00012670
      IPABA(1) = ITEMP(1)                 00012680
      DO 85 I=2,MXPL                     00012690
      LIMIT = 2000000000/ITEMP(I)        00012700
      IF (IPABA(I-1) .GT. LIMIT) GO TO 95 00012710
      IPABA(I) = IPABA(I-1) * (ITEMP(I-1) + 1) 00012720
      85 CONTINUE                          00012730
      ^^^ DROPPING THROUGH THE LOOP MEANS THAT THE UNIT LIMITS 00012740
      FOR THE CURRENT YEAR ARE ACCEPTABLE. ASSIGN THEM TO 00012750
      THE MAXUN ARRAY. 00012760
      DO 90 I=1,MXPL                     00012770
      IDP=IDEXP(I)+2                     00012780
      90 MAXUN(NYF, IDP) = ITEMP(I)       00012790
C   GO TO 100                           00012800
      95 WRITE(6,904)                     00012810
      904 FORMAT (34H UNIT LIMITS TOO BIG. ENTER AGAIN ) 00012820
      GO TO 80                            00012830
      100 CONTINUE                         00012840
CCCC
      LEAVE ROOM FOR ERROR. PROVIDE THE MEANS TO RESPECIFY 00012850
      THE UNIT NUMBERS. REQUEST THE FIRST YEAR OF CHANGES. 00012860
      00012870
      00012880
      00012890
      00012900
      00012910
      00012920

```

CHANGES SHOULD ALWAYS PROCEED UP TO AND INCLUDING THE
 LAST YEAR OF THE STUDY PERIOD, SINCE IT IS THE LAST
 YEARS LIMITS THAT DEFINE THE PACKING BASE VECTOR. 00012930
 00012940
 00012950
 00012960
 00012970
 00012980
 00012990
 00013000
 00013010
 00013020
 00013030
 00013040
 00013050
 00013060
 00013070
 00013080
 00013090
 00013100
 00013110
 00013120
 00013130
 00013140
 00013150
 00013160
 00013170
 00013180
 00013190
 00013200
 00013210
 00013220
 00013230
 00013240
 00013250
 00013260
 00013270
 00013280
 00013290
 00013300
 00013310
 00013320
 00013330
 00013340
 00013350
 00013360
 00013370
 00013380
 00013390
 00013400
 00013410
 00013420
 00013430
 00013440
 00013450
 00013460
 00013470
 00013480
 00013490
 00013500
 00013510
 00013520
 00013530
 00013540
 00013550
 00013560
 00013570
 00013580
 00013590
 00013600

CHANGES SHOULD ALWAYS PROCEEDED UP TO AND INCLUDING THE
 LAST YEAR OF THE STUDY PERIOD, SINCE IT IS THE LAST
 YEARS LIMITS THAT DEFINE THE PACKING BASE VECTOR. 00012930
 00012940
 00012950
 00012960
 00012970
 00012980
 00012990
 00013000
 00013010
 00013020
 00013030
 00013040
 00013050
 00013060
 00013070
 00013080
 00013090
 00013100
 00013110
 00013120
 00013130
 00013140
 00013150
 00013160
 00013170
 00013180
 00013190
 00013200
 00013210
 00013220
 00013230
 00013240
 00013250
 00013260
 00013270
 00013280
 00013290
 00013300
 00013310
 00013320
 00013330
 00013340
 00013350
 00013360
 00013370
 00013380
 00013390
 00013400
 00013410
 00013420
 00013430
 00013440
 00013450
 00013460
 00013470
 00013480
 00013490
 00013500
 00013510
 00013520
 00013530
 00013540
 00013550
 00013560
 00013570
 00013580
 00013590
 00013600

THIS SUBROUTINE DEFINES THE ARTIFICIAL CHANNEL WIDTH,
 THE MAXIMUM NUMBER OF STATES PER YEAR, THE INCREMENT BY
 WHICH THE ARTIFICIAL CHANNELS WILL BE MODIFIED AFTER EACH
 INNER ITERATION, THE MAXIMUM PLANT ADDITION PER YEAR FOR
 EXPANSION CANDIDATE AND THE ORIGINAL ARTIFICIAL CHANNELS. 00013300
 00013310
 00013320
 00013330
 00013340
 00013350
 00013360
 00013370
 00013380
 00013390
 00013400
 00013410
 00013420
 00013430
 00013440
 00013450
 00013460
 00013470
 00013480
 00013490
 00013500
 00013510
 00013520
 00013530
 00013540
 00013550
 00013560
 00013570
 00013580
 00013590
 00013600

COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
 FCR,HOURS,IBASYS,IMXPL,ITIN,ITMAX,
 ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
 MAXPLA,MAXPD,MXALL,MXPL,MXYEAR,
 RESMAR,UBOUND
 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30
 31
 32
 33
 34
 35
 36
 37
 38
 39
 40
 41
 42
 43
 44
 45
 46
 47
 48
 49
 50
 51
 52
 53
 54
 55
 56
 57
 58
 59
 60
 61
 62
 63
 64
 65
 66
 67
 68
 69
 70
 71
 72
 73
 74
 75
 76
 77
 78
 79
 80
 81
 82
 83
 84
 85
 86
 87
 88
 89
 90
 91
 92
 93
 94
 95
 96
 97
 98
 99
 100
 101
 102
 103
 104
 105
 106
 107
 108
 109
 110
 111
 112
 113
 114
 115
 116
 117
 118
 119
 120
 121
 122
 123
 124
 125
 126
 127
 128
 129
 130
 131
 132
 133
 134
 135
 136
 137
 138
 139
 140
 141
 142
 143
 144
 145
 146
 147
 148
 149
 150
 151
 152
 153
 154
 155
 156
 157
 158
 159
 160
 161
 162
 163
 164
 165
 166
 167
 168
 169
 170
 171
 172
 173
 174
 175
 176
 177
 178
 179
 180
 181
 182
 183
 184
 185
 186
 187
 188
 189
 190
 191
 192
 193
 194
 195
 196
 197
 198
 199
 200
 201
 202
 203
 204
 205
 206
 207
 208
 209
 210
 211
 212
 213
 214
 215
 216
 217
 218
 219
 220
 221
 222
 223
 224
 225
 226
 227
 228
 229
 230
 231
 232
 233
 234
 235
 236
 237
 238
 239
 240
 241
 242
 243
 244
 245
 246
 247
 248
 249
 250
 251
 252
 253
 254
 255
 256
 257
 258
 259
 260
 261
 262
 263
 264
 265
 266
 267
 268
 269
 270
 271
 272
 273
 274
 275
 276
 277
 278
 279
 280
 281
 282
 283
 284
 285
 286
 287
 288
 289
 290
 291
 292
 293
 294
 295
 296
 297
 298
 299
 300
 301
 302
 303
 304
 305
 306
 307
 308
 309
 310
 311
 312
 313
 314
 315
 316
 317
 318
 319
 320
 321
 322
 323
 324
 325
 326
 327
 328
 329
 330
 331
 332
 333
 334
 335
 336
 337
 338
 339
 340
 341
 342
 343
 344
 345
 346
 347
 348
 349
 350
 351
 352
 353
 354
 355
 356
 357
 358
 359
 360
 361
 362
 363
 364
 365
 366
 367
 368
 369
 370
 371
 372
 373
 374
 375
 376
 377
 378
 379
 380
 381
 382
 383
 384
 385
 386
 387
 388
 389
 390
 391
 392
 393
 394
 395
 396
 397
 398
 399
 400
 401
 402
 403
 404
 405
 406
 407
 408
 409
 410
 411
 412
 413
 414
 415
 416
 417
 418
 419
 420
 421
 422
 423
 424
 425
 426
 427
 428
 429
 430
 431
 432
 433
 434
 435
 436
 437
 438
 439
 440
 441
 442
 443
 444
 445
 446
 447
 448
 449
 450
 451
 452
 453
 454
 455
 456
 457
 458
 459
 460
 461
 462
 463
 464
 465
 466
 467
 468
 469
 470
 471
 472
 473
 474
 475
 476
 477
 478
 479
 480
 481
 482
 483
 484
 485
 486
 487
 488
 489
 490
 491
 492
 493
 494
 495
 496
 497
 498
 499
 500
 501
 502
 503
 504
 505
 506
 507
 508
 509
 510
 511
 512
 513
 514
 515
 516
 517
 518
 519
 520
 521
 522
 523
 524
 525
 526
 527
 528
 529
 530
 531
 532
 533
 534
 535
 536
 537
 538
 539
 540
 541
 542
 543
 544
 545
 546
 547
 548
 549
 550
 551
 552
 553
 554
 555
 556
 557
 558
 559
 560
 561
 562
 563
 564
 565
 566
 567
 568
 569
 570
 571
 572
 573
 574
 575
 576
 577
 578
 579
 580
 581
 582
 583
 584
 585
 586
 587
 588
 589
 590
 591
 592
 593
 594
 595
 596
 597
 598
 599
 600
 601
 602
 603
 604
 605
 606
 607
 608
 609
 610
 611
 612
 613
 614
 615
 616
 617
 618
 619
 620
 621
 622
 623
 624
 625
 626
 627
 628
 629
 630
 631
 632
 633
 634
 635
 636
 637
 638
 639
 640
 641
 642
 643
 644
 645
 646
 647
 648
 649
 650
 651
 652
 653
 654
 655
 656
 657
 658
 659
 660
 661
 662
 663
 664
 665
 666
 667
 668
 669
 670
 671
 672
 673
 674
 675
 676
 677
 678
 679
 680
 681
 682
 683
 684
 685
 686
 687
 688
 689
 690
 691
 692
 693
 694
 695
 696
 697
 698
 699
 700
 701
 702
 703
 704
 705
 706
 707
 708
 709
 710
 711
 712
 713
 714
 715
 716
 717
 718
 719
 720
 721
 722
 723
 724
 725
 726
 727
 728
 729
 730
 731
 732
 733
 734
 735
 736
 737
 738
 739
 740
 741
 742
 743
 744
 745
 746
 747
 748
 749
 750
 751
 752
 753
 754
 755
 756
 757
 758
 759
 760
 761
 762
 763
 764
 765
 766
 767
 768
 769
 770
 771
 772
 773
 774
 775
 776
 777
 778
 779
 780
 781
 782
 783
 784
 785
 786
 787
 788
 789
 790
 791
 792
 793
 794
 795
 796
 797
 798
 799
 800
 801
 802
 803
 804
 805
 806
 807
 808
 809
 810
 811
 812
 813
 814
 815
 816
 817
 818
 819
 820
 821
 822
 823
 824
 825
 826
 827
 828
 829
 830
 831
 832
 833
 834
 835
 836
 837
 838
 839
 840
 841
 842
 843
 844
 845
 846
 847
 848
 849
 850
 851
 852
 853
 854
 855
 856
 857
 858
 859
 860
 861
 862
 863
 864
 865
 866
 867
 868
 869
 870
 871
 872
 873
 874
 875
 876
 877
 878
 879
 880
 881
 882
 883
 884
 885
 886
 887
 888
 889
 890
 891
 892
 893
 894
 895
 896
 897
 898
 899
 900
 901
 902
 903
 904
 905
 906
 907
 908
 909
 910
 911
 912
 913
 914
 915
 916
 917
 918
 919
 920
 921
 922
 923
 924
 925
 926
 927
 928
 929
 930
 931
 932
 933
 934
 935
 936
 937
 938
 939
 940
 941
 942
 943
 944
 945
 946
 947
 948
 949
 950
 951
 952
 953
 954
 955
 956
 957
 958
 959
 960
 961
 962
 963
 964
 965
 966
 967
 968
 969
 970
 971
 972
 973
 974
 975
 976
 977
 978
 979
 980
 981
 982
 983
 984
 985
 986
 987
 988
 989
 990
 991
 992
 993
 994
 995
 996
 997
 998
 999
 1000

```

C          DOUBLE PRECISION TINT,TY
C          COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
C                         + HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
C                         + PSSCHD(20,8),SYSCUM(4)
C
C          DO 5 I=1,8
      5 IWIDTH(I) = 3
C          NID = MXPL
C          GO TO (10,20,30,40,50,60), NID
      10 IWIDTH(1) = 6
          MAXSTA = 6
          GO TO 70
      20 DO 25 I=1,MXPL
      25 IWIDTH(I) = 6
          MAXSTA = 36
          GO TO 70
      30 IWIDTH(1) = 4
          IWIDTH(2) = 4
          IWIDTH(3) = 5
          MAXSTA = 80
          GO TO 70
      40 IWIDTH(1) = 3
          IWIDTH(2) = 3
          IWIDTH(3) = 4
          IWIDTH(4) = 5
          MAXSTA = 180
          GO TO 70
      50 IWIDTH(1) = 3
          DO 55 I=2,MXPL
      55 IWIDTH(I) = 4
          IWIDTH(MXPL) = 5
          MAXSTA = 960
          GO TO 70
      60 DO 65 I = 1,MXPL
      65 IWIDTH(I) = 3
          IWIDTH(MXPL) = 4
          MAXSTA = 972
      70 CONTINUE
C          DEFINE CHANNEL INCREMENT
      DO 80 I = 1,8
      80 IPCH(I) = IWIDTH(I) - 2
C          DEFINE THE ORIGINAL ARTIFICIAL VCHANNELS.
      DO 100 NY=1,MXYEAR
      DO 90 IP = 1,MXPL
          LOWB(NY,IP) = 0
      90 NUPB(NY,IP) = IWIDTH(IP) - 1
      100 CONTINUE
      DO 130 NY = 1,MXYEAR
          PK=AUPEAK(NY)
C
      NYY = NY
      SYSDIF = CAPABS(NY) - PK
      DO 109 I = 1,MXPL
          SYSDIF = SYSDIF + PLACA(I,3)*LOWB(NY,I)
      109 CONTINUE
      110 CONTINUE
          RM = RESMIN(NY) * 0.95
          IF (SYSDIF .GT. RM) GO TO 130
      DO 120 IP = 1, MXPL
          IDP=IDEXP(IP)+2
          IF (LOWB(NY,IP).GE.MAXUN(NY,IP)) GO TO 125

```

```

LOWB(NY,IP) = 1 + LOWB(NY,IP)
SYSdif = SYSdif + PLACA(IP,3)
IF (SYSdif .GT. RM) GO TO 124
GO TO 120
125 IF(IP.LT.MXPL) GO TO 120
GO TO 200
120 CONTINUE
GO TO 110
124 DO 128 NY = NYV,MXYEAR
DO 126 I = 1,MXPL
LOWB(N,I) = LOWB(NY,I)
126 NUPB(N,I) = LOWB(N,I) + IWIDTH(I) - 1
128 CONTINUE
C 130 CONTINUE
C CALCULATE MAXIMUM UNIT ADDITION PER YEAR BY CANDIDATE TYPE
DO 135 IP=1,MXPL
135 MAXADD(IP) = IWIDTH(IP)/2+1
C PRINT THE RESULTS OF THIS SUBROUTINE
C WRITE(6,900)
900 FORMAT(1H //*, CHANNEL WIDTH, INCREMENT AND MAXIMUM *,
* NUMBER OF STATES*)
C WRITE(6,901) IWIDTH
C WRITE(6,901) IPCH
C WRITE(6,901) MAXSTA
C WRITE(6,901) IWIDTH
C WRITE(6,901) IPCH
C WRITE(6,901) MAXSTA
901 FORMAT(1H ,20I4)
C WRITE(6,902)
902 FORMAT(1H //*, LOWER AND UPPER CHANNEL BOUNDS PER YEAR*)
DO 140 NY = 1,MXYEAR
NYV = IBASYR + NY
WRITE(6,901) NYV, (LOWB(NY,I), I=1,MXPL)
WRITE(6,903) (NUPB(NY,I), I=1,MXPL)
WRITE(6,901) NYV, (LOWB(NY,I), I=1,MXPL)
WRITE(6,903) (NUPB(NY,I), I=1,MXPL)
140 CONTINUE
903 FORMAT(1M ,4X,19I4)
C 904 FORMAT(F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2)
905 FORMAT(1H ,F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2)
ICFLAG = 0
GO TO 300
200 ICFLAG = 1
WRITE(6,906)
906 FORMAT(1H /* EXPANSION PLANT TYPES SELECTED ARE TOO SMALL TO */
* COVER THE MINIMUM RESERVE MARGIN */
* ***** RERUN WITH MORE OR LARGER PLANTS *****)
ITOUT = 0
300 RETURN
END
C
C
C


---


SUBROUTINE CHOICE(TSTOP)


---


C
C
C
C
C
C
USER INTERACTION SUBROUTINE. THE USER CHOOSES UP TO SIX
EXPANSION PLANT CANDIDATES THAT WILL BE USED IN OPTIMIZA-
TION. THIS PLANT COMBINATION CAN BE CHANGED AFTER EACH
OPTIMIZATION. THE MAXIMUM NUMBER OF ITERATIONS IS ALSO
DEFINED BY THE USER. AT THE END OF EACH OPTIMIZATION THE UP
00014290
00014300
00014310
00014320
00014330
00014340
00014350
00014360
00014370
00014380
00014390
00014400
00014410
00014420
00014430
00014440
00014450
00014460
00014470
00014480
00014490
00014500
00014510
00014520
00014530
00014540
00014550
00014560
00014570
00014580
00014590
00014600
00014610
00014620
00014630
00014640
00014650
00014660
00014670
00014680
00014690
00014700
00014710
00014720
00014730
00014740
00014750
00014760
00014770
00014780
00014790
00014800
00014810
00014820
00014830
00014840
00014850
00014860
00014870
00014880
00014890
00014900
00014910
00014920
00014930
00014940
00014950
00014960

```

TO TEN BEST OPTIMAL SOLUTIONS WILL BE PRINTED DEPENDING ON
 USER REQUEST. 00014970
 00014980
 00014990
 00015000
 00015010
 00015020
 00015030
 00015040
 00015050
 00015060
 00015070
 00015080
 00015090
 00015100
 00015110
 00015120
 00015130
 00015140
 00015150
 00015160
 00015170
 00015180
 00015190
 00015200
 00015210
 00015220
 00015230
 00015240
 00015250
 00015260
 00015270
 00015280
 00015290
 00015300
 00015310
 00015320
 00015330
 00015340
 00015350
 00015360
 00015370
 00015380
 00015390
 00015400
 00015410
 00015420
 00015430
 00015440
 00015450
 00015460
 00015470
 00015480
 00015490
 00015500
 00015510
 00015520
 00015530
 00015540
 00015550
 00015560
 00015570
 00015580
 00015590
 00015600
 00015610
 00015620
 00015630
 00015640

DEFINITION OF LOCAL VARIABLES

NAME	TYPE	SIZE	DEFINITION
ISTOP	INT.	-	PROGRAM STOP FLAG. IF ISTOP = 0 PROGRAM CONTINUES IF ISTOP = 1 PROGRAM STOPS
NBEST	INT.	-	NUMBER OF BEST SOLUTIONS REQUESTED FOR REPORT
COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX, FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX, ITOUT,MAXALL,MAXI,MAXINP,MAXDR, MAXPLA,MAXPO,MXALL,MXPL,MXYEAR, RESMAR,UBOUND			
COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8), IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8), JWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200), MAXUN(20,20),MAXADD(8),MINUN(20,20), NEXPID(20),NSTPRE(20,20),NUPB(20,8), PEAKS(20,4),PLACA(20,14),PLANT(210,14), RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20), FATOPE(20),SOL(20,2),AUPEAK(20)			
COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260), MXELDC(20,4),X(1250)			
COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,6),ROM(40,4),PTCUMH(3,8)			
COMMON /TNS/ LBOULD(8),LIST(8,1000),MAXTUN(8), MINTUN(8),NUBOLD(8)			
COMMON /TBL/ TINT(120),TY(120)			
DOUBLE PRECISION TINT,TY			
COMMON /HYS/ CMULT(4),EVAL(3),EMULT(4),HYDR(3,11), HYEXP(11),HYSCHD(20,8),PSEXP(11), PSSCHD(20,8),SYSCUM(4)			
DIMENSION KEEP(10) IF (ITOUT .GT. 0) GO TO 10 PRINT INITIAL COMMENTS WRITE(6,900)			
900	FORMAT(1H1,10X,"CERES OPTIMIZATION BEGINS"// + " CERES HAS A BUILT IN SENSITIVITY ANALYSIS PROCESS."// + " FOR EACH SENSITIVITY ANALYSIS ITERATION (CALLED HERE"// + " "MAIN ITERATION"), THE USER MUST SPECIFY THE EXPANSION"// + " PLANT CANDIDATE COMBINATION THAT WILL BE USED IN"// + " OPTIMIZATION. SINCE THE CERES OPTIMIZATION ALGORITHM"// + " IS OF ITERATIVE NATURE THE MAXIMUM NUMBER OF "TUNNEL"// + " ITERATIONS" MUST BE PROVIDED") WRITE(6,901)		
901	FORMAT(1H1,10X,"FOR EACH MAIN ITERATION THE USER MUST SPECIFY:"// + " 1. THE DESIRED EXPANSION PLANT COMBINATION."// + " 2. THE MAXIMUM NUMBER OF TUNNEL ITERATIONS(MUST BE"// + " BETWEEN 10 AND 50. DEFAULT IS 10)"//		

```

+* 3. THE NUMBER OF BEST SOLUTIONS FOR WHICH A REPORT*/  

+* IS DESIRED (MAXIMUM IS 10 OR THE NUMBER OF TUNNEL*/  

+* ITERATIONS REQUIRED TO REACH THE OPTIMUM SOLUTION*/  

+* IF THIS IS LESS THAN 10, DEFAULT IS 11*/  

+* THERE IS NO LIMIT ON THE NUMBER OF MAIN ITERATIONS*/  

+* CERES WILL STOP WHEN THE USER INSTRUCTS TO DO SO.*/  

+* 4. FIXED CHARGE RATE OR SALVAGE VALUE OPTION FOR COST*/  

+* CALCULATIONS*/) 00015650  

+* 00015660  

+* 00015670  

+* 00015680  

+* 00015690  

+* 00015700  

+* 00015710  

+* 00015720  

+* 00015730  

+* 00015740  

+* 00015750  

+* 00015760  

+* 00015770  

C GO TO 100 00015780  

10 CONTINUE 00015790  

C  

C WRITE(6,902) ITOUT 00015800  

902 FORMAT(1H THE NUMBER OF MAIN ITERATIONS IS:,I3/) 00015810  

C  

C WRITE(6,903) ITIN 00015820  

903 FORMAT(1H ,I4,I TUNNEL ITERATIONS WERE REQUIRED*,  

+* TO FIND THE OPTIMAL SOLUTION*) 00015830  

C  

C NINIT = 10 00015840  

NIT = MIN0(NINIT,ITIN) 00015850  

WRITE(6,904) NIT 00015860  

904 FORMAT(1H INPUT THE NUMBER OF BEST SOLUTIONS YOU WANT*,  

+* REPORTED*/ THIS SHOULD BE LESS OR EQUAL TO:,I4) 00015880  

READ(5,*NBEST) 00015890  

C  

C CHECK NBEST AND SET TO 1 IF OUT OF RANGE 00015900  

IF (NBEST .LT. 1 .OR. NBEST .GT. NIT) NBEST = 1 00015920  

C  

C WRITE(40,920) MXPL 00015930  

920 FORMAT(1I2) 00015940  

930 FORMAT(1I2) 00015950  

C  

C WRITE(40,920) NBEST 00015960  

C CALL RESIMULATION AND REPORT ROUTINES NBEST TIMES. 00015970  

IRES = 1 00015980  

DO 20 N = 1,NBEST 00016000  

NN=N 00016010  

NB=NBEST 00016020  

CALL RESIM(NN,NB,IRES,KEEP) 00016030  

20 CONTINUE 00016040  

IRES = 0 00016050  

C  

C WRITE(6,905) 00016060  

905 FORMAT(1H IF YOU WANT TO STOP THE MAIN ITERATIONS*,  

+* (SENSITIVITY ANALYSIS), TYPE 1 /* 00016080  

+* IF YOU DESIRE TO CONTINUE, TYPE 0 /*) 00016090  

READ(5,*) ISTOP 00016100  

IF (ISTOP .LT. 1) GO TO 100 00016110  

RETURN 00016120  

C  

C 100 ITOUT = ITOUT+1 00016130  

C ASK FOR EXPANSION CANDIDATE COMBINATION 00016140  

C  

C WRITE(6,906) 00016150  

906 FORMAT(1H /, ENTER THE NUMBER OF EXPANSION CANDIDATES*,  

+* YOU WILL CONSIDER IN THIS ITERATION*/ 00016160  

+* IT SHOULD NOT BE MORE THAN 8. IT IS RECOMMENDED TO BE*,  

+* 4 OR 5*) 00016170  

C  

C READ(5,*) MXPL 00016180  

C CALCULATE NUMBER OF ALL PLANTS 00016190  

MAXALL = MAXPLA + MXPL 00016200  

C  

C WRITE(6,907) MXPL 00016210  

907 FORMAT(1H /, ENTER THE,I2,I EXPANSION PLANT ID'S./* 00016220  

+* THE PLANT ID'S ARE DEFINED FROM THE PLANT ORDER IN*,  

+* THE INPUT MODULE./* 00016230  

+* ENTER -1 FOR HYDRO EXPANSION, 0 FOR PUMPED STORAGE EXPANSION./* 00016240  

+* THE NUMBERS ENTERED SHOULD BE SEPARATED BY BLANKS./*) 00016250  

READ(5,*) (IDEXP(I),I=1,MXPL) 00016260  

00016270  

00016280  

00016290  

00016300  

00016310  

00016320

```

```

C      ASK FOR ITMAX
      WRITE(6,908)
908  FORMAT(1H /" ENTER THE MAXIMUM NUMBER OF TUNNELS",
+    " ITERATIONS",
+    " IT SHOULD BE BETWEEN 10 AND 50")
      READ(5,*), ITMAX
C      CHECK ITMAX
      IF (ITMAX .GE. 10 .AND. ITMAX .LE. 50) GO TO 110
      ITMAX = 10
110  CONTINUE
C      WRITE(6,909)
909  FORMAT(1H /" FOR SALVAGE VALUE OPTION ENTER -1.0",
+    " FOR FIXED CHARGE RATE ENTER THE FCR VALUE")
      READ(5,*), FCR
C      CLOLP = CPLOLP
      WRITE(6,910) CPLOLP
910  FORMAT(1H /" THE CRITICAL LOLP VALUE IN PREP MODULE IS", F9.6,
+    " IF YOU WANT TO INCREASE IT, ENTER THE NEW NUMBER. OTHER",
+    " WISE ENTER -1.0")
      READ(5,*), CHLOLP
      IF (CHLOLP .GT. -1.0) CLOLP = CHLOLP
C      WRITE(6,911)
911  FORMAT(1H /" ENTER THE MAXIMUM RESERVE MARGIN IN %",
+    " IT SHOULD BE BETWEEN 20 AND 50%. DEFAULT IS 40%")
      READ(5,*), RESMAR
C      WRITE(6,912)
912  FORMAT(1H /" ENTER THE DISCOUNT RATE IN %. DEFAULT IS 15%")
      READ(5,*), DISRAT
      IF (DISRAT .LT. 5.0 .OR. DISRAT .GT. 50.0) DISRAT = 15.0
      DISRAT = DISRAT/100.0
      IF (RESMAR .LT. 39.5 .AND. RESMAR .GT. 40.5) GO TO 140
      RESMAR = RESMAR/100.0
C      FIND THE YEARLY MAXIMUM RESERVE MARGINS
      DO 130 NY = 1,MXYEAR
      PYR=AUPEAK(NY)
C      130 RESMAX(NY)= PYR*RESMAR
C      140 CONTINUE
C      CC=9000.0
      DO 200 I=1,MXPL
      IP=IDEXP(I)
      IF(IP.GT.0) GO TO 160
      IF(IP.GT.-1) GO TO 170
      CX=HYEXP(9)
      GO TO 180
160  CX=PLANT(IP,6)
      GO TO 180
170  CX=PSEXPN(9)
180  IF(CX.LT.CC) GO TO 190
      GO TO 200
190  CC=CX
      IMXPL=I
200  CONTINUE
C      RESTRUCTURE AND PRINT THE EXPANSION CANDIDATE FILE.

```

```

C          DO 104 IP=1,8
104      NEXPID(IP) = -2
DO 107 IP=1,MXPL
      TOP=IDEXP(IP)
      IF(IDP.GT.0) GO TO 105
      IF(IDP.GT.-1) GO TO 106
      PLACA(IP,1)=HYEXPN(1)
      PLACA(IP,2)=HYEXPN(2)
      PLACA(IP,4)=HYEXPN(6)
      PLACA(IP,5)=HYEXPN(5)
      PLACA(IP,6)=HYEXPN(9)
      PLACA(IP,9)=30.0
      PLACA(IP,13)=HYEXPN(10)
      GO TO 107
105     REW IND 23
      DO 103 I=1, IDP
103      READ(23,914) (PLACA(IP,J),J=1,14)
914      FORMAT(F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2)
      GO TO 107
106      PLACA(IP,2)=0.0
      PLACA(IP,3)=PSEXPN(1)
      PLACA(IP,4)=PSEXPN(6)
      PLACA(IP,5)=PSEXPN(5)
      PLACA(IP,6)=PSEXPN(9)
      PLACA(IP,9)=30.0
      PLACA(IP,13)=PSEXPN(10)
107      CONTINUE
      RETURN
      END



---


SUBROUTINE COMPST(NSTCUR,NYNEXT)



---


C          COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1          FCR,HOUR$,IBASYS,IMXPL,ITIN,ITMAX,
2          ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3          MAXPLA,MXP0,MXALL,MXPL,MXYEAR,
4          RESMAR,UBOUND
C          COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
-          IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
-          INIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
-          MAXUN(20,20),MAXADD(8),MINUN(20,20),
-          NEXPID(20),NSTPRE(20,20),NUPB(20,8),
-          PEAKS(20,4),PLACA(20,14),PLANT(210,14),
-          RESMAX(20),RESHIN(20),SSCUM(4,20),UNE(20),
-          FATOPU(20),SOL(20,2),AUPEAK(20)
C          COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+          MXELDC(20,4),X(1250)
C          COMMON /CMS/ PTCUM(20,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)
C          COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+          MINTUN(8),NUBOLD(8)

```

```

C COMMON /TBL/ TINT(120),TY(120) 00017690
C DOUBLE PRECISION TINT,TY 00017700
C COMMON /HYS/ CMULT(4),EVAL(3),EMULT(4),HYDR(3,11), 00017710
C   HYEXPN(11),HYSCHD(20,8),PSEXPN(11), 00017720
C   PSSCHD(20,8),SYSCUM(4) 00017730
C
C DIMENSION NSTCUR(20) 00017740
C
C CCCCCC Initialization 00017750
C
C DO 10 J=1,11 00017760
C DO 10 I=1,3 00017770
C 10 HYDR(I,J)=0.0 00017780
C
C Hydro Scheduled Plant 00017790
C
C DO 15 J=1,8 00017800
C 15 HYDR(1,J)=HYSCHD(NYNEXT,J) 00017810
C
C Pumped Storage Scheduled Plant 00017820
C
C DO 16 J=1,8 00017830
C 16 HYDR(3,J)=PSSCHD(NYNEXT,J) 00017840
C
C Expansion Plants 00017850
C
C DO 20 I=1,MXPL 00017860
C IP=IDEXP(I) 00017870
C IF(IP.GT.0) GO TO 20 00017880
C IH=IP+2 00017890
C HMULT=FLOAT(NSTCUR(I)) 00017900
C IF(IH.GT.1) GO TO 40 00017910
C
C Hydro Plant 00017920
C
C DO 30 J=1,4 00017930
C 30 HYDR(2,J)=HMULT*HYEXPN(J) 00017940
C DO 35 J=5,11 00017950
C 35 HYDR(2,J)=HYEXPN(J) 00017960
C GO TO 20 00017970
C
C Pumped Storage 00017980
C
C 40 W1=PSSCHD(NYNEXT,1) 00017990
C W2=PSSCHD(NYNEXT,2) 00018000
C W3=HMULT*PSEXPN(1) 00018010
C W4=HMULT*PSEXPN(2) 00018020
C HYDR(3,1)=W1+W3 00018030
C HYDR(3,2)=W2+W4 00018040
C DO 45 J=3,7 00018050
C 45 HYDR(3,J)=(W1*PSSCHD(NYNEXT,J)+W3*PSEXPN(J))/HYDR(3,1) 00018060
C HYDR(3,8)=(W2*PSSCHD(NYNEXT,8)+W4*PSEXPN(8))/HYDR(3,2) 00018070
C DO 50 J=9,11 00018080
C 50 HYDR(3,J)=PSEXPN(J) 00018090
C
C

```

20 CONTINUE

1 RETURN
END

C SUBROUTINE CUNSTR (INSTATE, NY, CULOLP, IFLCOM)

	TITLE CONSTRAINTS CHECK	00018450
	FUNCTION	00018460
	THIS SUBROUTINE CHECKS THE ACCEPTABILITY OF STATE "INSTATE" WITH RESPECT TO MINIMUM AND MAXIMUM RESERVE MARGINS AND LOLP.	00018470
	THE LOLP IS CALCULATED WITH THE CUMULANT METHOD. THE LOLP CHECK IS FINAL ONLY WHEN EITHER THE STATE LOLP (DENOTED BY CULOLP) IS GREATER THAN 0.001 OR THE CRITICAL LOLP (DENOTED BY CLOLP) IS GREATER THAN 0.001. IF THESE CONDITIONS ARE NOT MET THE STATE IS CONDITIONALLY ACCEPTED AND LOLP IS RECALCULATED IN MAIN THROUGH THE PIECEWISE LINEAR APPROXIMATION. THE 0.001 LIMIT IS SET BECAUSE THE CUMULANT METHOD IS NOT ACCURATE FOR LOLP VALUES LESS THAN 0.001. CULOLP MUST ALSO BE SMALLER THAN THE CRITICAL LOLP UPPER BOUND (DENOTED BY CLOLPU). IT IS ASSUMED THAT CLOLPU IS ALWAYS GREATER THAN 0.001. THE CUMULANT LOLP CALCULATION OMMITS MAINTENANCE CONSIDERATIONS FOR THE EXPANSION CANDIDATES. THEREFORE, WHEN CLOLP>0.001, AND A STATE IS ACCEPTED LOLP WILL BE CALCULATED AND CHECKED AGAIN WHEN OPERATING COSTS ARE ESTIMATED AT THE OPERCO SUBROUTINE.	00018480
	COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX, FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX, ITOUT,MAXALL,MAXI,MAXINP,MAXOR, MAXPLA,MAXPO,MXALL,MXPL,MXYEAR, RESMAR,UBOUND	00018490
	COMMON /ARS/ AVSP(20,4),CAPAB(20),ENEDEM(20,4),EXPLCU(4,8), INDEX(20),IPABA(8),IPCH(8),ISOL(20,8), IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200), MAXUN(20,20),MAXADD(8),MINUN(20,20), NEXPID(20),NSTPRE(20,20),NUPB(20,8), PEAKS(20,4),PLACA(20,14),PLANT(210,14), RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20), FATOPE(20),SOL(20,2),AUPEAK(20)	00018500
	COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260), MXELDC(20,4),X(1250)	00018510
	COMMON /CMS/ PTCUM(20,8,4),PTCUMX(20,8),RDM(40,4),PTCUMH(3,8)	00018520
	COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8), MINTUN(8),NUBOLD(8)	00018530
	COMMON /TBL/ TINT(120),TY(120)	00018540
	DOUBLE PRECISION TINT,TY	00018550
	COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11), HYEXP(11),HYSCHD(20,8),PSEXP(11), PSSCHD(20,8),SYSCUM(4)	00018560
	SUBROUTINE ARGUMENTS	00018570
	NAME DESCRIPTION	00018580
	CULOLP CURRENT STATE LOLP	00018590
		00018600
		00018610
		00018620
		00018630
		00018640
		00018650
		00018660
		00018670
		00018680
		00018690
		00018700
		00018710
		00018720
		00018730
		00018740
		00018750
		00018760
		00018770
		00018780
		00018790
		00018800
		00018810
		00018820
		00018830
		00018840
		00018850
		00018860
		00018870
		00018880
		00018890
		00018900
		00018910
		00018920
		00018930
		00018940
		00018950
		00018960
		00018970
		00018980
		00018990
		00019000
		00019010
		00019020
		00019030
		00019040

```

C      IFLCON      CONSTRAINT FLAG.
C          IF = 0, STATE IS ACCEPTABLE
C          IF=1, STATE IS UNACCEPTABLE
C      NSTCUR      CURRENT STATE UNPACKED
C      NYCR       CURRENT YEAR
C
C      COMMON VARIABLES USED: CAPABS, CLOLP, CLOLPU,
C          MXPL, PLACA, RESMAX, RESMIN
C
C      LOCAL VARIABLES
C      NAME      DESCRIPTION
C          I      I-TH PLANT TYPE
C          SYSCAP   SYSTEM CAPACITY IN MW
C
C      DIMENSION NSTATE(20)
C          TOTAL SYSTEM CAPACITY
C          SYSCAP = CAPABS(NY)
C
C      DO 10 I=1,MXPL
C          SYSCAP = SYSCAP + PLACA(I,3) * NSTATE(I)
C          SYSDIF=SYSCAP-AUPEAK(NY)
C
C          CHECK RESERVE MARGINS
C          IF (SYSDIF .LT. RESMIN(NY)) GO TO 30
C          IF (SYSDIF .GT. RESMAX(NY)) GO TO 30
C
C          CALCULATE LOLP
C          CALL FLOLPC(CULOLP,NSTATE,NY,SYSCAP)
C
C          CHECK CULOLP AGAINST CLOLP AND 0.001 LIMIT
C          IF (CULOLP .GE. 0.001 .AND. CULOLP .GT. CLOLP) GO TO 30
C          IF (CULOLP .GE. 0.001 .AND. CULOLP .LE. CLOLP .AND.
C              *CULOLP .LT. CLOLPU) GO TO 20
C
C          NOTE THAT IT IS ASSUMED THAT CLOLPU>0.001
C          IF (CULOLP .LT. 0.001 .AND. CULOLP .GE. 0.001) GO TO 30
C
C 20  IF(IFLCON = 0
C      RETURN
C
C 30  IF(IFLCON = 1
C      RETURN
C      END
C
C-----SUBROUTINE CUCALI(NSTCUR,N,ICUM)
C-----
```

```

C
C      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
C          FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
C          ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
C          MAXPLA,MAXPD,MXALL,MXPL,MXYEAR,
C          RESMAR,UBOUND
C
C      COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
C          IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
C          IWIDTM(8),LRDERR(420,3),LOWB(20,8),MAINS(200),
C          MAXUN(20,20),MAXADD(8),MINUN(20,20),
C          NEXPID(20),NSTPRE(20,20),NUPB(20,8),
C          PEAKS(20,4),PLACA(20,14),PLANT(210,14),
C          RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
C          FATOPEN(20),SOL(20,2),AUPEAK(20)
C
C      COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
C          + MXELDC(20,4),X(1250)
C
```

```

C
C      COMMON /CMS/ PTCUM(20,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)      00019730
C      COMMON /TNS/ LBOOLD(8),LIST(8,1000),MAXTUN(8),                  00019740
C      + MINTLN(8),NUBOLD(8)                                              00019750
C      COMMON /TBL/ TINT(120),TY(120)                                         00019760
C      DOUBLE PRECISION TINT,TY                                         00019770
C      COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),          00019780
C      + HYEXP(11),HYSCHD(20,8),PSEXPN(11),                         00019790
C      + PSSCHD(20,8),SYSCUM(4)                                         00019800
C
C      DIMENSION R(4), NSTCUR(20)
C
C      INITIALIZATION
C
C      DO 10 I=1,20
C      DO 10 J=1,8
C      10 PTCUMX(I,J)=0.0
C      DO 15 J=1,8
C      DO 15 I=1,3
C      15 PTCUMH(I,J)=0.0
C      IF(ICUM.GT.0) GO TO 50
C      DO 20 I=1,MXPL
C      IP=IDEXP(I)+2
C      Q=PLAC(A(1,5))
C      C=PLAC(A(1,3))
C      PTCUMX(IP,1)=C*Q
C      PTCUMX(IP,2)=C*C*Q*(1.0-Q)
C      PTCUMX(IP,3)=C*C*C*Q*(1.0-(3.0-2.0*Q)*Q)
C      PTCUMX(IP,4)=C*C*C*C*Q*(1.0-(7.0-(12.0-6.0*Q)*Q)*Q)
C      20 CONTINUE
C      GO TO 200
C      50 CONTINUE
C
C      HYDRO CUMULANTS
C
C      DO 55 I=1,3
C      Q=HYDR(I,5)
C      DO 60 J=1,2
C      IF(J.GT.1) GO TO 65
C      C=HYDR(I,1)*CMULT(N)*(1.0-ROM(2,N))
C      IF(I.GE.3) C=0.0
C      GO TO 70
C      65 C=HYDR(I,2)*(1.0-ROM(1,N))
C      IF(I.GE.3) C=HYDR(I,1)*(1.0-ROM(3,N))
C      70 R(1)=C*Q
C      R(2)=C*C*Q*(1.0-Q)
C      R(3)=C*C*C*Q*(1.0-(3.0-2.0*Q)*Q)
C      R(4)=C*C*C*C*Q*(1.0-(7.0-(12.0-6.0*Q)*Q)*Q)
C      DO 75 K=1,4
C      KK=K
C      IF(J.GT.1) KK=K+4
C      75 PTCUMH(I,KK)=R(K)
C      60 CONTINUE
C      55 CONTINUE
C
C      THERMAL EXPN CUMULANTS
C
C      DO 80 I=1,MXPL
C      IP=IDEXP(I)
C      IF(IP.LE.0) GO TO 80
C      Q=PLAC(A(1,5))                                         00020340
C
C      00020350
C      00020360
C      00020370
C      00020380
C      00020390
C      00020400

```

```

DO 85 J=1,2
  IF(J.GT.1) GO TO 90
  IN=2*(IP+2)
  C=PLACA(1,2)*(1.0-ROM(IN,N))
  GO TO 95
90 IN=2*IP+3
  C=PLACA(1,3)*(1.0-ROM(IN,N))
95 R(1)=C*Q
  R(2)=C*C*Q*(1.0-Q)
  R(3)=C*C*C*Q*(1.0-(13.0-2.0*Q)*Q)
  R(4)=C*C*C*C*Q*(1.0-(7.0-(12.0-6.0*Q)*Q)*Q)
  DO 100 K=1,4
    KK=K
    IF(J.GT.1) KK=K+4
100 PTCUMX(IP,KK)=R(K)
85 CONTINUE
80 CONTINUE

200 RETURN
END



---


SUBROUTINE FATHOMINSTCUR(NY,OBJ,IFATH)


---


CALCULATES THE MINIMUM CONSTRUCTION COST FOR REACHING THE
FINAL YEARS MINIMUM RESERVE MARGIN. THE MINIMUM YEARLY
OPERATING COST FOR ALL FUTURE STUDY YEAR PERIODS IS ADDED
TO THE ABOVE CONSTRUCTION COST. BOTH THESE ARE ADDED TO THE
STATE'S OBJECTIVE FUNCTION AND COMPARED WITH THE UPPER BOUND
FOR THE OBJECTIVE FUNCTION. THE STATE "NSTCUR" IS ACCEPTABLE
IF THE UPPER BOUND IS NOT EXCEEDED BY THE ABOVE COSTS. THEN
FLAG IFATH = 1. OTHERWISE IFATH = 0.

COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
  FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
  ITOUT,MAXALL,MAXI,MAXINP,MAXUR,
  MAXPLA,MXP0,MXALL,MXPL,MXYEAR,
  RESMAR,UBOUND
COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
  IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
  IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
  MAXUN(20,20),MAXADD(8),MINUN(20,20),
  NEXPID(20),NSTPRE(20,20),NUPB(20,8),
  PEAKS(20,4),PLACA(20,14),PLANT(210,14),
  RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
  FATOPE(20),SOL(20,2),AUPEAK(20)
COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
  MXELDC(20,4),X(1250)
COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)
COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
  MINTUN(8),NUBOLD(8),
COMMON /TBL/ TINT(120),TY(120)
DOUBLE PRECISION TINT,TY

```

```

COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
+ HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
+ PSSCHD(20,8),SYSCUM(4)          00021090
                                         00021100
                                         00021110
                                         00021120
                                         00021130
                                         00021140
                                         00021150
                                         00021160
                                         00021170
                                         00021180
                                         00021190
                                         00021200
                                         00021210
                                         00021220
                                         00021230
                                         00021240
                                         00021250
                                         00021260
                                         00021270
                                         00021280
                                         00021290
                                         00021300
                                         00021310
                                         00021320
                                         00021330
                                         00021340
                                         00021350
                                         00021360
                                         00021370
                                         00021380
                                         00021390
                                         00021400
                                         00021410
                                         00021420
                                         00021430
                                         00021440
                                         00021450
                                         00021460
                                         00021470
                                         00021480
                                         00021490
                                         00021500
                                         00021510
                                         00021520
                                         00021530
                                         00021540
                                         00021550
                                         00021560
                                         00021570
                                         00021580
                                         00021590
                                         00021600
                                         00021610
                                         00021620
                                         00021630
                                         00021640
                                         00021650
                                         00021660
                                         00021670
                                         00021680
                                         00021690
                                         00021700
                                         00021710
                                         00021720
                                         00021730
                                         00021740
                                         00021750
                                         00021760

DIMENSION NSTCUR(20)          C
IF (OBJ .LE. UBOUND) GO TO 5
IFATH = 0
RETURN
5 CONTINUE
IF (NY .EQ. MXYEAR) GO TO 100
CALCULATE ADDITIONAL CAPACITY REQUIREMENTS
C = 0.0
DO 10 I=1,MXPL
10 C = C+PLACA(I,3)*NSTCUR(I)
ADDCAP = RESMIN(MXYEAR)-C
IF (ADDCAP .LE. 0.0) GO TO 100
THE MINIMUM UNDISCOUNTED CONSTRUCTION COST IS:
COST = ADDCAP*PLACA(IMXPL,6)*(1.0+PLACA(IMXPL,13))/1000.0
THE ADDITIONAL CAPACITY IS ASSUMED ADDED AT THE MIDDLE
OF THE REMAINING YEARS IN THE STUDY PERIOD.
YLEFT = (MXYEAR-NY+1) * 0.5
DISFAC = 1.0/(1.0+DISRAT)**(FLOAT(NY)+YLEFT))
IF (FCR .GT. 0.0) GO TO 12
SALV = COST*(1.0-YLEFT/PLACA(1,9))
COST = (COST-SALV)*DISFAC
GO TO 15
12 PWFC = (1.0-(1.0+DISRAT)**(-YLEFT))/DISRAT
COST = COST*FCR*PWFC*(1+DISRAT)**0.5*DISFAC
THE DISCOUNTED SUM OF THE YEARLY OPERATING COSTS IN THE
REMAINING STUDY PERIOD IS:          C
15 COP = 0.0
NS = NY+1
DO 20 N=NS,MXYEAR
DISFAC = 1.0/((1.0+DISRAT)**(FLOAT(N)+0.5))
20 COP = COP+FATOPE(N)*DISFAC
THE TOTAL COST IS:
COST = COST+COP+OBJ
CHECK THE TOTAL COST AGAINST THE OBJECTIVE FUNCTION UPPER
BOUND
IF (COST .GT. UBOUND) GO TO 30
IFATH = 1
RETURN
C
30 IFATH = 0
RETURN
100 COST = 0.0
IF (NY .LT. MXYEAR) GO TO 15
IFATH = 1
RETURN
END
SUBROUTINE FLOLP(NSTCUR, CULOLP, NSTOLD, OLOLP, NYNEXT)

```



```

      RETURN
20 CONTINUE
CUCU
      CALL MAINT2(NSTCUR,NYNEXT)
      CULOLP=0.0
      DO 40 N=1,4
      PK=PEAKS(NYNEXT,N)
      MAX=MXELOC(NYNEXT,N)
      DO 50 I=1,MAX
      ELL(I)=ELDC(N,I)
      SUMY=0.0
      DO 60 I=1,MXPL
      IF(NSTCUR(I).LE.0) GO TO 60
      JSTOP=NSTCUR(I)
      IP=I
      IN = 2*IDEXP(I)+2
      DO 70 J=1,JSTOP
      Y=PLACA(IP,3) * (1.0-RDM(IN,N))/PK
      P=1.0-PLACA(IP,5)
      SUMY=SUMY+PLACA(IP,3)/PK
      70 CALL PWADD(Y,P,MAXEL)
      60 CONTINUE
      Y=CAPAB(S(NYNEXT)/PK +SUMY
      PLP=PLOLP(Y)
      IF (PLP.GT.CULOLP) CULOLP = PLP
40 CONTINUE
      RETURN
      END
      SUBROUTINE FLOLPC(CULOLP, NSTCUR, NYCR, SYSCAP)
      CUMULANT LOLP
      FUNCTION: CUMULANT BASED CALCULATION OF LOLP. THIS
      CALCULATION DOES NOT CONSIDER MAINTENANCE
      OF THE EXPANSION PLANT TYPES, AND IS DONE
      ONLY FOR THE SEASON WITH THE SMALLEST RESERVE
      MARGIN.
      CALLED BY: CONSTR
      CALLS TO: VALUE
      SUBROUTINE ARGUMENTS
      NAME      DESCRIPTION
      CULOLP    LOLP FOR CURRENT STATE
      NSTCUR    CURRENT STATE (UNPACKED)
      NYCR     CURRENT YEAR
      SYSCAP    SYSTEM CAPACITY
      COMMON VARIABLES USED: EXPLCU, MXPL, SSCUM, SSCUM
      LOCAL VARIABLES
      NAME      DESCRIPTION
      CF3       THIRD EXPANSION COEFFICIENT
      CF4       FORTH EXPANSION COEFFICIENT
      JCUM     JCUM-TH CUMULANT
      SIGMA    STANDARD DEVIATION
      SYSCUM   CURRENT SYSTEM CUMULANTS
      Z        NORMALIZED SYSTEM CAPACITY
      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
      FCR,HOURS,IBASYSR,IMXPL,ITIN,ITMAX,

```

00022450
00022460
00022470
00022480
00022490
00022500
00022510
00022520
00022530
00022540
00022550
00022560
00022570
00022580
00022590
00022600
00022610
00022620
00022630
00022640
00022650
00022660
00022670
00022680
00022690
00022700
00022710
00022720
00022730
00022740
00022750
00022760
00022770
00022780
00022790
00022800
00022810
00022820
00022830
00022840
00022850
00022860
00022870
00022880
00022890
00022900
00022910
00022920
00022930
00022940
00022950
00022960
00022970
00022980
00022990
00023000
00023010
00023020
00023030
00023040
00023050
00023060
00023070
00023080
00023090
00023100
00023110
00023120

```

2      ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3      MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
4      RESMAR,UBOUND          00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350
00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

C      COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
1      INDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
2      IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
3      MAXUN(20,20),MAXADD(8),MINUN(20,20),
4      NEXPID(20),NSTPRE(20,20),NUPB(20,8),
5      PEAKS(20,4),PLACA(20,14),PLANT(210,14),
6      RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
7      FATOP(20),SOL(20,2),AUPEAK(20)          00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350
00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

C      COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+      MXELDC(20,4),X(1250)          00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350
00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

C      COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)
C      COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+      MINTUN(8),NUBOLD(8)          00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350
00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

C      COMMON /TBL/ TINT(120),TY(120)
C      DOUBLE PRECISION TINT,TY
C      COMMON /HYS/ CMULT(4),EVAL(3),EMULT(4),HYDR(3,11),
+      HYEXP(11),HYSCHD(20,8),PSEXP(11),
+      PSSCHD(20,8),SYSCUM(4)
C      DIMENSION NSTCUR(20), SYCUM(4)
C      DOUBLE PRECISION Z
C      CALCULATE THE CURRENT SYSTEM CUMULANTS BY ADDING THE
C      CURRENT STATE PLANT CUMULANTS TO THOSE OF THE SCHEDULED
C      SYSTEM          00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350
00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

C      DO 20 JCUM = 1,4
C      SYCUM(JCUM) = SSCUM(JCUM,NYCR)
C      DO 10 I = 1,MXPL
C      SYCUM(JCUM) = SYCUM(JCUM) + NSTCUR(I) * EXPLCU(JCUM,I)
C 20 CONTINUE          00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350
00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

C      CALCULATE THE GRAM-CHARLIER SERIES COEFFICIENTS THE
C      "MEAN" IS IDENTICAL WITH THE FIRST CUMULANT I.E.
C      MEAN = SYCUM(1)          00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350
00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

C      STANDARD DEVIATION
C      SIGMA = SQRT(SYCUM(2))
C      THIRD AND FORTH COEFFICIENTS
C      CF3 = SYCUM(3)/(SIGMA**3)
C      CF4 = SYCUM(4)/(SIGMA**4)
C      NORMALIZED SYSTEM CAPACITY
C      Z = DBLE((SYSCAP - SYCUM(1))/SIGMA)
C      FIND THE GRAM-CHARLIER SERIES VALUE THAT CORRESPONDS
C      TO Z
C      CALL VALUE(Z, CF3, CF4, CULOLP)
C      RETURN
C      END          00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350
00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

C      SUBROUTINE FORM(IPAGE,K,N,LN,I,FMT1,FMT2,FMT3,
+      FMT4,FMT5,FMT6,FMT7,C,IYR,NP,MX1)          00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350
00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800

```

```

C      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1      FCR,HOURS,IBASRY,IMXPL,ITIN,ITMAX,
2      ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3      MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
4      RESMAR,UBOUND           00023810
                                00023820
                                00023830
                                00023840
                                00023850
                                00023860
                                00023870
C      COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+      MXELDC(20,4),X(1250)          00023880
                                00023890
                                00023900
C      COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
+      HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
+      PSSCHD(20,8),SYSCUM(4)        00023910
                                00023920
                                00023930
                                00023940
                                00023950
+      DIMENSION A1(250),A2(20),FMT1(18),FMT2(18),FMT3(18),
+      FMT4(18),FMT5(18),FMT6(18),FMT7(18),C(60),IYR(20)
                                00023960
      LMX=51                      00023970
      M=N-K+1                      00023980
      N1=(M+11-45)/2+11           00023990
      N2=N1-11
      N3=N1+4
      FMT1(3)=C(N1)               00024000
      FMT2(3)=C(N1)               00024010
      FMT1(14)=C(N2)              00024020
      FMT2(14)=C(N2)              00024030
      FMT3(3)=C(N3)               00024040
      FMT4(5)=C(M)                00024050
      FMT5(5)=C(M)                00024060
      IF (LN.EQ.0) GO TO 350       00024070
      LK=LMX-LN*2-9                00024080
      IF (I.EQ.NP) LK=LK-3         00024090
      IF (NP.EQ.MXPL) GO TO 355   00024100
      GO TO 360                   00024110
      355 WRITE (6,365)             00024120
      365 FORMAT (1H //)            00024130
      WRITE (6,366)               00024140
      366 FORMAT (36H *PLANT COSTS ARE GIVEN AS THE TOTAL,
+      & 8H WORTH OF THE PLANT AS IT COMES ON LINE LESS THE //    00024150
+      & 6H SALVAGE VALUE AT THE END OF THE STUDY PERIOD., //
+      & 4H IF THE FIXED CHARGE RATE OPTION IS USED, //
+      & 3H CONSTRUCTION COSTS REPRESENT THE FIXED,               00024160
+      & 24H CHARGES FOR EACH PLANT.)                          00024170
      LK=LK-8
      360 CONTINUE                 00024180
      DO 320 IJK=1,LK              00024190
      WRITE (6,330)                00024200
      330 FORMAT (1H )              00024210
      320 CONTINUE                 00024220
      350 CONTINUE                 00024230
      MN1=MXALL+3
      IF (NP.EQ.MXPL) WRITE (6,FMT1) IPAGE
      IF (NP.EQ.MN1) WRITE (6,FMT2) IPAGE
      WRITE (6,FMT3) IBASRY
      IF (N.EQ.MX1) WRITE (6,FMT4) (IYR(J),J=K,N)
      IF (N.EQ.MXYEAR) WRITE (6,FMT5) (IYR(J),J=K,N)
      RETURN
      END                         00024280
                               00024290
                               00024300
                               00024310
                               00024320
                               00024330
                               00024340
                               00024350
                               00024360
                               00024370
                               00024380
                               00024390
                               00024400
                               00024410
                               00024420
                               00024430
                               00024440
C      SUBROUTINE MAINTZ(NSTCUR,NVNEXT)


---


      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1      FCR,HOURS,IBASRY,IMXPL,ITIN,ITMAX,
2      ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3      MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,          00024450
                                00024460
                                00024470
                                00024480

```

```

4          RESMAR , UBOUND                               00024490
C          COMMON /ARS/ AVSP(20,4),CAPAB S(20),ENEDEM(20,4),EXPLCU(4,8),
1           IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),               00024500
2           IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),      00024510
3           MAXUN(20,20),MAXADD(8),MINUN(20,20),                00024520
4           NEXPID(20),NSTPRE(20,20),NUPB(20,8),                00024530
5           PEAKS(20,4),PLACA(20,14),PLANT(20,14),              00024540
6           RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),            00024550
7           FATOPE(20),SOL(20,2),AUPEAK(20)                   00024560
00024570
00024580
C          COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+           MXELDC(20,4),X(1250)                                00024590
00024600
00024610
C          COMMON /CMS/ PTCUM(201,8,4),PTCUMXI(20,8),ROM(40,4),PTCUMH(3,8) 00024620
00024630
C          COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+           MINTUN(8),NUBOLD(8)                                00024640
00024650
00024660
C          COMMON /TBL/ TINT(120),TY(120)                      00024670
00024680
C          DOUBLE PRECISION TINT,TY                           00024690
00024700
C          COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
+           HYEXP(11),HYSCHD(20,8),PSEXPN(11),
+           PSSCHD(20,8),SYSCUM(4)                            00024710
00024720
00024730
00024740
00024750
00024760
00024770
00024780
DIMENSION NSTCUR(20),ASP(4)                         00024790
RLT=365./4.
DO 50 J=1,4                                         00024800
DO 50 I=1,40                                         00024810
50 ROM(I,J)=0.0                                       00024820
DO 100 NP=1,4                                         00024830
NP=NP
ASP(N)=AVSP(NYNEXT,N)                                00024840
00024850
00024860
00024870
IF(NSTCUR(I).LE.0) GO TO 110                        00024880
NEXP=NSTCUR(I)
ASP(N)=ASP(N)+FLOAT(NEXP)*PLACA(I,3)                00024890
00024900
110 CONTINUE                                         00024910
100 CONTINUE                                         00024920
00024930
00024940
DO 120 I=1,MXPL                                     00024950
IF(NSTCUR(I).LE.0) GO TO 120                        00024960
NEXP=NSTCUR(I)
IP=IDEXP(I)+2                                       00024970
DO 130 J=1,2                                         00024980
IF(J.GT.1) GO TO 140                                00024990
C=PLACA(I,2)
IF(C.LE.0.0) GO TO 120                             00025000
IN=2*IP
GO TO 150                                         00025030
140 C=PLACA(I,3)-PLACA(I,2)
IF(C.LE.0.0) GO TO 120
IN=2*IP - 1
150 EMXAS=0.0
DO 160 L=1,4
IF(ASP(L).GT.EMXAS) GO TO 170
GO TO 160
170 EMXAS=ASP(L)
LL=L
160 CONTINUE
ROM(IN,LL)=PLACA(I,4)/RLT
130 ASP(LL)=ASP(LL)-FLOAT(NEXP)*ROM(IN,LL)*C
00025040
00025050
00025060
00025070
00025080
00025090
00025100
00025110
00025120
00025130
00025140
00025150
00025160

```

120 CONTINUE
RETURN
END

C
C SUBROUTINE OBJFUN(NSTCUR,NY,OPEC,NSTATE,CAPCO,OBJORI,OBJ,
+S,RESCAP,IRES)

C
C CALCULATES THE DISCOUNTED CONSTRUCTION COST "CAPCO" AND THE
C OBJECTIVE FUNCTION (INCLUDING SALVAGE VALUE) FOR STATE
C "NSTCUR" THAT ORIGINATED FROM STATE "NSTATE"

C
COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1 FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
2 ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3 MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
4 RESMAR,UBOUND

C
COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
1 IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
2 IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
3 MAXUN(20,20),MAXADD(8),MINUN(20,20),
4 NEXPID(20),NSTPRE(20,20),MUPB(20,8),
5 PEAKS(20,4),PLACA(20,14),PLANT(210,14),
6 RESMAX(20),RESMIN(20),SSCLUM(4,20),UNE(20),
7 FATOP(20),SOL(20,2),AUPEAK(20)

C
COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+ MXELDC(20,4),X(1250)

C
COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),RDM(40,4),PTCUMH(3,8)

C
COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+ MINTUN(8),NUBOLD(8)

C
COMMON /TBL/ TINT(120),TY(120)

C
DOUBLE PRECISION TINT,TY

C
COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
+ HYEXP(11),HYSCHO(20,8),PSEXP(11),
+ PSSCHO(20,8),SYSCUM(4)

C
.....

C
DIMENSION NSTCUR(20),NSTATE(20),NADD(20),CONCO(20),RESCAP(20,20)
CALCULATE THE PLANT ADDITIONS

C
DO 10 I=1,MXPL

10 NADD(I) = NSTCUR(I)-NSTATE(I)

C
CALCULATE THE UNDISCOUNTED CONSTRUCTION COST FOR EACH
CANDIDATE TYPE IN MILLION OF DOLLARS. ALSO CALCULATE
THE TOTAL CONSTRUCTION COST

C
C = 0.0

C
DO 20 I=1,MXPL

C
CONCO(I) = NADD(I)*PLACA(I,3)*PLACA(I,6)/1000.0*

20 C = (1.0+PLACA(I,13))**NY

C
C = C+CONCO(I)

C
CALCULATE THE DISCOUNT FACTOR

C
DISFAC = 1.0/(1.0+DISRAT)**NY

C
CALCULATE THE SALVAGE VALUE FOR EACH CANDIDATE USING
STRAIGHT LINE DEPRECIATION. ALSO CALCULATE TOTAL SALVAGE

00025170
00025180
00025190
00025200
00025210
00025220
00025230
00025240
00025250
00025260
00025270
00025280
00025290
00025300
00025310
00025320
00025330
00025340
00025350
00025360
00025370
00025380
00025390
00025410
00025420
00025430
00025440
00025450
00025460
00025470
00025480
00025490
00025500
00025510
00025520
00025530
00025540
00025550
00025560
00025570
00025580
00025590
00025600
00025610
00025620
00025630
00025640
00025650
00025660
00025670
00025680
00025690
00025700
00025710
00025720
00025730
00025740
00025750
00025760
00025770
00025780
00025790
00025800
00025810
00025820
00025830
00025840

```

C      VALUE.
C      S = 0.0
C      DO 30 I=1,MXPL
C      NUMBER OF YEARS OF THE PLANT'S LIFE IN STUDY PERIOD.      00025850
C      NYLEFT = MXYEAR-NY+1                                         00025860
C      PLANT SALVAGE VALUE                                         00025870
C      SALV = CONCO(I)*(1.0-FLOAT(NYLEFT)/PLACA(I,9))           00025880
C      RESCAP(NY,I) = (CONCO(I) - SALV)*DISFAC                  00025890
C      30 S = S+Salv*DISFAC                                         00025900
C
C      IF (FCR .GT. 0.0) GO TO 40
C      CALCULATE DISCOUNTED CAPITAL COST AND OBJECTIVE FUNCTION 00025910
C      CAPCO = C*DISFAC                                           00025920
C      GO TO 50                                                 00025930
C      40 NYLEFT = MXYEAR-NY+1                                     00025940
C      PWFC = (1.0-(1.0+DISRAT)**(-NYLEFT))/DISRAT               00025950
C      DISFAC = 1.0/(1.0+DISRAT)**(NY+0.5)                         00025960
C      C = C*FCR*PWFC                                           00025970
C      CAPCO = C*DISFAC                                         00025980
C      S = 0.0
C
C      50 OBJ = OBJOR I+CAPCO-S+OPEC
C      IF (FCR .LE. 0.0 .OR. IRES .LT. 1) GO TO 100
C      DISFAC = 1.0/(1.0+DISRAT)**NY
C      DO 60 I=1,MXPL
C      CNEW = CONCO(I)*FCR*DISFAC
C      60 RESCAP(NY,I)=RESCAP(NY,I)+CNEW
C      100 RETURN
C      END
C
C      SUBROUTINE OPERCO(NSTCUR, NYNEXT, OPEC, IRES)
C
C
C PURPOSE: CALCULATES THE DISCOUNTED OPERATING COSTS "OPEC"
C FOR STATE "NSTCUR(I)" IN STUDY YEAR "NYNEXT". THE METHOD
C USED IS "PLANT DERATINGS" WITH 100 POINTS P-W REPRESENTATION
C OF THE ORIGINAL LOAD PROBABILITY CURVE(INVERTED LOAD DURA-
C TION CURVE).
C
C THE COMMON VARIABLES ARE DEFINED IN THE MAIN PROGRAM
C
C      *** DEFINITION OF LOCAL VARIABLES ***
C      FAC    REAL    =    DISCOUNT FACTOR
C      IDP    INT.    =    PLANT ID IN SCHEDULED OR CANDIDATE
C
C      P      REAL    =    PLANT AVAILABILITY
C      Y      REAL    =    NORMALIZED AND DERATED PLANT CAPACITY
C      COST   REAL    =    PLANT OPERATING COST IN $/MWH
C
C      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,Dx,
C      FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
C      ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
C      MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
C      RESMAR,UBOUND
C
C      COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
C      IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
C      IWIDTH(8),LOKDER(420,3),LOWB(20,8),MAINS(200),
C      MAXUN(20,20),MAXADD(8),MINUN(20,20),

```

```

4      NEXPID(20),NSTPRE(20,20),NUPB(20,8),          00026530
5      PEAKS(20,4),PLACAI(20,14),PLANT(210,14),    00026540
6      RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),   00026550
7      FATOPE(20),SOL(20,2),AUPEAK(20)            00026560
+
COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),      00026570
+      MXELDC(20,4),X(1250)                      00026580
+
COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8) 00026600
+
COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),      00026620
+      MINFUN(8),NUBOLD(8)                      00026630
+
COMMON /STBL/ TINT(120),TY(120)                  00026640
+
DOUBLE PRECISION TINT,TY                         00026650
+
COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11), 00026660
+      HYEXP(11),HYSCHD(20,8),PSEXP(11),          00026670
+      PSSCHD(20,8),SYSCUM(4)                    00026680
+
+
DIMENSION NSTCUR(20),ITK(420,3),POPEC(250,20),CFH(3),ENRT(220,2), 00026700
1      ENRH(3,2),ENY(2,30),ENP(2,30),NORDER(4)     00026710
EQUIVALENCE (ELF(1)),ITK(1),(ELDC(1),POPEC(1))  00026720
+
NCUM=4
T=HOURS
+
DETERMINE EFFECTIVE LOADING ORDER
+
DO 10 I=1,MXPL                                00026730
IP=IDEXP(I)
IF(IP.LE.0) GO TO 10
DO 11 K=1,MAXOR                                00026740
IF(LORDER(K,1).NE.IP) GO TO 11
LORDER(K,3)=NSTCUR(I)
J1=K+1
DO 12 J=J1,MAXOR                                00026750
IF(LORDER(J,1).NE.IP) GO TO 12
LORDER(J,3)=NSTCUR(I)
GO TO 10
12 CONTINUE
GO TO 10
11 CONTINUE
10 CONTINUE
K1=0
DO 13 K=1,MAXOR                                00026760
IF(LORDER(K,3).LE.0) GO TO 13
K1=K+1
DO 14 J=1,3
14 ITK(K1,J)=LORDER(K,J)
13 CONTINUE
KNMAX=K1
+
EXPN MAINTENANCE ALLOCATION
+
CALL MAINT2(NSTCUR,NYNEXT)                   00027130
+

```

```

        HYDRO COMPOSITE CHARACTERISTICS          00027210
        CALL COMPST(NSTCUR,NYNEXT)                00027220
                                                00027230
                                                00027240
                                                00027250
                                                00027260
                                                00027270
                                                00027280
                                                00027290
                                                00027300
                                                00027310
                                                00027320
                                                00027330
                                                00027340
                                                00027350
                                                00027360
                                                00027370
                                                00027380
                                                00027390
                                                00027400
                                                00027410
                                                00027420
                                                00027430
                                                00027440
                                                00027450
                                                00027460
                                                00027470
                                                00027480
                                                00027490
                                                00027500
                                                00027510
                                                00027520
                                                00027530
                                                00027540
                                                00027550
                                                00027560
                                                00027570
                                                00027580
                                                00027590
                                                00027600
                                                00027610
                                                00027620
                                                00027630
                                                00027640
                                                00027650
                                                00027660
                                                00027670
                                                00027680
                                                00027690
                                                00027700
                                                00027710
                                                00027720
                                                00027730
                                                00027740
                                                00027750
                                                00027760
                                                00027770
                                                00027780
                                                00027790
                                                00027800
                                                00027810
                                                00027820
                                                00027830
                                                00027840
                                                00027850
                                                00027860
                                                00027870
                                                00027880

        DO 15 J=1,2
        DO 15 I=1,220
15 ENRT(I,J)=0.0
        GENTOT=0.0
        HYTOT=0.0
        DTOT=0.0
        UNE(NYNEXT)=0.0
        OPEC=0.0
        CTFUL=0.0
        CTOM=0.0
                                                00027260
                                                00027270
                                                00027280
                                                00027290
                                                00027300
                                                00027310
                                                00027320
                                                00027330
                                                00027340
                                                00027350
                                                00027360
                                                00027370
                                                00027380
                                                00027390
                                                00027400
                                                00027410
                                                00027420
                                                00027430
                                                00027440
                                                00027450
                                                00027460
                                                00027470
                                                00027480
                                                00027490
                                                00027500
                                                00027510
                                                00027520
                                                00027530
                                                00027540
                                                00027550
                                                00027560
                                                00027570
                                                00027580
                                                00027590
                                                00027600
                                                00027610
                                                00027620
                                                00027630
                                                00027640
                                                00027650
                                                00027660
                                                00027670
                                                00027680
                                                00027690
                                                00027700
                                                00027710
                                                00027720
                                                00027730
                                                00027740
                                                00027750
                                                00027760
                                                00027770
                                                00027780
                                                00027790
                                                00027800
                                                00027810
                                                00027820
                                                00027830
                                                00027840
                                                00027850
                                                00027860
                                                00027870
                                                00027880

        *** SECTION 1 ***
        ENERGY & COST CALCULATION FOR EACH SEASON
        LOOP 60
                                                00027440
                                                00027450
                                                00027460
                                                00027470
                                                00027480
                                                00027490
                                                00027500
                                                00027510
                                                00027520
                                                00027530
                                                00027540
                                                00027550
                                                00027560
                                                00027570
                                                00027580
                                                00027590
                                                00027600
                                                00027610
                                                00027620
                                                00027630
                                                00027640
                                                00027650
                                                00027660
                                                00027670
                                                00027680
                                                00027690
                                                00027700
                                                00027710
                                                00027720
                                                00027730
                                                00027740
                                                00027750
                                                00027760
                                                00027770
                                                00027780
                                                00027790
                                                00027800
                                                00027810
                                                00027820
                                                00027830
                                                00027840
                                                00027850
                                                00027860
                                                00027870
                                                00027880

        *** SECTION 2 ***
        BASE LOAD RUN-OF-RIVER CAPACITY
                                                00027660
                                                00027670
                                                00027680
                                                00027690
                                                00027700
                                                00027710
                                                00027720
                                                00027730
                                                00027740
                                                00027750
                                                00027760
                                                00027770
                                                00027780
                                                00027790
                                                00027800
                                                00027810
                                                00027820
                                                00027830
                                                00027840
                                                00027850
                                                00027860
                                                00027870
                                                00027880

        IH=4
        DO 20 K=1,2
        P=1.0-HYDR(K,5)
        Y=HYDR(K,1)*CMULT(N)*(1.0-RDM(2,N))
        IF(Y.LE.0.0) GO TO 20
        Y1=SUMP
        SUMP=SUMP+Y
        Y2=SUMP
        IF(K.EQ.1) GO TO 25
                                                00027660
                                                00027670
                                                00027680
                                                00027690
                                                00027700
                                                00027710
                                                00027720
                                                00027730
                                                00027740
                                                00027750
                                                00027760
                                                00027770
                                                00027780
                                                00027790
                                                00027800
                                                00027810
                                                00027820
                                                00027830
                                                00027840
                                                00027850
                                                00027860
                                                00027870
                                                00027880

        CONVOLVE THE FIRST HYDRO
                                                00027790
                                                00027800
                                                00027810
                                                00027820
                                                00027830
                                                00027840
                                                00027850
                                                00027860
                                                00027870
                                                00027880

        DO 24 J=1,NCUM
24 SYSCUM(J)=SYSCUM(J)+PTCUMH(1,J)
                                                00027830
                                                00027840
                                                00027850
                                                00027860
                                                00027870
                                                00027880

        EVALUATE AREA UNDER EL-CURVE

```

```

C   25 CALL AREA(Y1,Y2,P,EH,CF,CFP,IM,N)          00027890
C     ENRH(K,1)=EH                                00027900
C
C   MAX. HYDRO ENERGY AVAILABLE                  00027920
C
C     EAVAL(K)=HYDR(K,4)*EMULT(N)*1000.0-EH      00027930
C     C=HYDR(K,2)-HYDR(K,1)*CMULT(N)              00027940
C     IF(C.LE.0.0) GO TO 20                         00027950
C     CFH(K)=EAVAL(K)/(C*T)                        00027960
C   20 CONTINUE                                     00027970
C     EPS=HYDR(3,2)*1000.0                          00027980
C
C   CONVOLVE THE SECOND HYDRO                   00028000
C
C     DO 27 J=1,NCUM                             00028010
C   27 SYSCUM(J)=SYSCUM(J)+PTCUMH(2,J)           00028020
C
C   ORDER OF HYDRO PLANT IN DECREASING CAPACITY FACTOR
C
C     NUMBER=0                                     00028030
C     IF(CFM(1).GE.CFH(2)) GO TO 800             00028040
C     NORDER(1)=2                                 00028050
C     IF(CFH(1).LE.0.0) GO TO 802                 00028060
C     NORDER(2)=1                                 00028070
C     NUMBER=2                                 00028080
C     GO TO 801                                 00028090
C   800 IF(CFH(1).LE.0.0) GO TO 801             00028100
C     NORDER(1)=1                               00028110
C     IF(CFH(2).LE.0.0) GO TO 802                 00028120
C     NORDER(2)=2                               00028130
C     NUMBER=2                                 00028140
C     GO TO 801                                 00028150
C   801 IF(EPS.LE.0.0) GO TO 803                 00028160
C     NUMBER=NUMBER+1                           00028170
C     NORDER(NUMBER)=3                         00028180
C   803 NUMBER=NUMBER+1                           00028190
C     NORDER(NUMBER)=4                         00028200
C
C     GO TO 801                                 00028210
C   802 NUMBER=1                                 00028220
C   801 IF(EPS.LE.0.0) GO TO 803                 00028230
C     NUMBER=NUMBER+1                           00028240
C     NORDER(NUMBER)=3                         00028250
C   803 NUMBER=NUMBER+1                           00028260
C     NORDER(NUMBER)=4                         00028270
C
C   802 NUMBER=1                                 00028280
C   801 IF(EPS.LE.0.0) GO TO 803                 00028290
C     NUMBER=NUMBER+1                           00028300
C     NORDER(NUMBER)=3                         00028310
C   803 NUMBER=NUMBER+1                           00028320
C     NORDER(NUMBER)=4                         00028330
C
C   INITIALIZE INDEX FOR HYDRO LOADING ORDER
C
C     NOHY=1                                    00028340
C     IH=NORDER(NOHY)                           00028350
C     IPUMP=-1                                 00028360
C     IF(EPS.LE.0.0) IPUMP=1                     00028370
C     KLOW1=0                                 00028380
C     KLOW2=0                                 00028390
C     SUEN2=0 .0                                00028400
C
C     IPUMP=0                                 00028410
C     IF(EPS.LE.0.0) IPUMP=1                     00028420
C     KLOW1=0                                 00028430
C     KLOW2=0                                 00028440
C     SUEN2=0 .0                                00028450
C
C     *** SECTION 3 ***
C
C     START LOADING THE THERMAL PLANTS ,
C     LOOP 70
C
C     DO 70 KC=1,KNMAX
C       K=KC

```

```

I01=ITK(K,2)          00028570
IP=ITK(K,1)           00028580
IUNITB=1              00028590
IUNITE=ITK(K,3)       00028600
C
DO 630 J=1,30         00028610
ENP(2,J)=0.0          00028620
630 ENY(2,J)=0.0       00028630
C
INDEX FOR EMOR ARRAY 00028640
C
IF(IP.GT.MAXINP) GO TO 55 00028650
IF(ID1.GT.1) GO TO 50     00028660
IMANT=2*(IP+2)            00028670
R=1.0-RUM(IMANT,N)        00028680
GO TO 56                 00028690
50 IMANT=2*IP+3           00028700
R=1.0-RUM(IMANT,N)        00028710
GO TO 56                 00028720
55 JP=IP-MAXINP          00028730
R=1.0                   00028740
IF(MAINS(JP).EQ.N) R=1.0-PLANT(IP,4)/(365.0/4.0) 00028750
56 PT=1.0-PLANT(IP,5)    00028760
IF(ID1.GT.1) GO TO 57     00028770
YT=PLANT(IP,2)*R          00028780
GO TO 74                 00028790
57 YT=(PLANT(IP,3)-PLANT(IP,2))*R      00028800
C
LOOP FOR IDENTICAL UNITS 00028810
C
LOOP 75                 00028820
C
74 DO 75 ISTAR=IUNITB,IUNITE 00028830
I=ISTAR                 00028840
Y1=SUMP                  00028850
SUMP=SUMP+YT             00028860
Y2=SUMP                  00028870
IF(I.EQ.1) GO TO 76       00028880
IQ=IP                    00028890
ID2=ID1                  00028900
GO TO 79                 00028910
76 IF(K.EQ.1) GO TO 150   00028920
K1X=K-1                  00028930
IQ=ITK(K1X,1)             00028940
ID2=ITK(K1X,2)            00028950
C
PLANT CONVOLUTION        00028960
C
79 DO 90 J=1,NCUM         00028970
J1=J                     00028980
IF(ID2.GT.1) J1=J+4       00028990
IF(IQ.GT.MAXINP) GO TO 92 00029000
SYSCUM(J)=SYSCUM(J)+PTCUMX(IW,J1) 00029010
GO TO 90                 00029020
92 JC=IQ-MAXINP+1         00029030
SYSCUM(J)=SYSCUM(J)+PTCUM(JC,J1,N) 00029040
90 CONTINUE                00029050
IF(ID1.LE.1) GO TO 150   00029060
C
PLANT DECONVOLUTION      00029070
C

```

```

C          DO 100 J=1,NCUM           00029250
C          IF(IP.GT.MAXINP) GO TO 101 00029260
C          SYSCUM(J)=SYSCUM(J)-PTCUMX(IP,J) 00029270
C          GO TO 100 00029280
101 JC=IP-MAXINP+1 00029290
C          SYSCUM(J)=SYSCUM(J)-PTCUM(JC,J,N) 00029300
100 CONTINUE 00029310
C
C          150 CONTINUE 00029320
C
C          EVALUATE AREA UNDER THE EL-CURVE 00029330
C
C          152 CALL AREA(Y1,Y2,PT,ET,CF,CFP,IH,N) 00029340
C          ENY(2,I)=ET 00029350
C
C          ANY MORE HYDRO LEFT 00029360
C
C          IF(IH.GE.4) GO TO 75 00029370
C
C          CHECK HYDRO GENERATION FEASIBILITY 00029380
C
C          IF(CF.LE.CFH(IH)) GO TO 200 00029400
C
C          CHECK PUMPING DUTY ASSIGNMENT 00029410
C
C          IF(IPUMP.GT.0.OR.CFP.GE.1.0) GO TO 75 00029420
C
C          PUMPING ASSIGNMENT 00029430
C
C          PPP=1.0-HYDR(3,5) 00029440
C          YP1=Y1-HYDR(3,1)*(1.0-ROM(3,N)) 00029450
C          YP2=Y2-HYDR(3,1)*(1.0-ROM(3,N)) 00029460
C          CALL AREA(YP1,YP2,PT,EP,CF,CFP,IH,N) 00029470
C          ENP(2,I)=PP*(EP-ET) 00029480
C          EPMP=EPMP+ENP(2,I)*HYDR(3,3) 00029490
C          IF(EPMP.LT.EPS) GO TO 570 00029500
C
C          STOP PUMPING 00029510
C
C          IPUMP=1 00029520
C          ENP(2,I)=ENP(2,I)-(EPMP-EPS)/HYDR(3,3) 00029530
C          EPMP=EPS 00029540
C          EAVAL(3)=EPS*HYDR(3,4) 00029550
C          CFH(3)=EAVAL(3)/(T*HYDR(3,1)*(1.0-HYDR(3,5))) 00029560
C          GO TO 75 00029570
C
C          570 EAVAL(3)=EPMP*HYDR(3,4) 00029580
C          CFH(3)=EAVAL(3)/(T*HYDR(3,1)*(1.0-HYDR(3,5))) 00029590
C          GO TO 75 00029600
C
C          HYDRO GENERATION 00029610
C
C

```

```

C
200 CONTINUE          00029930
ICNTL=-1              00029940
C
REPLACE THE THERMAL BY HYDRO          00029950
C
SUMP=SUMP-YT          00029960
ENY(2,I)=0.0           00029970
ENP(2,I)=0.0           00029980
IF(IDI.LE.1) GO TO 215          00029990
C
RETRIEVE EL-CURVE INFO          00030000
C
DO 210 J=1,NCUM          00030010
IF(IP.GT.MAXINP) GO TO 211          00030020
SYSCUM(J)=SYSCUM(J)+PTCUMX(IP,J)
GO TO 210          00030030
211 JC=IP-MAXINP+1          00030040
SYSCUM(J)=SYSCUM(J)+PTCUM(JC,J,N)
210 CONTINUE          00030050
215 CONTINUE          00030060
C
DECONVNE BASE LOADED HYDRO          00030070
C
216 DO 220 J=1,NCUM          00030080
220 SYSCUM(J)=SYSCUM(J)-PTCUMH(IH,J)
C
*          00030090
PH=1.0-HYDR(IH,5)          00030100
YH=(HYDR(IH,2)-HYDR(IH,1))*CMULT(N)*(1.0-ROM(1,N))
IF(IH.EQ.3) YH=HYDR(3,1)*(1.0-ROM(3,N))
Y1=SUMP          00030110
SUMP=SUMP+YH          00030120
Y2=SUMP          00030130
CALL AREA(Y1,Y2,PH,EH,CF,CFP,IH,N)
ENRH(IH,2)=EH          00030140
IF(ICNTL.GT.0) GO TO 240          00030150
KLOW1=K          00030160
KLOW2=I-1          00030170
C
CONVOLVE THIS HYDRO          00030180
C
DO 230 J=1,NCUM          00030190
J1=J+4          00030200
230 SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J1)
C
NEXT HYDRO          00030210
C
IF(IH.EQ.3) GO TO 620          00030220
NOHY=NOHY+1          00030230
IH=NORDER(NOHY)
ICNTL=1          00030240
GO TO 216          00030250
240 IF(EH.GT.EAVAL(IH)) GO TO 250          00030260
ENRH(IH,2)=EH          00030270
.KLOW1=K          00030280
          00030290
          00030300
          00030310
          00030320
          00030330
          00030340
          00030350
          00030360
          00030370
          00030380
          00030390
          00030400
          00030410
          00030420
          00030430
          00030440
          00030450
          00030460
          00030470
          00030480
          00030490
          00030500
          00030510
          00030520
          00030530
          00030540
          00030550
          00030560
          00030570
          00030580
          00030590
          00030600

```

```

KLOW2=I-1          00030610
GO TO 300          00030620
250 CONTINUE        00030630
                      00030640
                      00030650
                      00030660
                      00030670
                      00030680
REPLACE THIS HYDRO BY THERMAL 00030690
                               00030700
                               00030710
                               00030720
                               00030730
                               00030740
                               00030750
270 DO 270 J=1,NCUM 00030760
      SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J)
      NOHY=NOHY-1    00030770
      IH=NORDER(NOHY) 00030780
                      00030790
      620 IBLK=-1     00030800
      CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP,
      +NORDER,EPMP)   00030810
      IPUMP=1         00030820
      GO TO 455       00030830
                      00030840
                      00030850
                      00030860
                      00030870
300 CONTINUE          00030880
FORM CLUSTER AND TEST SWAP LOADING POSITION 00030890
NPAST=NOHY-1        00030900
IHPY=NORDER(NPAST) 00030910
IBLK=1              00030920
IF(KLOW2.LE.0) GO TO 301 00030930
KIX=KLOW1           00030940
ISWP=2              00030950
ITRM=KLOW2          00030960
GO TO 302          00030970
301 KIX=KLOW1-1    00030980
ISWP=1              00030990
ITRM=ITK(KIX,3)    00031000
302 II1=ITK(KIX,1)  00031010
II2=ITK(KIX,2)    00031020
DO 310 J=1,NCUM   00031030
J1=J
IF(II2.GT.1) J1=J+4
IF(II1.GT.MAXINP) GO TO 311
SYSCUM(J)=SYSCUM(J)-PTCUMX(II1,J1)
GO TO 310
311 JC=II1-MAXINP+1 00031040
SYSCUM(J)=SYSCUM(J)-PTCUM(JC,J1,N)
310 CONTINUE        00031050
DO 320 J=1,NCUM   00031060
J1=J+4
320 SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J1)
P=1.0-PLANT(II1,5)
IF(II1.GT.MAXINP) GO TO 322
IF(II2.GT.1) GO TO 321
IMANT=2*(II1+2)
R=1.0-PLANT(IMANT,N)
GO TO 325
321 IMANT=2*II1+3
R=1.0-PLANT(IMANT,N)
GO TO 325
322 JP=II1-MAXINP
R=1.0
IF(MAINS(JP).EQ.N) R=1.0-PLANT(II1,4)/(365.0/4.0)
325 IF(II2.GT.1) GO TO 326
Y=PLANT(II1,2)*R
GO TO 327
326 Y=(PLANT(II1,3)-PLANT(II1,2))*R

```

```

327 Y2=SUMP          00031290
Y1=SUMP-Y           00031300
CALL AREA(Y1,Y2,P,ET,CF,CFP,IH,N) 00031310
SUEN1=ENY(ISWP,ITRM)+ENRH(IH,2)+ENRH(IHPT,2) 00031320
SUEN2=SUEN1-ET      00031330
ETOT=EAVAL(IH)+EAVAL(IHPT) 00031340
IF(SUEN2.GT.ETOT) GO TO 400 00031350
00031360
00031370
00031380
00031390
00031400
00031410
00031420
00031430
00031440
00031450
00031460
00031470
00031480
00031490
00031500
00031510
00031520
00031530
00031540
00031550
00031560
00031570
00031580
00031590
00031600
00031610
00031620
00031630
00031640
00031650
00031660
00031670
00031680
00031690
00031700
00031710
00031720
00031730
00031740
00031750
00031760
00031770
00031780
00031790
00031800
00031810
00031820
00031830
00031840
00031850
00031860
00031870
00031880
00031890
00031900
00031910
00031920
00031930
00031940
00031950
00031960

ACCEPT SWAP TEST

ENY(ISWP,ITRM)=ET

KLOW1=K
KLOW2=I-2

EPMP=EPMP-ENP(ISWP,ITRM)*HYDR(3,3)
ENP(ISWP,ITRM)=0.0
CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP,
+NORDER,EPMP)
IPUMP=1
GO TO 450

400 CONTINUE

REJECT SWAP TEST

SUEN2=ENRH(IH,2)+ENRH(IHPT,2)
CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP,
+NORDER,EPMP)
IPUMP=1
450 CONTINUE

LOAD THE CURRENT OFF-LOADED THERMAL PLANT

DO 460 J=1,NCUM
J1=J
IF(III2.GT.1) J1=J+4
IF(III1.GT.MAXINP) GO TO 461
SYSCUM(J)=SYSCUM(J)+PTCUMX(III,J1)
GO TO 460
461 JC=III-MAXINP+1
SYSCUM(J)=SYSCUM(J)+PTCUM(JC,J1,N)
460 CONTINUE

455 NOHY=NOHY+1
IH=NORDER(NOHY)

Y1=SUMP
SUMP=SUMP+YT
Y2=SUMP
IF(ID1.NE.2) GO TO 470
DO 468 J=1,NCUM
IF(IP.GT.MAXINP) GO TO 469
SYSCUM(J)=SYSCUM(J)-PTCUMX(IP,J)
GO TO 468
469 JC=IP-MAXINP+1
SYSCUM(J)=SYSCUM(J)-PTCUM(JC,J,N)
468 CONTINUE

```

```

470 CALL AREA(Y1,Y2,PT,ET,CF,CFP,IM,N)           00031970
ENY(2,I)=ET

75 CONTINUE                                     00031980
                                                 00031990
                                                 00032000
                                                 00032010
                                                 00032020
                                                 00032030
                                                 00032040
                                                 00032050
                                                 00032060
                                                 00032070
                                                 00032080
                                                 00032090
                                                 00032100
                                                 00032110
                                                 00032120
                                                 00032130
                                                 00032140
                                                 00032150
                                                 00032160
                                                 00032170
                                                 00032180
                                                 00032190
                                                 00032200
                                                 00032210
                                                 00032220
                                                 00032230
                                                 00032240
                                                 00032250
                                                 00032260
                                                 00032270
                                                 00032280
                                                 00032290
                                                 00032300
                                                 00032310
                                                 00032320
                                                 00032330
                                                 00032340
                                                 00032350
                                                 00032360
                                                 00032370
                                                 00032380
                                                 00032390
                                                 00032400
                                                 00032410
                                                 00032420
                                                 00032430
                                                 00032440
                                                 00032450
                                                 00032460
                                                 00032470
                                                 00032480
                                                 00032490
                                                 00032500
                                                 00032510
                                                 00032520
                                                 00032530
                                                 00032540
                                                 00032550
                                                 00032560
                                                 00032570
                                                 00032580
                                                 00032590
                                                 00032600
                                                 00032610
                                                 00032620
                                                 00032630
                                                 00032640

```

END OF IDENTICAL UNITS

IF(K.LE.-1) GO TO 80

KIX=K-1
L1=ITK(KIX,1)
L2=ITK(KIX,2)
L3=ITK(KIX,3)
DO 77 L=L3

GENTOT=GENTOT+ENV(1,L)

77 ENRT(L1,L2)=ENRT(L1,L2)+ENV(1,L)+ENP(1,L)

80 DO 78 I=IUNITB,IUNITE

ENP(1,I)=ENP(2,I)

78 ENV(1,I)=ENY(2,I)

IF(K.LT.KNMAX) GO TO 70

DO 81 I=IUNITB,IUNITE

GENTOT=GENTOT+ENV(2,I)

81 ENRT(IP, ID1)=ENRT(IP, ID1)+ENV(2,I)+ENP(2,I)

70 CONTINUE

END OF SECTION 3

IF(IM.GE.4) GO TO 66

IP=ITK(KNMAX,1)
ID=ITK(KNMAX,2)
DO 600 J=1,NCUM

J1=J

IF(ID.GT.1) J1=J+4

IF(IP.GT.MAXINP) GO TO 601

SYSCUM(J)=SYSCUM(J)+PTCUMX(IP,J1)

GO TO 600

601 JC=IP-MAXINP+1

SYSCUM(J)=SYSCUM(J)+PTCUM(JC,J1,N)

600 CONTINUE

VH=HYDR(3,1)*(1.0-ROM(3,N))

PH=1.0-HYDR(3,5)

V1=SUMP

V2=SUMP+VH

CALL AREA(Y1,Y2,PH,EH,CF,CFP,IM,N)

IF(EH.GT.EAVAL(3)) GO TO 610

ENRH(3,2)=EAVAL(3)

ENRT(IP, ID)=ENRT(IP, ID)-(EAVAL(3)-EH)

GO TO 66

610 ENRH(3,2)=EAVAL(3)

66 DO 67 J=1,2

DO 67 I=1,2

67 HYTOT=HYTOT+ENRH(I,J)

GENTOT=GENTOT+ENRH(3,2)

DO 350 I=1,3

```

      FIX=HYDR(I,7)
      VAB=HYDR(I,8)
      FAC=(1.0/(1.0+DISRAT))**FLOAT(NYNEXT)+0.5
      DO 351 J=1,2
      IF(I.GE.3) GO TO 352
      C=HYDR(I,1)*CMULT(N)
      IF(J.GT.1) C=HYDR(I,2)-C
      GO TO 353
352  IF(J.LE.1) GO TO 351
      C=HYDR(I,1)
353  COSTOP=(ENRH(I,J)*VAB+C*FIX)*FAC/1.0E+06
      IF(IRES.NE.1) GO TO 354
      IX=MXALL+I
      CTOM=CTOM+COSTOP
      POPEC(IX,NYNEXT)=POPEC(IX,NYNEXT)+COSTOP
354  OPEC=OPEC+COSTOP
351  CONTINUE
350  CONTINUE
C
C       60 CONTINUE
C
C       END OF SECTION 2
C
C
C       *** S E C T I O N   4 ***
C
C       COST CALCULATION
C
      DO 700 I=1,MXALL
      IF(I.GT.MAXINP) GO TO 718
      DO 717 J=1,MXPL
      IP=IDEXP(J)
      IF(I.EQ.IP.AND.NSTCUR(J).GT.0) GO TO 718
717  CONTINUE
      GO TO 700
C
C       718 FIX=PLANT(I,10)
      VAB=PLANT(I,11)
      DO 710 J=1,2
      IF(J.GT.1) GO TO 715
      CFL=PLANT(I,7)
      C=PLANT(I,2)
      GO TO 720
715  CFL=PLANT(I,8)
      C=PLANT(I,3)-PLANT(I,2)
720  FAC=((1.0+PLANT(I,14))/(1.0+DISRAT))**FLOAT(NYNEXT)+0.5
      CFUEL=CFL*ENRT(I,J)*FAC/1.0E+06
      COM=(VAB*ENRT(I,J)+FIX*C)*FAC/1.0E+06
      IF(IRES.NE.1) GO TO 716
      CTFUL=CTFUL+CFUEL
      CTOM=CTOM+COM
      POPEC(I,NYNEXT)=POPEC(I,NYNEXT)+CFUEL+COM
716  OPEC=OPEC+CFUEL+COM
710  CONTINUE
700  CONTINUE
C
C
C       GENTOT=GENTOT+HYTOT
C
C       UNE(NYNEXT)=DTOT-GENTOT

```

```

IF(UNE(NYNEXT).LT.0.0) UNE(NYNEXT)=0.0          00033330
DO 500 I=1,MXPL                               00033340
IP=IDEXP(I)                                     00033350
IF(IP.LE.0) GO TO 500                         00033360
DO 510 K=1,MAXOR                             00033370
IF (LORDER(K,1).NE.IP) GO TO 510             00033380
LORDER(K,3)=-1                                00033390
JJ=K+1                                         00033400
DO 520 J=JJ,MAXOR                            00033410
IF(LORDER(J,1).NE.IP) GO TO 520             00033420
LORDER(J,3)=-1                                00033430
GO TO 500                                      00033440
520 CONTINUE                                    00033450
GO TO 500                                      00033460
510 CONTINUE                                    00033470
500 CONTINUE                                    00033480
IF(IRES.NE.1) GO TO 1500                      00033490
WRITE(40,1400) CTFUL
WRITE(40,1400) CTOM
1400 FORMAT(E12.5)
1500 RETURN
END

```

SUBROUTINE PACK(NUNP, NPA)

TITLE: STATE PACKING
FUNCTION: THIS ROUTINE PACKS CURRENT STATE "NSTCUR" INTO
ONE WORD "NST" USING PACKING BASE VECTOR IPABA

CALLED BY: MAIN
CALLS TO: NONE

SUBROUTINE ARGUMENTS
NAME DESCRIPTION
NPA PACKED STATE
NUNP UNPACKED STATE

COMMON VARIABLES USED: IPABA, MXPL

LOCAL VARIABLES
NAME DESCRIPTION
I I-TH EXPANSION PLANT TYPE

COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
ITOUT,MAXALL,MAXI,MAXIMP,MAXOK,
MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
RESMAR,UBOUND

COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
MAXUN(20,20),MAXADD(8),MINUN(20,20),
NEXPID(20),NSTPRE(20,20),NUPB(20,8),
PEAKS(20,4),PLACA(20,14),PLANT(210,14),

```

6           RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),          00034010
7           FATOPE(20),SOL(20,2),AUPEAK(20)                   00034020
C   COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),          00034030
+           MXELDC(20,4),X(1250)
C   COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8) 00034040
C   COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),          00034050
+           MINTUN(8),NUBOLD(8)                           00034060
C   COMMON /TBL/ TINT(120),TY(120)                         00034070
C   DOUBLE PRECISION TINT,TY                               00034080
C   COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),      00034090
+           HYEXPN(11),HYSCHD(20,8),PSEXPN(11),            00034100
+           PSSCHD(20,8),SYSCUM(4)                         00034110
C   DIMENSION NUNP(20)                                     00034120
C   INITIALIZE NST                                         00034130
C   NPA = 0                                                 00034140
C   PACK NSTCUR                                         00034150
C   DO 10 I=1,MXPL                                      00034160
10  NPA = NPA+NUNP(I)*IPABA(I)
    RETURN
    END
C   -----
C   FUNCTION PLOLP(Y)
C   -----
C   FIND LOLP. I.E. THE EL(I) POINT THAT CORRESPONDS TO Y
C   Y: NORMALIZED MW CAPACITY
C   COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,          00034350
1    FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,          00034360
2    ITOUT,MAXALL,MAXI,MAXINP,MAXOR,          00034370
3    MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,          00034380
4    RESMAR,UBOUND                                     00034390
C   COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8), 00034400
1    IDEXP(20),IPABA(6),IPCH(8),ISOL(20,8),          00034410
2    IWIDTH(8),LORDER(420,3),LOW8(20,8),MAINS(200), 00034420
3    MAXUN(20,20),MAXADD(8),MINUN(20,20),          00034430
4    NEXPID(20),NSTPRE(20,20),NUPB(20,8),          00034440
5    PEAKS(20,4),PLACA(20,14),PLANT(210,14),        00034450
6    RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),        00034460
7    FATOPE(20),SOL(20,2),AUPEAK(20)                  00034470
C   COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),          00034480
+           MXELDC(20,4),X(1250)
C   COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8) 00034490
C   COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),          00034500
+           MINTUN(8),NUBOLD(8)                           00034510
C   COMMON /TBL/ TINT(120),TY(120)                         00034520
C   DOUBLE PRECISION TINT,TY                               00034530
C   COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),      00034540
+           HYEXPN(11),HYSCHD(20,8),PSEXPN(11),            00034550
+           PSSCHD(20,8),SYSCUM(4)                         00034560
C   -----

```

```

* PSSCHD(20,8),SYSCUM(4)

YY = Y*MAXPO
IY = YY
MAXIM = MAXI-1
IF (IY .GT. MAXIM) GO TO 100
DY = IY+1-Y
PLOLP = (EL(IY)-EL(IY+1))*DY+EL(IY+1)
RETURN
PLOLP = 0.0
RETURN
END

SUBROUTINE PWADD(Y,P,NEWMAX)

DETERMINES THE NEW ELDC AFTER THE ADDITION OF THE UNIT WITH
NORMALIZED MW CAPACITY Y AND RELIABILITY P.

COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1 FCR,HOURS,IBASYSR,IMXPL,ITIN,ITMAX,
2 ITOUT,MAXALL,MAXI,MAXINP,MAXDR,
3 MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
4 RESMAR,UBOUND

COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
1 IDEXP(20),IPABA(8),IPCH18!,ISDL(20,8),
2 INIDTH(8),LORDR(420,3),LOWB(20,8),MAINS(200),
3 MAXUN(20,20),MAXADD(8),MINUN(20,20),
4 NEXPID(20),NSTPRE(20,20),NUPB(20,8),
5 PEAKS(20,4),PLACA(20,14),PLANT(210,14),
6 RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
7 FATOPE(20),SOL(20,2),AUPEAK(20)

COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+ MXELDC(20,4),X(1250)

COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)

COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+ MINTUN(8),NLBOLD(8)

COMMON /TBL/ TINT(120),TY(120)

DOUBLE PRECISION TINT,TY

COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
+ HYEXP(11),HYSCHD(20,98),PSEXPN(11),
+ PSSCHD(20,8),SYSCUM(4)

DEFINITION OF TERMS:

X: X-AXIS I.E. NORMALIZED MW AXIS
EL: LOAD PROBABILITY MATRIX
ELF: TEMPORARY LOAD PROBABILITY MATRIX. STORES THE NEW
EL TEMPORARILY
Y: NORMALIZED MW CAPACITY OF ADDED UNIT: Y=MWC/PERPK

```

```

P=1-FOR, LOAD PROBABILITY          00035370
Q=FOR  FORCED ADUTAGE RATE        00035380
MAXI: EXPECTED MAXIMUM DIMENSION OF EL MATRIX. 00035390
PMAX: =0.999 IS CONSIDERED A REALISTIC LIMIT OF UNIT 00035400
      AVAILABILITY ESTIMATE. I.E. IF P>PMAX 00035410
      IT IS ASSUMED THAT P=1.0 AND THE ELD 00035420
      CURVE DOES NOT CHANGE 00035430
      ELMIN=0.0000001, IS THE LIMIT OF LOLP ACCURACY. 00035440
      IF LOLP < ELMIN THEN LOLP = 0.0 00035450
00035460
00035470
00035480
00035490
00035500
00035510
00035520
00035530
00035540
00035550
00035560
00035570
00035580
00035590
00035600
00035610
00035620
00035630
00035640
00035650
00035660
00035670
00035680
00035690
00035700
00035710
00035720
00035730
00035740
00035750
00035760
00035770
00035780
00035790
00035800
00035810
00035820
00035830
00035840
00035850
00035860
00035870
00035880
00035890
00035900
00035910
00035920
00035930
00035940
00035950
00035960
00035970
00035980
00035990
00036000
00036010
00036020
00036030
00036040

IF(P.GE.PMAX) GO TO 100
FORM THE NEW ELD CURVE. STORE IN ELF
Q=ONE-P
J=1
DO 50 I=1,MAXI
XY=X(I)-Y
IF(XY.LE.X(1)) GO TO 30
J=J+1
25 IF(ABS(XY-X(J)).LE.ELMIN) GO TO 40
ELY=(X(J-1)-XY)/(X(J-1)-X(J))*EL(J)+*
(XY-X(J))/(X(J-1)-X(J))*EL(J-1)
GO TO 45
30 ELY=ONE
GO TO 45
40 ELY=EL(J)
45 CONTINUE
ELF(I)=P*EL(I)+Q*ELY
MAXIMU=I
IF (ELF(I).LE.ELMIN) GO TO 55
50 CONTINUE
55 NEWMAX=MAXIMU
IF (NEWMAX.GT.MAXI) NEWMAX=MAXI
ALL ELD VALUES FOR WHICH I SATISFIED THE :
NEWMAX<I<MAXI, HAVE EL(I)<ELMIN AND
ARE LEFT TO BE ZERO. (NOTE THAT THE EL MATRIX
IS INITIALIZED TO ZERO IN LOADSY)
REDEFINE THE EL MATRIX AS THE NEW ELD CURVE
THAT IS TEMPORARILY STORED IN ELF.
DO 60 I=1,NEWMAX
EL(I)=ELF(I)
60 CONTINUE
100 CONTINUE
RETURN
END

SUBROUTINE READIN


---


COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1 FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
2 ITOUT,MAXALL,MAX1,MAXINP,MAXOR,
3 MAXPLA,MAXPO,MAXLL,MXPL,MXYEAR,
4 RESMAR,UBOUND
COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),

```

```

1 INDEX(20),IPABA(8),IPCH(8),ISQL(20,8),  

2 IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),  

3 MAXUN(20,20),MAXADD(8),MINUN(20,20),  

4 NEXPID(20),NSTPRE(20,20),NUPB(20,8),  

5 PEAKS(20,4),PLACAA(20,14),PLANT(210,14),  

6 RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),  

7 FATOPe(20),SOL(20,2),AUPEAK(20) 00036050  

00036060  

00036070  

00036080  

00036090  

00036100  

00036110  

00036120  

00036130  

00036140  

00036150  

00036160  

00036170  

00036180  

00036190  

00036200  

00036210  

00036220  

00036230  

00036240  

00036250  

00036260  

00036270  

00036280  

00036290  

00036300  

00036310  

00036320  

00036330  

00036340  

00036350  

00036360  

00036370  

00036380  

00036390  

00036400  

00036410  

00036420  

00036430  

00036440  

00036450  

00036460  

00036470  

00036480  

00036490  

00036500  

00036510  

00036520  

00036530  

00036540  

00036550  

00036560  

00036570  

00036580  

00036590  

00036600  

00036610  

00036620  

00036630  

00036640  

00036650  

00036660  

00036670  

00036680  

00036690  

00036700  

00036710  

00036720

```

COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+ MXELDC(20,4),X(1250) 00036130
COMMON /CMS/ PTCUM(20,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8) 00036140
COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+ MINTUN(8),NUBOLD(8) 00036150
COMMON /TBL/ TINT(120),TY(120) 00036170
DOUBLE PRECISION TINT,TY 00036180
COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
+ HYEXP(11),HYSCHD(20,8),PSEXPN(11),
+ PSSCHD(20,8),SYSCUM(4) 00036190
00036200
00036210
00036220
00036230
00036240
00036250
00036260
00036270
00036280
00036290
00036300
00036310
00036320
00036330
00036340
00036350
00036360
00036370
00036380
00036390
00036400
00036410
00036420
00036430
00036440
00036450
00036460
00036470
00036480
00036490
00036500
00036510
00036520
00036530
00036540
00036550
00036560
00036570
00036580
00036590
00036600
00036610
00036620
00036630
00036640
00036650
00036660
00036670
00036680
00036690
00036700
00036710
00036720

CRITICAL LOAD, HOURS PER SEASON
READ(10,902) CPLDLP, HOURS
WRITE(11,902) CPLDLP, HOURS
BASE YEAR (CALENDAR), MAX.ELDC DIMENSION, NUMBER OF NEW PLANTS
NUMBER OF SCHEDULED PLANTS, ORIGINAL ELOC DIMENSION,
NUMBER OF YEARS IN THE STUDY PERIOD
READ(10,900) IBASYR,MAXI,MAXINP,MAXPLA,MAXPO,MYEAR,MAXOR
WRITE(11,900) IBASYR,MAXI,MAXINP,MAXPLA,MAXPO,MYEAR,MAXOR
MAXIMUM NUMBER OF POINTS OF SCHEDULED SYSTEM FINAL
ELDC'S. (BY SEASON PER YEAR)
DO 10 NY = 1, MYEAR
READ(10,900) (MXELDC(NY,I),I = 1,4) 00036420
10 WRITE(11,900) (MXELDC(NY,I),I = 1,4) 00036430
READ FOR EACH YEAR: SCHEDULED SYSTEM CAPACITY,
MIN. & MAX. RES. MARGINS, MINIMUM POSSIBLE YEARLY
PRODUCTION COST
DO 20 NY = 1, MYEAR
READ(10,902) CAPABS(NY),RESMIN(NY),RESMAX(NY),FATOPe(NY) 00036440
20 WRITE(11,902) CAPABS(NY),RESMIN(NY),RESMAX(NY),FATOPe(NY)
DEFINE AND PRINT THE NORMALIZED X-AXIS.
DX = 1.0 / MAXPU
DO 30 I = 1,MAXI
30 X(I) = I * DX 00036450
WRITE(11,906) (X(I),I = 1,MAXI)
PEAK LOAD DEMAND (BY SEASON PER YEAR).
DO 40 NY = 1,MYEAR
READ(10,902) (PEAKS(NY,I),I = 1,4) 00036460
40 WRITE(11,902) (PEAKS(NY,I),I = 1,4)
SCHEDED SYSTEM TOTAL ENERGY DEMAND (BY SEASON PER YEAR)
DO 50 NY = 1,MYEAR
READ(10,902) (ENEDEM(NY,I),I = 1,4) 00036470
50 WRITE(11,902) (ENEDEM(NY,I),I = 1,4)
EXPANSION CANDIDATE DATA
MXEXPN=MAXINP+2
CLOADP=0.01
READ(10,911) (HYEXP(11),J=1,11)
READ(10,911) (PSEXPN(11),J=1,11),
DO 140 I=1,2
IF(I.GT.1) GO TO 130 00036480
00036490
00036500
00036510
00036520
00036530
00036540
00036550
00036560
00036570
00036580
00036590
00036600
00036610
00036620
00036630
00036640
00036650
00036660
00036670
00036680
00036690
00036700
00036710
00036720

```

PLACA(I,2)=HYEXP(1)          00036730
PLACA(I,3)=HYEXP(2)          00036740
PLACA(I,4)=HYEXP(6)          00036750
PLACA(I,5)=HYEXP(5)          00036760
PLACA(I,6)=HYEXP(9)          00036770
PLACA(I,9)=30.0              00036780
PLACA(I,13)=HYEXP(10)         00036790
130  GO TO 140                00036800
     PLACA(I,2)=0.0            00036810
     PLACA(I,3)=PSEXP(1)        00036820
     PLACA(I,4)=PSEXP(6)        00036830
     PLACA(I,5)=PSEXP(5)        00036840
     PLACA(I,6)=PSEXP(9)        00036850
     PLACA(I,9)=30.0            00036860
     PLACA(I,13)=PSEXP(10)       00036870
140  CONTINUE                  00036880
     DO 60 IP=3,MXEXP
       READ(10,904) (PLACA(IP,J),J=1,14)
       WRITE(11,904) (PLACA(IP,J),J=1,14)
C   60  CONTINUE                  00036890
C   MIN & MAX. NUMBER OF UNITS ALLOWED FOR EACH CANDIDATE PER YEAR. 00036900
C   DD 80 NY =1,MXYEAR          00036910
C   READ(10,900) (MINUN(NY,IP),IP = 1,MXEXP) 00036920
C   WRITE(11,900) (MINUN(NY,IP),IP = 1,MXEXP) 00036930
C   READ(10,900) (MAXUN(NY,IP),IP=1,MXEXP)    00036940
C   WRITE(11,900) (MAXUN(NY,IP),IP =1,MXEXP)    00036950
C   READ PREP MODULE SOLUTION FOR CALCULATION OF OBJ UPPER BOUND 00036960
C   DO 90 NY=1,MXYEAR          00036970
C   READ(10,900) (NSTPRE(NY,IP),IP=1,MXEXP)    00036980
C   90  WRITE(11,901) (NSTPRE(NY,IP),IP=1,MXEXP) 00036990
C   READ SPACE AVAILABLE FOR MAINTENANCE          00037000
C   DO 100 NY=1,MXYEAR          00037010
C   READ(10,902) (AVSP(NY,J), J=1,4)           00037020
C   100 WRITE(11,903) (AVSP(NY,J), J=1,4)        00037030
C   DO 110 NY=1,MXYEAR          00037040
C   READ(10,909) (SSCUM(J,NY),J=1,4)           00037050
C   110 WRITE(11,910) (SSCUM(J,NY),J=1,4)        00037060
C   READ NORMAL DISTRIBUTION TABLE               00037070
C   READ(19,907) (TYIK),TINT(K),K = 1,120      00037080
C   WRITE(11,908) (TY(K),TINT(K),K = 1,120)      00037090
C   120  CONTINUE                  00037100
     DO 120 NY=1,MXYEAR          00037110
     READ(31,1010) NNDUM          00037120
     READ(31,911) (HYSCHD(NY,J),J=1,8)          00037130
     READ(33,1010) NNDUM          00037140
     READ(33,911) (PSSCHD(NY,J),J=1,8)          00037150
     900  FORMAT(10I4)             00037160
     901  FORMAT(1H ,20I4)          00037170
     902  FORMAT(5E16.8)           00037180
     903  FORMAT(1H ,4E16.8)        00037190
     904  FORMAT(F3.0 ,2F5.0 ,F3.0 ,F8.6 ,3F7.2 ,F3.0 ,2F7.3 ,E10.3 ,2F4.2) 00037200
     905  FORMAT(1H ,F3.0 ,2F5.0 ,F3.0 ,F8.6 ,3F7.2 ,F3.0 ,2F7.3 ,E10.3 ,2F4.2) 00037220
     906  FORMAT(10F11.6)           00037230
     907  FORMAT(F10.4 ,F12.8)      00037240
     908  FORMAT(1H ,5(F10.4 ,F12.8))    00037250
     909  FORMAT(4E16.8)            00037260
     910  FORMAT(1H ,4E16.8)        00037270
     911  FORMAT(5E12.5)           00037280
     1010 FORMAT(7,3I3)             00037290
C   RETURN                      00037300
C   END                         00037310
C   00037320
C   00037330
C   00037340
C   00037350
C   00037360
C   00037370
C   00037380
C   00037390
C   00037400

```

```

C----- SUBROUTINE REPORT (SALV,RESCAP) -----
C----- COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
C----- FCR,HOURS,IEASYR,IMXPL,ITIN,ITMAX,
C----- ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
C----- MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
C----- RESMAR,UBOUND
C----- COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
C----- IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
C----- IWIDTH(8),LDRDER(420,3),LOWB(20,8),MAINS(200),
C----- MAXUNI(20,20),MAXADD(8),MINUNI(20,20),
C----- NEXPID(20),NSTPRE(20,20),NUPB(20,8),
C----- PEAKSI(20,4),PLACAI(20,14),PLANT(210,14),
C----- RESMAX(20),RESMINI(20),SSCUM(4,20),UNE(20),
C----- FATOPE(20),SOL(20,2),AUPEAK(20)
C----- COMMON /LCS/ EL(1250),POPEC(250,20),ELF(1260),
C----- MXELDC(20,4),X(1250)
C----- COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)
C----- COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
C----- MINTUN(8),NUBOLD(8)
C----- COMMON /TBL/ TINT(120),TY(120)
C----- DOUBLE PRECISION TINT,TY
C----- COMMON /HYS/ CMULT(4),EVAL(3),EMULT(4),HYDR(3,11),
C----- HYEXP(11),HYSCHD(20,8),PSEXPN(11),
C----- PSSCHD(20,8),SYSCUM(4)
C----- DIMENSION SALV(20),RESCAP(20,20),OPTOT(20),CAP(20),
C----- ETABLE(20,10),RSCAP1(20,10),POPEC1(10,20),
C----- PL(10),ALPHA(20)
C----- DIMENSION A1(250),A2(20),FMT1(18),FMT2(18),FMT3(18),
C----- FMT4(18),FMT5(18),FMT6(18),FMT7(18),C(60),IYR(20)
C----- EQUIVALENCE (EL(1),A1(1)),(EL(251),ETABLE(1)),
C----- (EL(451),POPEC1(1)),(EL(651),RSCAP1(1)),
C----- (EL(851),OPTOT(1)),(EL(871),CAP(1))
C----- (EL(891),IYR(1)),(EL(911),PL(1)),(EL(921),A2(1))
C----- EQUIVALENCE (EL(941),C(1)),(EL(1001),FMT1(1)),
C----- (EL(1021),FMT2(1)),(EL(1041),FMT3(1)),
C----- (EL(1061),FMT4(1)),(EL(1081),FMT5(1)),
C----- (EL(1101),FMT6(1)),(EL(1121),FMT7(1))
C----- DATA ALPHA/3H 1,3H 2,3H 3,3H 4,3H 5,3H 6,3H 7,3H 8,
C----- +3H 9,3H 10,3H 11,3H 12,3H 13,3H 14,3H 15,3H 16,3H 17,
C----- +3H 18,3H 19,3H 20/
C----- DATA HY/3H HY/, PS/3H PS/, ZERO/3H 0/, BLNK/3H /
C----- REWIND 19
C----- MX1=MXYEAR-MXYEAR/2
C----- *C*
C----- DD 10 I=1,8
10 PL(I)=BLNK
C----- *
C----- DO 14 I=1,MXPL
11 IP=IDEXP(I)
IF(IP.GT.0) GO TO 12
IF(IP.GT.-1) GO TO 11
12 PL(I)=HY
GO TO 14
13 PL(I)=PS
14

```

```

12 GO TO 14          00038090
12 DO 13 J=1,20      00038100
   IF(IP.NE.J) GO TO 13
   PL(I)=ALPHA(J)
   GO TO 14          00038110
13 CONTINUE          00038120
14 CONTINUE          00038130
14 CONTINUE          00038140
14 CONTINUE          00038150
14 CONTINUE          00038160
14 CONTINUE          00038170
14 CONTINUE          00038180
14 CONTINUE          00038190
14 CONTINUE          00038200
14 CONTINUE          00038210
14 CONTINUE          00038220
14 CONTINUE          00038230
14 CONTINUE          00038240
14 CONTINUE          00038250
14 CONTINUE          00038260
14 CONTINUE          00038270
14 CONTINUE          00038280
14 CONTINUE          00038290
14 CONTINUE          00038300
14 CONTINUE          00038310
14 CONTINUE          00038320
14 CONTINUE          00038330
14 CONTINUE          00038340
14 CONTINUE          00038350
14 CONTINUE          00038360
14 CONTINUE          00038370
14 CONTINUE          00038380
14 CONTINUE          00038390
14 CONTINUE          00038400
14 CONTINUE          00038410
14 CONTINUE          00038420
14 CONTINUE          00038430
14 CONTINUE          00038440
14 CONTINUE          00038450
14 CONTINUE          00038460
14 CONTINUE          00038470
14 CONTINUE          00038480
14 CONTINUE          00038490
14 CONTINUE          00038500
14 CONTINUE          00038510
14 CONTINUE          00038520
14 CONTINUE          00038530
14 CONTINUE          00038540
14 CONTINUE          00038550
14 CONTINUE          00038560
14 CONTINUE          00038570
14 CONTINUE          00038580
14 CONTINUE          00038590
14 CONTINUE          00038600
14 CONTINUE          00038610
14 CONTINUE          00038620
14 CONTINUE          00038630
14 CONTINUE          00038640
14 CONTINUE          00038650
14 CONTINUE          00038660
14 CONTINUE          00038670
14 CONTINUE          00038680
14 CONTINUE          00038690
14 CONTINUE          00038700
14 CONTINUE          00038710
14 CONTINUE          00038720
14 CONTINUE          00038730
14 CONTINUE          00038740
14 CONTINUE          00038750
14 CONTINUE          00038760

DO 16 NY=1,MXYEAR
DO 15 I=1,8
15 ETABLE(NY,I)=BLNK
OPTOT(NY)=0.0
CAP(NY)=0.0
IYR(NY)=IBASYR+NY
16 CONTINUE

DO 20 NY=1,MXYEAR
DO 19 I=1,MXPL
IS=ISOL(NY,I)
IF(IS.GT.0) GO TO 17
ETABLE(NY,I)=ZERO
GO TO 19
17 DO 18 J=1,20
IF(IIS.NE.J) GO TO 18
ETABLE(NY,I)=ALPHA(J)
GO TO 19
18 CONTINUE
19 CONTINUE
20 CONTINUE

MN1=MXALL+3
DO 23 NY=1,MXYEAR
DO 21 I=1,MN1
21 OPTOT(NY)=OPTOT(NY)+POPEC(1,NY)
DO 22 I=1,MXPL
22 CAP(NY)=CAP(NY)+RESCAP(NY,I)
23 CONTINUE

WRITE (6,41)
41 FORMAT (1H1)
DO 35 L=1,10
WRITE (6,71)
35 CONTINUE
WRITE (6,43)
43 FORMAT (1H ,30X,23H CHARACTERISTICS OF THE,
+31H OPTIMAL OR SUBOPTIMAL SOLUTION//)
WRITE (6,48)
48 FORMAT (1H ,15X,11H PLANT TYPE,18X,9H UNSERVED,5X,6H TOTAL)
WRITE (6,51)
51 FORMAT (1H ,45X,7H ENERGY,4X,10H OPERATING,
+ 1X,8H CAPITAL,3X,8H SALVAGE,2X,10H OBJECTIVE)
WRITE (6,55) (PL(I),I=1,8)
55 FORMAT (1H ,5H YEAR,1X,8A3,6X,6H LOLP+,5X,6H(MWH), ,5X,
+ 6H COST*,4X,7H CUST**,4X,7H VALUE*,3X,10HFUNCTION* )
DO 60 NY=1,MXYEAR
WRITE (6,61) IYR(NY),(ETABLE(NY,I),I=1,8),SOL(NY,1),
+ UNE(NY),OPTOT(NY),CAP(NY),SALV(NY),SOL(NY,2)
61 FORMAT (/1H ,15,1X,8A3,1X,F11.6,1X,E12.5,2F10.3,1X,F10.3,
+ 2X,F10.3)
60 CONTINUE

```

```

      WRITE(6,65)
66 FORMAT(1H ,4H + ZERO MEANS LOLP VALUE SMALLER THAN 0.00001) 00038770
      WRITE(6,67) IBASYR 00038780
67 FORMAT(1//1H ,16H * MILLIONS OF ,14,8H DOLLARS) 00038790
      WRITE(6,69) 00038800
69 FORMAT(1//1H ,38H ** PLANT COSTS ARE GIVEN AS THE TOTAL,
+48H WORTH OF THE PLANT AS IT COMES ON LINE LESS THE // 00038810
+46H SALVAGE VALUE AT THE END OF THE STUDY PERIOD.,/
+41H IF THE FIXED CHARGE RATE OPTION IS USED, // 00038820
+34H CAPITAL COSTS REPRESENT THE FIXED, 00038830
+24H CHARGES FOR EACH PLANT.) 00038840
      WRITE(6,41) 00038850
      DO 74 I=1,10 00038860
      WRITE(6,71) 00038870
71 FORMAT(1H /) 00038880
74 CONTINUE 00038890
      READ(19,99) C 00038910
99 FORMAT(30A2) 00038920
      WRITE(6,310) 00038930
310 FORMAT(1H1)
      READ(19,100) FMT1 00038940
      READ(19,100) FMT2 00038950
      READ(19,105) FMT3 00038960
      READ(19,110) FMT4 00038970
      READ(19,110) FMT5 00038980
      READ(19,115) FMT6 00038990
      READ(19,116) FMT7 00038990
100 FORMAT(2A4,A2,10A4,A2,4A4) 00039000
105 FORMAT(A3,A2,A2,13A4) 00039010
110 FORMAT(4A4,A2,13A4) 00039020
115 FORMAT(3A4,A1,A2,13A4) 00039030
116 FORMAT(5A4,A2,12A4) 00039040
      NP=MN1 00039050
      DO 122 NY=1,MXYEAR 00039060
122 A2(NY)=OPTOT(NY) 00039070
      CALL TABLE(FMT1,FMT2,FMT3,FMT4,FMT5,FMT6,
+ FMT7,C,A1,A2,1YR,NP,MX1) 00039080
      NP=MXPL 00039090
      DO 130 I=1,NP 00039100
      DO 140 J=1,MXYEAR 00039110
140 POPEC(I,J)=RESCAP(J,1) 00039120
130 CONTINUE 00039130
      DO 125 NY=1,MXYEAR 00039140
125 A2(NY)=CAP(NY) 00039150
      CALL TABLE(FMT1,FMT2,FMT3,FMT4,FMT5,FMT6,
+ FMT7,C,A1,A2,1YR,NP,MX1) 00039160
      RETURN 00039170
      END 00039180
C
C-----SUBROUTINE RESMIN, NBEST, IRES, KEEP)
C
      RESIMULATES THE NBEST SOLUTIONS 00039190
C
COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
ITOUT,MAXALL,MAXI,MAXINP,MAXUR,
MAXPLA,MAXPD,MXALL,MXPL,MXYEAR,
RESMAR,UBOUND 00039200
C
COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
IWIDTH(8),LDRDER(420,3),LOWB(20,8),MAINS(200),
MAXUN(20,20),MAXADD(8),MINUN(20,20),
NEXPID(20),NSTPRE(20,20),NUPB(20,8),
PEAKS(20,4),PLACAA(20,14),PLANT(210,14), 00039210

```



```

REWIND 24
REWIND 27
DO 200 NY=1,MXYEAR
128 = ISTAR + NY - 1
READ(28,128,900) (ISOL(NY,I),I=1,8),SOL(NY,1),SOL(NY,2)
WRITE(40,910) (ISOL(NY,I),I=1,MXPL)
910 FORMAT(814)
C WRITE(11,*)(ISOL(NY,I),I=1,8),SOL(NY,1),SOL(NY,2)
DO 205 I = 1,250
205 POPEC(I,NY) = 0.0
C DO 210 I=1,20
210 RESCAP(NY,I) = 0.0
C DO 215 N=1,4.
READ(27) ((PTCUM(I,J,N), I=1,201), J=1,8)
215 CONTINUE
READ(24,902) ((LDRDER(J,K),K=1,3),J=1,420)
READ(24,921) (MAINS(J),J=1,200)
902 FORMAT(125(13,12,13))
921 FORMAT(200I2)
DO 230 I=1,MXPL
NSTATE(I) = 0
NSTCUR(I) = ISOL(NY,I)
IF (NY .NE. 1) NSTCUR(I) = ISOL(NY-1,I)
230 CONTINUE
CALL OPERCO(NSTCUR,NY,OPEC,IRES)
CALL OBJFUN(NSTCUR,NY,OPEC,NSTATE,CAPCO,OBJ,OBJ1,S,
+RESCAP,IRES)
SALV(NY) = S
WRITE(6,*)(NSTCUR,NSTATE
WRITE(6,*)(OPEC,CAPCO,OBJ,J,OBJ1,SALV(NY)
WRITE(11,*)(OPEC,CAPCO,OBJ,OBJ1,SALV(NY)
WRITE(40,940) (RESCAP(NY,I),I=1,MXPL)
940 FORMAT(5E12.5)
200 OBJ=OBJ + OBJ1
DO 300 NY = 1,MXYEAR
WRITE(11,*)(RESCAP(NY,K),K=1,MAXIPN)
WRITE(11,*)(POPEC(I,NY),I=1,40)
C 300 CONTINUE
CALL REPORT(SALV,RESCAP)
C RETURN
END
C

```

SUBROUTINE SEARCH (NST, NST2, NUMACC, NYCR, NACCEP, NTH)

TITLE:	STATE SEARCH SUBROUTINE	00040600
FUNCTION:	THIS SUBROUTINE SEARCHES THE FIRST SLOT OF ARRAY NST2 TO FIND WEATHER STATE "NST" IS ALREADY CONTAINED IN NST2. IF SO, NTH IS THE STATE INDEX NUMBER, IF NOT, NTH IS 0.	00040610 00040620 00040630 00040640 00040650 00040660 00040670 00040680 00040690 00040700 00040710 00040720 00040730 00040740 00040750 00040760 00040770 00040780 00040790 00040800
CALLED BY:	MAIN	
CALLS TO:	NONE	
 SUBROUTINE ARGUMENT		
NAME	DESCRIPTION	00040710
NACCEP	NUMBER OF ACCEPTED STATES	00040720
NST	CURRENT STATE PACKED	00040730
NST2	ARRAY WITH ACCEPTED STATES (TO BE SEARCHED)	00040740
NTH	INDEX OF STATE FOUND IN NST ARRAY. IF 0, STATE WAS NOT FOUND	00040750
NUMACC	(NYCR) NUMBER OF STATES ACCEPTED IN YEAR	00040760
NYCR	"NYCR" AFTER LAST FATHOMING	00040770
	CURRENT YEAR	00040780

```

COMMON VARIABLES USED: NONE
LOCAL VARIABLES
N      TEMPORARY INDEX OF SEARCH POINTER
NBEGIN BEGINNING INDEX OF SEARCH RANGE
NEND   ENDING INDEX OF SEARCH RANGE
NRANGE SEARCH RANGE
DIMENSION NUMACC(20), NST2(2,1000)          00040810
                                              00040820
                                              00040830
                                              00040840
                                              00040850
                                              00040860
                                              00040870
                                              00040880
                                              00040890
                                              00040900
                                              00040910
                                              00040920
                                              00040930
                                              00040940
                                              00040950
                                              00040960
                                              00040970
                                              00040980
                                              00040990
                                              00041000
                                              00041010
DO 10 NBEGIN, NEND
IF (NST .NE. NST2(1,N)) GO TO 10          00041020
STATE "NST" IS FOUND IN "NST2" ARRAY IDENTIFY THE
SEQUENCE NUMBER AND RETURN                  00041030
NTH = N                                     00041040
GO TO 20                                     00041050
10 CONTINUE                                     00041060
DROPPING THROUGH THE LOOP MEANS THAT STATE "NST" WAS
NOT FOUND IN THE "NST2" ARRAY. THIS FACT IS INDICATED
BY SETTING NTH = 0                           00041070
NTH = 0                                     00041080
20 RETURN                                     00041090
END                                         00041100
                                              00041110
                                              00041120
                                              00041130
                                              00041140
                                              00041150
SUBROUTINE STAGEN(NSTCNT, CUML0L, NYCR)      00041160
                                              00041170
TITLE    STAGE GENERATION
FUNCTION  THIS SUBROUTINE GENERATES ALL THE ACCEPTABLE
          STATES WITHIN THE TUNNEL DEFINED BY THE
          "TUNNEL" SUBROUTINE. STATE ACCEPTABILITY
          IS CHECKED THROUGH CALLS TO THE "CONSTR"
          SUBROUTINE. THE ACCEPTED STATES ARE STORED
          IN THE "LIST" ARRAY.          00041180
          00041190
          00041200
          00041210
          00041220
          00041230
          00041240
CALLED BY MAIN                                00041250
CALLS TO CONSTR                               00041260
                                              00041270
SUBROUTINE ARGUMENTS
NAME      DESCRIPTION
NSTCNT   COUNTER OF THE NUMBER OF ACCEPTED
          STATES          00041280
          00041290
COMMON VARIABLES USED. LIST, MAXTUN, MINTUN, MXPL
LOCAL VARIABLES
NAME      TYPE SIZE DESCRIPTION
I        DUMMY PLANT TYPE INDEX
IFLCON   CONSTRAINT FLAG
          IF =0, STATE IS ACCEPTABLE
          IF =1, STATE IS UNACCEPTABLE
NSTCUR   NSTCUR(I) DENOTES THE NUMBER OF
          UNITS OF THE I-TH PLANT TYPE
          CONTAIN IN STATE "NSTCUR".          00041300
          00041310
          00041320
          00041330
          00041340
          00041350
          00041360
          00041370
          00041380
          00041390
          00041400
          00041410
          00041420
          00041430
          00041440
          00041450
ALGORITHM LOGIC
THE FIRST STATE IS IDENTICAL TO THE TUNNEL LOWER BOUND.
A NEW STATE IS GENERATED RECURSIVELY FROM THE PREVIOUS-
LY GENERATED STATE BY INCREASING THE NUMBER OF UNITS 00041460
          00041470
          00041480

```

IN THE FIRST PLANT TYPE WHOSE UNIT NUMBER HAS NOT YET
 REACHED THE TUNNEL UPPER LIMIT. EACH TIME THE UPPER
 LIMIT FOR A PLANT TYPE (I) IS REACHED, RESET UNIT
 NUMBER FOR PLANT TYPES I THROUGH 1 TO THE LOWER TUNNEL
 BOUND.
 00041490
 00041500
 00041510
 00041520
 00041530
 00041540
 00041550
 00041560
 00041570
 00041580
 00041590
 00041600
 00041610
 00041620
 00041630
 00041640
 00041650
 00041660
 00041670
 00041680
 00041690
 00041700
 00041710
 00041720
 00041730
 00041740
 00041750
 00041760
 00041770
 00041780
 00041790
 00041800
 00041810
 00041820
 00041830
 00041840
 00041850
 00041860
 00041870
 00041880
 00041890
 00041900
 00041910
 00041920
 00041930
 00041940
 00041950
 00041960
 00041970
 00041980
 00041990
 00042000
 00042010
 00042020
 00042030
 00042040
 00042050
 00042060
 00042070
 00042080
 00042090
 00042100
 00042110
 00042120
 00042130
 00042140
 00042150
 00042160

COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
 FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
 ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
 MAXPLA,MAXPU,MAXALL,MAXPL,MAXYEAR,
 RESMAR,UBOUND
 00041490
 00041500
 00041510
 00041520
 00041530
 00041540
 00041550
 00041560
 00041570
 00041580
 00041590
 00041600
 00041610
 00041620
 00041630
 00041640
 00041650
 00041660
 00041670
 00041680
 00041690
 00041700
 00041710
 00041720
 00041730
 00041740
 00041750
 00041760
 00041770
 00041780
 00041790
 00041800
 00041810
 00041820
 00041830
 00041840
 00041850
 00041860
 00041870
 00041880
 00041890
 00041900
 00041910
 00041920
 00041930
 00041940
 00041950
 00041960
 00041970
 00041980
 00041990
 00042000
 00042010
 00042020
 00042030
 00042040
 00042050
 00042060
 00042070
 00042080
 00042090
 00042100
 00042110
 00042120
 00042130
 00042140
 00042150
 00042160

COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
 INDEXP(20),IPABA(8),IPCH(8),ISOL(20,8)
 IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
 MAXUN(20,20),MAXADD(8),MINUN(20,20),
 NEXPID(20),NSTPRE(20,20),NUPB(20,8),
 PEAKS(20,4),PLACA(20,14),PLANT(210,14),
 RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
 FATOP(20),SOL(20,2),AUPEAK(20)
 00041490
 00041500
 00041510
 00041520
 00041530
 00041540
 00041550
 00041560
 00041570
 00041580
 00041590
 00041600
 00041610
 00041620
 00041630
 00041640
 00041650
 00041660
 00041670
 00041680
 00041690
 00041700
 00041710
 00041720
 00041730
 00041740
 00041750
 00041760
 00041770
 00041780
 00041790
 00041800
 00041810
 00041820
 00041830
 00041840
 00041850
 00041860
 00041870
 00041880
 00041890
 00041900
 00041910
 00041920
 00041930
 00041940
 00041950
 00041960
 00041970
 00041980
 00041990
 00042000
 00042010
 00042020
 00042030
 00042040
 00042050
 00042060
 00042070
 00042080
 00042090
 00042100
 00042110
 00042120
 00042130
 00042140
 00042150
 00042160

COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
 + MXELDC(20,4),X(1250)
 00041490
 00041500
 00041510
 00041520
 00041530
 00041540
 00041550
 00041560
 00041570
 00041580
 00041590
 00041600
 00041610
 00041620
 00041630
 00041640
 00041650
 00041660
 00041670
 00041680
 00041690
 00041700
 00041710
 00041720
 00041730
 00041740
 00041750
 00041760
 00041770
 00041780
 00041790
 00041800
 00041810
 00041820
 00041830
 00041840
 00041850
 00041860
 00041870
 00041880
 00041890
 00041900
 00041910
 00041920
 00041930
 00041940
 00041950
 00041960
 00041970
 00041980
 00041990
 00042000
 00042010
 00042020
 00042030
 00042040
 00042050
 00042060
 00042070
 00042080
 00042090
 00042100
 00042110
 00042120
 00042130
 00042140
 00042150
 00042160

COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)
 COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
 + MINTUN(8),NUBOLD(8)
 COMMON /TBL/ TINT(120),TY(120)
 DOUBLE PRECISION TINT,TY
 COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
 + HYEXP(11),HSCHD(20,8),PSEXP(11),
 + PSSCHD(20,8),SYSCUM(4)
 00041490
 00041500
 00041510
 00041520
 00041530
 00041540
 00041550
 00041560
 00041570
 00041580
 00041590
 00041600
 00041610
 00041620
 00041630
 00041640
 00041650
 00041660
 00041670
 00041680
 00041690
 00041700
 00041710
 00041720
 00041730
 00041740
 00041750
 00041760
 00041770
 00041780
 00041790
 00041800
 00041810
 00041820
 00041830
 00041840
 00041850
 00041860
 00041870
 00041880
 00041890
 00041900
 00041910
 00041920
 00041930
 00041940
 00041950
 00041960
 00041970
 00041980
 00041990
 00042000
 00042010
 00042020
 00042030
 00042040
 00042050
 00042060
 00042070
 00042080
 00042090
 00042100
 00042110
 00042120
 00042130
 00042140
 00042150
 00042160

DIMENSION NSTCUR(20), CUMLOL(1000)

INITIALIZE STATE COUNTER
 NSTCNT = 0

INITIALIZE NSTCUR TO THE LOWER TUNNEL BOUNDARY
 DO 10 I=1,MXPL
 10 NSTCUR(I) = MINTUN(I)
 CHECK IF NEW STATE IS ACCEPTABLE
 20 CALL CONSTR(NSTCUR, NYCR, CULOLP, IFLCON)
 IF (IFLCON .GE. 1) GO TO 40
 NOT BRANCHING TO 40 MEANS NSTCUR IS ACCEPTABLE
 INCREMENT STATE COUNTER AND STORE IN LIST ARRAY
 NSTCNT = NSTCNT + 1
 DO 30 I=1,MXPL
 30 LIST(I,NSTCNT) = NSTCUR(I)
 STORE CUMULANT LOLP IN CORE
 CUMLOL(NSTCNT) = CULOLP
 FIND THE FIRST PLANT TYPE WHOSE UNIT NUMBER IN THE LAST
 STATE GENERATED HAS NOT REACHED THE TUNNEL UPPER BOUND.
 INITIALIZE ALL OTHERS TO THE TUNNEL LOWER BOUND.
 40 DO 50 I=1,MXPL
 IF (NSTCUR(I) .LT. MAXTUN(I)) GO TO 60
 NSTCUR(I) = MINTUN(I)

50 CONTINUE
 DROPPING THROUGH THE LOOP MEANS THAT THE TUNNEL UPPER
 BOUND IS REACHED FOR ALL PLANT TYPES, I.E. ALL
 POSSIBLE STATES WERE GENERATED.

```

C      WRITE(11,*) ((LIST(I,N),I=1,MXPL),N=1,NSTCNT)          00042170
C      RETURN          00042180
C      GENERATE NEW STATE          00042190
C      60 NSTCUR(I) = NSTCUR(I) + 1          00042200
C      GO TO 20          00042210
C      END          00042220
C      C          00042230
C      C          00042240
C      C          00042250
C      C          00042260
C      C          00042270
C      C          00042280
C      C          00042290
C      C          00042300
C      C          00042310
C      SUBROUTINE TABLE(FMT1,FMT2,FMT3,FMT4,FMT5,FMT6,          00042320
C      + FMT7,C,A1,A2,IYR,NP,MX1)          00042330
C      C          00042340
C      COMMON /VRS/ CLDLR,CLDLPU,CPLDLP,DISRAT,DX,          00042350
C      1      FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,          00042360
C      2      ITOUT,MAXALL,MAXI,MAXINP,MAXDR,          00042370
C      3      MAXPLA,MAXPD,MXALL,MXPL,MXYEAR,          00042380
C      4      RESMAR,UBOUND          00042390
C      C          00042400
C      COMMON /LCS/ EL(1250),POPEC(250,20),ELF(1260),          00042410
C      + MXELDC(20,4),X(1250)          00042420
C      C          00042430
C      COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),          00042440
C      + HYEXP(11),HYSCHD(20,8),PSEXP(11),          00042450
C      + PSSCHD(20,8),SYSCUM(4)          00042460
C      DIMENSION A1(250),A2(20),FMT1(18),FMT2(18),FMT3(18),          00042470
C      + FMT4(18),FMT5(18),FMT6(18),FMT7(18),C(60),IYR(20)
C      LMX=51          00042480
C      LN=0          00042490
C      K=1          00042500
C      N=MX1          00042510
C      TOT=0.0          00042520
C      IF (MXYEAR.LE.10) N=MXYEAR          00042530
C      IPAGE=1          00042540
C      DO 170 I=1,NP          00042550
C      170 A1(I)=0.0          00042560
C      DO 190 I=1,NP          00042570
C      DO 200 J=1,MXYEAR          00042580
C      200 A1(I)=A1(I)+POPEC(I,J)          00042590
C      190 CONTINUE          00042600
C      DO 195 I=1,NP          00042610
C      195 TOT=TOT+A1(I)          00042620
C      CALL FORM(IPAGE,K,N,LN,I,FMT1,FMT2,FMT3,          00042630
C      + FMT4,FMT5,FMT6,FMT7,C,IYR,NP,MX1)          00042640
C      LN=0          00042650
C      M=N-K+1          00042660
C      DO 250 I=1,NP          00042670
C      LN=LN+1          00042680
C      FMT6(5)=C(M)          00042690
C      IF (N.EQ.MX1) WRITE (6,FMT6) I,(POPEC(I,J),J=K,N)          00042700
C      IF (N.EQ.MXYEAR) WRITE (6,FMT6) I,(POPEC(I,J),J=K,N),A1(I)          00042710
C      IF (LN.EQ.15.AND.I.NE.NP) GO TO 280          00042720
C      GO TO 250          00042730
C      280 IPAGE=IPAGE+1          00042740
C      CALL FORM(IPAGE,K,N,LN,I,FMT1,FMT2,FMT3,          00042750
C      + FMT4,FMT5,FMT6,FMT7,C,IYR,NP,MX1)          00042760
C      LN=0          00042770
C      250 CONTINUE          00042780
C      FMT7(6)=C(M)          00042790
C      WRITE (6,FMT7) (A2(J),J=K,N),TOT          00042800
C      IF (N.EQ.MXYEAR) GO TO 300          00042810
C      K=MX1+1          00042820
C      N=MXYEAR          00042830
C      IPAGE=IPAGE+1          00042840
C      GO TO 240
C      300 CONTINUE

```

```

LK=LMX-LN*2-9          00042850
IF (I.EQ.NP) LK=LK-3    00042860
IF (NP.EQ.MXPL) GO TO 355 00042870
GO TO 360               00042880
355 WRITE (6,365)        00042890
365 FORMAT (1H //)       00042900
      WRITE (6,366)        00042910
366 FORMAT (36H *PLANT COSTS ARE GIVEN AS THE TOTAL,
+48H WORTH OF THE PLANT AS IT COMES ON LINE LESS THE // 00042920
+46H SALVAGE VALUE AT THE END OF THE STUDY PERIOD., //
+41H IF THE FIXED CHARGE RATE OPTION IS USED, // 00042930
+39H CONSTRUCTION COSTS REPRESENT THE FIXED, // 00042940
+24H CHARGES FOR EACH PLANT.) 00042950
      LK=LK-8               00042960
360 CONTINUE              00042970
DO 302 IK=1,LK           00042980
      WRITE (6,304)         00042990
304 FORMAT (1H //)        00043000
302 CONTINUE              00043010
      RETURN                00043020
      END                   00043030
                                         00043040
                                         00043050
                                         00043060
                                         00043070
                                         00043080
                                         00043090
                                         00043100
                                         00043110
                                         00043120
                                         00043130
                                         00043140
                                         00043150
                                         00043160
                                         00043170
                                         00043180
                                         00043190
                                         00043200
                                         00043210
                                         00043220
                                         00043230
                                         00043240
                                         00043250
                                         00043260
                                         00043270
                                         00043280
                                         00043290
                                         00043300
                                         00043310
                                         00043320
                                         00043330
                                         00043340
                                         00043350
                                         00043360
                                         00043370
                                         00043380
                                         00043390
                                         00043400
                                         00043410
                                         00043420
                                         00043430
                                         00043440
                                         00043450
                                         00043460
                                         00043470
                                         00043480
                                         00043490
                                         00043500
                                         00043510
                                         00043520

```

SUBROUTINE TALOOK(Z,ZINTEG)

```

COMMON /TBL/ YINT(120),Y(120)
DOUBLE PRECISION AVZ,Y,YINT,Z,ZINTEG,DY,PK,ZINT,FACTOR,
+R1,R2,R3,R4,Q12,Q13,Q14,Q23,Q24,Q34,Q21,Q31,Q41,Q32,Q42,Q43
C
C   LOOKUP IN NORMAL DISTRIBUTION FUNCTION THE VALUE OF
C   THE INTEGRAL OF EXP(-Z2/2) FROM MINUS INFINITY TO Z.
C   INTERPOLATE WITH CUBIC FIT, BETWEEN THE 120 DATA POINTS
C   COPIED FROM TABLES. THE REFERENCE USED IS: TABLES OF
C   NORMAL PROBABILITY FUNCTIONS BY THE U.S. DEPARTMENT
C   OF COMMERCE.
C
DEFINITION OF KEY VARIABLES:
NAME   TYPE   SIZE   MEANING
Z      REAL   -      NORMALIZED INDEPENDENT VARIABLE
ZINTEG REAL   -      INTEGRAL VALUE
Y      REAL   120    NORMALIZE INDEPENDENT VARIABLE
      (INPUT DATA)
YINT   REAL   120    INTEGRAL VALUES CORRESPONDING
                      TO Y (INPUT DATA)
K      INT.   -      GRID POINT CORRESPONDING TO Z
DY     REAL   -      Y INCREMENT
C
IF (Z.GE.-5.9D00) GO TO 5
ZINTEG = 0.0D00
RETURN
5 CONTINUE
IF (DABS(Z).GT.1.0D-10) GO TO 10
ZINTEG = 0.5D00
RETURN
10 CONTINUE
DY = Y(2) - Y(1)
PK = DABS(Z)/DY + 1.0D00
K = PK
AVZ = DABS(Z)
R1 = 0.0D00
R2 = 0.0D00
R3 = 0.0D00
R4 = 0.0D00
IF (AVZ.NE.Y(K)) R1 = AVZ - Y(K)

```

```

IF ( AVZ.NE.Y(K+1))R2 = AVZ - Y(K+1) 00043530
IF ( AVZ.NE.Y(K+2))R3 = AVZ - Y(K+2) 00043540
IF ( AVZ.NE.Y(K+3))R4 = AVZ - Y(K+3) 00043550
C
Q12= Y(K) - Y(K+1) 00043560
Q13= Y(K) - Y(K+2) 00043570
Q14= Y(K) - Y(K+3) 00043580
Q23= Y(K+1) - Y(K+2) 00043590
Q24= Y(K+1) - Y(K+3) 00043600
Q34= Y(K+2) - Y(K+3) 00043610
Q21= - Q12 00043620
Q31= -Q13 00043630
Q41= -Q14 00043640
Q32= -Q23 00043650
Q42= -Q24 00043660
Q43= -Q34 00043670
IF ( DABS(R1).GT.1.0D-10)GO TO 20 00043680
ZINT = YINT(K)*R2*R3*R4/(Q12*Q13*Q14) 00043690
GO TO 30 00043700
20 IF ( DABS(R2).GT.1.0D-10)GO TO 22 00043710
ZINT = YINT(K+1)*R1*R3*R4/(Q21*Q23*Q24) 00043720
GO TO 30 00043730
22 IF ( DABS(R3).GT.1.0D-10)GO TO 24 00043740
ZINT = YINT(K+2)*R1*R2*R4/(Q31*Q32*Q34) 00043750
GO TO 30 00043760
24 IF ( DABS(R4).GT.1.0D-10)GO TO 26 00043770
ZINT = YINT(K+3)*R1*R2*R3/(Q41*Q42*Q43) 00043780
GO TO 30 00043790
C
26 ZINT = YINT(K)*R2*R3*R4/(Q12*Q13*Q14) +
+ YINT(K+1)*R1*R3*R4/(Q21*Q23*Q24) +
+ YINT(K+2)*R1*R2*R4/(Q31*Q32*Q34) +
+ YINT(K+3)*R1*R2*R3/(Q41*Q42*Q43) 00043800
00043810
00043820
00043830
00043840
00043850
00043860
C
30 FACTOR = 1.0D00 00043870
IF(Z.LT.0.0D00)FACTOR = -1.0D00 00043880
ZINTEG = 0.5D00 * ( 1.0D00 + FACTOR * ZINT ) 00043890
IF ( ZINTEG.GT.1.0D00 ) ZINTEG = 1.0D00 00043900
RETURN 00043910
END 00043920
00043930
00043940
00043950
00043960
00043970
00043980
00043990
00044000
00044010
00044020
00044030
00044040
00044050
00044060
00044070
00044080
00044090
00044100
00044110
00044120
00044130
00044140
00044150
00044160
00044170
00044180
00044190
00044200
CUC
SUBROUTINE TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,
+IPUMP,NORDER,EPMP)
CUC
COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1 FCR,HOURS,IBASYSR,IMXPL,ITIN,ITMAX,
2 ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3 MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
4 RESMAR,UBOUND 00044200
00044300
00044400
00044500
00044600
00044700
00044800
00044900
00044950
00044960
00044970
00044980
00044990
00045000
00045010
00045020
00045030
00045040
00045050
00045060
00045070
00045080
00045090
00045100
00045110
00045120
00045130
00045140
00045150
00045160
00045170
00045180
00045190
00045200
C
COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
1 IDEXP(20),IPABA(8),IPCH(8),ISUL(20,8),
2 IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
3 MAXUN(20,20),MAXADD(8),MINUN(20,20),
4 NEXPID(20),NSTPRE(20,20),NUPB(20,8),
5 PEAKS(20,4),PLACA(20,14),PLANT(210,14),
6 RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
7 FATOPE(20),SOL(20,2),AUPEAK(20) 00045210
00045220
00045230
00045240
00045250
00045260
00045270
00045280
00045290
00045300
00045310
00045320
00045330
00045340
00045350
00045360
00045370
00045380
00045390
00045400
00045410
00045420
00045430
00045440
00045450
00045460
00045470
00045480
00045490
00045500
00045510
00045520
00045530
00045540
00045550
00045560
00045570
00045580
00045590
00045600
00045610
00045620
00045630
00045640
00045650
00045660
00045670
00045680
00045690
00045700
00045710
00045720
00045730
00045740
00045750
00045760
00045770
00045780
00045790
00045800
00045810
00045820
00045830
00045840
00045850
00045860
00045870
00045880
00045890
00045900
00045910
00045920
00045930
00045940
00045950
00045960
00045970
00045980
00045990
00046000
00046010
00046020
00046030
00046040
00046050
00046060
00046070
00046080
00046090
00046100
00046110
00046120
00046130
00046140
00046150
00046160
00046170
00046180
00046190
00046200
C
COMMON /LCS/ EL(1250),ELDC(4,1250),ITK(420,3),
+ MXELDC(20,4),X(1250) 00046210
00046220
00046230
00046240
00046250
00046260
00046270
00046280
00046290
00046300
00046310
00046320
00046330
00046340
00046350
00046360
00046370
00046380
00046390
00046400
00046410
00046420
00046430
00046440
00046450
00046460
00046470
00046480
00046490
00046500
00046510
00046520
00046530
00046540
00046550
00046560
00046570
00046580
00046590
00046600
00046610
00046620
00046630
00046640
00046650
00046660
00046670
00046680
00046690
00046700
00046710
00046720
00046730
00046740
00046750
00046760
00046770
00046780
00046790
00046800
00046810
00046820
00046830
00046840
00046850
00046860
00046870
00046880
00046890
00046900
00046910
00046920
00046930
00046940
00046950
00046960
00046970
00046980
00046990
00047000
00047010
00047020
00047030
00047040
00047050
00047060
00047070
00047080
00047090
00047100
00047110
00047120
00047130
00047140
00047150
00047160
00047170
00047180
00047190
00047200
00047210
00047220
00047230
00047240
00047250
00047260
00047270
00047280
00047290
00047300
00047310
00047320
00047330
00047340
00047350
00047360
00047370
00047380
00047390
00047400
00047410
00047420
00047430
00047440
00047450
00047460
00047470
00047480
00047490
00047500
00047510
00047520
00047530
00047540
00047550
00047560
00047570
00047580
00047590
00047600
00047610
00047620
00047630
00047640
00047650
00047660
00047670
00047680
00047690
00047700
00047710
00047720
00047730
00047740
00047750
00047760
00047770
00047780
00047790
00047800
00047810
00047820
00047830
00047840
00047850
00047860
00047870
00047880
00047890
00047900
00047910
00047920
00047930
00047940
00047950
00047960
00047970
00047980
00047990
00048000
00048010
00048020
00048030
00048040
00048050
00048060
00048070
00048080
00048090
00048100
00048110
00048120
00048130
00048140
00048150
00048160
00048170
00048180
00048190
00048200
00048210
00048220
00048230
00048240
00048250
00048260
00048270
00048280
00048290
00048300
00048310
00048320
00048330
00048340
00048350
00048360
00048370
00048380
00048390
00048400
00048410
00048420
00048430
00048440
00048450
00048460
00048470
00048480
00048490
00048500
00048510
00048520
00048530
00048540
00048550
00048560
00048570
00048580
00048590
00048600
00048610
00048620
00048630
00048640
00048650
00048660
00048670
00048680
00048690
00048700
00048710
00048720
00048730
00048740
00048750
00048760
00048770
00048780
00048790
00048800
00048810
00048820
00048830
00048840
00048850
00048860
00048870
00048880
00048890
00048900
00048910
00048920
00048930
00048940
00048950
00048960
00048970
00048980
00048990
00049000
00049010
00049020
00049030
00049040
00049050
00049060
00049070
00049080
00049090
00049100
00049110
00049120
00049130
00049140
00049150
00049160
00049170
00049180
00049190
00049200
00049210
00049220
00049230
00049240
00049250
00049260
00049270
00049280
00049290
00049300
00049310
00049320
00049330
00049340
00049350
00049360
00049370
00049380
00049390
00049400
00049410
00049420
00049430
00049440
00049450
00049460
00049470
00049480
00049490
00049500
00049510
00049520
00049530
00049540
00049550
00049560
00049570
00049580
00049590
00049600
00049610
00049620
00049630
00049640
00049650
00049660
00049670
00049680
00049690
00049700
00049710
00049720
00049730
00049740
00049750
00049760
00049770
00049780
00049790
00049800
00049810
00049820
00049830
00049840
00049850
00049860
00049870
00049880
00049890
00049900
00049910
00049920
00049930
00049940
00049950
00049960
00049970
00049980
00049990
00049999
CUC

```

```

C
C      COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+      MINTUN(8),NUBOLD(8)                                00044210
00044220
00044230
00044240
00044250
00044260
00044270
00044280
00044290
00044300
00044310
00044320
00044330
00044340
00044350
00044360
00044370
00044380
00044390
00044400
00044410
00044420
00044430
00044440
00044450
00044460
00044470
00044480
00044490
00044500
00044510
00044520
00044530
00044540
00044550
00044560
00044570
00044580
00044590
00044600
00044610
00044620
00044630
00044640
00044650
00044660
00044670
00044680
00044690
00044700
00044710
00044720
00044730
00044740
00044750
00044760
00044770
00044780
00044790
00044800
00044810
00044820
00044830
00044840
00044850
00044860
00044870
00044880

C      COMMON /TBL/ TINT(120),TY(120)
C      DOUBLE PRECISION TINT,TY
C      COMMON /HYS/ CMULT(4),EAVAL(3),EMUL,T(4),HYDR(3,11),
+      HYEXP(11),HYSCHD(20,8),PSEXPN(11),
+      PSSCHD(20,8),SYSCUM(4)                            00044210
00044220
00044230
00044240
00044250
00044260
00044270
00044280
00044290
00044300
00044310
00044320
00044330
00044340
00044350
00044360
00044370
00044380
00044390
00044400
00044410
00044420
00044430
00044440
00044450
00044460
00044470
00044480
00044490
00044500
00044510
00044520
00044530
00044540
00044550
00044560
00044570
00044580
00044590
00044600
00044610
00044620
00044630
00044640
00044650
00044660
00044670
00044680
00044690
00044700
00044710
00044720
00044730
00044740
00044750
00044760
00044770
00044780
00044790
00044800
00044810
00044820
00044830
00044840
00044850
00044860
00044870
00044880

C      DIMENSION CFH(3),ENY(2,30),ENP(2,30),ENRH(3,2),NORDER(4)

C      IF(KLOW2.LE.0) GO TO 10
C      ISWP=2
C      ITRM=KLOW2
C      L=KLOW1
C      GO TO 15
10     L=KLOW1-1
C      ISWP=1
C      ITRM=ITK(L,3)+KLOW2
15     IP=ITK(L,1)
ID=ITK(L,2)
IF(IBLK.GT.0) GO TO 30

C      TRIM HYDRO BLOCK

C      IH=NORDER(NOHY)
C      DIFF=EAVAL(IH)-ENRH(IH,2)
C      ENRH(IH,2)=EAVAL(IH)
C      IF(NOHY.LE.1) GO TO 40
C      ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF
C      GO TO 60

C      TRIM HYDRO CLUSTER

30     IH=NORDER(NOHY)
NPAST=NOHY-1
IMPT=NORDER(NPAST)
C      DIFF=EAVAL(IH)+EAVAL(IMPT)-SUEN2
ENRH(IH,2)=EAVAL(IH)
ENRH(IMPT,2)=EAVAL(IMPT)
C      IF(NPAST.LE.1) GO TO 40
C      ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF
C      GO TO 60

C      PUMPING ENERGY ADJUSTMENT

40     IF(IPUMP.GT.0) GO TO 50
EDF=ENP(ISWP,ITRM)*(DIFF/ENY(ISWP,ITRM))
ENP(ISWP,ITRM)=ENP(ISWP,ITRM)-EDF
ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF
EPMP=EPMP-EDF*HYDR(3,3)
EAVAL(3)=EPMP*HYDR(3,4)
P=1.0-HYDR(3,5)
CP=HYDR(3,1)
IF(CP.LE.0.0) GO TO 60
CFH(3)=EAVAL(3)/(2190.0*P*CP)

```

GO TO 60
 50 ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF
 60 RETURN
 END

00044890
 00044900
 00044910
 00044920
 00044930
 00044940
 00044950
 00044960
 00044970
 00044980
 00044990
 00045000
 00045010
 00045020
 00045030
 00045040
 00045050
 00045060
 00045070
 00045080
 00045090
 00045100
 00045110
 00045120
 00045130
 00045140
 00045150
 00045160
 00045170
 00045180
 00045190
 00045200
 00045210
 00045220
 00045230
 00045240
 00045250
 00045260
 00045270
 00045280
 00045290
 00045300
 00045310
 00045320
 00045330
 00045340
 00045350
 00045360
 00045370
 00045380
 00045390
 00045400
 00045410
 00045420
 00045430
 00045440
 00045450
 00045460
 00045470
 00045480
 00045490
 00045500
 00045510
 00045520
 00045530
 00045540
 00045550

SUBROUTINE TUNNEL(NSTATE, NYNEXT, IFLTUN)

TITLE: TUNNEL
 FUNCTION: THIS SUBROUTINE DEFINES THE RANGE (CALLED
 TUNNEL) IN THE NUMBER OF UNITS THAT EXPANSION
 PLANT TYPES CAN HAVE IN STATE NYNEXT WHEN
 THEY ORIGINATE FROM STATE "NSTATE" IN STAGE
 "NY". THE CONSTRAINTS THAT ARE CONSIDERED IN
 THE "TUNNEL" DEFINITION ARE BOTH REAL AND
 ARTIFICIAL.
 1. REAL CONSTRAINTS
 A. MINIMUM NUMBER OF UNITS REQUIRED
 (MINUN)
 B. MAXIMUM NUMBER OF UNITS ALLOWED (MAXUN)
 C. MAXIMUM NUMBER OF UNITS THAT CAN BE
 CONSTRUCTED IN A YEAR FOR EACH EXPANSION
 PLANT TYPE (MAXADD)
 2. ARTIFICIAL CONSTRAINTS
 A. ARTIFICIAL UPPER BOUNDARY (NUPB)
 B. ARTIFICIAL LOWER BOUNDARY (LOWB)

IF IFLTUN>=0 THE TUNNEL FOR STATE "NSTATE"
 HAS ALREADY BEEN DEFINED BUT THE NUMBER OF
 STATES ACCEPTED IN STAGE "NYNEXT" IS SMALLER
 THAN THE MINIMUM NUMBER ALLOWED. THEREFORE,
 ADDITIONAL STATES ARE GENERATED BY RELAXING
 THE ARTIFICIAL CONSTRAINTS (2A AND 2B).
 IN THIS CASE ONLY THE REGION NOT INCLUDED IN
 THE OLD TUNNELS WILL CONSTITUTE THE SUPPLE-
 MENTAL TUNNEL.

SUBROUTINE ARGUMENTS
 NAME DESCRIPTION
 IFLTUN FLAG -1 = ORIGINAL TUNNELS
 0 = NO TUNNEL ADJUSTMENTS POSSIBLE
 1 = LOWER BOUNDARY DECREASED
 2 = UPPER BOUNDARY INCREASED
 NSTATE UNPACKED ORIGIN STATE IN STATE NY=NYNEXT-1
 NYNEXT CURRENT STAGE

CALLED BY : MAIN
 CALLS : NONE

COMMON VARIABLES USED: IDEXP, LBOLD, MAXTUN, MINTUN,
MXPL, NUBOLD

LOCAL VARIABLES

NAME	DESCRIPTION
I	EXPANSION PLANT TYPE INDEX
IADD	MAXIMUM NUMBER OF UNITS ALLOWED FOR EXPANSION PLANT TYPE I DUE TO LIMIT OF UNIT ADDITION EACH YEAR
IP	I-TH EXPANSION PLANT TYPE ID IN THE ORIGINAL EXPANSION PLANT TYPES LIST

00045320
 00045330
 00045340
 00045350
 00045360
 00045370
 00045380
 00045390
 00045400
 00045410
 00045420
 00045430
 00045440
 00045450
 00045460
 00045470
 00045480
 00045490
 00045500
 00045510
 00045520
 00045530
 00045540
 00045550

```

      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1      FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
2      ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3      MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
4      RESMAR,UBOUND          00045570
00045580
00045590
00045600
00045610
00045620
00045630
00045640
00045650
00045660
00045670
00045680
00045690
00045700
00045710
00045720
00045730
00045740
00045750
00045760
00045770
00045780
00045790
00045800
00045810
00045820
00045830
00045840
00045850
00045870
00045880
00045890
00045900
00045910
00045920
00045930
00045940
00045950
00045960
00045970
00045980
00045990
00046000
00046010
00046020
00046030
00046040
00046050
00046060
00046070
00046080
00046090
00046100
00046110
00046120
00046130
00046140
00046150
00046160
00046170
00046180
00046190
00046200
00046210
00046220
00046230
00046240

      COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
1      IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
2      IWIDTH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
3      MAXUN(20,20),MAXADD(8),MINUN(20,20),
4      NEXPID(20),NSTPRE(20,20),NUPB(20,8),
5      PEAKS(20,4),PLACA(20,14),PLANT(210,14),
6      RESHAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
7      FATOPE(20),SOL(20,2),AUPEAK(20) 00045600
00045610
00045620
00045630
00045640
00045650
00045660
00045670
00045680
00045690
00045700
00045710
00045720
00045730
00045740
00045750
00045760
00045770
00045780
00045790
00045800
00045810
00045820
00045830
00045840
00045850
00045870
00045880
00045890
00045900
00045910
00045920
00045930
00045940
00045950
00045960
00045970
00045980
00045990
00046000
00046010
00046020
00046030
00046040
00046050
00046060
00046070
00046080
00046090
00046100
00046110
00046120
00046130
00046140
00046150
00046160
00046170
00046180
00046190
00046200
00046210
00046220
00046230
00046240

      COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+      MXELDC(20,4),X(1250) 00045700
00045710
00045720
00045730
00045740
00045750
00045760
00045770
00045780
00045790
00045800
00045810
00045820
00045830
00045840
00045850
00045870
00045880
00045890
00045900
00045910
00045920
00045930
00045940
00045950
00045960
00045970
00045980
00045990
00046000
00046010
00046020
00046030
00046040
00046050
00046060
00046070
00046080
00046090
00046100
00046110
00046120
00046130
00046140
00046150
00046160
00046170
00046180
00046190
00046200
00046210
00046220
00046230
00046240

      COMMON /CMS/ PTCUM(20,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)
      COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+      MINTUN(8),NUBOLD(8) 00045700
00045710
00045720
00045730
00045740
00045750
00045760
00045770
00045780
00045790
00045800
00045810
00045820
00045830
00045840
00045850
00045870
00045880
00045890
00045900
00045910
00045920
00045930
00045940
00045950
00045960
00045970
00045980
00045990
00046000
00046010
00046020
00046030
00046040
00046050
00046060
00046070
00046080
00046090
00046100
00046110
00046120
00046130
00046140
00046150
00046160
00046170
00046180
00046190
00046200
00046210
00046220
00046230
00046240

      COMMON /TBL/ TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
      COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
+      HYEXP(11),HVSCHD(20,8),PSEXP(11),
+      PSSCHD(20,8),SYSCUM(4) 00045700
00045710
00045720
00045730
00045740
00045750
00045760
00045770
00045780
00045790
00045800
00045810
00045820
00045830
00045840
00045850
00045870
00045880
00045890
00045900
00045910
00045920
00045930
00045940
00045950
00045960
00045970
00045980
00045990
00046000
00046010
00046020
00046030
00046040
00046050
00046060
00046070
00046080
00046090
00046100
00046110
00046120
00046130
00046140
00046150
00046160
00046170
00046180
00046190
00046200
00046210
00046220
00046230
00046240

      DIMENSION NSTATE(20)

      IF THIS IS NOT THE FIRST TIME THIS ROUTINE IS CALLED.
      FOR STATE "NSTATE", BRANCH TO SUPPLEMENTAL TUNNEL
      DEFINITION
      IF (IFLTUN .GT. 0) GO TO 20
      SET ORIGINAL TUNNEL
      DO 10 I=1,MXPL
      IP=IDEXP(I)+2
      LOWER BOUND
      MINTUN(I) = MAXO(NSTATE(I), LOWB(NYNEXT,I), MINUN(NYNEXT,IP)) 00045900
00045910
00045920
00045930
00045940
00045950
00045960
00045970
00045980
00045990
00046000
00046010
00046020
00046030
00046040
00046050
00046060
00046070
00046080
00046090
00046100
00046110
00046120
00046130
00046140
00046150
00046160
00046170
00046180
00046190
00046200
00046210
00046220
00046230
00046240
      UPPER BOUND
      IADD = MAXADD(I) + NSTATE(I)
10 MAXTUN(I) = MINO(IADD, NUPB(NYNEXT,I), MAXUN(NYNEXT,IP)) 00046040
00046050
00046060
00046070
00046080
00046090
00046100
00046110
00046120
00046130
00046140
00046150
00046160
00046170
00046180
00046190
00046200
00046210
00046220
00046230
00046240
      RETURN
20 CONTINUE
      SET SUPPLEMENTAL TUNNEL
      IF THE UPPER BOUNDARY WAS CHANGED BRANCH TO 40
      IF (IFLTUN .GT. 1) GO TO 40
      LOWER BOUND WAS CHANGED
      DO 30 I=1,MXPL
      IP=IDEXP(I)+2
      LOWER BOUND
      MINTUN(I) = MAXO(NSTATE(I), LOWB(NYNEXT,I), MINUN(NYNEXT,IP)) 00046130
00046140
00046150
00046160
00046170
00046180
00046190
00046200
00046210
00046220
00046230
00046240
      UPPER BOUND
30 MAXTUN(I) = LBOLD(I)
      RETURN
      UPPER BOUNDARY WAS CHANGED 00046140
00046150
00046160
00046170
00046180
00046190
00046200
00046210
00046220
00046230
00046240

```

```

40 DO 50 I=1,MXPL
      IP=IDEXP(I)+2
      LOWER BOUND
      MINTUN(I) = NUBOLD(I)
      UPPER BOUND
      IADD = MAXADD(I) + NSTATE(I)
50 MAXTUN(I) = MIN0(IADD, NUPB(NYNEXT,I), MAXUN(NYNEXT,IP))
C      RETURN
C      END
C


---


C      SUBROUTINE UFIRST
C
C      CALCULATES THE FIRST ESTIMATE OF THE OBJECTIVE FUNCTION. BASED
C      ON TRADITIONAL OBJ DEFINITION AND DISCOUNT FACTOR OF 10%.
C
C      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1      FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
2      ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3      MAXPLA,MAXPO,MXALL,MXPL,MXYEAR,
4      RESMAR,UBOUND
C      COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
1      IDEXP(20),IPABA(8),IPCCH(8),ISOL(20,8),
2      IWIDTH(8),LORDER(420,3),LOWBL(20,8),MAINS(200),
3      MAXUN(20,20),MAXADD(8),MINUN(20,20),
4      NEXPID(20),NSTPRE(20,20),NUPB(20,8),
5      PEAKS(20,4),PLACA(20,14),PLANT(210,14),
6      RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
7      FATOPE(20),SOL(20,2),AUPEAK(20)
C      COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+      MXELDC(20,4),X(1250)
C      COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),ROM(40,4),PTCUMH(3,8)
C      COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+      MINTUN(8),NUBOLD(8)
C      COMMON /TBL/ TINT(120),TY(120)
C      DOUBLE PRECISION TINT,TY
C      COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
+      HYEXP(11),HYSCHD(20,8),PSEXP(11),
+      PSSCHD(20,8),SYSCUM(4)
C
C      DIMENSION NSTATE(20), NSTCUR(20), RESCP(20,20),NUF(20)
C      IF (ITOUT .LE. 0) GO TO 10
C      RETURN
10 CONTINUE
      OBJ = 0.0
C      ANNUAL PEAK DEMAND
C      DO 130 NY=1,MXYEAR

```

```

00046250
00046260
00046270
00046280
00046290
00046300
00046310
00046320
00046330
00046340
00046350
00046360
00046370
00046380
00046390
00046400
00046410
00046420
00046430
00046440
00046450
00046460
00046470
00046480
00046490
00046500
00046510
00046520
00046530
00046540
00046550
00046560
00046570
00046580
00046590
00046600
00046610
00046620
00046630
00046640
00046650
00046660
00046670
00046680
00046690
00046700
00046710
00046720
00046730
00046740
00046750
00046760
00046770
00046780
00046790
00046800
00046810
00046820
00046830
00046840
00046850
00046860
00046870
00046880
00046890
00046900
00046910
00046920

```

```

PK=0.0
DO 135 J=1,4
IF(PK.GT.PEAKS(NY,J)) GO TO 135
PK=PEAKS(NY,J)
135 CONTINUE
AUPEAK(NY)=PK
130 CONTINUE
CCCCC
MXPL=MAXINP+2
IDEXP(1)=-1
IDEXP(2)=0
DO 20 I=3,MXPL
II=I-2
IDEXP(I)=II
20 NEXPID(I)=II
MXALL = MAXPLA + MAXINP
REWIND 23
DO 40 IP = 1,MXALL
READ(23,900) (PLANT(IP,J),J=1,14)
40 CONTINUE
C
940 WRITE(40,940) MAXPLA,MAX INP,MXYEAR
940 FORMAT(3I5)
C
DO 45 I=1,MAXPLA
READ(23,950) (NUF(N),N=1,MXYEAR)
WRITE(40,950) (NUF(N),N=1,MXYEAR)
45 CONTINUE
950 FORMAT(40I2)
C
EHY=30.0
EPS=30.0
I1=MAXINP+1
960 WRITE(40,960) EHY,EPS,(PLANT(I,9),I=I1,MXALL)
960 FORMAT(20F3.0)
C
970 WRITE(40,970) (PLANT(I,6),I=I1,MXALL)
970 FORMAT(10F7.2)
980 WRITE(40,970) HYEXP(9),PSEXP(9),(PLANT(I,6),I=1,MAXINP)
980 FORMAT(F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2)
990 FORMAT(F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2)
901 FORMAT(1H ,F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2)
FCR = -1.0
DISRAT = 0.10
IRES = 0
REWIND 24
REWIND 27
DO 50 NY = 1,MXYEAR
DO 30 IP = 1,MXPL
NSTCUR(IP) = NSTPRE(NY,IP)
NSTATE(IP) = 0
IF (NY .NE. 1) NSTATE(IP) = NSTPRE(NY-1,IP)
30 CONTINUE
DO 35 N = 1,4
READ(27)((PTCUM(I,J,N),I=1,201),J=1,8)
35 WRITE(11,*)(PTCUM(I,1,N),I=1,31)
35 CONTINUE
READ(24,902)((LORDER(J,K),K=1,3),J=1,420)
902 FORMAT(125(13,12,13))

```

```

C      WRITE(11,*)
      READ(24,921) ((LORDER(J,K),K=1,3),J=1,100)
921    FORMAT(200I2)
C      WRITE(11,*)
      CALL OPERCD(NSTCUR, NY, OPEC, IRES)
      CALL OBJFUN(NSTCUR, NY, OPEC, NSTATE, CAPCO, OBJJ, OBJ1, S,
      +RESCAP, IRES)
      WRITE(6,*)
      + (NSTCUR(I),I=1,MXPL),NY,NSTATE,IRES
      WRITE(6,*)
      + OPEC,CAPCO,OBJ1,S

      WRITE(11,*)
      + (NSTCUR(I),I=1,MXPL),NY,NSTATE,IRES
      WRITE(11,*)
      + OPEC,CAPCO,OBJ1,S
50     OBJ = OBJ+OBJ1
      UBOUND = OBJ*1.5
      WRITE(6,*)
      + UBOUND
      WRITE(11,*)
      + UBOUND
200    RETURN
END

SUBROUTINE UNPACK(NPA, NUNP)

TITLE: STATE UNPACKING
FUNCTION: UNPACKS STATE "NST" INTO "NSTCUR" USING
          THE PACKING BASE IPABA.
CALLED BY: MAIN
CALLS TO: NONE
SUBROUTINE ARGUMENTS
NPA      PACKED STATE
NUNP     UNPACKED STATE
COMMON VARIABLES USED: IPABA, MXPL
LOCAL VARIABLES
NAME      DESCRIPTION
I         I-TH PLANT TYPE
II        BACKWARD PLANT TYPE INDEX
NSTEMP   PACKED STATE REMAINDER
COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
1       FCR,HOURS,IBASYSR,IMXPL,ITIN,ITMAX,
2       ITOUT,MAXALL,MAXI,MAXINP,MAXOR,
3       MAXPLA,MAXPU,MXALL,MXPL,MYEAR,
4       RESMAR,UBOUND
COMMON /ARS/ AVSP(20,4),CAPABS(20),ENEDEM(20,4),EXPLCU(4,8),
1       IDEXP(20),IPABA(8),IPCH(8),ISOL(20,8),
2       IWIOH(8),LORDER(420,3),LOWB(20,8),MAINS(200),
3       MAXUN(20,20),MAXADD(8),MINUN(20,20),
4       NEXPID(20),NSTPRE(20,20),NUPB(20,8),
5       PEAKS(20,4),PLACAI(20,14),PLANT(210,14),
6       RESMAX(20),RESMIN(20),SSCUM(4,20),UNE(20),
7       FATOPE(20),SQL(20,2),AUPEAK(20)
COMMON /LCS/ EL(1250),ELDC(4,1250),ELF(1260),
+             MXELDC(20,4),X(1250)
COMMON /CMS/ PTCUM(201,8,4),PTCUMX(20,8),RDW(40,4),PTCUMH(3,8)
COMMON /TNS/ LBOLD(8),LIST(8,1000),MAXTUN(8),
+             MINTUN(8),NUBOLD(8)

```

```

C COMMON /TBL/ TINT(120),TY(120)          00048290
C DOUBLE PRECISION TINT,TY               00048300
C COMMON /MYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11), 00048310
C      + HYEXPN(11),MYSCHD(20,8),PSEXPN(11), 00048320
C      + PSSCHD(20,8),SYSCUM(4)           00048330
C
C DIMENSION NUNP(20)                      00048340
C NSTEMP = NPA                           00048350
C DO 10 I=1,MXPL                         00048360
C     II = MXPL - I + 1                   00048370
C     NUNP(II) = NSTEMP/IPABA(II) * IPABA(II) 00048380
C 10 NSTEMP = NSTEMP - NUNP(II) * IPABA(II) 00048390
C RETURN
C END
C
C SUBROUTINE VALUET(Z,C3,C4,FRESGL)
C
C COMMON /TBL/ TINT(120),TY(120)
C DOUBLE PRECISION TY,TINT
C
C DOUBLE PRECISION Z,ZN(8),FRE,F3,F4,F6,E,ZINTEG,ZSQ
C IF ( Z.GT.5.90D00 ) GO TO 40
C ZSQ = Z*Z/2.0D00
C E = DEXP(-ZSQ)/2.506628275D00
C ZN(1) = -Z * E
C ZN(2) = E * (Z*Z - 1.0D00)
C ZN(3) = -Z*E*(Z*Z - 3.0D00)
C ZN(4) = E*(Z*Z*(Z*Z - 6.0D00) + 3.0D00)
C ZN(5) = -Z*E*(Z*Z*(Z*Z - 10.0D00) + 15.0D00)
C F3 = 6.0D00
C F4 = 24.0D00
C F6 = 720.0D00
C CALL TALOOK(Z,ZINTEG)
C FRE = 1.0D00 - ZINTEG
C FRE = FRE + DBLE(C3)*ZN(2)/F3
C      + DBLE(C4)*ZN(3)/F4
C      + DBLE(C3)*DBLE(C3)*ZN(5)*10.0D00/F6
C IF(FRE.LT.0.0D00) FRE=0.0D00
C IF(FRE.GT.1.0D00) FRE=1.0D00
C FRESGL = SNGL(FRE)
C
C RETURN
40 FRESGL = 0.0
C RETURN
C END

```

MAIN PROGRAM		
		00000010
THIS PROGRAM PERFORMS FINANCIAL CALCULATIONS FOR THE OPTIMAL OR SUBOPTIMAL CAPACITY EXPANSION PLAN OBTAINED FROM THE DYNOMODULE.		00000020
		00000030
		00000040
		00000050
		00000060
		00000070
		00000080
		00000090
		00000100
		00000110
		00000120
		00000130
		00000140
		00000150
		00000160
		00000170
		00000180
		00000190
		00000200
		00000210
		00000220
		00000230
		00000240
		00000250
		00000260
		00000270
		00000280
		00000290
		00000300
		00000310
		00000320
		00000330
		00000340
		00000350
		00000360
		00000370
		00000380
		00000390
		00000400
		00000410
		00000420
		00000430
		00000440
		00000450
		00000460
		00000470
		00000480
		00000490
		00000500
		00000510
		00000520
		00000530
		00000540
		00000550
		00000560
		00000570
		00000580
		00000590
		00000600
		00000610
		00000620
		00000630
		00000640
		00000650
COMMON /DEBTS/ DEBS(20),DEBL(20),DEBR(20),CDEBS(20), + CDEBL(20),RDEBS(20),RDEBL(20),DEBT(20), + DEBT(20),DEBN(20),FDEBT(20),FDEBS(20),		00000660
		00000670
		00000680

```

+      FDEBL(20),FDEBR(20,20,2)          00000690
+      STCOM(20),STPK(20),FSTCOM(20),FSTPR(20), 00000700
+      DIVCOM(20),DIVPR(20),KIVC(20),RDIVP(20), 00000710
+      FDIVC(20),FDIVP(20),KFDEBL(20)          00000720
+      GREV(20),OPINC(20),TXINC(20),ARATE(20), 00000730
+      AINC(20),BINC(20),UMC(20),FERN(20), 00000740
+      ANINC(20),ERNET(20),ERNRT(20), 00000750
+      OMCF(20),UMCX(20)          00000760
+      COMMON /INPUTS/ FINDAT(20,30),BDAT(20),KEYFIN(140,18) 00000770
+      COMMON /PLANTS/ PLGS1(20),PLGA1(20),PLNS1(20),PLNA1(20), 00000780
+      PLNS2(20),ITPLX(20),LPLX(20),IPLEX(22,20), 00000790
+      PLX(22,20),LEPA(22,20),CEPLX(22,20), 00000800
+      CEX(20),PLGSP(20),CEP(20),CWIP(20), 00000810
+      CWIPC(20),DEPLX(22,20),DEPLX1(22,20), 00000820
+      DEPLX2(22,20),DEPX1(20),DEPX2(20),DEPF1(20), 00000830
+      DEPF2(20),DEP1(20),DEP2(20),AFDCF2(20), 00000840
+      AFDCR(20),AFDC1(20),AFDC2(20),AFCX2(22,20), 00000850
+      CAPRO(20),RATB(20),CE1(20),AFDCX2(20), 00000860
+      CEPXC(20,20),CE1(20)          00000870
+      SRCIN(20),SKREX(20),SRC(20),RWKCAP(20), 00000871
+      USE(20),CAPIN(20),SRCTOT(20),USETOT(20), 00000880
+      WKCAP(20),DWKCAP(20),WKAPI(20)          00000881
+      KTXFD(20),TXFD(20),KTXST(20),TXST(20), 00000890
+      RTXP(20),TXP(20),RTXCR(20),TXCR(20)          00000900
+      PLGS2(20),DEPIC(20),DEP2C(20),BAL(20), 00000910
+      ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20), 00000920
+      TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20), 00000930
+      OPEXP(20),TOTEXP(20),PLGAZ(20),PLNV(20), 00000940
+      PLNUT(20),STCOMC(20),STPRC(20),ERNRTC(20), 00000950
+      FDEBLR(20,20),FDEBSR(20,20),TXREV(20),RTXREV(20), 00000960
+      CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20), 00000970
+      PLNAZ(20),DEBLT(20),DEBSTR(20),DEBLR(20), 00000980
+      DEBSR(20),FC5(20),LCR(20),TXCRC(20), 00000990
+      OPCOST(20),PLGSX(20),TXDEF(20), 00001000
+      DSLIM(20),DLLIM(20),COMLIM(20), 00001010
+      PRLLIM(20),FCWIP(20),BINT(20),CWIPC1(20), 00001020
+      TXSPD(20),FUEL(20),ITPL(20),PLC(200), 00001030
+      IPL(200,20),PLI(200,20),ELPLI(200),LPLI(200), 00001040
+      DEPLI(200,20),DEPL2(200,20),DEPL(200,20), 00001050
+      FSUM(20,20),EQTD(20),DEBASS(20), 00001060
+      DEBEQ(20),COVRG(20),KYR(20,2)          00001070
+      PLGSP,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB, 00001080
+      DEBSB,DEBLB,DEBSRB,DEBLRB,STCOMB,STPRB, 00001090
+      ERNRTB,AFDC1B,TXDEFB,TXCRB,WKCAPB, 00001100
+      AFDC2B          00001110
+      NF1,PLW(200,20),ELHF,ELSF,ELHX, 00001111
+      ELSX,PLCC(200),HXCC,SXCC,HFCAP(20), 00001120
+      SFCAP(20),PLLAP(200),IDEXP(20),NEXP, 00001130
+      NBEST,HXC,SXC,ISOL(20,20)          00001140
+      COMMON /PARM/ NBYK,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC 00001150
+      COMMON /FLAGS/ 1STOP          00001160
+      INTEGER RESP,YES,YE,NEG,NO          00001170
+      00001180          00001190
+      00001190          00001200
+      DATA YES,YE/3HYES,1HY/          00001210
+      DATA NEG,NE/2HNO,1HN/          00001220
+      1STOP=0          00001230
+      CCC          00001240
+      CCC          00001250
+      READ DATA FROM DYNO OUTPUT FILE          00001260
+      WRITE (6,11)          00001270
11 FORMAT (//,27H BEGINNING OF FINAN MODULE)          00001280
READ (21,15) MXPLF,MXPLX,MXYS          00001290
15 FORMAT (315)
MXALL=MXPLF+MXPLX          00001300
NF1=MXPLX+1          00001310
DO 17 I=NF1,MXALL          00001320
DO 17 I=NF1,MXALL          00001330

```

```

17 READ (21,20) (IPLW(I,NY),NY=1,MXYR)          00001340
18 CONTINUE
19 FORMAT (200I2)
20 READ (21,22) ELHF,ELSF,(ELPL(I),I=NFI,MXALL) 00001350
21 READ (21,22) ELHX,ELSX,(ELPL(I),I=1,MXPLX)   00001360
22 FORMAT (20F3.0)                                00001370
23 READ (21,25) (PLCC(I),I=NFI,MXALL)            00001380
24 READ (21,25) HXCC,SXCC,(PLCC(I),I=1,MXPLX)   00001390
25 FORMAT (10F7.2)                                00001400
26 READ (21,27) (HFCAP(NY),NY=1,MXYR)            00001410
27 READ (21,27) (SFCAP(NY),NY=1,MXYR)            00001420
28 READ (21,27) (PLCAP(I),I=NFI,MXALL)           00001430
29 READ (21,27) HXCAP,SXCAP,(PLCAP(I),I=1,MXPLX) 00001440
30 FORMAT (16F5.0)                                00001450
31 READ (21,30) MXPL                            00001460
32 FORMAT (12)                                    00001470
33 READ (21,32) (IDEXP(I),I=1,MXPL)             00001480
34 WRITE (6,31) MXPL,(IDEXP(I),I=1,MXPL)         00001490
35 FORMAT (//,1H ,I3,23H OPTIMAL SOLUTIONS ARE,    00001500
*33H AVAILABLE FROM DYNQ OUTPUT FILE.)          00001510
36 *12H UNIT TYPES./24H THEY HAVE THE FOLLOWING, 00001520
*19H UNIT CODE NUMBERS:/814)                   00001530
37 FORMAT (8I12)                                 00001540
38 READ (21,30) NBEST                          00001550
39 WRITE (6,33) NBEST                          00001560
40 FORMAT (//,1H ,I3,23H COMPUTATIONS TO BEGIN FOR, 00001570
*9H SOLUTION,I2,/20H IF YOU WANT TO STOP,       00001580
*8H ENTER 1/2H ?)                           00001590
41 READ (5,38) 1STOP                         00001600
42 FORMAT (11)                                    00001610
43 IF (1STOP.EQ.1) GO TO 99                  00001620
44 DO 40 NY=1,MXYR                           00001630
45 READ (21,47) (ISOL(I,NY),I=1,MXPL)        00001640
46 FORMAT (8I4)                                00001650
47 READ (21,42) FUEL(NY)                      00001660
48 READ (21,42) OMC(NY)                      00001670
49 READ (21,46) (PLX(I,NY),I=1,MXPL)        00001680
50 CONTINUE
51 CONTINUE
52 FORMAT (E12.5)                                00001690
53 FORMAT (5E12.3)                                00001700
54 HXC=HXCC*MXCAP                            00001710
55 SXC=SXCC*SXCAP                            00001720
56 DO 50 I=1,MXPL                            00001730
57 J=IDEXP(I)
58 IF (J) 52,54,56                            00001740
59 PLC(I)=MXCC*MXCAP*0.001
60 LPL(I)=ELHX
61 GO TO 58
62 PLC(I)=SXC*SXCAP*0.001
63 LPL(I)=ELSX
64 GO TO 58
65 PLC(I)=PLCC(J)*PLCAP(J)*0.001
66 LPL(I)=ELPL(J)
67 IPLX(I,1)=ISOL(I,1)
68 PLX(I,1)=PLX(I,1)*PLC(I)
69 DO 60 NY=2,MXYR
70 IPLX(I,NY)=ISOL(I,NY)-ISOL(I,NY-1)
71 PLX(I,NY)=IPLX(I,NY)*PLC(I)
72 CONTINUE
73 CONTINUE
74 IF (NSOL.GT.1) GO TO 70
75 CALL INFIN
76 CALL RATBAS
77 CALL CAPTL

```

```

IF (ISTOP.EQ.1) GO TO 99
CALL REPORT
35 CONTINUE
99 CONTINUE
WRITE (6,115)
115 FORMAT (//,24H END OF PROGRAM RUN WITH,
+23H FINAN MODULE OF CERES.)
STOP
END

```

SUBROUTINE INFIN

THIS SUBROUTINE ACCEPTS USER SUPPLIED VALUES OF
THE INPUT PROCEDURE IS ENTIRELY INTERACTIVE.

```

COMMON /DEBTS/ DEBS(20),DEBL(20),DEBK(20),CDEBS(20),
+ COEBS(20),KDEBS(20),FDEBL(20),DEBT(20),
+ CDEBT(20),DEBN(20),FDEBT(20),FDEBS(20),
+ FDEBL(20),FDEBR(20,20),
+ STCOM(20),STPR(20),FSTCOM(20),FSTPR(20),
+ DIVCOM(20),DIVPR(20),KDIVC(20),RDIVP(20),
+ FDIVC(20),FDIVP(20),RFDEBL(20),
+ GREVL(20),UPINC(20),TXINC(20),ARATE(20),
+ AINC(20),BINC(20),UMC(20),FEKN(20),
+ ANINC(20),ERNET(20),ERNKT(20),
+ OMCF(20),UMCX(20)
COMMON /INPUTS/ FINUAT(20,30),BUAT(20),KEYFIN(140,18)
COMMON /PLANTS/ PLGS1(20),PLGA1(20),PLNS1(20),PLNA1(20),
+ PLNS2(20),ITPLX(20),LPLX(20),IPLEX(22,20),
+ PLX(22,20),CEPX(22,20),CEPLX(22,20),
+ CEX(20),PLGSF(20),CEF(20),CWIP(20),
+ CWIPC(20),DEPLX(22,20),DEPLX1(22,20),
+ DEPLX2(22,20),DEPX1(20),DEPX2(20),DEPF1(20),
+ DEPF2(20),DEPI(20),DEP2(20),AFDCF2(20),
+ AFDCR(20),AFDC1(20),AFDC2(20),AFDCX2(20),
+ CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
+ CEPXC(20,20),CE1(20)
COMMON /FUNDS/ SRCIN(20),SKCEX(20),SRC(20),RWKCAP(20),
+ USE(20),CAPIN(20),SRCTOT(20),USETOT(20),
+ WKCAP(20),DWKCAP(20),WKCAP1(20)
COMMON /TAXES/ RIXFD(20),TXFD(20),RTAST(20),TXST(20),
+ RTXP(20),TXP(20),RTXCR(20),TXCR(20)
COMMON /AUX/ PLGS2(20),DEPIC(20),DEP2C(20),BAL(20),
+ ASSET(20),EQCOM(20),ERKNCOM(20),CAPTOT(20),
+ TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
+ OPEXP(20),TDXEXP(20),PLGA2(20),PLNV(20),
+ PLNUT(20),STCUMC(20),STPRC(20),ERNRTC(20),
+ FDEBLR(20,20),FDEBSR(20,20),TXREV(20),RTXREV(20),
+ CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20),
+ PLNA2(20),DEBLT(20),DEGST(20),DEBLR(20),
+ DEBSR(20),FCE(20),LCR(20),TXCRC(20),
+ OPCOST(20),PLGSX(20),TXDEFCT(20)
COMMON /AUX2/ DSLIM(20),DLIM(20),CLMLIM(20),
+ PRLIM(20),FCWIP(20),BINT(20),CWIPC1(20),
+ TXSPD(20),FUEL(20),ITPL(20),PLC(200),
+ IPL(200,20),PL(200,20),ELPL(200),LPL(200),
+ DEPL1(200,20),DEPL2(200,20),DEPL(200,20),
+ FSUM(20,20),EQTOT(20),DEBASS(20),
+ DEBEQ(20),COVKG(20),KYK(20,2)

```

```

COMMON /BVAL/ PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
+ DEBSB,DESLB,DEBSKB,DEBLRB,STCOMB,STPRB,
+ ERNRTB,AFDC1B,TXDEFB,TXCRB,WKCAPB,
+ AFDC2B
COMMON /INDYN/ NF1,IPLW(200,20),ELHF,ELSF,ELHX,
+ ELSX,PLCC(200),HXCC,SXCC,HFCAP(20),
+ SFCAP(20),PLCAP(200),LUEXP(20),NEXP,
+ NBEST,HXC,SXC,ISCL(20,20)
COMMON /PARM/ NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
COMMON /FLAGS/ ISTOP
INTEGER RESP,YES,YE,NEG,NO
C
C DATA YES,YE/3HYES,1HY/
C DATA NEG,NE/2HNO,1HN/
C
C NBYR=1960
C NY=1
C NYDT=28
C NBDT=17
C NYRU=NYDT*3
C NBRD=NBRD*3
C NRD=NYRU+NBRD
C NBYR=1980
C IFILE=1
C NY=1
105 CONTINUE
C
C      READ THE KEY OF VARIABLE DEFINITIONS
C
C      DO 110 I=1,NRD
C      READ (16,120) (KEYFIN(I,J),J=1,18)
110 CONTINUE
120 FORMAT (18A4)
C
C 140 FORMAT (A3)
C
C      IFILE=1
C 149 CONTINUE
C
C      BEGINNING OF INPUT SESSION
C
C      WRITE (6,210)
210 FORMAT (//,39H INPUT SESSION FOR THE FINANCIAL MODULE,
+ 17H OF CERES BEGINS./28H 3 SETS OF DATA SHOULD BE,
+ 9H ON FILE./22H OTHERWISE ENTER THEM.,
+ //37H 1 BEGINNING VALUES OF FINANCIAL DATA,
+ //24H 2 YEARLY FINANCIAL DATA,
+ //51H 3 RETIREMENT SCHEDULES OF SHORT AND LONG TERM DEBT)
C
C      WRITE (6,215)
215 FORMAT (//,40H THE DATA TO BE ENTERED ARE LISTED BELOW)
C
C      WRITE (6,220)
220 FORMAT (//,35H BEGINNING VALUES OF FINANCIAL DATA,
+ /35H _____)
C
C      DO 225 I=1,NBDT
225 CONTINUE
C
C      WRITE (6,230)
230 FORMAT (//,22H YEARLY FINANCIAL DATA,
+ /22H _____)
C
C      DO 235 I=1,NYDT
235 CONTINUE
C
C      WRITE (6,237)
237 FORMAT (//,35H RETIREMENT SCHEDULES FOR SHORT AND,
+ 15H LONG TERM DEBT,
+ /50H _____)

```

```

        WRITE (6,240)
240 FORMAT (35H 1 YEAR OF MATURITY FOR DEBT ISSUED,
+13H IN EACH YEAR,
+13H 2 PERCENT TO BE RETIRED EACH YEAR FOR,
+16H EACH DEBT ISSUE)          00003390
00003400
00003410
00003420
00003430
00003440
00003450
00003460
00003470
00003480
00003490
00003500
00003510
00003520
00003530
00003540
00003550
00003560
00003570
00003580
00003590
00003600
00003610
00003620
00003630
00003640
00003650
00003660
00003670
00003680
00003690
00003700
00003710
00003720
00003730
00003740
00003750
00003760
00003770
00003780
00003790
00003800
00003810
00003820
00003830
00003840
00003850
00003860
00003870
00003880
00003890
00003900
00003910
00003920
00003930
00003940
00003950
00003960
00003970
00003980
00003990
00004000
00004010
00004020
00004030
00004040
00004050
00004060

9000 CONTINUE
      WRITE (6,258)
258 FORMAT (//,31H DO YOU HAVE A FILE ALREADY FOR,
+1CH FINANCIAL DATA?/17H ENTER YES OR NO./2H ?)
      READ (5,140) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) IFILE=2
      IF (IFILE.EQ.2) GO TO 800
C
C      INPUT BEGINNING VALUES OF
C      FINANCIAL DATA
C
260 CONTINUE
C
      WRITE (6,350)
350 FORMAT (//,26H ENTER BEGINNING VALUES OF,
+16H FINANCIAL DATA.)
      DO 352 I=1,NBDT
      WRITE (6,434)
      WRITE (6,120) (KEYFIN(I+34,J),J=1,18)
      WRITE (6,445)
      READ (5,470) BDAT(I)
352 CONTINUE
C
C      CHANGES/CORRECTIONS OF BEGINNING VALUES
C      OF FINANCIAL DATA
C
      WRITE (6,354)
354 FORMAT (//,25H ENTER YES IF YOU WANT TO,
+27H CHANGE ANY BEGINNING DATA./2H ?)
      READ (5,140) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 358
      GO TO 360
358 CONTINUE
      WRITE (6,359)
359 FORMAT (//,22H DO YOU WANT TO CHANGE,
+24H ALL THE BEGINNING DATA?,
+17H ENTER YES OR NO./2H ?)
      READ (5,140) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 260
      CALL FBCHNG(NBDT)
360 CONTINUE
      IF (IFILE.EQ.2) GO TO 1200
      KY=NY+NBYR
381 CONTINUE
      WRITE (6,398)
398 FORMAT (//,29H ENTER YEARLY FINANCIAL DATA.)
      WRITE (6,402) KY
402 FORMAT (//,23H ENTER ALL DATA FOR THE,
+3H YEAR,I5)
      DO 423 I=1,NYDT
      WRITE (6,434)
434 FORMAT (1M //)
      WRITE (6,120) (KEYFIN(I+107,J),J=1,18)
      WRITE (6,445)
445 FORMAT (11H -----)
      READ (5,470) FINDAT(NY,I)
423 CONTINUE
470 FORMAT (F10.5)

C
C      CHANGES/CORRECTIONS OF YEARLY FINANCIAL DATA
      WRITE (6,475) KY

```

```

475 FORMAT (//,3OH ANY CHANGES IN FINANCIAL DATA,  

+13H FOR THE YEAR,15,1H?/17H ENTER YES OR NO./2H ?)  

READ (5,140) RESP  

IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 490  

480 IF (NY.EQ.MAXYR.OR.IYCH.EQ.2) GO TO 555  

NY=NY+1  

GO TO 360  

490 CONTINUE  

WRITE (6,505) KY  

505 FORMAT (//,26H DO YOU WANT TO CHANGE ALL,  

+18H THE DATA FOR YEAR,15,1H?/17H ENTER YES OR NO./2H ?)  

READ (5,140) RESP  

IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 381  

CALL FYCHNG(NY,NYDT)  

GO TO 480  

555 CONTINUE  

C CHANGES/CORRECTIONS OF YEARLY FINANCIAL DATA  

IYCH=2  

NY=1  

WRITE (6,560)  

560 FORMAT (//,33H DO YOU WANT TO CHANGE ANY YEARLY,  

+10H FINANCIAL DATA?/17H ENTER YES OR NO.)  

READ (5,140) RESP  

IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 570  

GO TO 590  

570 CONTINUE  

IYCH=2  

WRITE (6,575)  

575 FORMAT (//,32H ENTER 1 TO REINPUT ALL DATA FOR,  

+11H ALL YEARS./29H ENTER 2 TO CHANGE DATA FOR A,  

+13H SINGLE YEAR.)  

READ (5,726) IFCH  

GO TO (360,577),IFCH  

577 CONTINUE  

WRITE (6,580)  

580 FORMAT (//,36H ENTER YEAR(4 DIGITS) OF DATA CHANGE,  

+/2H ?)  

READ (5,585) KY  

585 FORMAT (14)  

NY=KY-NBYR  

GO TO 490  

590 CONTINUE  

IF (IFILE.EQ.2) GO TO 1200  

C ENTER DEBT RETIREMENT SCHEDULES  

IDEBT=1  

IDCH=1  

600 CONTINUE  

NY=1  

WRITE (6,606)  

606 FORMAT (//,36H ENTER DEBT RETIREMENT SCHEDULES FOR)  

IF (IDEBT.EQ.1) WRITE (6,608)  

IF (IDEBT.EQ.2) WRITE (6,609)  

608 FORMAT (16H SHORT TERM DEBT)  

609 FORMAT (15H LONG TERM DEBT)  

610 CONTINUE  

KY=NBYR+NY  

NY1=NY+1  

615 CONTINUE  

WRITE (6,620)  

620 FORMAT (//,38H ENTER YEAR OF MATURITY (4 DIGITS) FOR)  

IF (IDEBT.EQ.1) WRITE (6,608)  

IF (IDEBT.EQ.2) WRITE (6,609)  

WRITE (6,622) KY  

622 FORMAT (15H ISSUED IN YEAR,15/5H ----)  

READ (5,630) KYR(NY,IDEBT)

```

```

630 FORMAT (I4)
NYR=KYR(NY, IDEBT)-NBYR
DO 635 K=NY1,NYR
KY=K+NBYR
WRITE (6,640) KY
640 FORMAT (//,24H PERCENT RETIRED IN YEAR,I5,2H ?)
WRITE (6,443)
READ (5,470) FDEBR(NY,K,IDEFT)
635 CONTINUE
IF (IDCH.EQ.2) GO TO 710
IF (NY.EQ.MAYR) GO TO 700
NY=NY+1
GO TO 610
700 CONTINUE
IF (IDEFT.EQ.2) GO TO 710
IDEFT=2
GO TO 600
710 CONTINUE
WRITE (6,720)
720 FORMAT (//,33H DO YOU WANT TO CHANGE ANY OF THE,
+/24H DEBT RETIREMENT SCHEDULES?/17H ENTER YES OR NO.,
+/2H ?)
READ (5,140) RESP
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 722
GO TO 741
722 CONTINUE
WRITE (6,725)
725 FORMAT (//,30H ENTER 1 IF YOU WANT TO CHANGE,
+/38M SHORT TERM DEBT RETIREMENT SCHEDULES.,
+/30H ENTER 2 IF YOU WANT TO CHANGE,
+/37H LONG TERM DEBT RETIREMENT SCHEDULES./2H ?)
READ (5,726) IDEFT
726 FORMAT(1I)
WRITE (6,730)
730 FORMAT(//,32H ENTER YES IF YOU WANT TO CHANGE,
+/33M ALL THE RETIREMENT SCHEDULES FOR)
IF (IDEFT.EQ.1) WRITE (6,608)
IF (IDEFT.EQ.2) WRITE (6,609)
READ (5,140) RESP
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 600
IDCH=2
WRITE (6,740)
740 FORMAT (//,30H YEAR OF DEBT ISSUE (4 DIGITS)/2H ?)
READ (5,630) KY
NY=KY-NBYR
GO TO 615
741 CONTINUE
C
ENTER CONSTRUCTION EXPENDITURE FOR EXPANSION PLANTS
800 CONTINUE
IPCH=1
I=1
805 CONTINUE
WRITE (6,810) IDEXP(I)
810 FORMAT (//,44H ENTER CONSTRUCTION TIME IN YEARS (2 DIGITS),
+/33H FOR A SINGLE UNIT WITH UNIT CODE,I4/3H --)
READ (5,815) ITPL(I)
815 FORMAT( I2)
830 CONTINUE
JY=ITPL(I)
WRITE (6,845) IDEXP(I),PLC(1),JY
845 FORMAT (//,32H THE TOTAL CONSTRUCTION COST FOR,
+/6H PLANT,I3,3H IS,$8.3,17H MILLION DOLLARS.,
+/26H THE CONSTRUCTION TIME IS ,12,7H YEARS.,
+/23H ENTER CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS),
+/34H DURING EACH YEAR OF CONSTRUCTION.)
DO 850 IY=1,JY
00004750
00004760
00004770
00004780
00004790
00004800
00004810
00004820
00004830
00004840
00004850
00004860
00004870
00004880
00004890
00004900
00004910
00004920
00004930
00004940
00004950
00004960
00004970
00004980
00004990
00005000
00005010
00005020
00005030
00005040
00005050
00005060
00005070
00005080
00005090
00005100
00005110
00005120
00005130
00005140
00005150
00005160
00005170
00005180
00005190
00005200
00005210
00005220
00005230
00005240
00005250
00005260
00005270
00005280
00005290
00005300
00005310
00005320
00005330
00005340
00005350
00005360
00005370
00005380
00005390
00005400
00005410
00005420

```

```

      WRITE (6,848) IY
848 FORMAT (//,25H CONSTRUCTION EXPENDITURE,
+31H (MILLIONS OF DOLLARS) FOR YEAR,13,1H?,
+/11H ____)
      READ (5,855) CEPLX(I,IY)
855 FORMAT (F10.5)
850 CONTINUE
      WRITE (6,852) IDEXP(I)
852 FORMAT (//,36H DO YOU WANT TO REINPUT CONSTRUCTION,
+17H EXPENDITURE DATA/23H JUST ENTERED FOR UNIT,
+5H CODE,14,2H ?/17H ENTER YES OR NO./2H ?)
      READ (5,140) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 805
      IF (IPCH.EQ.2) GO TO 985
      IF (I.GE.MXPL) GO TO 857
      I=I+1
      GO TO 805
857 CONTINUE
      WRITE (6,890)
890 FORMAT (//,33H ENTER YES IF YOU WANT TO CHANGE,
+35H ANY CONSTRUCTION EXPENDITURE DATA,
+21H FOR EXPANSION UNITS./2H ?)
      READ (5,140) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 950
      GO TO (1200,1100),IFILE
950 CONTINUE
      WRITE (6,955)
955 FORMAT (//,26H DO YOU WANT TO CHANGE ALL,
+31H CONSTRUCTION EXPENDITURE DATA?,
+17H ENTER YES OR NO./2H ?)
      READ (5,140) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 800
960 CONTINUE
      WRITE (6,965)
965 FORMAT (//,34H ENTER UNIT CODE NUMBER (2 DIGITS),
+42H FOR WHICH YOU WANT TO CHANGE CONSTRUCTION,
+17H EXPENDITURE DATA)
970 CONTINUE
      WRITE (6,975) (IDEXP(I),I=1,MXPL)
975 FORMAT (/,26H ENTER ONE OF THE FOLLOWING./8I4,
+3H --)
      READ (5,990) J
      DO 977 K=1,MXPL
      IF (J.EQ.IDEXP(K)) GO TO 986
977 CONTINUE
      WRITE (6,982)
982 FORMAT (//,7H ERROR!)
      GO TO 970
986 I=K
      IPCH=2
      GO TO 805
985 CONTINUE
      WRITE (6,987)
987 FORMAT (//,36H DO YOU WANT TO REINPUT CONSTRUCTION,
+17H EXPENDITURE DATA/35H FOR ANOTHER EXPANSION UNIT TYPE?,
+17H ENTER YES OR NO./2H ?)
      READ (5,140) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 960
990 FORMAT (I2)
      GO TO (1200,1100),IFILE
1100 CONTINUE
      READ DATA FROM EXISTING DATA FILE
      IFILE=2
      READ (18,1225) (BDAT(I),I=1,NBDT)
      READ (18,1215) MXYR
      DU 1120 NY=1,MXYR
      00005430
      00005440
      00005450
      00005460
      00005470
      00005480
      00005490
      00005500
      00005510
      00005520
      00005530
      00005540
      00005550
      00005560
      00005570
      00005580
      00005590
      00005600
      00005610
      00005620
      00005630
      00005640
      00005650
      00005660
      00005670
      00005680
      00005690
      00005700
      00005710
      00005720
      00005730
      00005740
      00005750
      00005760
      00005770
      00005780
      00005790
      00005800
      00005810
      00005820
      00005830
      00005840
      00005850
      00005860
      00005870
      00005880
      00005890
      00005900
      00005910
      00005920
      00005930
      00005940
      00005950
      00005960
      00005970
      00005980
      00005990
      00006000
      00006010
      00006020
      00006030
      00006040
      00006050
      00006060
      00006070
      00006080
      00006090
      00006100

```

CC

```

READ (18,1225) (FINDAT(NY,I),I=1,NYDT)          00006110
READ (18,1225) ((FDEBR(NY,K,IDEBT),IDEBT=1,2),K=1,MXYR) 00006120
1120 CONTINUE                                     00006130
1215 FORMAT (I2)                                    00006140
1225 FORMAT (5E12.5)                                00006150
REWIND 18                                         00006160
00006170
C      CHANGES/CORRECTIONS OF DATA ALREADY
C      ON FILE.
00006180
00006190
00006200
00006210
00006220
00006230
00006240
00006250
00006260
00006270
00006280
00006290
00006300
00006310
00006320
00006330
00006340
00006350
00006360
00006370
00006380
00006390
00006400
00006410
00006420
00006430
00006440
00006450
00006460
00006470
00006480
00006490
00006500
00006510
00006520
00006530
00006540
00006550
00006560
00006570
00006580
00006590
00006600
00006610
00006620
00006630
00006640
00006650
00006660
00006670
00006680
00006690
00006700
00006710
00006720
00006730
00006740
00006750
00006760
00006770
00006780
00006791
00006790

1200 CONTINUE
WRITE (0,1250)
1250 FORMAT (//,31H ANY CHANGES IN FINANCIAL DATA?,
+/17H ENTER YES OR NO./2H ?)
READ (5,140) RESP
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 1400
WRITE (18,1225) (BDAT(I),I=1,NBDT)
WRITE (18,1215) MXYR
DO 1320 NY=1,MXYR
WRITE (18,1225) (FINDAT(NY,I),I=1,NYDT)
WRITE (18,1225) ((FDEBR(NY,K,IDEBT),IDEBT=1,2),K=1,MXYR)
1320 CONTINUE
FILE=2
REWIND 18
1260 CONTINUE
1255 CONTINUE
GO TO 1450
1400 CONTINUE
WRITE (0,1430)
1430 FORMAT (//,44H ENTER 1 TO CHANGE BEGINNING VALUES OF DATA.,
+/41H ENTER 2 TO CHANGE YEARLY FINANCIAL DATA.,
+/45H ENTER 3 TO CHANGE DEBT RETIREMENT SCHEDULES.,
+/43H ENTER 4 TO CHANGE CONSTRUCTION EXPENDITURE.,
+/25H DATA OF EXPANSION UNITS.,
+/42H ENTER 5 TO RE INPUT ALL OF THE ABOVE DATA./2H ?)
READ (5,726) ICHNG
GO TO (358,570,722,950,149),ICHNG
1450 CONTINUE
WRITE (0,1480)
1480 FORMAT (//,33H IS YOUR INPUT FILE FOR FINANCIAL,
+/15H DATA COMPLETE?/17H ENTER YES OR NO./2H ?)
READ (5,140) RESP
IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 1485
GO TO 1200
C      TRANSFER THE INPUT DATA INTO APPROPRIATE
C      ARRAYS OF FINANCIAL VARIABLES.
1485 CONTINUE
PLGSB=BDAT(1)
DEP1B=BDAT(2)
DEP2B=BDAT(3)
CWIPFB=BDAT(4)
DEBSB=BDAT(5)
DEBLB=BDAT(6)
CDEBSB=BDAT(7)
CDEBLB=BDAT(8)
DEBSRB=BDAT(9)
DEBLRB=BDAT(10)
STCOMB=BDAT(11)
STPRB=BDAT(12)
ERNRTB=BDAT(13)
AFDC1B=BDAT(14)
TXDEFB=BDAT(15)
TXCRB=BDAT(16)
AFDC2B=BDAT(17)
DD 1490 NY=1,MXYR

```

```

PLGSF(NY)=FINDAT(NY,1)          00006800
DEPF1(NY)=FINDAT(NY,2)          00006810
DEPF2(NY)=FINDAT(NY,3)          00006820
CEF(NY)=FINDAT(NY,4)            00006830
FDEBS(NY)=FINDAT(NY,5)*0.01     00006840
DSLIM(NY)=FINDAT(NY,6)          00006850
FDEBL(NY)=FINDAT(NY,7)*0.01     00006860
DLIM(NY)=FINDAT(NY,8)          00006870
FSTCGM(NY)=FINDAT(NY,9)*0.01     00006880
COMLIM(NY)=FINDAT(NY,10)        00006890
FSTPR(NY)=FINDAT(NY,11)*0.01     00006900
PRLIM(NY)=FINDAT(NY,12)        00006910
RDEBS(NY)=FINDAT(NY,13)*0.01     00006920
RDEBL(NY)=FINDAT(NY,14)*0.01     00006930
RDIVP(NY)=FINDAT(NY,15)*0.01     00006940
FDIVC(NY)=FINDAT(NY,16)*0.01     00006950
RTXFD(NY)=FINDAT(NY,17)*0.01     00006960
RTXST(NY)=FINDAT(NY,18)*0.01     00006970
RTXREV(NY)=FINDAT(NY,19)*0.01     00006980
RTXP(NY)=FINDAT(NY,20)*0.01     00006990
RTXCR(NY)=FINDAT(NY,21)*0.01     00007000
FCE(NY)=FINDAT(NY,22)*0.01     00007010
FCWIP(NY)=FINDAT(NY,23)*0.01     00007020
AFDCR(NY)=FINDAT(NY,24)*0.01     00007030
ARATE(NY)=FINDAT(NY,25)*0.01     00007040
AFDCF2(NY)=FINDAT(NY,26)        00007041
RWKCAP(NY)=FINDAT(NY,27)*0.01     00007050
RFDEBL(NY)=FINDAT(NY,28)        00007053
C .... ASSIGN SOME TEMPORARY VALUES HERE
C ****
1490 CONTINUE
DO 1745 NY=1,MXYR
NY1=NY+1
DO 1750 K=NY1,MXYR
FDEBSR(NY,K)=FDEBR(NY,K,1)*0.01
FDEBLR(NY,K)=FDEBR(NY,K,2)*0.01
1750 CONTINUE
1745 CONTINUE
C END OF INPUT PROCEDURE
C
1900 CONTINUE
WRITE(6,1930)
1930 FORMAT(//,34H END OF INPUT PROCEDURE.INPUT DATA,
+15H STORED ON UNIT 18.)
WRITE(6,2500)
2500 FORMAT(//,37H ENTER ACCOUNTING METHOD FOR TREATING,
+15H DEFERRED TAXES/26H 1 : NORMALIZED ACCOUNTING/
+28H 0 : FLOW THROUGH ACCOUNTING/2H ?)
C READ(5,726) IDEF
C
2600 FORMAT(//,35H ENTER 1 IF AFUDC SHOULD BE TREATED,
+10H AS INCOME/18H ENTER 0 OTHERWISE/2H ?)
READ(5,726) IAPC
IF (IDEF.EQ.0) TXDEFB=0
C
4000 CONTINUE
RETURN
END
C
C -----
SUBROUTINE FYCHNG(NY,NYDT)
C -----
C
C

```

THIS SUBPROGRAM IS USED TO CHANGE OR CORRECT
 FINANCIAL PARAMETERS ENTERED DURING THE CURRENT
 SESSION OR ALREADY EXISTING ON A DATA FILE.

COMMON /DEBTS/	DEBS(20), DEBL(20), DEBR(20), CDEBS(20), + CDEBL(20), RDEBS(20), KDEBL(20), DEBT(20), + CDEBT(20), DEBN(20), FDEBT(20), FDEBS(20), + FDEBL(20), FDCLR(20,20), + STCOM(20), SPRT(20), FSTPR(20), + DIVCOM(20), DIVPK(20), RDIVC(20), RDIVP(20), + FDIVC(20), FDIVP(20), RFDEBL(20), + GREV(20), OPINC(20), TXINC(20), ARATE(20), + AINC(20), BINC(20), OMCI(20), FERN(20), + ANINC(20), ERNET(20), ERNRT(20), + OMCF(20), OMCX(20), + FINDAT(20,30), BDAT(20), KEYFIN(140,18), + PLGS1(20), PLGAI(20), PLNS1(20), PLNA1(20), + PLNS2(20), ITPLX(20), LPLX(22,20), + PLX(22,20), CEPLX(22,20), + CEX(20), PLGSF(20), CEF(20), CWIPC(20), + CWIPC(20), DEPLX(22,20), DEPLX1(22,20), + DEPLX2(22,20), DEPL1(20), DEPL2(20), DEPF1(20), + DEPF2(20), DEP1(20), DEP2(20), AFDCF2(20), + AFDCR(20), AFDC1(20), AFDC2(20), AFDCX2(22,20), + CAPRO(20), RATB(20), CE(20), AFDCX(20), + CEPC(20,20), CE(20), + SRCIN(20), SRCEX(20), SRC(20), RWKCAP(20), + USE(20), CAPIN(20), SRC(TOT)(20), USE(TOT)(20), + WKCAP(20), DWKCAP(20), WKCAP(20), + RTXF0(20), TXFD(20), RTAST(20), TXST(20), + RTXP(20), TXP(20), RTXCR(20), TXCR(20), + PLGS2(20), DEP1C(20), DEP2C(20), BAL(20), + ASSET(20), EQCLUM(20), ERNCUM(20), CAPTOT(20), + TXDEF(20), UEF(20), TLIAB(20), FTXDEF(20), + OPEXP(20), TOTEXP(20), PLGA2(20), PLNV(20), + PLNUT(20), STCDMC(20), STPRC(20), ERNKTC(20), + FDEBLR(20,20), FDEBSR(20,20), TXREV(20), RTXREV(20), + CEC(20), AFDC1C(20), AFDC2C(20), FDEP(20), + PLNA2(20), DEBLT(20), DEBTL(20), DEBLR(20), + DEBSR(20), FCE(20), LCR(20), TXCRC(20), + QPCOST(20), PLGSX(20), TXDEF(20), + DLSIM(20), DLLIM(20), CQMLIM(20), + PRLLIM(20), FCWIPC(20), BINT(20), CWIPC1(20), + TXSPD(20), FUEL(20), ITPL(20), PLC(200), + IPL(200,20), PL(200,20), ELPL(200), LPL(200), + DEPL1(200,20), DEPL2(200,20), DEPL(200,20), + FSUM(20,20), EQTOT(20), DEBASS(20), + DEBEQ(20), COVRG(20), KYR(20,2), + PLGS8, DEP1B, DEP2B, CWIPFB, CWIPXB, CWIPB, + DEBS, DEBLB, DEBSRB, DEBLRB, STCOMB, STPRB, + ERNRKT, AFDC1B, TXDFB, TXCRB, WKCAPB, + AFDC2B, + COMMON /INDYN/ NF1, IPLW(200,20), ELHF, ELSF, ELHX, + ELSA, PLCC(200), MXCC, SXCC, HFCAP(20), + SFCAP(20), PLCAP(200), IDEXP(20), NEXP, + NBEST, HXC, SXC, ISOL(20,20), + COMMON /PARM/ NBYR, MXYR, MXPLF, MXPLX, MXPL, MXALL, IDEF, IAFC + COMMON /FLAGS/ 1STOP + INTEGER RESP, YES, YE, NEG, NO CCC DATA YES, YE/3HYES, 1HY/ DATA NEG, NE/2HNO, 1HN/ CCC KY=NBYR + NY CCC	00007880 00007690 00007900 00007910 00007920 00007930 00007940 00007950 00007960 00007970 00007980 00007990 00008000 00008010 00008020 00008030 00008040 00008050 00008060 00008070 00008080 00008090 00008100 00008110 00008120 00008130 00008140 00008150 00008151 00008160 00008170 00008180 00008190 00008200 00008210 00008220 00008230 00008240 00008250 00008260 00008270 00008280 00008290 00008300 00008310 00008320 00008330 00008340 00008350 00008360 00008370 00008371 00008380 00008390 00008400 00008410 00008420 00008430 00008440 00008450 00008460 00008470 00008480 00008490 00008500 00008510 00008520
----------------	---	--

```

105 FORMAT (A3) 00008530
106 CONTINUE 00008540
107 WRITE (6,110) 00008550
110 FORMAT (//,31H WHICH DATA YOU WANT TO CHANGE?, 00008560
    +/43H ENTER 2 DIGIT DATA CODE. FOR HELP,ENTER 99., 00008570
    +/3H --> 00008580
    READ (5,120) IP 00008590
120 FORMAT (I2) * 00008600
    IF (IP.EQ.99) GO TO 160 00008610
C     ENTER CHANGED/CORRECTED PARAMETER 00008620
C
125 CONTINUE 00008630
    WRITE (6,135) (KEYFIN(IP+107,J),J=1,16) 00008640
135 FORMAT (18A4) 00008650
    WRITE (6,142) FINDAT(NY,IP) 00008660
142 FORMAT (16H PREVIOUS VALUE=,F10.4) 00008670
    WRITE (6,145) 00008680
145 FORMAT (11H _____._____) 00008690
    READ (5,155) FINDAT(NY,IP) 00008700
155 FORMAT (F10.5) 00008710
    WRITE (6,159) KY 00008720
159 FORMAT (//,30H ANY MORE CHANGES IN FINANCIAL, 00008730
    +9H DATA FOR,15, 00008740
    +/17H ENTER YES OR NO./2H ?) 00008750
    READ (5,105) RESP 00008760
    IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 106 00008770
    GO TO 190 00008780
160 CONTINUE 00008790
C     PRINT 2 DIGIT DATA CODES AND 00008800
C     DEFINITONS. 00008810
C
    DO 180 IP=1,NYDT 00008820
    WRITE (6,135) (KEYFIN(IP+79,J),J=1,18) 00008830
180 CONTINUE 00008840
    GO TO 106 00008850
190 CONTINUE 00008860
C
C     RETURN 00008870
C     END 00008880
C
C----- SUBROUTINE INCOM(NY) 00008890
C----- 00008900
C----- 00008910
C----- 00008920
C----- 00008930
C----- 00008940
C----- 00008950
C----- 00008960
C----- 00008970
C----- 00008980
C----- 00008990
C----- 00009000
C----- 00009010
C----- THIS SUBPROGRAM PERFORMS ALL CALCULATIONS OF 00009020
C----- EARNINGS AND EXPENSES. THESE INCLUDE OPERATING 00009030
C----- REVENUES AND COSTS, FEDERAL, STATE AND OTHER TAXES, 00009040
C----- AND INTEREST CHARGES ON DEBT. 00009050
C----- 00009060
C----- 00009070
C----- COMMON /DEBTS/ DEBS(20),DEBL(20),UEBR(20),CDEBS(20), 00009080
C----- + CDEBL(20),RDEBS(20),RDEBL(20),DEBT(20), 00009090
C----- + CDEBT(20),DEBN(20),FDEBT(20),FDEBS(20), 00009100
C----- + FDEBL(20),FDEBR(20,20,2) 00009110
C----- COMMON /EQUITY/ STCOM(20),STPR(20),FSTCUM(20),FSTPR(20), 00009120
C----- + DIVCOM(20),DIVPR(20),RDIVC(20),RDIVP(20), 00009130
C----- + RDIVC(20),FDIVP(20),RFDEBL(20) 00009140
C----- COMMON /INCOME/ GREV(20),OPINC(20),AINC(20),ARATE(20), 00009150
C----- + AINC(20),BINC(20),OMC(20),FERN(20), 00009160
C----- + ANINC(20),ERNET(20),ERNRT(20), 00009170
C----- + OMCF(20),OMCX(20) 00009180
C----- COMMON /INPUTS/ FINDAT(20,30),BDAT(20),KEYFIN(140,18) 00009190
C----- COMMON /PLANTS/ PLGS1(20),PLGA1(20),PLNS1(20),PLNA1(20), 00009200

```

+	PLNS2(20),ITPLX(20),LPLX(20),IPLX(22,20),	00009210
+	PLX(22,20),CEPX(22,20),CEPLX(22,20),	00009220
+	CEX(20),PLGSF(20),CEF(20),CWIP(20),	00009230
+	CWIPC(20),DEPLX(22,20),DEPLX1(22,20),	00009240
+	DEPLX2(22,20),DEPX1(20),DEPX2(20),DEPF1(20),	00009250
+	DEPF2(20),DEP1(20),DEP2(20),AFDCF2(20),	00009260
+	AFUCR(20),AFDC1(20),AFDC2(20),AFCX2(22,20),	00009270
+	CAPRQ(20),RATB(20),CE(20),AFDCX2(20),	00009280
+	CEPC(20,20),CE1(20)	00009281
COMMON /FUNDS/	SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),	00009290
+	USE(20),CAPIN(20),SRCTOT(20),USETOT(20),	00009300
+	WKCAP(20),DWKCAP(20),WKCAP1(20)	00009301
COMMON /TAXES/	RTXF0(20),TXFD(20),RTXST(20),TXST(20),	00009310
+	RTXP(20),TXP(20),RTXCR(20),TXCR(20)	00009320
COMMON /AUX/	PLGS2(20),DEPIC(20),DEPZC(20),BAL(20),	00009330
+	ASSET(20),EQCOM(20),ERNCLM(20),CAPTOT(20),	00009340
+	TXDEF(20),DEFCCR(20),TLIAB(20),FTXDEF(20),	00009350
+	OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),	00009360
+	PLNUT(20),STCOMC(20),STPRC(20),ERNRTC(20),	00009370
+	FDEBLR(20,20),FDEBSR(20,20),TXREV(20),RTXREV(20),	00009380
+	CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20),	00009390
+	PLNAZ(20),DEBLT(20),DEEST(20),DEBLR(20),	00009400
+	DEBSR(20),FCET(20),LCR(20),TXCRC(20),	00009410
COMMON /AUX2/	DPCOST(20),PLGSX(20),TXDEF(20)	00009420
+	DSLIM(20),DLLIM(20),COMLIM(20),	00009430
+	PKLIM(20),FCWIP(20),BINT(20),CWIPC1(20),	00009440
+	TXSPD(20),FUEL(20),ITPL(20),PLC(200),	00009450
+	IPL(200,20),PL(200,20),ELPL(200),LPL(200),	00009460
+	DEPL1(200,20),DEPL2(200,20),DEPL(200,20),	00009470
+	FSUM(20,20),EQTOT(20),DEBASS(20),	00009480
+	DEBEG(20),COVRG(20),KYR(20,2)	00009490
COMMON /BVAL/	PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,	00009500
+	DEBSB,DEBLB,DEBSKB,DEBLKB,STCOMB,STPRB,	00009510
+	ERNRTC,AFDC1B,TXDEFB,TXCRB,WKCAPB,	00009520
+	AFDC2B	00009521
COMMON /INDYN/	NF1,IPLW(200,20),ELHF,ELSF,ELHX,	00009530
+	ELSX,PLCC(200),HXC(SXCC,HFCAP(20),	00009540
+	SFCAP(20),PLCAP(200),IDEXP(20),NEXP,	00009550
+	NBEST,HXC,SXC,ISOL(20,20)	00009560
COMMON /PARM/	NBYR,MXYN,MXPLF,MXPLX,MXALL,IDEF,IAFC	00009570
COMMON /FLAGS/	ISTOP	00009580
INTEGER RESP,YES,YE,NEG,NO		00009590
		00009600
		00009610
		00009620
		00009630
		00009640
		00009650
		00009655
		00009660
		00009664
		00009666
		00009670
		00009680
		00009680
		00009690
		00009700
		00009710
		00009720
		00009730
		00009740
		00009741
		00009742
		00009743
		00009744
		00009750
		00009760
102	CONTINUE	00009770
	OPINC(NY)=RATB(NY)*ARATE(NY)	00009771
	STCOMC(NY)=STCOMC(NY-1)+STCOM(NY)	00009771

```

STPRC(NY)=STPRC(NY-1)+STPR(NY)
TXUEFC(NY)=TXUEFC(NY-1)+TXDEF(NY)
DO 110 M=1,NY
DIVPR(NY)=DIVPR(NY)+STPR(M)*RDIVP(M)
RFD=RFD+KFDEBL(M)
FACT1=0
FACT2=0
IF (M.EQ.NY) GO TO 122
DO 120 K=M,NY1
FACT1=FACT1+FUEBSR(M,K)
120 FACT2=FACT2+FDEBLR(M,K)
PRINT 2001, FACT1,FACT2
2001 FORMAT (1X, "FACT1=", E12.5, 2X, "FACT2=", E12.5)
122 CDEBS(NY)=CDEBS(NY)+DEBS(M)*(1.-FACT1)*RDEBS(M)
CDEBL(NY)=CDEBL(NY)+DEBL(M)*(1.-FACT2)*RDEBL(M)
110 CONTINUE
CDEBL(NY)=CDEBL(NY)+(DEBLB-RFD)*CDEBLB/DEBLB
CDEBT(NY)=CDEBS(NY)+CDEBL(NY)

00009772
00009774
00009780
00009782
00009792
00009793
00009800
00009810
00009820
00009830
00009840
00009850
00009860
00009870
00009880
00009900
00009902
00009910
00009920
00009930
00009940
00009950
00009970
00009972
00009980
00009990
00010000
00010001
00010030
00010050
00010051
00010060
00010070
00010080
00010090
00010092
00010094
00010095
00010096
00010097
00010098
00010110
00010120
00010130
00010140
00010150
00010160
00010170
00010380
00010390
00010400
00010410
00010420
00010430
00010440
00010450
00010460
00010470
00010480
00010490
00010500
00010510
00010520
00010530

115 CONTINUE
BINT(NY)=OPINC(NY)+AFDC1(NY)*IAFC
ERNET(NY)=BINT(NY)-CDBT(NY)
ERNCOM(NY)=ERNET(NY)-DIVPR(NY)
IF (ERNCOM(NY).LT.0.) ERNCOM(NY)=0.0
DIVCOM(NY)=ERNCOM(NY)*FDIVC(NY)
ERNRTC(1)=ERNRT(1)+EKNRTB
IF (NY.GT.1) ERNRTC(1)=ERNRTC(NY-1)+ERNRT(NY)
TXSPD(NY)=(ERNET(NY)-AFDC1(NY)*IAFC)*(RTXST(NY)+RTXFD(NY))/(
1.-RTXST(NY)-RTXFD(NY))-(TXCR(NY))/(
1.-RTXST(NY)-RTXFD(NY))-(DEP2(NY)-DEP1(NY))*(
RTXFD(NY)+RTXST(NY))/(1.-RTXFD(NY)-RTXST(NY))
3 IF (TXSPD(NY).LT.0.) GO TO 125
TXST(NY)=TXSPD(NY)*RTXST(NY)/(RTXST(NY)+RTXFD(NY))
TXFD(NY)=TXSPD(NY)*RTXFD(NY)/(RTXST(NY)+RTXFD(NY))
BINC(NY)=ERNET(NY)-AFDC1(NY)*IAFC+TXSPL(NY)
TXINC(NY)=BINC(NY)-DEP2(NY)+DEP1(NY)
GO TO 132
125 DEFCR(NY)=-TXSPD(NY)
TXSPD(NY)=0.0
BINC(NY)=ERNET(NY)-AFDC1(NY)*IAFC
TXINC(NY)=BINC(NY)-DEP2(NY)+DEP1(NY)
132 CONTINUE
TXP(NY)=(PLGSI(NY)+AFDC2C(NY))*RTXP(NY)
OPEXP(NY)=OMC(NY)+FUEL(NY)
GREV(NY)=(OPEXP(NY)+TXSPD(NY)+TXDEF(NY)+DEP1(NY)+(
TXP(NY)+TXCR(NY)+UPINC(NY)))/(1.-RTXREV(NY))
TXREV(NY)=GREV(NY)-RTXREV(NY)
TOTEXP(NY)=OPEXP(NY)+TXSPD(NY)+TXDEF(NY)+(
TXP(NY)+TXREV(NY)+DEP1(NY))

RETURN
END

SUBROUTINE RATBAS
COMMON /DEBTS/ DEBS(20),DEBL(20),DEBR(20),CDEBS(20),
+ CDEBL(20),RDEBS(20),RDEBL(20),DEBT(20),
+ CDEBT(20),DEBN(20),FDEBT(20),FDEBS(20),
+ FDEBL(20),FDEBR(20),DEBR(20),20,21
COMMON /EQUITY/ STCOM(20),STPR(20),FSTCOM(20),FSTPR(20),
+ DIVCOM(20),DIVPR(20),RDIVC(20),RDIVP(20),
+ FDIVC(20),FDIVP(20),RFDEBL(20)

```

COMMON /INCOME/ GREV(20),OPINC(20),TXINC(20),ARATE(20),	00010540
* AINC(20),BINC(20),UMC(20),FERN(20),	00010550
* ANINC(20),ERNET(20),ERNRT(20),	00010560
* UMC(20),OMCA(20)	00010570
COMMON /INPUTS/ FINDAT(20,30),BDAT(20),KEYFIN(140,18)	00010580
COMMON /PLANTS/ PLGS1(20),PLGA1(20),PLNS1(20),PLNA1(20),	00010590
* PLNS2(20),ITPLX(20),LPLX(20),IPLEX(22,20),	00010600
* PLX(22,20),CEPX(22,20),CEPLX(22,20),	00010610
* CEX(20),PLGSF(20),CEF(20),CWIPC(20),	00010620
* CWIPC(20),DEPLA(22,20),DEPLX1(22,20),	00010630
* DEPLX2(22,20),DEPX1(20),DEPX2(20),DEPF1(20),	00010640
* DEPF2(20),DEP1(20),DEP2(20),AFDCF2(20),	00010650
* AFDCR(20),AFDC1(20),AFDC2(20),AFCX2(22,20),	00010660
* CAPRU(20),RATB(20),CE(20),AFDCX2(20),	00010670
* CEPXL(20,20),CE1(20)	00010671
COMMON /FUNDS/ SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),	00010680
* USE(20),CAPIN(20),SRCTOT(20),USETOT(20),	00010690
* WKCAP(20),DWKCAP(20),WKCAP1(20)	00010691
COMMON /TAXES/ RTXF(20),TXFD(20),RTAST(20),TXST(20),	00010700
* RTXP(20),TXP(20),RTXCR(20),TXCR(20)	00010710
COMMON /AUX/ PLGS2(20),DEP1C(20),DEP2C(20),BAL(20),	00010720
* ASSET(20),EWCOM(20),ERNCOM(20),CAPTOT(20),	00010730
* TXDEF(20),UEFCR(20),TLIAB(20),FTXDEF(20),	00010740
* OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),	00010750
* PLNUT(20),STCCMC(20),STPRC(20),ERNRTC(20),	00010760
* FDEBLR(20,20),FDEBSR(20,20),TXREVI(20),RTXREV(20),	00010770
* CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20),	00010780
* PLNA2(20),DEBLT(20),DEB5T(20),DEBLR(20),	00010790
* DEBSK(20),FCE(20),LCR(20),TXCRC(20),	00010800
* OPCOST(20),PLGSX(20),TXDEF(20)	00010810
COMMON /AUX2/ DSLIM(20),DLLIM(20),COMLIM(20),	00010820
* PRLIM(20),FCWIP(20),BINT(20),CWIPC1(20),	00010830
* TXSPD(20),FUEL(20),ITPLI(20),PLC(200),	00010840
* IPL(200,20),PLI(200,20),ELPL(200),LPL(200),	00010850
* DEPLI(200,20),DEPL2(200,20),DEPL(200,20),	00010860
* FSUM(20,20),EQTOT(20),DEBASS(20),	00010870
* DEBEQ(20),COVRG(20),KYR(20,2)	00010880
* PLGSB,DEP1B,CWIPFB,CWIPXB,CWIPB,	00010890
* DEBSB,DEBLB,DEBSR,B,DEBLRB,STCMB,STPRB,	00010900
* ERNRTB,AFDC1B,TXDEFB,TXCKB,WKCAPB,	00010910
* AFDC2B	00010911
COMMON /INDYN/ NF1,PLW(200,20),ELHF,ELSF,ELHX,	00010920
* ELSX,PLCC(200),HXCC,SXCC,HFCAP(20),	00010930
* SFCAP(20),PLCAP(200),IDEXP(20),NEXP,	00010940
* NBEST,HXC,SXC,ISUL(20,20)	00010950
COMMON /PARM/ NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC	00010960
COMMON /FLAGS/ ISTOP	00010970
INTEGER RESP,YES,YE,NEG,NO	00010980
	00010990
	00011000
	00011010
	00011020
	00011030
	00011040
	00011050
	00011060
	00011070
DO 110 NY=1,MXYR	00011080
PLGSX(NY)=0	00011090
CEX(NY)=0	00011100
CEC(NY)=0	00011102
OPCOST(NY)=OMC(NY)+FUEL(NY)	00011110
110 CONTINUE	00011120
	00011130
COMPUTE GROSS PLANT VALUE OF EXPANSION PLANTS	00011140
DO 125 NY=1,MXYR	00011150
DO 130 I=1,MXPL	00011160
	00011170

```

PLGSX(NY)=PLGSX(NY)+PLC(I)*ISOL(I,NY)          00011180
130 CONTINUE                                     00011190
125 CONTINUE                                     00011200
                                                 00011210
                                                 00011220
CALCULATE GROSS PLANT VALUE OF ALL PLANTS      00011230
DO 135 NY=1,MXYR                                00011240
135 PLGS1(NY)=PLGSF(NY)+PLGSX(NY)               00011250
                                                 00011260
CALCULATE GROSS PLANT ADDITIONS                00011270
PLGA1(1)=PLGS1(1)-PLGSB                         00011280
DO 140 NY=2,MXYR                                00011290
140 PLGA1(NY)=PLGS1(NY)-PLGS1(NY-1)             00011300
                                                 00011310
CALCULATE DEPRECIATION FOR BOOK PURPOSES       00011320
255 FORMAT (//,34H ENTER ACCORDING TO THE FOLLOWING./,
+23H 2: SUM OF YEARS DIGITS/
+20H 3: DOUBLE DECLINING)                      00011330
257 FURMAT (11)
CALL DEPR(1)
DO 260 I=1,MXPL
DO 270 NY=1,MXYR
DEPLX1(I,NY)=DEPLX(I,NY)
270 CONTINUE
260 CONTINUE                                     00011340
                                                 00011350
                                                 00011360
                                                 00011370
CALCULATE DEPRECIATION FOR TAX PURPOSES        00011380
275 FORMAT (//,32H ACCELERATED DEPRECIATION METHOD,
+18H FOR TAX PURPOSES?)                         00011390
WRITE (6,275)
READ (5,255) IDEP
CALL DEPR(IDEPE)
DO 265 I=1,MXPL
DO 277 NY=1,MXYR
DEPLX2(I,NY)=DEPLX(I,NY)
277 CONTINUE
265 CONTINUE                                     00011400
                                                 00011410
                                                 00011420
                                                 00011430
                                                 00011440
                                                 00011450
                                                 00011460
                                                 00011470
                                                 00011480
                                                 00011490
                                                 00011500
                                                 00011510
                                                 00011520
                                                 00011530
                                                 00011540
                                                 00011550
                                                 00011560
                                                 00011570
                                                 00011580
                                                 00011590
                                                 00011600
                                                 00011610
                                                 00011620
                                                 00011630
                                                 00011640
                                                 00011650
                                                 00011660
                                                 00011670
                                                 00011680
                                                 00011690
                                                 00011700
                                                 00011710
                                                 00011720
DEP1C(1)=DEP1B+DEP1(1)                         00011730
DEP2C(1)=DEP2B+DEP2(1)                         00011740
DO 315 NY=2,MAYR                                00011750
DEP1C(NY)=DEP1C(NY-1)+DEP1(NY)
DEP2C(NY)=DEP2C(NY-1)+DEP2(NY)
315 CONTINUE                                     00011760
                                                 00011770
                                                 00011780
                                                 00011790
                                                 00011800
                                                 00011810
CALCULATE NET PLANT IN SERVICE                 00011820
AND NET PLANT ADDED                            00011830
PLNS1(1)=PLGS1(1)-DEP1C(1)                      00011840
                                                 00011850

```

```

PLNA1(1)=PLNS1(1)-(PLGS8-DEP1B)          00011860
PLNS2(1)=PLGS1(1)-DEP2C(1)                00011870
PLNA2(1)=PLNS1(1)-(PLGS8-DEP2B)          00011880
DO 360 NY=2,MXYR                          00011890
PLNS1(NY)=PLGS1(NY)-DEPIC(NY)            00011900
PLNA1(NY)=PLNS1(NY)-PLNS1(NY-1)          00011910
PLNS2(NY)=PLGS1(NY)-DEP2C(NY)            00011920
PLNA2(NY)=PLNS2(NY)-PLNS2(NY-1)          00011930
360 CONTINUE                                00011940
                                                00011950
                                                00011960
                                                00011970
                                                00011980
                                                00011990
                                                00012000
                                                00012010
                                                00012011
                                                00012020
                                                00012030
                                                00012040
                                                00012050
                                                00012060
                                                00012070
                                                00012080
                                                00012090
                                                00012100
                                                00012110
                                                00012111
                                                00012112
                                                00012113
                                                00012114
                                                00012115
                                                00012116
                                                00012117
                                                00012118
                                                00012119
                                                00012120
                                                00012121
                                                00012122
                                                00012123
                                                00012125
                                                00012127
                                                00012128
                                                00012129
                                                00012131
                                                00012133
                                                00012135
                                                00012137
                                                00012139
                                                00012141
                                                00012143
                                                00012145
                                                00012147
                                                00012160
                                                00012170
                                                00012180
                                                00012190
                                                00012200
                                                00012210
                                                00012220
                                                00012230
                                                00012240
                                                00012250
                                                00012260
                                                00012270
                                                00012280
                                                00012290
                                                00012300

```

FIND CWIP FOR EXPANSION UNITS

```

DO 370 I=1,MXPL                         00011980
DO 375 M=1,MXYR                         00011990
L=ITPL(1)-M+1
IF (L.LT.1) GO TO 377
DO 376 K=1,L
CWIPXB=CWIPXB+CEPLX(I,K)*IPLX(I,M)
377 CONTINUE
DO 380 NY=1,MXYR
J=ITPL(1)-M+1+NY
IF (J.LT.1.OR.J.GT.ITPL(1)) GO TO 380
CEPX(I,NY)=CEPX(I,NY)+CEPLX(I,J)*IPLX(I,M)
380 CONTINUE
375 CONTINUE
370 CONTINUE
                                                00012000
                                                00012010
                                                00012011
                                                00012020
                                                00012030
                                                00012040
                                                00012050
                                                00012060
                                                00012070
                                                00012080
                                                00012090
                                                00012100
                                                00012110
                                                00012111
                                                00012112
                                                00012113
                                                00012114
                                                00012115
                                                00012116
                                                00012117
                                                00012118
                                                00012119
                                                00012120
                                                00012121
                                                00012122
                                                00012123
                                                00012125
                                                00012127
                                                00012128
                                                00012129
                                                00012131
                                                00012133
                                                00012135
                                                00012137
                                                00012139
                                                00012141
                                                00012143
                                                00012145
                                                00012147
                                                00012160
                                                00012170
                                                00012180
                                                00012190
                                                00012200
                                                00012210
                                                00012220
                                                00012230
                                                00012240
                                                00012250
                                                00012260
                                                00012270
                                                00012280
                                                00012290
                                                00012300

```

ADD CE FOR ALL EXPANSION PLANTS

```

DO 430 M=1,MXYR
AFDCX2(M)=0
DO 440 I=1,MXPL
AFCX2(I,M)=0
DO 450 J=1,NMAX
NY=J+M-NMAX-1
CEP=CEPXC(I,J)*IPLX(I,M)
AFCR=AFDCR(I)*(1.-FCWIP(I))
IF (NY.LE.0) AFDC1B=AFDC1B+CEP*AFCR
IF (NY.GT.0) AFR=AFDCR(NY)*(1.-FCWIP(NY))
450 AFCX2(I,M)=AFCX2(I,M)+CEP*AFCR
440 CONTINUE
430 AFDCX2(M)=AFDCX2(M)+AFCX2(I,M)
DO 490 NY=1,MXYR
DO 500 I=1,MXPL
CEX(NY)=CEX(NY)+CEPX(I,NY)
500 CONTINUE
490 CONTINUE
                                                00012143
                                                00012145
                                                00012147
                                                00012160
                                                00012170
                                                00012180
                                                00012190
                                                00012200
                                                00012210
                                                00012220
                                                00012230
                                                00012240
                                                00012250
                                                00012260
                                                00012270
                                                00012280
                                                00012290
                                                00012300

```

ADD CE FOR SCHEDULED AND EXPANSION PLANTS TO FIND TOTAL CE

```

DO 510 NY=1,MXYR
510 CEF(NY)=CEF(NY)+CEX(NY)

```

CALCULATE ACCUMULATED CONSTRUCTION EXPENDITURES

```

CWIPB=CWIPFB+CWIPXB
CEB=PLGSB+CWIPB

```

```

      CEC(1)=CEB+CE(1)
      DO 520 NY=2,MXYS
 520  CEC(NY)=CEC(NY-1)+CE(NY)

      CALCULATE ACCUMULATED AND
      INCREMENTAL CWIP
      DO 530 NY=1,MXYS
      CWIPC(NY)=CEC(NY)-PLGS1(NY)
      IF (CWIPC(NY).LT.0.0) CWIPC(NY)=0
 530  CONTINUE
      CWIP(1)=CWIPC(1)-CWIPB
      DO 540 NY=2,MXYS
      CWIP(NY)=CWIPC(NY)-CWIPC(NY-1)
 540  CONTINUE

      CALCULATE ALLOWANCE FOR FUNDS USED DURING
      CONSTRUCTION(AFUDC). AFDC1 IS ASSOCIATED WITH
      THE CWIP ACCOUNT. AFDC2 BELONGS TO THE PLANT
      IN SERVICE ACCOUNT.
      DO 550 NY=1,MXYS
      AFDC1(NY)=CWIPC(NY)*AFDCR(NY)*(1.-FCWIP(NY))
      AFDC2(NY)=AFDCF2(NY)+AFDCX2(NY)
 550  CONTINUE

      CALCULATE CUMULATIVE AFUDC
      AFDC1C(1)=AFDC1B+AFDC1(1)-AFDC2(1)
      AFDC2C(1)=AFDC2B+AFDC2(1)
      DO 560 NY=2,MXYS
      AFDC1C(NY)=AFDC1C(NY-1)+AFDC1(NY)-AFDC2(NY)
      AFDC2C(NY)=AFDC2C(NY-1)+AFDC2(NY)
 560  CONTINUE

      CALCULATE THE RATEBASE
      DO 600 NY=1,MXYS
      RATB(NY)=PLNS1(NY)+AFDC2C(NY)+  

      1     FCWIP(NY)*(CWIPC(NY)+AFDC1C(NY))  

      TXCR(NY)=CE(NY)*FCE(NY)*RTXCR(NY)  

      TXDEF(NY)=(DEP2(NY)-DEP1(NY))*  

      1     (RTXST(NY)+RTXF(D(NY))*IDEF
 600  CONTINUE

      RETURN
      END



---


      SUBROUTINE DEPR(IDEF)


---


      THIS SUBPROGRAM CALCULATES THE TOTAL DEPRECIATION
      OF ALL PLANTS BASED ON A SPECIFIED METHOD OF
      DEPRECIATION
      COMMON /DEBTS/ DEBS(20),DEBL(20),DEBR(20),CDEBS(20),
      +           CDEBL(20),RDEBS(20),RDEBL(20),DEBT(20),
      +           CDEBT(20),DEBN(20),FDEBT(20),FUEBS(20),
      +           FDEBL(20),FDEBR(20,20,2)
      COMMON /EQUITY/ STCOM(20),STPR(20),FSTCOM(20),FSTPR(20),
      +           DIVCOM(20),DIVPR(20),RDIVC(20),RDIVP(20),
      +           FDIVC(20),FDIVP(20),RFDEBL(20)
      COMMON /INCOME/ GREV(20),UPINC(20),TXINC(20),ARATE(20),
      +           AINC(20),BINC(20),UMC(20),FERN(20),

```

```

*+
*+ COMMON /INPUTS/ ANINC(20),ERNET(20),ERNRT(20),
*+ COMMON /PLANTS/ MCDF(20),UMCX(20) 00013670
*+ FINDAT(20,30),BDAT(20),KEYFIN(140,18) 00013680
*+ PLGS1(20),PLGAI(20),PLNS1(20),PLNA1(20), 00013700
*+ PLNS2(20),ITPLX(20),PLX(20),IPLX(22,20), 00013710
*+ PLX(22,20),CEPX(22,20),CEPLX(22,20), 00013720
*+ CEX(20),PLGSF(20),CEF(20),CWIP(20), 00013730
*+ CWIPC(20),DEPLX(22,20),DEPLX1(22,20), 00013740
*+ DEPLX2(22,20),DEPA1(20),DEPX2(20),DEPF1(20), 00013750
*+ DEPF2(20),UEP1(20),DEP2(20),AFDC2(20), 00013760
*+ AFDCR(20),AFDC1(20),AFDC2(20),AFCX2(22,20), 00013770
*+ CAPRQ(20),RATB(20),CE(20),AFDCX(20), 00013780
*+ CEPXC(20,20),CE1(20) 00013781
*+ SRCIN(20),SRGEX(20),SRC(20),RWKCAP(20), 00013790
*+ USEI(20),CAPIN(20),SRCTUT(20),USETOT(20), 00013800
*+ WKCAP(20),DWKCAP(20),WKCAP1(20) 00013801
*+ RTXFD(20),TXFD(20),RTXST(20),TXST(20), 00013810
*+ RTXP(20),TXP(20),RTXCR(20),TXCR(20), 00013820
*+ PLGS2(20),DEP1C(20),DEP2C(20),BAL(20), 00013830
*+ ASSET(20),EUCOM(20),ERNCOM(20),CAPTOT(20), 00013840
*+ TXDEF(20),DEFCR(20),TL1AB(20),FTXDEF(20), 00013850
*+ OPEXP(20),TUTEEXP(20),PLGA2(20),PLNV(20), 00013860
*+ PLNUT(20),STCOMC(20),STPRC(20),ERNRTC(20), 00013870
*+ FDEBLR(20,20),FDEBSK(20,20),TXREV(20),RTXREV(20), 00013880
*+ CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20), 00013890
*+ PLNA2(20),DEBLT(20),DEBST(20),DEBLR(20), 00013900
*+ DEBSR(20),FCE(20),LCR(20),TXCRC(20), 00013910
*+ OPCOST(20),PLGSX(20),TXDEF(20) 00013920
*+ DSLIM(20),DLLIM(20),COMLIM(20), 00013930
*+ PRLLIM(20),FCWIP(20),BINT(20),CWIPC1(20), 00013940
*+ TXSPD(20),FUEL(20),ITPL(20),PLC(200), 00013950
*+ IPL(200,20),PL(200,20),ELPL(200),LPL(200), 00013960
*+ DEPL1(200,20),DEPL2(200,20),DEPL(200,20), 00013970
*+ FSUM(20,20),EQTOT(20),DEBASS(20), 00013980
*+ DEBEQ(20),CUVRG(20),KXR(20,2) 00013990
*+ PLGS8,DEPIB,DEP2B,CWIPFB,CWIPXB,CWIPB, 00014000
*+ DEBSB,DEBLB,DEBSRB,DEBLRB,STCOMB,STPRB, 00014010
*+ ERNRTB,AFDC1B,TXDEFB,TXCRB,WKCAPB, 00014020
*+ AFDC2B 00014021
*+ COMMON /INDYN/ NF1,1PLW(200,20),ELHF,ELSF,ELHX, 00014030
*+ ELSX,PLCC(200),MXCC,SXCC,MFCAP(20), 00014040
*+ SFCA(20),PLLAP(200),IUEXP(20),NEXP, 00014050
*+ NBEST,HXC,SXC,ISOL(20,20) 00014060
*+ NBRY,MXYK,MXPLF,MAPLX,MXPL,MXALL,IDEF,IAFC 00014070
*+ COMMON /PARM/ 00014080
*+ COMMON /FLAGS/ 00014090
*+ ISTOP 00014100
*+ DIMENSION LSUM(200),BPL(200,20) 00014110
*+ INTEGER RESP,YES,YE,NEG,NO 00014120
*+
*+ DATA YES,YE/3HYES,1HY/ 00014130
*+ DATA NEG,NE/2HNO,1HN/ 00014140
*+
*+ MXPL1=MXPLX+1 00014150
*+ DO 80 NY=1,MXYR 00014160
*+ DO 85 I=1,MXPL1 00014170
*+ DEPLX(I,NY)=0 00014180
*+ 85 CONTINUE 00014190
*+ 80 CONTINUE 00014200
*+ GO TO (90,135,185),IDEP 00014210
*+
*+ 90 CONTINUE 00014220
*+ CALCULATE DEPRECIATION BY USING THE 00014230
*+ LINEAR METHOD 00014290
*+
*+ DO 110 I=1,MXPL 00014300
*+ FDEP(I)=1./FLGAT(LPL(I)) 00014320
*+ 00014330
*+ 00014340
*+ 00014350
*+ 00014360

```

```

DO 120 NY=1,MXYR          00014370
DO 125 M=1,NY             00014380
DEPLX(I,NY)=DEPLX(I,NY)+PLX(I,M)*FDEP(I)
125 CONTINUE                00014390
120 CONTINUE                00014400
110 CONTINUE                00014410
GO TO 999                  00014420
00014430
00014440
00014450
00014460
00014470
00014480
00014490
00014500
00014510
00014520
00014530
00014540
00014550
00014560
00014570
00014580
00014590
00014600
00014610
00014620
00014630
00014640
00014650
00014660
00014670
00014680
00014690
00014700
00014710
00014720
00014730
00014740
00014750
00014760
00014770
00014780
00014790
00014800
00014810
00014820
00014830
00014840
00014850
00014860
00014870
00014880
00014890
00014900
00014910
00014920
00014930
00014940
00014950
00014960
00014970
00014980
00014990
00015000
00015010
00015020
00015030
00015040
00015050
00015060
00015070
00015080
00015090

CALCULATE DEPRECIATION BY USING THE
SUM OF YEARS DIGITS METHOD

135 CONTINUE
DO 140 I=1,MXPL
LSUM(I)=0
LP=LPL(I)
DO 150 LY=1,LP
LSUM(I)=LSUM(I)+LY
150 CONTINUE
140 CONTINUE

DO 160 I=1,MXPL
DO 170 NY=1,MXYR
DO 175 M=1,NY
J=NY-M
FSUM(I,J)=FLOAT(LPL(I)-J)/FLOAT(LSUM(I))
DEPLX(I,NY)=DEPLX(I,NY)+PLX(I,M)*FSUM(I,J)
175 CONTINUE
170 CONTINUE
160 CONTINUE
GO TO 999

CALCULATE DEPRECIATION BY USING THE
DOUBLE DECLINING METHOD

185 CONTINUE
DO 190 I=1,MXPL
FDEP(I)=2./FLOAT(LPL(I))
DEPLX(I,I)=PLX(I,I)*FDEP(I)
BPL(I,I)=PLX(I,I)-DEPLX(I,I)
DO 200 NY=2,MXYR
DO 205 M=1,NY
IF (M.EQ.NY) GO TO 206
BPL(M,NY)=BPL(M,NY-1)*(1.-FDEP(I))
DEPLX(I,NY)=DEPLX(I,NY)+(BPL(M,NY-1)-BPL(M,NY))
GO TO 205
206 BPL(M,NY)=PLX(I,M)*(1.-FDEP(I))
DEPLX(I,NY)=DEPLX(I,NY)+PLX(I,M)*FDEP(I)
205 CONTINUE
200 CONTINUE
190 CONTINUE

999 CONTINUE
RETURN
END

SUBROUTINE CAPTL
-----
```

THIS SUBPROGRAM CALCULATES THE CAPITAL REQUIREMENTS
AND DETERMINES THE CAPITAL BUDGET FROM THE INPUT
CAPITAL STRUCTURE FOR EACH YEAR

COMMON /DEB TS/	DEBS(20),DEBL(20),DEBR(20),CDEBS(20), CDEBL(20),RDEBS(20),RDEBL(20),DEBT(20), FDEBT(20),DEBN(20),FDEBT(20),FDEBS(20), FDEBL(20),FDEBR(20,20,2)	00015100 00015110 00015120 00015130 00015140 00015150 00015160 00015170 00015180 00015190 00015200 00015210 00015220 00015230 00015240 00015250 00015260 00015270 00015280 00015290 00015300 00015310 00015320 00015321 00015330 00015340 00015350 00015360 00015370 00015380 00015390 00015400 00015410 00015420 00015430 00015440 00015450 00015460 00015470 00015480 00015490 00015500 00015510 00015520 00015530 00015540 00015541 00015550 00015560 00015570 00015580 00015590 00015600 00015610 00015620 00015630 00015640 00015650 00015660 00015670 00015680 00015690 00015700 00015710 00015711 00015712 00015713
COMMON /EQUITY/	STCOM(20),STPR(20),FSTCOM(20),FSTPR(20), DIVCOM(20),DIVPR(20),RDIVC(20),RDIVP(20), FDIVC(20),FDIVP(20),FDEBL(20)	
COMMON /INCOME/	GREV(20),UPINC(20),TXINC(20),ARATE(20), AINC(20),BINC(20),OMC(20),FERN(20), ANINC(20),ERNET(20),ERNRT(20), OMCF(20),UMCX(20)	
COMMON /INPUTS/	FINDAT(20,30),BUAT(20),KEYFIN(140,18)	
COMMON /PLANTS/	PLGS1(20),PLGA1(20),PLNS1(20),PLNA1(20), PLN2(20),ITPLX(20),PLPX(20),IPLEX(22,20), PLX(22,20),CEPX(22,20),CEPLX(22,20), CEX(20),PLGSF(20),CEF(20),CWIP(20), CWIPC(20),DEPLX(22,20),DEPLX1(22,20), DEPLX2(22,20),DEPX1(20),DEPX2(20),DEPF1(20), DEPF2(20),DEP1(20),DEP2(20),AFDCF2(20), AFDCR(20),AFDC1(20),AFDC2(20),AFCX2(22,20), CAPRQ(20),KATB(20),CE(20),AFDCX2(20), CEPXC(20,20),CE1(20)	
COMMON /FUNDS/	SRCIN(20),SRCEX(20),SRC(20),RKWCAP(20), USE(20),LAPIN(20),SRCTOT(20),USETOT(20), WKCAP(20),DWKCAP(20),WKCAP1(20)	
COMMON /TAXES/	KTFD(20),TXFD(20),RTAST(20),TXST(20), RTXP(20),TXP(20),RTXCR(20),TXCR(20)	
COMMON /AUX/	PLGS2(20),DEPIC(20),DEP2C(20),BAL(20), ASSET(20),EQCOM(20),ERNCOM(20),CAPTUT(20), TXDEF(20),DEFCR(20),TLIAB(20),PTXDEF(20), OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20), PLNUT(20),STCLUMC(20),STPRC(20),ERNRTC(20), FDEBLR(20,20),FDEBSR(20,20),TXREV(20),RTXREV(20), CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20), PLNAZ(20),DEBLT(20),DEBSC(20),DEBLR(20), DEBSR(20),FCE(20),LCR(20),TXCRC(20), OPCOST(20),PLGSX(20),TXDEF(20), DSLIM(20),DLLIM(20),CMLIM(20), PRLIM(20),PCWIP(20),BINT(20),CWIPC1(20), TXSPD(20),FUEL(20),IPL(20),PLC(200), IPL(200,20),PL(200,20),ELPL(200),LPL(200), DEPL1(200,20),DEPL2(200,20),DEPL(200,20), FSUM(20,20),EQTOT(20),DEBASS(20), DEBQ(20),COVRG(20),KYR(20,2)	
COMMON /AUX2/	PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB, DEBSB,DEBLB,DEBSR,B,DEBLR,B,STCOMB,STPRB, ERNRTB,AFDC1B,TXDEFB,TXCRB,WKCAPB, AFDC2B	
COMMON /BVAL/	NFL,IPLW(200,20),ELHF,ELSF,ELHX, ELSX,PLCC(200),MXCC,SXCC,HFCAP(20), SFCAP(20),PLCAP(200),IDEXP(20),NEXP, NBEST,MHXC,SXC,ISOL(20,20)	
COMMON /INDYN/	NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC	
COMMON /PARM/	1STOP	
COMMON /FLAGS/	INTEGER RESP,YES,YE,NEG,NO	
DATA YES,YE/3HYES,1HY/		
DATA NEG,NE/ZHNU,1HN/		
FIND CAPITAL REQUIREMENTS AND CAPITAL BUDGET FOR THE BASE YEAR		
A1=STCOMB+ERNRTB+STPRB+DEBLB+DEBSB+TXDEFB+TXCRB	00015710	
A2=PLGSB-DEP1B+CWIPB+AFDC1B+AFDC2B	00015711	
WKCAB=A1-A2	00015712	
	00015713	

```

NY=1          00015720
KY=NBYR+NY   00015730
LY=1          00015740
DEBSR(1)=DEBSRB 00015750
DEBLR(1)=RFDEBL(1) 00015760
DEBR(1)=DEBLR(1)+DEBSR(1) 00015770
WKCAP(1)=GREVB*RWKCAP(1) 00015772
CAPRQ(1)=CE(1)+AFDC1(1)- 00015774
    DEP1(1))-TXDEF(1) 00015776
    IF (CAPRQ(1).LT.0.) CAPRQ(1)=0.0 00015779
    DEBS(1)=CAPRQ(1)*FDEBS(1)+DEBSR(1) 00015800
80 IF (DEBS(1).GT.DSLIM(1)) GO TO 201 00015801
    DEBL(1)=CAPRQ(1)*FDEBL(1)+DEBLR(1) 00015820
85 IF (DEBL(1).GT.DLIM(1)) GO TO 211 00015821
    DEBN(1)=DEBS(1)+DEBL(1) 00015830
    STCOM(1)=CAPRQ(1)*FSTCOM(1) 00015840
90 IF (STCOM(1).GT.COMLIM(1)) GO TO 221 00015850
    STPR(1)=CAPRQ(1)*FSTPR(1) 00015860
95 IF (STPR(1).GT.PRLIM(1)) GO TO 231 00015870
    DEBTS(1)=DEBSB-DEBSR(1)+DEBS(1) 00015880
    DEBLT(1)=DEBLS-DEBLR(1)+DEBL(1) 00015890
    DEBT(1)=DEBTS(1)+DEBLT(1) 00015900
    CALL INCOM(NY) 00015910
    CALL FUND(NY) 00015912
    WKCAP(1)=WKCAPB+CAPIN(1) 00015914
    CALL BALANC(NY) 00015916
    IF (ISTUP.EQ.1) GO TO 400 00015920
CALCULATE CAPITAL BUDGET FOR OTHER 00015930
YEARS IN THE STUDY PERIOD 00015940
00015950
00015960
110 CONTINUE 00015970
NY=NY+1 00015980
KY=NY+NBYR 00015990
NY1=NY-1 00016000
RFD1=0 00016002
RFD2=0 00016004
DO 120 M=1,NY1 00016010
    RFD1=RFD1+FDEBS(M)*DEBSR(M,NY) 00016020
120 RFD2=RFD2+DEBL(M)*DEBLR(M,NY) 00016030
    DEBSR(NY)=RFD1 00016032
    DEBLR(NY)=RFD2+RFDEBL(NY) 00016034
    DEBR(NY)=DEBSR(NY)+DEBLR(NY) 00016040
    WKCAP1(NY)=GREV(NY-1)*RWKCAP(NY) 00016042
    CAPRQ(NY)=CE(NY)+AFDC1(NY)- 00016044
    DEP1(NY))-TXDEF(NY)+ 00016064
    (WKCAP1(NY)-WKCAP(NY-1)) 00016066
    IF (CAPRQ(NY).LT.0.) CAPRQ(NY)=0.0 00016068
    DEBS(NY)=CAPRQ(NY)*FDEBS(NY)+DEBSR(NY) 00016070
130 IF (DEBS(NY).GT.DSLIM(NY)) GO TO 201 00016080
    DEBL(NY)=CAPRQ(NY)*FDEBL(NY)+DEBLR(NY) 00016100
135 IF (DEBL(NY).GT.DLIM(NY)) GO TO 211 00016101
    DEBN(NY)=DEBS(NY)+DEBL(NY) 00016110
    DEBTS(NY)=DEBTS(NY-1)+DEBS(NY)-DEBSR(NY) 00016120
    DEBLT(NY)=DEBLT(NY-1)+DEBL(NY)-DEBLR(NY) 00016130
    DEBT(NY)=DEBTS(NY)+DEBLT(NY) 00016140
    STCOM(NY)=CAPRQ(NY)*FSTCOM(NY) 00016150
140 IF (STCOM(NY).GT.COMLIM(NY)) GO TO 221 00016160
    STPR(NY)=CAPRQ(NY)*FSTPR(NY) 00016170
145 IF (STPR(NY).GT.PRLIM(NY)) GO TO 231 00016180
    CALL INCOM(NY) 00016190
    CALL FUND(NY) 00016192
    WKCAP(NY)=WKCAP(NY-1)+CAPIN(NY) 00016194
    CALL BALANC(NY) 00016196
    IF (ISTUP.EQ.1) GO TO 400 00016200
    IF (NY.EQ.MAYK) GO TO 400 00016210
    GO TO 110 00016220
00016230

```

IF ANY OF THE DEBT AND EQUITY ISSUES EXCEEDS
 SPECIFIED UPPER LIMITS, RESPECIFY THESE LIMITS
 OR STOP THE PROGRAM.

```

201 CONTINUE
  WRITE (6,203) KY
203 FORMAT (//,26H SHORT TERM DEBT FOR YEAR ,I4)
  WRITE (6,205) DEBS(NY),DSLIM(NY)
205 FORMAT (3H 15,F10.2,17M MILLION DOLLARS./14H EXCEEDS UPPER,
+4H LIMIT OF ,F10.2,17M MILLION DOLLARS./
+36H REENTER LIMIT (IN MILLION DOLLARS)./
+31H TO STOP THE PROGRAM,ENTER -1.0//11H ----)
  READ (5,300) DSLIM(NY)
  IF (DSLIM(NY).EQ.-1.0) GO TO 350
  IF (NY-1) 80,80,130
211 CONTINUE
  WRITE (6,213) KY
213 FORMAT (//,25H LONG TERM DEBT FOR YEAR ,I4)
  WRITE (6,205) DEBL(NY),DLLIM(NY)
  READ (5,300) DLLIM(NY)
  IF (DLLIM(NY).EQ.-1.0) GO TO 350
  IF (NY-1) 85,85,135
221 CONTINUE
  WRITE (6,223) KY
223 FORMAT (//,23H COMMON STOCK FOR YEAR ,I4)
  WRITE (6,205) STCOM(NY),COMLIM(NY)
  READ (5,300) COMLIM(NY)
  IF (COMLIM(NY).EQ.-1.0) GO TO 350
  IF (NY-1) 90,90,140
231 CONTINUE
  WRITE (6,233) KY
233 FORMAT (//,26H PREFERRED STOCK FOR YEAR ,I4)
  WRITE (6,205) STPR(NY),PRIM(NY)
  READ (5,300) PRIM(NY)
  IF (PRIM(NY).EQ.-1.0) GO TO 350
  IF (NY-1) 95,95,145
300 FORMAT (F10.5)
  GO TO 400
350 ISTOP=1
400 CONTINUE
  RETURN
END
  
```

SUBROUTINE REPORT

THIS SUBPROGRAM PRINTS THE BALANCE SHEET,
 THE INCOME STATEMENT AND THE
 SOURCES AND USE OF FUNDS STATEMENT.

```

COMMON /DEBTS/ DEBS(20),DEBL(20),DEBR(20),CDEBS(20),
+ CDEBL(20),RDEBS(20),RDEBL(20),DEBT(20),
+ CDEBT(20),DEBN(20),FDEBT(20),FDEBS(20),
+ FDEBL(20),FDEBR(20,20,2)
COMMON /EQUITY/ STCOM(20),STPR(20),FSTCOM(20),FSTPR(20),
+ DIVCOM(20),DIVPK(20),RDIVC(20),RDIVP(20),
+ FDIVC(20),FDIVP(20),RFDEBL(20)
COMMON /INCOME/ GREV(20),OPINC(20),TXINC(20),ARATE(20),
+ AINC(20),BINC(20),UMC(20),FERN(20),
+ ANINC(20),ERNET(20),ERNRT(20),
+ OMCF(20),UMCX(20)
COMMON /INPUTS/ FINDAT(20,30),BUAT(20),KEYFIN(140,18)
COMMON /PLANTS/ PLGS1(20),PLGAL(20),PLNSI(20),PLNAI(20),
  
```

```

+ PLNS2(20),ITPLX(20),LPLX(20),IPLX(22,20), 00017450
+ PLX(22,20),CEPX(22,20),CEPLX(22,20), 00017460
+ CEX(20),PLGSE(20),CEF(20),CWIP(20), 00017470
+ CWIPC(20),DEPLX(22,20),DEPLX1(22,20), 00017480
+ DEPLX(22,20),DEPX1(20),DEPX2(20),DEPF1(20), 00017490
+ DEPF2(20),DEP1(20),DEP2(20),AFDCF2(20), 00017500
+ AFDCR(20),AFDC1(20),AFDC2(20),AFCX2(22,20), 00017510
+ CAPRQ(20),RATB(20),CE(20),AFDCX2(20), 00017520
+ CEPC(20,20),CE1(20) 00017521
+ SRCIN(20),SKCSEX(20),SRC(20),RWKCAP(20), 00017530
+ USE(20),CAPIN(20),SHCTOT(20),USETOT(20), 00017540
+ WKCAP(20),DWKCAP(20),WKCAP1(20) 00017541
+ RTXFU(20),TXFD(20),RTXST(20),TXST(20), 00017550
+ RTXP(20),TXP(20),RTXCR(20),TXCR(20) 00017560
+ PLGS2(20),DEP1C(20),DEP2C(20),BAL(20), 00017570
+ ASSET(20),EQCLM(20),ERNCCM(20),CAPTOT(20), 00017580
+ TXDEF(20),DEFCCR(20),TLIAB(20),FTXDEF(20), 00017590
+ OPEXP(20),TUTEXP(20),PLGAZ(20),PLNV(20), 00017600
+ PLNUT(20),STCCUM(20),STPRC(20),ERNRTC(20), 00017610
+ FDEBLR(20,20),FDEBSR(20,20),TXREV(20),RTXREV(20), 00017620
+ CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20), 00017630
+ PLNAZ(20),DEBLT(20),DEBST(20),DEBLR(20), 00017640
+ DEBSR(20),FCE(20),LCR(20),TXCRC(20), 00017650
+ OPCOST(20),PLGSX(20),TXDEF(20) 00017660
+ DSLIM(20),DLL1M(20),CDML1M(20), 00017670
+ PRIM(20),FCWIP(20),BINT(20),CWIPC1(20), 00017680
+ TXSPD(20),FUEL(20),ITPL(20),PLC(200), 00017690
+ IPL(200,20),PL(200,20),ELPL(200),LPL(200), 00017700
+ DEPL1(200,20),DEPL2(200,20),DEPL(200,20), 00017710
+ FSUM(20,20),EZTOT(20),DEBASS(20), 00017720
+ DEBEQ(20),COVRG(20),SKYR(20,2) 00017730
+ PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB, 00017740
+ DEBSB,DEBLSB,DEBSRB,DEBLRB,STCUMB,STPRB, 00017750
+ ERNRIB,AFDC1B,TXDEFB,TXCRB,WKCAPB, 00017760
+ AFDC2B 00017761
+ COMMON /INDYN/ NF1,IPLW(200,20),ELHF,ELSF,ELHX, 00017770
+ ELSX,PLCC(200),HXCC,SXCC,HFCAP(20), 00017780
+ SFCAP(20),PLCAP(200),TUEXP(20),NEXP, 00017790
+ NBEST,HXL,SXL,ISOL(20,20) 00017800
+ COMMON /PARM/ NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC 00017810
+ COMMON /FLAGS/ 1STOP 00017820
+ DIMENSION NYR(20),HYPH(20) 00017830
+ INTEGER RESP,YES,YE,NEG,NO 00017840
+ 00017850
+ 00017860
+ DATA YES,YE/3HYES,1HY/ 00017870
+ DATA NEG,NE/2HN0,1HN/ 00017880
+ DATA DASH/4H—/ 00017890
+ MAX=8 00017900
+ N1=1 00017910
+ DO 110 NY=1,MXYR 00017920
+ HYPH(NY)=DASH 00017930
+ 110 NYR(NY)=NBYR+NY 00017940
+ 00017950
+ 00017960
+ 00017970
+ 00017980
+ 00017990
+ 00018000
+ 112 CONTINUE
+ WRITE (6,99)
+ WRITE (6,115) 00018010
+ 115 FORMAT (//,40X,21H XYZ ELECTRIC COMPANY) 00018020
+ WRITE (6,125) 00018030
+ 125 FORMAT (40X,14H BALANCE SHEET) 00018040
+ WRITE (6,145) 00018050
+ 145 FORMAT (//,40X,29H MILLIONS OF CONSTANT DOLLARS) 00018060
+ WRITE (6,155) (NYR(I),I=N1,MAX) 00018070
+ 00018080
+ 00018090

```

```

155 FORMAT (//,9X,12H DECEMBER 31,14X,8I10)          00018100
WRITE (6,165) (HYPH(1),I=N1,MAX)                   00018110
165 FORMAT (35X,8(6X,A4))                         00018120
WRITE (6,170)                                     00018130
170 FORMAT (//,7H ASSETS)                         00018140
WRITE (6,178) (PLGS2(NY),NY=N1,MAX)               00018150
178 FORMAT (//,26H GR PLANT INS (INCL AFUDC),9X,8(1X,F9.2)) 00018160
WRITE (6,180) (DEPIC(NY),NY=N1,MAX)               00018170
180 FORMAT (21H - ACCUM DEPRECIATION,14X,8(1X,F9.2)) 00018180
WRITE (6,182) (PLNV(NY),NY=N1,MAX)               00018190
182 FORMAT (16H NET PLANT VALUE,19X,8(1X,F9.2))   00018200
WRITE (6,185) (CWIPC1(NY),NY=N1,MAX)             00018210
185 FORMAT (6H +CWIP,29X,8(1X,F9.2))            00018220
WRITE (6,195) (PLNUT(NY),NY=N1,MAX)              00018230
195 FORMAT (18H NET UTILITY PLANT,17X,8(1X,F9.2)) 00018240
WRITE (6,190) (WKCAP(NY),NY=N1,MAX)              00018250
190 FORMAT (20H NET WORKING CAPITAL,15X,8(1X,F9.2)) 00018260
WRITE (6,200) (ASSET(NY),NY=N1,MAX)              00018270
200 FORMAT (//,13H TOTAL ASSETS,22X,8(1X,F9.2))   00018280
WRITE (6,210)                                     00018290
210 FORMAT (//,12H LIABILITIES)                  00018300
WRITE (6,220) (STCOCM(NY),NY=N1,MAX)             00018310
220 FORMAT (//,13H COMMON STOCK,22X,8(1X,F9.2))   00018320
WRITE (6,230) (TENKTC(NY),NY=N1,MAX)             00018330
230 FORMAT (20H + RETAINED EARNINGS,15X,8(1X,F9.2)) 00018340
WRITE (6,240) (EQCOM(NY),NY=N1,MAX)              00018350
240 FORMAT (20H TOTAL COMMON EQUITY,15X,8(1X,F9.2)) 00018360
WRITE (6,250) (STPRC(NY),NY=N1,MAX)              00018370
250 FORMAT (18H + PREFERRED STOCK,17X,8(1X,F9.2)) 00018380
WRITE (6,260) (DEBLT(NY),NY=N1,MAX)              00018390
260 FORMAT (17H + LONG TERM DEBT,15X,8(1X,F9.2))  00018400
WRITE (6,270) (CAPTOT(NY),NY=N1,MAX)             00018410
270 FORMAT (14H TOTAL CAPITAL,21X,8(1X,F9.2))    00018420
WRITE (6,280) (DEBHT(NY),NY=N1,MAX)              00018430
280 FORMAT (18H + SHORT TERM DEBT,17X,8(1X,F9.2)) 00018440
WRITE (6,290) (TXDEFCC(NY),NY=N1,MAX)            00018450
290 FORMAT (24H + DEFERRED INCOME TAXES,11X,8(1X,F9.2)) 00018460
WRITE (6,295) (TXCRC(NY),NY=N1,MAX)              00018470
295 FORMAT (26H + DEFERRED INV TAX CREDIT,9X,8(1X,F9.2)) 00018480
WRITE (6,300) (TLIAS(NY),NY=N1,MAX)              00018490
300 FORMAT (//,18H TOTAL LIABILITIES,17X,8(1X,F9.2)) 00018500
IF (MAX.EQ.MAXYR) GO TO 302
N1=MAX+1
MAX=MAXYR
GO TO 112
302 N1=1
MAX=8
305 CONTINUE

PRINT THE INCOME STATEMENT

99 FORMAT (1HE)
WRITE (6,110)
WRITE (6,310)
310 FORMAT (40X,17H INCOME STATEMENT)
WRITE (6,145)
WRITE (6,155) (NYH(1),I=N1,MAX)                00018680
WRITE (6,165) (HYPH(1),I=N1,MAX)                00018690
WRITE (6,170) (GKEV(NY),NY=N1,MAX)              00018700
320 FORMAT (//,19H OPERATING REVENUES,16X,8(1X,F9.2)) 00018710
WRITE (6,320) (OMC(NY),NY=N1,MAX)              00018720
330 FORMAT (23H - OPER AND MAINT COSTS,12X,8(1X,F9.2)) 00018730
WRITE (6,335) (FUEL(NY),NY=N1,MAX)              00018740
335 FORMAT (13H - FUEL COSTS,22X,8(1X,F9.2))    00018750
WRITE (6,350) (OPEXP(NY),NY=N1,MAX)             00018760
                                         00018790

```

```

350 FORMAT (//,25H TOTAL OPERATING EXPENSES,10X,8(1X,F9.2))      00018800
    WRITE (6,340) (DEPL(NY),NY=N1,MAX)                                00018805
340 FORMAT (//,15H - DEPRECIATION,20X,8(1X,F9.2))                  00018810
    WRITE (6,360) (TXREV(NY),NY=N1,MAX)                                00018811
360 FORMAT (15H - OPER REV TAX,20X,8(1X,F9.2))                   00018820
    WRITE (6,363) (TXP(NY),NY=N1,MAX)                                00018830
363 FORMAT (15H - PROPERTY TAX,20X,8(1X,F9.2))                   00018840
    WRITE (6,370) (TXSPD(NY),NY=N1,MAX)                                00018850
370 FORMAT (15H - INCOME TAX PAID,17X,8(1X,F9.2))                 00018860
    WRITE (6,373) (TXDEF(NY),NY=N1,MAX)                                00018870
373 FORMAT (22H - DEFERRED INCOME TAX,13X,8(1X,F9.2))              00018880
    WRITE (6,377) (TXCR(NY),NY=N1,MAX)                                00018890
377 FORMAT (22H - DEFERRED INV TAX CR,13X,8(1X,F9.2))              00018900
    WRITE (6,380) (TOTEEXP(NY),NY=N1,MAX)                                00018910
380 FORMAT (//,15H TOTAL EXPENSES,20X,8(1X,F9.2))                  00018920
    WRITE (6,383) (OPINC(NY),NY=N1,MAX)                                00018930
383 FORMAT (//,17H OPERATING INCOME,18X,8(1X,F9.2))                00018940
    IF (IAFC.EQ.1) WRITE (6,387) (AFDC1(NY),NY=N1,MAX)               00018950
387 FORMAT (8H + AFUDC,27X,8(1X,F9.2))                            00018960
    WRITE (6,390) (BINT(NY),NY=N1,MAX)                                00018970
390 FORMAT (//,23H INCOME BEFORE INTEREST,12X,8(1X,F9.2))          00018980
    WRITE (6,400) (CDEBS(NY),NY=N1,MAX)                                00018990
400 FORMAT (28H - INTRST ON SHORT TERM DEBT,7X,8(1X,F9.2))          00019000
    WRITE (6,410) (CDEBL(NY),NY=N1,MAX)                                00019010
410 FORMAT (27H - INTRST ON LONG TERM DEBT,8X,8(1X,F9.2))          00019020
    WRITE (6,420) (ERNET(NY),NY=N1,MAX)                                00019030
420 FORMAT (//,11H NET INCOME,24X,8(1X,F9.2))                      00019040
    WRITE (6,430) (DIVPR(NY),NY=N1,MAX)                                00019050
430 FORMAT (//,22H - PREFERRED DIVIDENDS,15X,8(1X,F9.2))              00019060
    WRITE (6,440) (ERNCOM(NY),NY=N1,MAX)                                00019070
440 FORMAT (25H EARNINGS AVAIL TO COMMON,10X,8(1X,F9.2))              00019080
    WRITE (6,450) (DIVCOM(NY),NY=N1,MAX)                                00019090
450 FORMAT (//,19H - COMMON DIVIDENDS,16X,8(1X,F9.2))                00019100
    WRITE (6,460) (ERNKT(NY),NY=N1,MAX)                                00019110
460 FORMAT (18H RETAINED EARNINGS,17X,8(1X,F9.2))                  00019120
    IF (MAX.EQ.MXYR) GO TO 462
N1=MAX+1
MAX=MXYR
GO TO 305
462 N1=1
MAX=8
465 CONTINUE
C
CCCC PRINT THE SOURCES AND USES OF FUNDS STATEMENT
C
        WRITE (6,99)
        WRITE (6,115)
        WRITE (6,470)
470 FORMAT (40X,26H SOURCES AND USES OF FUNDS)                      00019280
        WRITE (6,145)
        WRITE (6,152) (NYR(I),I=N1,MAX)                                00019290
        WRITE (6,165) (HYPM(I),I=N1,MAX)                                00019310
        WRITE (6,480)
480 FORMAT (//,8H SOURCES)                                         00019330
        WRITE (6,490)
490 FORMAT (//,9H INTERNAL)                                         00019340
        WRITE (6,500) (ERNET(NY),NY=N1,MAX)                                00019360
500 FORMAT (12H NET INCOME,23X,8(1X,F9.2))                          00019370
        WRITE (6,510) (DEPL(NY),NY=N1,MAX)                                00019380
510 FORMAT (14H DEPRECIATION,21X,8(1X,F9.2))                          00019390
        WRITE (6,520) (TXDEF(NY),NY=N1,MAX)                                00019400
520 FORMAT (23H DEFERRED INCOME TAXES,12X,8(1X,F9.2))                00019410
        WRITE (6,530) (TXCR(NY),NY=N1,MAX)                                00019420
530 FORMAT (20H DEF INV TAX CREDIT,15X,8(1X,F9.2))                  00019430
        WRITE (6,540) (SRCIN(NY),NY=N1,MAX)                                00019440
540 FORMAT (15H TOTAL INTERNAL,20X,8(1X,F9.2))                      00019450

```

```

      WRITE (6,550)
550 FORMAT (//,9H EXTERNAL)
      WRITE (6,560) (STCOM(NY),NY=N1,MAX)
560 FORMAT (14H COMMON STOCK,21X,8(1X,F9.2))
      WRITE (6,570) (STPR(NY),NY=N1,MAX)
570 FORMAT (17H PREFERRED STOCK,18X,8(1X,F9.2))
      WRITE (6,580) (DEBL(NY),NY=N1,MAX)
580 FORMAT (16H LONG TERM DEBT,19X,8(1X,F9.2))
      WRITE (6,585) (DEBS(NY),NY=N1,MAX)
585 FORMAT (17H SHORT TERM DEBT,18X,8(1X,F9.2))
      WRITE (6,590) (SRCEX(NY),NY=N1,MAX)
590 FORMAT (16H TOTAL EXTERNAL,19X,8(1X,F9.2))
      WRITE (6,600) (SRCTOT(NY),NY=N1,MAX)
600 FORMAT (//,14H TOTAL SOURCES,21X,8(1X,F9.2))
      WRITE (6,610)
610 FORMAT (//,5H USES)
      WRITE (6,640) (CE1(NY),NY=N1,MAX)
640 FORMAT (//,27H TOT CONS EXPN (INCL AFUDC),8X,8(1X,F9.2))
      WRITE (6,650) (DIVPR(NY),NY=N1,MAX)
650 FORMAT (21H PREFERRED DIVIDENDS,14X,8(1X,F9.2))
      WRITE (6,660) (DIVCOM(NY),NY=N1,MAX)
660 FORMAT (18H COMMON DIVIDENDS,17X,8(1X,F9.2))
      WRITE (6,670) (DEBK(NY),NY=N1,MAX)
670 FORMAT (17H DEBT RETIREMENT,18X,8(1X,F9.2))
      WRITE (6,680) (CAPIN(NY),NY=N1,MAX)
680 FORMAT (25M NET INCR WORKING CAPITAL,10X,8(1X,F9.2))
      WRITE (6,690) (USETCT(NY),NY=N1,MAX)
690 FORMAT (//,12H TOTAL USES,23X,8(1X,F9.2))
      IF (MAX.EQ.MXYR) GO TO 710
      N1=MAX+1
      MAX=MXYR
      GO TO 465
710 CONTINUE
C     PRINT THE RATIO ANALYSIS
      N1=1
      MAX=8
720 CONTINUE
      WRITE (6,991)
      WRITE (6,115)
      WRITE (6,720)
725 FORMAT (40X,15H RATIO ANALYSIS)
      WRITE (6,155) (NYR(I),I=N1,MAX)
      WRITE (6,165) (HYPH(I),I=N1,MAX)
      WRITE (6,714)
714 FORMAT (1M,/)
      WRITE (6,730) (DEBASS(NY),NY=N1,MAX)
730 FORMAT (/,14H DEBT TO ASSET,21X,8(1X,F9.2))
      WRITE (6,740) (DEBEQ(NY),NY=N1,MAX)
740 FORMAT (/,15H DEBT TO EQUITY,20X,8(1X,F9.2))
      WRITE (6,750) (COVRG(NY),NY=N1,MAX)
750 FORMAT (/,24H INTEREST COVERAGE KATID,11X,8(1X,F9.2))
      IF (MAX.EQ.MXYR) GO TO 790
      N1=MAX+1
      MAX=MXYR
      GO TO 720
790 CONTINUE
      RETURN
      END
C
C-----SUBROUTINE BALANC(NY)
C-----FIND CUMULATIVE VALUES OF COMMON STOCKS, RETAINED EARNINGS
C-----AND PREFERRED STOCKS.

```

C COMMON /DEBTS/ DEBS(20),DEBL(20),DEBK(20),CDEBS(20),
 + CDEBL(20),RDEBS(20),RDEBL(20),DEBT(20),
 + CDEBT(20),DEBN(20),FDEBT(20),FDEBS(20),
 + FDEBL(20),FDEBK(20,20,2)
 COMMON /EQUITY/ STCOM(20),STPR(20),FSTCOM(20),FSTPR(20),
 + DIVCOM(20),DIVPR(20),RLIVC(20),RDIVP(20),
 + FDIVC(20),FDIVP(20),RFDEBL(20)
 COMMON /INCOME/ GREV(20),OPINC(20),TXINC(20),ARATE(20),
 + AINC(20),BINC(20),OMC(20),ERN(20),
 + ANINC(20),ERNET(20),ERNRT(20),
 + OMCF(20),OMCX(20)
 COMMON /INPUTS/ FINDAT(20,30),BUAT(20),KEYFIN(140,18)
 COMMON /PLANTS/ PLGS1(20),PLGAI(20),PLNS1(20),PLNAI(20),
 + PLNS2(20),ITPLX(20),LPLX(20),IPLX(22,20),
 + PLX(22,20),CEPX(22,20),CEPLX(22,20),
 + CEX(20),PLGSF(20),CEF(20),CWIP(20),
 + CWIPC(20),DEPLX(22,20),DEPLX1(22,20),
 + DEPLX2(22,20),DEPX1(20),DEPX2(20),DEPF1(20),
 + DEPF2(20),DEP1(20),DEP2(20),AFDCF2(20),
 + AFDCR(20),AFDCI(20),AFDC2(20),AFCX2(22,20),
 + CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
 + CEPXC(20,20),CE1(20)
 COMMON /FUNDS/ SRCIN(20),SRCLEX(20),SRC(20),RWKCAP(20),
 + USE(20),CAPIN(20),SRCTOT(20),USEDT(20),
 + WKCAP(20),DWKCAP(20),WKCAP1(20)
 COMMON /TAXES/ RTXF(20),TXFD(20),RTXST(20),TXST(20),
 RTXP(20),TXP(20),RTXCR(20),TXCR(20)
 COMMON /AUX/ PLGS2(20),DEP1C(20),DEP2C(20),BAL(20),
 + ASSET(20),EQCUM(20),EKNCUM(20),CAPTOT(20),
 + TXDEF(20),DEFCR(20),TL1A(20),FTXDEF(20),
 + OPEXP(20),TUTEXP(20),PLGA2(20),PLNV(20),
 + PLNUT(20),STCUMC(20),STPRC(20),ERNRTC(20),
 + FDEBLR(20,20),FDEBSR(20,20),TXREV(20),RTXREV(20),
 + CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20),
 + PLNA2(20),DEBLT(20),DEBSC(20),DEBLR(20),
 + DEBSR(20),FCE(20),LCR(20),TXCRC(20),
 + UPCOST(20),PLGSX(20),TXDFC(20)
 COMMON /AUX2/ DSLIM(20),DLLIM(20),LUMLIM(20),
 + PKLIM(20),FCWIP(20),BINT(20),CWIPC1(20),
 + TXSPD(20),FUEL(20),ITPL(20),PLC(200),
 + IPL(200,20),PL(200,20),ELPL(200),LPL(200),
 + DEPL1(200,20),DEPL2(200,20),DEPL(200,20),
 + FSUM(20,20),EQTOT(20),DEBASS(20),
 + DEBEQ(20),COVRG(20),KYL(20,2)
 COMMON /BVAL/ PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
 + DEBSB,DEBLB,DEBSR,DEBLRB,STCOMB,STPRB,
 + ERNRTB,AFDC1B,TXUEFB,TXCRB,WKCAPB,
 + AFDC2B
 COMMON /INDYN/ NF1,1PLW(200,20),ELHF,ELSF,ELHX,
 + ELSX,PLCC(200),HXCC,SXCC,HFCAP(20),
 + SFCAP(20),PLCAP(200),FDEXP(20),NEXP,
 + NBEST,HXC,SXC,ISOL(20,20)
 COMMON /PARM/ NYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
 COMMON /FLAGS/ ISTOP
 INTEGER RESP,YES,YE,NEG,NO
 CCC
 DATA YES,YE/3HYES,1HY/
 DATA NEG,NE/2HNU,1HN/
 FIND TOTAL ASSETS AND LIABILITIES
 PLGS2(NY)=PLGS1(NY)+AFDC2C(NY)
 PLNV(NY)=PLGS2(NY)-DEP1C(NY)
 CWIPC1(NY)=CWIPC(NY)+AFDC1C(NY)
 PLNUT(NY)=PLNV(NY)+CWIPC1(NY)

```

EQCOM(NY)=STCOMC(NY)+ERNRTC(NY) 00020940
EQTOT(NY)=EQCOM(NY)+STPRC(NY) 00020950
CAPTOT(NY)=EWTOT(NY)+DEBLT(NY) 00020960
TLIAB(NY)=CAPTOT(NY)+DEBST(NY)+TXDEF(C(NY)) 00020970
+ TXCRC(NY) 00020980
BAL(NY)=TLIAB(NY)-PLNUT(NY) 00020990
IF (NY.EQ.1) WKCAP(NY)=BAL(NY) 00020992
ASSET(NY)=PLNUT(NY)+WKCAP(NY) 00021000
DEBASS(NY)=DEBT(NY)/ASSET(NY) 00021030
DEBEQ(NY)=DEBT(NY)/ETOT(NY) 00021040
COVRG(NY)=BINT(NY)/CDEBT(NY) 00021050
00021070
00021080
00021090
00021100
00021110
00021120
00021130
00021140
00021150
00021160
00021170
00021180
00021190
00021200
00021210
00021220
00021230
00021240
00021250
00021260
00021270
00021280
00021290
00021300
00021310
00021320
00021330
00021340
00021350
00021360
00021370
00021380
00021390
00021391
00021400
00021410
00021411
00021420
00021430
00021440
00021450
00021460
00021470
00021480
00021490
00021500
00021510
00021520
00021530
00021540
00021550
00021560
00021570
00021580
00021590
00021600
00021610

C C
RETURN
END
C C
SUBROUTINE FUND(NY)
C C
COMMON /DEBTS/ DEBS(20),DEBL(20),DEBR(20),CDEBS(20),
+ CDEBL(20),RDEBS(20),KDEBL(20),DEBT(20),
+ CDEBT(20),DEBN(20),FDEBT(20),FDEBS(20),
+ FDEBL(20),FDEBR(20,20,2)
COMMON /EQUITY/ STCOM(20),STPR(20),FSTCOM(20),FSTPR(20),
+ DIVCOM(20),DIVPR(20),RDIVC(20),RDIVP(20),
+ FDIVC(20),FDIVP(20),FDEBL(20)
COMMON /INCOME/ GREV(20),UPINC(20),TXINC(20),ARATE(20),
+ AINC(20),BINC(20),UMC(20),FERN(20),
+ ANINC(20),ERNET(20),ERNRT(20)
COMMON /INPUTS/ FINDAT(20,30),BDAT(20),KEYFIN(140,18)
COMMON /PLANTS/ PLGS1(20),PLGA1(20),PLNS1(20),PLNA1(20),
+ PLNS2(20),ITPLX(20),LPLX(20),IPLEX(22,20),
+ PLX(22,20),CEPX(22,20),CEPLX(22,20),
+ CEX(20),PLGSPF(20),CEF(20),CWIP(20),
+ CWIPC(20),DEPLX(22,20),DEPLX1(22,20),
+ DEPLX2(22,20),DEPX1(20),DEPX2(20),DEPF1(20),
+ DEPF2(20),DEP1(20),DEP2(20),AFDC(2120),
+ AFDCR(20),AFDC1(20),AFDC2(20),AFCX(22,20),
+ CAPRO(20),RATB(20),CE(20),AFDCX(20),
+ CEPXC(20,20),CE1(20)
COMMON /FUNDS/ SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
+ USE(20),CAPIN(20),SRCTDT(20),USEDT(20),
+ WKCAP(20),DWKCAP(20),WKCAP1(20)
COMMON /TAXES/ RTXFD(20),TXFD(20),RTXST(20),TXST(20),
+ RTXP(20),TXP(20),RTXCR(20),TXCR(20)
COMMON /AUX/
+ PLGS2(20),DEPIC(20),DEPZC(20),BAL(20),
+ ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20),
+ TXDEF(20),DEFPCR(20),TLIAB(20),FTXDEF(20),
+ QPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
+ PLNUT(20),STCML(20),STPRC(20),ERNRT(20),
+ FDEBLR(20,20),FDEBSR(20,20),TXREV(20),RTXREV(20),
+ CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20),
+ PLNA2(20),DEBLT(20),DEBST(20),DEBLR(20),
+ DEBSR(20),FCE(20),LCR(20),TXCRC(20),
+ OPCOST(20),PLGSX(20),TXDEF(20)
COMMON /AUX2/
+ DSLIM(20),DLIM(20),CMLIM(20),
+ PRLIM(20),FCWIP(20),BINT(20),CWIPC(20),
+ TXSPD(20),FUEL(20),IP(20),PLC(200),
+ IPL(200,20),PL(200,20),ELPL(200),LPL(200),
+ DEPL1(200,20),DEPL2(200,20),DEPL(200,20),
+ FSUM(20,20),EWTOT(20),DEBASS(20),
+ DEBEW(20),COKRG(20),KYR(20,2)
COMMON /BVAL/
+ PLGSB,DEP1B,DEP2B,CW1PFB,CWIPXB,CWIPB,

```

```

+ DEBSB,DEBLB,DEBSRB,DEBLRB,STCOMB,STPRB, 00021620
+ ERNRTB,AFDC1B,TXDEFB,TXCRB,WKCAPB, 00021630
+ AFDC2B 00021631
+ COMMON /INDYN/ NFL,IPLW(200,20),ELHF,ELSF,ELHX, 00021640
+ ELSX,PLCC(200),HACC,SXCC,HFCAP(20), 00021650
+ SFCA(20),PLLCAP(200),IDEXP(20),NEXP, 00021660
+ NBEST,HXC,SXC,ISOL(20,20) 00021670
+ COMMON /PARM/ NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC 00021680
+ COMMON /FLAGS/ ISTOP 00021690
+ INTEGER RESP,YES,YE,NEG,NO 00021700
+ 00021710
+ 00021720
+ 00021730
+ 00021740
+ 00021750
+ 00021760
+ DATA YES,YE,BHYES,1HY/ 00021770
+ DATA NEG,NE/2HNU,1HN/ 00021780
C SRCIN(NY)=ERNET(NY)+DEP1(NY)+TXDEF(NY) 00021780
C SRCEX(NY)=STCOM(NY)+STPR(NY)+DEBS(NY)+ 00021800
+ DEBL(NY) 00021810
+ CE1(NY)=CE(NY)+AFDC1(NY) 00021812
+ SRCTOT(NY)=SRCIN(NY)+SRCEX(NY) 00021820
+ USE(NY)=CE1(NY)+DIVPR(NY)+DIVCOM(NY)+DEBR(NY) 00021840
+ CAPIN(NY)=SRCTOT(NY)-USE(NY) 00021860
+ USETOT(NY)=USE(NY)+CAPIN(NY) 00021870
+ 00021890
+ 00021900
+ 00021910
+ 00021920
+ 00021930
+ 00021940
+ 00021950
+ 00021960
+ 00021970
+ 00021980
+ 00021990
+ 00022000
+ 00022010
+ 00022020
+ COMMON /DEBTS/ DEBS(20),DEBL(20),DEBR(20),CDEBS(20), 00022030
+ CDEBL(20),RDEBS(20),RDEBL(20),DEBT(20), 00022040
+ CDEBT(20),DEBN(20),FDEBT(20),FDEBS(20), 00022050
+ FDEBL(20),FDEBR(20,20,2) 00022060
+ COMMON /EQUITY/ STCOM(20),STPR(20),FSTCOM(20),FSTPR(20), 00022070
+ DIVCOM(20),DIVPR(20),RDIVC(20),RDIVP(20), 00022080
+ FDIVC(20),FDIVP(20),RFDEBL(20) 00022090
+ COMMON /INCOME/ GREV(20),UPINC(20),TXINC(20),ARATE(20), 00022100
+ AINC(20),BINC(20),UMC(20),ERNI(20), 00022110
+ ANINC(20),BKNET(20),ERNRT(20), 00022120
+ OMCF(20),UMCX(20) 00022130
+ COMMON /INPUTS/ FINDAT(20,30),BDAT(20),KEYFIN(140,18) 00022140
+ COMMON /PLANTS/ PLGS1(20),PLGA1(20),PLNS1(20),PLNA1(20), 00022150
+ PLNS2(20),ITPLX(20),LPLX(20),IPLX(22,20), 00022160
+ PLX(22,20),CEPX(22,20),CEPLX(22,20), 00022170
+ CEX(20),PLGSF(20),CEF(20),CWIP(20), 00022180
+ CWIPC(20),DEPLX(22,20),DEPLX1(22,20), 00022190
+ DEPLX2(22,20),DEPA1(20),DEPA2(20),DEPF1(20), 00022200
+ DEPF2(20),DEP1(20),DEP2(20),AFDCF2(20), 00022210
+ AFDCR(20),AFDC1(20),AFDC2(20),AFCX2(22,20), 00022220
+ CAPRQ(20),RATB(20),CE(20),AFDCX2(20), 00022230
+ CEPXC(20,20),CE1(20) 00022231
+ COMMON /FUNDS/ SRCIN(20),SRCEX(20),SKC(20),RWKCAP(20), 00022240
+ USE(20),CAPIN(20),SRCTOT(20),USETOT(20), 00022250
+ WKCAP(20),DWKCAP(20),WKCAP1(20) 00022251
+ RTXFD(20),TXFD(20),KAST(20),TXST(20), 00022260
+ RTXP(20),TXP(20),RTXCR(20),TXCR(20) 00022270
+ COMMON /AUX/ PLGS2(20),DEP1C(20),DEP2C(20),BAL(20), 00022280
+ ASSET(20),EWCOM(20),ERNCUM(20),CAPTOT(20), 00022290
+ TXDEF(20),DEFCR(20),TL1AB(20),FTXDEF(20), 00022300

```

```

+
+ OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20), 00022310
+ PLNUT(20),STCUMC(20),STPRC(20),ERNRTC(20), 00022320
+ FDEBLR(20,20),FDEBSR(20,20),TXREV(20),RTXREV(20), 00022330
+ CEC(20),AFDC1C(20),AFDC2C(20),FDEP(20), 00022340
+ PLNA2(20),DEBLT(20),DEBSC(20),DEBLR(20), 00022350
+ DEBSR(20),FCE(20),LCR(20),TXCRC(20), 00022360
+ OPCOST(20),PLGSX(20),TXDEF(20), 00022370
+ DLSIM(20),DLIM(20),COMLIM(20), 00022380
+ PRLIM(20),FCWIP(20),BINT(20),CWIPC1(20), 00022390
+ TXSPD(20),FUEL(20),ITPL(20),PLC(200), 00022400
+ IPL(200,20),PL(200,20),ELPL(200),LPL(200), 00022410
+ DEPL1(200,20),DEPL2(200,20),DEPL(200,20), 00022420
+ FSUM(20,20),EQUT(20),DEBASS(20), 00022430
+ DEBEQ(20),COVRG(20),KYR(20,2), 00022440
+ PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB, 00022450
+ DEBSB,DEBLB,DEBSRB,DEBLRB,STCUMB,STPRB, 00022460
+ ERNRTB,AFDC1B,TXDEFB,TXCRB,WKCAPB, 00022470
+ AFDC2B, 00022480
+ NFI,IPLW(200,20),ELHF,ELSF,ELHX, 00022481
+ ELSX,PLCC(200),HXC,SXC,HFCAP(20), 00022490
+ SFCAP(20),PLCAP(200),IDEXP(20),NEXP, 00022500
+ NBEST,HXC,SXC,ISOL(20,20), 00022510
+ NBXR,MXZR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC 00022520
+ COMMON /PARM/ 00022530
+ COMMON /FLAGS/ 00022540
C INTEGER RESP,YES,YE,NEG,NO 00022550
CCC
C DATA YES,YE/3HYES,1HY/ 00022560
C DATA NEG,NE/2HNO,1HN/ 00022570
C
C 105 FORMAT (A3) 00022610
C 106 CONTINUE 00022620
C WRITE (6,110) 00022630
C 110 FORMAT (//,3H WHICH DATA YOU WANT TO CHANGE?, 00022640
C +/4H ENTER 2 DIGIT DATA CODE. FOR HELP,ENTER 99., 00022650
C +/3H --) 00022660
C READ (5,120) IP 00022670
C 120 FORMAT (I2) 00022680
C IF (IP.EQ.99) GO TO 160 00022690
C
C ENTER CHANGED/CORRECTED PARAMETER 00022700
C
C 125 CONTINUE 00022710
C WRITE (6,135) (KEYFIN(IP+34,J),J=1,18) 00022720
C 135 FORMAT (18A4) 00022730
C WRITE (6,142) BDAT(IP) 00022740
C 142 FORMAT (16H PREVIOUS VALUE=,F10.4) 00022750
C WRITE (6,145) 00022760
C 145 FORMAT (11H ----) 00022770
C READ (5,155) BDAT(IP) 00022780
C 155 FORMAT (F10.5) 00022790
C WRITE (6,159) 00022800
C 159 FORMAT (//,26H ANY MORE CHANGES IN DATA?, 00022810
C +/17H ENTER YES OR NO./2H ?) 00022820
C READ (5,105) RESP 00022830
C IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 100 00022840
C GO TO 190 00022850
C 160 CONTINUE 00022860
C
C PRINT 2 DIGIT DATA CODES AND 00022870
C DEFINITIONS. 00022880
C
C DO 180 IP=1,NBDT 00022890
C WRITE (6,135) (KEYFIN(IP+17,J),J=1,18) 00022900
C 180 CONTINUE 00022910
C GO TO 106 00022920
C
C

```

190 CONTINUE

C
C RETURN
C END

00022980
00022990
00023000
00023010
00023020
00023030

Appendix G
SAMPLE OUTPUT

***** INPUT PROCEDURE FOR PLANT DATA STARTS *****

UNLESS OTHERWISE SPECIFIED,
A PLANT CODE IDENTIFIES A THERMAL PLANT.
DO YOU HAVE A FILE ALREADY FOR THE PLANT DATA?
ENTER YES OR NO.

?

S

ARE ANY CHANGES IN PLANT DATA NECESSARY? ENTER YES OR NO.

N

DO YOU WANT TO ADD PLANTS? ENTER YES OR NO.

N

**** IF PRINT OF DATA IS NEEDED, ENTER YES.

S

ENTER 1,2,3,4 OR 5 ACCORDING TO PRINT OPTIONS

1 PRINT ALL DATA

2 PRINT ALL DATA FOR A PARTICULAR PLANT

3 PRINT THE SAME DATA FOR ALL PLANTS

4 PRINT SPECIFIC DATA FOR A PARTICULAR PLANT

5 EARLIER REQUEST FOR PRINTING CANCELLED

1

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 1 (EXPANSION PLANT).
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 10 15 15 15 15 15 15

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 2 (EXPANSION PLANT).
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 10 15 15 15 15 15 15

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 3 (EXPANSION PLANT).
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 10 15 15 15 15 15 15

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 4 (EXPANSION PLANT).
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 10 15 15 15 15 15 15

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 5 (EXPANSION PLANT).
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 10 15 15 15 15 15 15

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 6 (EXPANSION PLANT).
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 10 15 15 15 15 15 15

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 7 (EXPANSION PLANT).
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 10 15 15 15 15 15 15

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 8 (EXPANSION PLANT).

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 15 15 15 15 15 15

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 9 (EXPANSION PLANT).

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 15 15 15 15 15 15

LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)
OF PLANT CODE 10 (EXPANSION PLANT).

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5 5 5 5 10 10 10 10 10 15 15 15 15 15 15

NUMBER OF UNITS OF PLANT CODE 11 WORKING IN EACH YEAR

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

NUMBER OF UNITS OF PLANT CODE 12 WORKING IN EACH YEAR

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

NUMBER OF UNITS OF PLANT CODE 13 WORKING IN EACH YEAR

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 14 WORKING IN EACH YEAR

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 15 WORKING IN EACH YEAR

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 16 WORKING IN EACH YEAR

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 17 WORKING IN EACH YEAR

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 18 WORKING IN EACH YEAR

81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
--- --- --- --- --- --- --- --- --- --- --- ---
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

NUMBER OF UNITS OF PLANT CODE 19 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 20 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 21 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 22 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

NUMBER OF UNITS OF PLANT CODE 23 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 24 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

NUMBER OF UNITS OF PLANT CODE 25 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

NUMBER OF UNITS OF PLANT CODE 26 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5

NUMBER OF UNITS OF PLANT CODE 27 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

NUMBER OF UNITS OF PLANT CODE 28 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

NUMBER OF UNITS OF PLANT CODE 29 WORKING IN EACH YEAR
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

NUMBER OF UNITS OF PLANT CODE 30 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

NUMBER OF UNITS OF PLANT CODE 31 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

NUMBER OF UNITS OF PLANT CODE 32 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

NUMBER OF UNITS OF PLANT CODE 33 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

NUMBER OF UNITS OF PLANT CODE 34 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

NUMBER OF UNITS OF PLANT CODE 35 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5

NUMBER OF UNITS OF PLANT CODE 36 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5

NUMBER OF UNITS OF PLANT CODE 37 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

NUMBER OF UNITS OF PLANT CODE 38 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5

NUMBER OF UNITS OF PLANT CODE 39 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4

NUMBER OF UNITS OF PLANT CODE 40 WORKING IN EACH YEAR
 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5

PLANT CODE	1 EXPN	2 EXPN	3 EXPN	4 EXPN	5 EXPN
02 BASE CAPCTY MW	600.000	600.000	400.000	250.000	300.000
03 MAX CAPACITY MW	1200.000	1200.000	800.000	400.000	300.000
04 MAINT DTG DYS/YR	40.000	30.000	25.000	25.000	15.000
05 FORCED DTG.RATE	0.300	0.250	0.200	0.200	0.150
06 CAP. COST \$/KW	400.000	300.000	330.000	180.000	160.000
07 BASE FL CST \$/MWH	3.900	7.000	7.700	9.100	11.910
08 MAX FL CST \$/MWH	7.000	10.000	11.000	13.000	17.020
10 FIX O.M.C. \$/MWY	5.000	5.000	4.000	2.000	2.000
11 VAR O.M.C. \$/MWH	0.005	0.012	0.010	0.010	0.010
12 SALVG VALUE K\$	0.0	0.0	0.0	0.0	0.0
13 CAP. ESC. RATE	0.080	0.080	0.080	0.080	0.080
14 FUEL ESC. RATE	0.100	0.100	0.100	0.100	0.100
PLANT CODE	6 EXPN	7 EXPN	8 EXPN	9 EXPN	10 EXPN
02 BASE CAPCTY MW	200.000	200.000	200.000	100.000	70.000
03 MAX CAPACITY MW	200.000	500.000	200.000	100.000	70.000
04 MAINT DTG DYS/YR	10.000	25.000	0.800	5.000	2.000
05 FORCED DTG.RATE	0.150	0.120	0.100	0.070	0.050
06 CAP. COST \$/KW	140.000	200.000	120.000	130.000	90.000
07 BASE FL CST \$/MWH	12.600	8.400	17.500	18.900	24.500
08 MAX FL CST \$/MWH	18.000	12.000	25.000	27.000	35.000
10 FIX O.M.C. \$/MWY	1.000	3.000	0.500	0.500	0.200
11 VAR O.M.C. \$/MWH	0.010	0.010	0.020	0.020	0.020
12 SALVG VALUE K\$	0.0	0.0	0.0	0.0	0.0
13 CAP. ESC. RATE	0.080	0.080	0.080	0.080	0.080
14 FUEL ESC. RATE	0.100	0.100	0.100	0.100	0.100
PLANT CODE	11 SCHD	12 SCHD	13 SCHD	14 SCHD	15 SCHD
02 BASE CAPCTY MW	203.000	212.000	80.000	94.000	96.000
03 MAX CAPACITY MW	653.000	650.000	330.000	276.000	255.000
04 MAINT DTG DYS/YR	26.000	26.000	21.000	25.000	24.000
05 FORCED DTG.RATE	0.117	0.117	0.045	0.046	0.039
06 CAP. COST \$/KW	300.000	300.000	150.000	140.000	140.000
07 BASE FL CST \$/MWH	8.600	8.540	14.700	15.400	16.800
08 MAX FL CST \$/MWH	12.400	12.200	21.000	22.000	24.000
10 FIX O.M.C. \$/MWY	3.500	3.400	2.500	2.100	2.000
11 VAR O.M.C. \$/MWH	0.010	0.010	0.010	0.010	0.010
12 SALVG VALUE K\$	0.0	0.0	0.0	0.0	0.0
13 CAP. ESC. RATE	0.080	0.080	0.080	0.080	0.080
14 FUEL ESC. RATE	0.100	0.100	0.100	0.100	0.100
PLANT CODE	16 SCHD	17 SCHD	18 SCHD	19 SCHD	20 SCHD
02 BASE CAPCTY MW	76.000	109.000	159.000	77.000	47.000
03 MAX CAPACITY MW	243.000	232.000	570.000	225.000	191.000
04 MAINT DTG DYS/YR	20.000	26.000	25.000	14.000	21.000
05 FORCED DTG.RATE	0.034	0.052	0.046	0.023	0.035
06 CAP. COST \$/KW	135.000	134.000	280.000	130.000	120.000
07 BASE FL CST \$/MWH	17.000	17.100	9.000	17.500	18.200
08 MAX FL CST \$/MWH	24.500	24.600	12.500	25.000	27.000
10 FIX O.M.C. \$/MWY	2.000	2.000	3.100	2.000	2.000
11 VAR O.M.C. \$/MWH	0.010	0.010	0.010	0.010	0.009
12 SALVG VALUE K\$	0.0	0.0	0.0	0.0	0.0
13 CAP. ESC. RATE	0.080	0.080	0.080	0.080	0.080
14 FUEL ESC. RATE	0.100	0.100	0.100	0.100	0.100

PLANT CODE	21	22	23	24	25
	SCHD	SCHD	SCHD	SCHD	SCHD
02 BASE CAPCTY MW	80.000	51.000	72.000	45.000	47.000
03 MAX CAPACITY MW	176.000	142.000	142.000	161.000	110.000
04 MAINT OTG DYS/YR	20.000	20.000	14.000	21.000	21.000
05 FORCED OTG.RATE	0.034	0.034	0.023	0.035	0.035
06 CAP. COST \$/KW	120.000	115.000	117.000	119.000	105.000
07 BASE FL CST \$/MWH	18.200	19.000	18.600	18.400	20.000
08 MAX FL CST \$/MWH	27.000	28.000	27.500	27.000	29.000
10 FIX O.M.C. \$/MWY	2.000	2.000	2.000	2.000	2.000
11 VAR O.M.C. \$/MWH	0.009	0.009	0.009	0.009	0.009
12 SALVG VALUE L\$	0.0	0.0	0.0	0.0	0.0
13 CAP. ESC. RATE	0.080	0.080	0.080	0.080	0.080
14 FUEL ESC. RATE	0.100	0.100	0.100	0.100	0.100
PLANT CODE	26	27	28	29	30
	SCHD	SCHD	SCHD	SCHD	SCHD
02 BASE CAPCTY MW	38.000	40.000	27.000	28.000	20.000
03 MAX CAPACITY MW	106.000	106.000	104.000	80.000	66.000
04 MAINT OTG DYS/YR	20.000	22.000	19.000	16.000	21.000
05 FORCED OTG.RATE	0.034	0.039	0.035	0.027	0.035
06 CAP. COST \$/KW	105.000	105.000	100.000	90.000	80.000
07 BASE FL CST \$/MWH	20.000	20.000	21.000	21.000	22.000
08 MAX FL CST \$/MWH	29.000	29.000	29.500	29.500	30.000
10 FIX O.M.C. \$/MWY	2.000	2.000	1.500	1.500	1.500
11 VAR O.M.C. \$/MWH	0.009	0.009	0.008	0.007	0.005
12 SALVG VALUE K\$	0.0	0.0	0.0	0.0	0.0
13 CAP. ESC. RATE	0.080	0.080	0.080	0.080	0.080
14 FUEL ESC. RATE	0.100	0.100	0.100	0.100	0.100
PLANT CODE	31	32	33	34	35
	SCHD	SCHD	SCHD	SCHD	SCHD
02 BASE CAPCTY MW	73.000	21.000	25.000	19.000	27.000
03 MAX CAPACITY MW	333.000	235.000	69.000	47.000	27.000
04 MAINT OTG DYS/YR	15.000	15.000	22.000	22.000	25.000
05 FORCED OTG.RATE	0.046	0.046	0.039	0.039	0.053
06 CAP. COST \$/KW	140.000	120.000	100.000	90.000	80.000
07 BASE FL CST \$/MWH	28.000	30.000	32.500	35.000	42.000
08 MAX FL CST \$/MWH	36.000	37.000	46.000	50.000	60.000
10 FIX O.M.C. \$/MWY	0.700	0.600	0.600	0.500	0.500
11 VAR O.M.C. \$/MWH	0.020	0.020	0.010	0.007	0.007
12 SALVG VALUE K\$	0.0	0.0	0.0	0.0	0.0
13 CAP. ESC. RATE	0.080	0.080	0.080	0.080	0.080
14 FUEL ESC. RATE	0.100	0.100	0.100	0.100	0.100
PLANT CODE	36	37	38	39	40
	SCHD	SCHD	SCHD	SCHD	SCHD
02 BASE CAPCTY MW	17.000	35.000	35.000	25.000	28.000
03 MAX CAPACITY MW	17.000	35.000	35.000	25.000	28.000
04 MAINT OTG DYS/YR	0.0	0.0	0.0	0.0	0.0
05 FORCED OTG.RATE	0.016	0.012	0.032	0.026	0.032
06 CAP. COST \$/KW	65.000	85.000	85.000	80.000	81.000
07 BASE FL CST \$/MWH	44.000	36.200	38.500	44.500	42.700
08 MAX FL CST \$/MWH	62.000	53.000	55.000	63.000	61.000
10 FIX O.M.C. \$/MWY	0.500	0.200	0.200	0.200	0.200
11 VAR O.M.C. \$/MWH	0.005	0.005	0.005	0.005	0.005
12 SALVG VALUE K\$	0.0	0.0	0.0	0.0	0.0
13 CAP. ESC. RATE	0.080	0.080	0.080	0.080	0.080
14 FUEL ESC. RATE	0.100	0.100	0.100	0.100	0.100

DO YOU WANT TO PRINT MORE DATA? ENTER YES OR NO.

11

END OF PRINTING

IS YOUR PLANT DATA COMPLETE ? ENTER YES OR NO.

n ARE ANY CHANGES IN PLANT DATA NECESSARY? ENTER YES OR NO.

s DO YOU WANT TO CHANGE ALL THE DATA OF A PLANT?
ENTER YES OR NO.

n ENTER THE 3 DIGIT PLANT CODE FOR THE SINGLE DATA CHANGE.

005 ENTER DATA CODE IN 2 DIGIT, OR 99 FOR HELP.

--
99 DATA CODE, CONTENT

01 NUMBER OF UNITS IN EACH YEAR
02 BASE CAPCTY MW
03 MAX CAPACITY MW
04 MAINT OTG DYS/YR
05 FORCED OTG.RATE
06 CAP. COST \$/KW
07 BASE FL CST \$/MWH
08 MAX FL CST \$/MWH
09 (NOT USED)
10 FIX O.M.C. \$/MWY
11 VAR O.M.C. \$/MWH
12 SALVG VALUE K\$
13 CAP. ESC. RATE
14 FUEL ESC. RATE

ENTER DATA CODE IN 2 DIGIT, OR 99 FOR HELP.

05 FORCED OUTAGE RATE ?
(THE VALUE BEFORE CHANGE= 0.15000)

0.145 ANY MORE CHANGE FOR PLANT CODE 5? ENTER YES OR NO.

n DO YOU WANT TO CHANGE DATA FOR ANOTHER PLANT ? ENTER YES OR NO.

n DO YOU WANT TO ADD PLANTS? ENTER YES OR NO.

n **** IF PRINT OF DATA IS NEEDED, ENTER YES.

n IS YOUR PLANT DATA COMPLETE ? ENTER YES OR NO.

ATTN: THERMAL PLANT DATA FILE IS CREATED
OR REVISED ON UNIT 11
***** END OF THERMAL PLANT INPUT PROCEDURE *****

DO YOU WANT TO INCLUDE HYDROELECTRIC
AND PUMPED STORAGE PLANTS
IN YOUR INPUT DATA FILE?
ENTER YES OR NO.

s

INPUT PROCEDURE FOR HYDROELECTRIC PLANT SYSTEM BEGINS.

DO YOU HAVE A FILE FOR EXISTING HYDROELECTRIC PLANTS?
ENTER YES OR NO.

Y

THE TOTAL NO. OF EXISTING HYDROELECTRIC PLANTS IS 3

ENTER YES IF YOU WANT TO PRINT EXISTING SYSTEM DATA.
N

DO YOU WANT TO CHANGE ANY DATA FOR EXISTING
HYDROELECTRIC PLANTS? ENTER YES OR NO.

Y

ENTER PLANT CODE NUMBER(2 DIGITS)
FOR WHICH YOU WANT TO CHANGE DATA.

--
02

DO YOU WANT TO CHANGE ALL THE DATA FOR PLANT 2?
ENTER YES OR NO.

N

WHICH DATA YOU WANT TO CHANGE FOR PLANT 2 ?
ENTER 2 DIGIT DATA CODE NUMBER.

IF YOU NEED HELP,ENTER 99.

--

99

- 01 NO. OF UNITS
- 02 BASE CAPACITY,MW
- 03 MAX. AVAILABLE CAPACITY,MW
- 04 STORAGE ENERGY LIMIT,GWH
- 05 TOTAL INFLOW ENERGY IN THE PERIOD,GWH
- 06 FORCED OUTAGE RATE (FRACTION)
- 07 MAINT. OUTAGE, DAYS/YR
- 08 FIXED OPER. AND MAINT. COSTS, \$(THOUSANDS)/MW/YR
- 09 VARIABLE OPER. AND MAINT. COSTS, \$/MWH

WHICH DATA YOU WANT TO CHANGE FOR PLANT 2 ?
ENTER 2 DIGIT DATA CODE NUMBER.

IF YOU NEED HELP,ENTER 99.

--

02

BASE CAPACITY,MW ?

(PREVIOUS VALUE= 20.00000)

21.0

DO YOU WANT TO CHANGE ANY MORE DATA FOR PLANT 2 ?
ENTER YES OR NO.

n

THIS IS A SEASONAL REGULATING PLANT

DO YOU WANT TO CHANGE DATA FOR ANOTHER PLANT?
ENTER YES OR NO.

n

IS YOUR INPUT FILE FOR EXISTING HYDRO PLANTS COMPLETE?
ENTER YES OR NO.

y

DO YOU HAVE A FILE FOR EXPANSION HYDROELECTRIC PLANTS?
ENTER YES OR NO.

y

NO. OF PLANT TYPES CONSIDERED FOR EXPANSION IS 3
THEY ARE ASSIGNED PLANT CODE NUMBERS 4 THROUGH 6

ENTER YES IF YOU WANT TO PRINT EXPANSION SYSTEM DATA.
n

DO YOU WANT TO CHANGE ANY DATA FOR EXPANSION HYDRO PLANTS?
ENTER YES OR NO.

n

IS YOUR INPUT FILE FOR EXPANSION HYDRO PLANTS COMPLETE?
ENTER YES OR NO.

y

DO YOU HAVE A FILE ALREADY FOR SEASONAL MULTIPLIERS
OF BASE CAPACITY AND TOTAL INFLOW ENERGY?
ENTER YES OR NO.

?

y

ENTER YES IF YOU WANT TO PRINT THE SEASONAL MULTIPLIERS.
?

n

ENTER YES IF YOU WANT TO RENTER SEASONAL MULTIPLIERS.
?

n

PRINTING OF THE ENTIRE HYDROELECTRIC SYSTEM DATA FOLLOWS.
IF YOU WANT THIS PRINTING,ENTER YES.
y

BASE CAPACITY MULTIPLIERS:

WINTER	SPRING	SUMMER	AUTUMN
1.00000	0.90000	1.00000	0.90000

INFLOW ENERGY MULTIPLIERS:

WINTER	SPRING	SUMMER	AUTUMN
1.00000	1.05000	1.10000	1.00000

NO. OF UNITS OF EACH PLANT CODE WORKING EACH YEAR:
(EXISTING PLANTS)

HYD PLT CODE	1	2	3
1981	1	1	1
1982	1	1	1
1983	1	1	1
1984	1	1	1
1985	1	1	1
1986	1	1	1
1987	1	1	1
1988	1	1	1
1989	1	1	1
1990	1	1	1
1991	1	1	1
1992	1	1	1
1993	1	1	1
1994	1	1	1
1995	1	1	1

INDIVIDUAL PLANT DATA:
(EXISTING PLANTS)

HYD PLT CODE	1	2	3
BASE CAP(MW)	50.00000	21.00000	12.50000
MAX AVAIL CAP(MW)	300.00000	100.00000	100.00000
STR. ENRG. LIMIT(GWH)	50.00000	50.00000	35.00000
TOTAL INFL ENRG(GWH)	150.00000	100.00000	100.00000
FORCED OUTAGE RATE	0.01000	0.01000	0.00900
MAINT OUTG(D/YR)	2.00000	2.50000	2.60000
*FIXED O.M.C.(K\$/MW/YR)	200.00000	23.00000	25.60001
VAR O.M.C(\$/MWH)	20.00000	12.50000	13.45000

LOWER AND UPPER BOUNDS ON NO. OF UNITS OF
EACH PLANT CODE EACH YEAR:
(EXPANSION PLANTS)

HYD PLT CODE	4		5		6	
	L	U	L	U	L	U
1981	0	5	0	5	0	5
1982	0	5	0	5	0	5
1983	0	5	0	5	0	5
1984	0	5	0	5	0	5
1985	0	5	0	5	0	5
1986	0	5	0	5	0	5
1987	0	5	0	5	0	5
1988	0	5	0	5	0	5
1989	0	5	0	5	0	5
1990	0	5	0	5	0	5
1991	0	5	0	5	0	5
1992	0	5	0	5	0	5
1993	0	5	0	5	0	5
1994	0	5	0	5	0	5
1995	0	5	0	5	0	5

INDIVIDUAL PLANT DATA:
(EXPANSION PLANTS)

HYD PLT CODE	4			5			6		
	4	5	6	4	5	6	4	5	6
BASE CAP(MW)	12.50000	12.50000	10.00000						
MAX AVAIL CAP(MW)	55.00000	75.00000	100.00000						
STR. ENRG. LIMIT(GWH)	25.00000	5.00000	5.00000						
TOTAL INFL. ENRG(GWH)	67.00000	100.00000	50.00000						
FORCED OUTAGE RATE	0.00900	0.00800	0.00800						
MATNT OUTG.(\$/YR)	2.50000	2.00000	2.00000						
*FIXED O.M.C.(\$/MW/YR)	12.50000	10.50000	10.50000						
VAR O.M.C(\$/MW)	5.50000	5.50000	4.50000						
CAP COST(\$/KW)	23.50000	4.50000	23.80000						
CAP COST ESC RT	0.08000	0.08000	0.09000						
*SALVAGE VALUE(K\$)	1600.00000	2000.00000	1900.00000						

* K\$ MEANS DOLLARS IN THOUSANDS

END OF INPUT PROCEDURE FOR HYDROELECTRIC PLANTS.
THEIR DATA STORED IN UNITS 35, 38, 39 AND 40.

INPUT PROCEDURE FOR PUMPED STORAGE PLANT SYSTEM BEGINS

DO YOU HAVE A FILE FOR EXISTING PUMPED STORAGE PLANTS?
ENTER YES OR NO.

Y

THE TOTAL NO. OF EXISTING STORAGE PLANTS IS 3

ENTER YES IF YOU WANT TO PRINT EXISTING SYSTEM DATA.
N

DO YOU WANT TO CHANGE ANY DATA FOR EXISTING STORAGE PLANTS?
ENTER YES OR NO.

N

IS YOUR INPUT FILE FOR EXISTING STORAGE PLANTS COMPLETE?
ENTER YES OR NO.

Y

DO YOU HAVE A FILE FOR EXPANSION PUMPED STORAGE PLANTS?
ENTER YES OR NO.

Y

ONLY 1 PLANT TYPE IS CONSIDERED FOR EXPANSION.
IT IS ASSIGNED PLANT CODE NO. 4

ENTER YES IF YOU WANT TO PRINT EXPANSION SYSTEM DATA.
N

DO YOU WANT TO CHANGE ANY DATA FOR PLANT 4?
ENTER YES OR NO.

N

IS YOUR INPUT FILE FOR EXPANSION STORAGE PLANTS COMPLETE?
ENTER YES OR NO.

Y

PRINTING OF THE ENTIRE PUMPED STORAGE SYSTEM DATA FOLLOWS.
IF YOU WANT THIS PRINTING, ENTER YES.

Y

NO. OF UNITS OF EACH PLANT CODE WORKING EACH YEAR:
(EXISTING PLANTS)

PS PLT CODE	1	2	3
1981	1	1	1
1982	1	1	1
1983	1	1	1
1984	1	1	1
1985	1	1	1
1986	1	1	1
1987	1	1	1
1988	1	1	1
1989	1	1	1
1990	1	1	1
1991	1	1	1
1992	1	1	1
1993	1	1	1
1994	1	1	1
1995	1	1	1

INDIVIDUAL PLANT DATA:
(EXISTING PLANTS)

PS PLT CODE	1	2	3
CAPACITY(MW)	50.00000	25.00000	30.00000
STR ENRG LIMIT(GWH)	10.00000	20.00000	20.00000
PUMPING EFFECIENCY	0.90000	0.90000	0.90000
GENERATING EFFECIENCY	0.90000	0.90000	0.90000
FORCED OUTAGE RATE	0.00900	0.00800	0.00800
MAINT OUTG(D/YR)	2.00000	2.00000	2.50000
*FIXED O.M.C.(K\$/MW/YR)	10.50000	12.50000	12.50000
VAR O.M.C(\$/MWH)	5.50000	5.00000	5.00000

* K\$ MEANS DOLLARS IN THOUSANDS

LOWER AND UPPER BOUNDS ON NO. OF UNITS OF
THE EXPANSION PLANT EACH YEAR:

PS PLT CODE	L	U
1981	0	5
1982	0	5
1983	0	5
1984	0	5
1985	0	5
1986	0	5
1987	0	5
1988	0	5
1989	0	5
1990	0	5
1991	0	5
1992	0	5
1993	0	5
1994.	0	5
1995	0	5

INDIVIDUAL PLANT DATA:
(EXPANSION PLANT)

PS PLT CODE	4
CAPACITY(MW)	35.00000
STR ENRG LIMIT(GWH)	25.00000
PUMPING EFFECIENCY	0.90000
GENERATING EFFECIENCY	0.90000
FORCED OUTAGE RATE	0.00800
MAINT OUTG(D/YR)	2.40000
*FIXED O.M.C.(K\$/MW/YR)	12.50000
VAR O.M.C(\$/MWH)	5.50000
CAP COST(\$/KW)	20.50000
CAP COST ESC RT	0.09000
*SALVAGE VALUE(K\$)	500.00000

* K\$ MEANS DOLLARS IN THOUSANDS

END OF INPUT PROCEDURE FOR PUMPED STORAGE PLANTS.
INPUT DATA STORED ON UNITS 45,48 AND 49.
READY

exec load.clist

***** START OF LOAD INPUT PROCEDURE *****

DOES A FILE OF ENERGY MULTIPLIER AND LOAD FACTOR
EXISTS? ENTER YES OR NO.

YEAR	ENERGY MULTIPLIER FOR EACH SEASON EACH YEAR	YEAR	ENERGY MULTIPLIER FOR EACH SEASON EACH YEAR	YEAR	ENERGY MULTIPLIER FOR EACH SEASON EACH YEAR
81 :	1.07000	81 :	1.07000	81 :	1.07000
82 :	1.14000	82 :	1.14000	82 :	1.14000
83 :	1.22500	83 :	1.22500	83 :	1.22500
84 :	1.31000	84 :	1.31000	84 :	1.31000
85 :	1.40000	85 :	1.40000	85 :	1.40000
86 :	1.50000	86 :	1.50000	86 :	1.50000
87 :	1.60000	87 :	1.60000	87 :	1.60000
88 :	1.72000	88 :	1.72000	88 :	1.72000
89 :	1.83000	89 :	1.83000	89 :	1.83000
90 :	1.97000	90 :	1.97000	90 :	1.97000
91 :	2.10000	91 :	2.10000	91 :	2.10000
92 :	2.25000	92 :	2.25000	92 :	2.25000
93 :	2.40000	93 :	2.40000	93 :	2.40000
94 :	2.58000	94 :	2.58000	94 :	2.58000
95 :	2.76000	95 :	2.76000	95 :	2.76000

YEAR	LOAD FACTOR FOR EACH SEASON EACH YEAR	YEAR	LOAD FACTOR FOR EACH SEASON EACH YEAR	YEAR	LOAD FACTOR FOR EACH SEASON EACH YEAR
81 :	0.60000	81 :	0.60000	81 :	0.60000
82 :	0.60000	82 :	0.60000	82 :	0.60000
83 :	0.60000	83 :	0.60000	83 :	0.60000
84 :	0.60000	84 :	0.60000	84 :	0.60000
85 :	0.60000	85 :	0.60000	85 :	0.60000
86 :	0.60000	86 :	0.60000	86 :	0.60000
87 :	0.60000	87 :	0.60000	87 :	0.60000
88 :	0.60000	88 :	0.60000	88 :	0.60000
89 :	0.60000	89 :	0.60000	89 :	0.60000
90 :	0.60000	90 :	0.60000	90 :	0.60000
91 :	0.60000	91 :	0.60000	91 :	0.60000
92 :	0.60000	92 :	0.60000	92 :	0.60000
93 :	0.60000	93 :	0.60000	93 :	0.60000
94 :	0.60000	94 :	0.60000	94 :	0.60000
95 :	0.60000	95 :	0.60000	95 :	0.60000

IS YOUR INPUT COMPLETE ? ENTER YES OR NO.

DOES A FILE OF REFERENCE LOAD DATA EXIST? ENTER YES OR NO.

SSN PEAK LD(MW) MIN LD(MW) AVE LD(MW) TOT ENER(MWH) LOAD FACTOR
WI 7.37000E+03 2.77300E+03 5.07329E+03 1.10801E+07 6.88370E-01
SF 7.23500E+03 2.43200E+03 4.08594E+03 8.92370E+06 5.64746E-01
SU 7.80500E+03 2.63200E+03 4.95954E+03 1.08316E+07 6.35431E-01
AU 6.43900E+03 2.57100E+03 4.29443E+03 9.48210E+06 6.66940E-01
THE ANNUAL PEAK OF THE REFERENCE LOAD DATA IS 7.8050E+03 MW

YEAR OF 81
SSN PEAK LD(MW) MIN LD(MW) AVE LD(MW) TOT ENER(MWH) LOAD FACTOR
1 9.05093E+03 2.46071E+03 5.42882D+03 1.18565D+07 5.99808E-01
2 7.30691E+03 2.69436E+03 4.37216D+03 9.54880D+06 5.98360E-01
3 8.86041E+03 2.63889E+03 5.30701D+03 1.15905D+07 5.98957E-01
4 7.66659E+03 2.45871E+03 4.59547D+03 1.01468D+07 5.99416E-01

YEAR OF 82
SSN PEAK LD(MW) MIN LD(MW) AVE LD(MW) TOT ENER(MWH) LOAD FACTOR
1 9.64305E+03 2.62169E+03 5.78398D+03 1.26322D+07 5.99808E-01
2 7.78494E+03 2.87063E+03 4.65820D+03 1.01735D+07 5.98360E-01
3 9.44007E+03 2.81153E+03 5.65420D+03 1.23488D+07 5.98957E-01
4 8.16814E+03 2.61956E+03 4.89611D+03 1.08106D+07 5.99416E-01

YEAR OF 83
SSN PEAK LD(MW) MIN LD(MW) AVE LD(MW) TOT ENER(MWH) LOAD FACTOR
1 1.03621E+04 2.81717E+03 6.21524D+03 1.35741D+07 5.99808E-01
2 8.36541E+03 3.08467E+03 5.00552D+03 1.09321D+07 5.98360E-01
3 1.01439E+04 3.02116E+03 6.07578D+03 1.32695D+07 5.98957E-01
4 8.77717E+03 2.81489E+03 5.26117D+03 1.16167D+07 5.99416E-01

YEAR OF 84

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	1.10810E+04	3.01264E+03	6.64650D+03	1.45160D+07	5.99808E-01
2	8.94585E+03	3.29871E+03	5.35284D+03	1.16906D+07	5.98360E-01
3	1.08478E+04	3.23079E+03	6.49737D+03	1.41903D+07	5.98957E-01
4	9.38620E+03	3.01020E+03	5.62623D+03	1.24227D+07	5.99416E-01

YEAR OF 85

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	1.18423E+04	3.21961E+03	7.10312D+03	1.55132D+07	5.99808E-01
2	9.56045E+03	3.52534E+03	5.72059D+03	1.24938D+07	5.98360E-01
3	1.15931E+04	3.45275E+03	6.94375D+03	1.51452D+07	5.98957E-01
4	1.00311E+04	3.21701E+03	6.01277D+03	1.32762D+07	5.99415E-01

YEAR OF 86

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	1.26882E+04	3.44959E+03	7.61049D+03	1.66213D+07	5.99808E-01
2	1.02433E+04	3.77715E+03	6.12921D+03	1.33862D+07	5.98360E-01
3	1.24211E+04	3.69938E+03	7.43974D+03	1.62484D+07	5.98957E-01
4	1.07476E+04	3.44680E+03	6.44225D+03	1.42245D+07	5.99415E-01

YEAR OF 87

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	1.35341E+04	3.67956E+03	8.11787D+03	1.77294D+07	5.99808E-01
2	1.09262E+04	4.02896E+03	6.53782D+03	1.42786D+07	5.98360E-01
3	1.32492E+04	3.94600E+03	7.93572D+03	1.73316D+07	5.98957E-01
4	1.14641E+04	3.67658E+03	6.87174D+03	1.51728D+07	5.99415E-01

YEAR OF 88

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	1.45492E+04	3.95553E+03	8.72671D+03	1.90591D+07	5.99808E-01
2	1.17457E+04	4.33113E+03	7.02816D+03	1.53495D+07	5.98360E-01
3	1.42429E+04	4.24195E+03	8.53090D+03	1.86315D+07	5.98957E-01
4	1.23239E+04	3.95233E+03	7.38712D+03	1.63108D+07	5.99415E-01

YEAR OF 89

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	1.54796E+04	4.20850E+03	9.28481D+03	2.02780D+07	5.99808E-01
2	1.24969E+04	4.60812E+03	7.47763D+03	1.63312D+07	5.98360E-01
3	1.51538E+04	4.51323E+03	9.07648D+03	1.98230D+07	5.98957E-01
4	1.31120E+04	4.20509E+03	7.85955D+03	1.73539D+07	5.99416E-01

YEAR OF 90

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	1.66639E+04	4.53046E+03	9.99512D+03	2.16293D+07	5.99808E-01
2	1.34529E+04	4.96065E+03	8.04969D+03	1.75805D+07	5.98360E-01
3	1.63131E+04	4.05852E+03	9.77086D+03	2.13396D+07	5.98957E-01
4	1.41115E+04	4.52680E+03	8.46083D+03	1.86815D+07	5.99415E-01

YEAR OF 91

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	1.77635E+04	4.82943E+03	1.06547D+04	2.32699D+07	5.99808E-01
2	1.43407E+04	5.28800E+03	8.58089D+03	1.87407D+07	5.98360E-01
3	1.73896E+04	5.17912E+03	1.04156D+04	2.27477D+07	5.98957E-01
4	1.50466E+04	4.82552E+03	9.01915D+03	1.99143D+07	5.99415E-01

YEAR OF 92

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	1.90323E+04	5.17439E+03	1.14158D+04	2.49320D+07	5.99808E-01
2	1.53650E+04	5.66572E+03	9.19381D+03	2.00793D+07	5.98360E-01
3	1.86317E+04	5.54906E+03	1.11596D+04	2.43726D+07	5.98957E-01
4	1.61213E+04	5.17020E+03	9.66338D+03	2.13367D+07	5.99415E-01

YEAR OF 93

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	2.03011E+04	5.51934E+03	1.21768D+04	2.65941D+07	5.99808E-01
2	1.63894E+04	6.04344E+03	9.80673D+03	2.14179D+07	5.98360E-01
3	1.98738E+04	5.91900E+03	1.19036D+04	2.59974D+07	5.98957E-01
4	1.71961E+04	5.51487E+03	1.03076D+04	2.27592D+07	5.99416E-01

YEAR OF 94

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	2.18237E+04	5.93330E+03	1.30901D+04	2.85887D+07	5.99808E-01
2	1.76186E+04	6.49669E+03	1.05422D+04	2.30242D+07	5.98360E-01
3	2.13644E+04	6.36293E+03	1.27964D+04	2.79472D+07	5.98957E-01
4	1.84858E+04	5.92849E+03	1.10807D+04	2.44661D+07	5.99415E-01

YEAR OF 95

SSN	PEAK LD(MW)	MIN LD(MW)	AVE LD(MW)	TOT ENER(MWH)	LOAD FACTOR
1	2.33463E+04	6.34725E+03	1.40033D+04	3.05833D+07	5.99808E-01
2	1.88478E+04	6.94995E+03	1.12777D+04	2.46306D+07	5.98360E-01
3	2.28549E+04	6.80685E+03	1.36891D+04	2.98970D+07	5.98957E-01
4	1.97755E+04	6.34211E+03	1.18538D+04	2.61731D+07	5.99415E-01

*****END OF INPUT PROCEDURE FOR LOAD DATA****
 LOAD MULTIPLIERS STORED IN UNIT 21.
 LOAD DURATION CURVES STORED IN UNIT 25.
 CUMULANTS STORED IN UNIT 26

READY

exec prep.hydrs.clist

SPECIFY THE CRITICAL LOLP VALUE TO BE USED IN THE
MINIMUM RESERVE CALCULATION IN DAYS PER TEN YEARS

?

1.0
SPECIFY THE LOLP VALUE TO BE USED IN THE MAXIMUM
RESERVE CALCULATION IN DAYS PER TEN YEARS

?

0.2

YEAR	MINIMUM RESERVE (MW)*	MAXIMUM RESERVE (MW)	LOWER BOUND OF OPERATING COST **	ANNUAL PEAK (MW)
1981	2300.07	2530.08	639.714	9050.8
1982	1707.35	1878.75	640.223	8843.0
1983	888.95	2188.85	648.855	10362.1
1984	659.95	845.95	580.760	11081.0
1985	238.68	1438.68	603.452	11842.3
1986	0.0	2072.73	586.030	12686.2
1987	1168.89	2366.89	381.675	13534.1
1988	1451.84	2651.84	310.571	14349.2
1989	1521.37	2721.37	267.413	15478.8
1990	1637.14	3037.14	218.374	16683.9
1991	877.48	885.24	230.411	17763.5
1992	0.0	2218.67	242.718	18032.3
1993	2349.86	3549.86	150.151	20301.1
1994	2527.23	3727.23	137.108	21823.7
1995	2704.57	3904.57	127.873	23346.3

* CRITICAL LOLP ASSIGNED IS 0.000274

** MILLIONS OF 1980 DOLLARS

*** END OF PREP SUBMODULE SIMULATION ***

READY

exec dyno2.clist

CERES OPTIMIZATION BEGINS

CERES HAS A BUILT IN SENSITIVITY ANALYSIS PROCESS.
FOR EACH SENSITIVITY ANALYSIS ITERATION (CALLED HERE
"MAIN ITERATION"), THE USER MUST SPECIFY THE EXPANSION
PLANT CANDIDATE COMBINATION THAT WILL BE USED IN
OPTIMIZATION. SINCE THE CERES OPTIMIZATION ALGORITHM
IS OF ITERATIVE NATURE THE MAXIMUM NUMBER OF "TUNNEL
ITERATIONS" MUST BE PROVIDED.

- FOR EACH MAIN ITERATION THE USER MUST SPECIFY:
1. THE DESIRED EXPANSION PLANT COMBINATION.
 2. THE MAXIMUM NUMBER OF TUNNEL ITERATIONS (MUST BE
BETWEEN 10 AND 50. DEFAULT IS 10)
 3. THE NUMBER OF BEST SOLUTIONS FOR WHICH A REPORT
IS DESIRED (MAXIMUM IS 10 OR THE NUMBER OF TUNNEL
ITERATIONS REQUIRED TO REACH THE OPTIMUM SOLUTION
IF THIS IS LESS THAN 10. DEFAULT IS 1)
THERE IS NO LIMIT ON THE NUMBER OF MAIN ITERATIONS
CERES WILL STOP WHEN THE USER INSTRUCTS TO DO SO.
 4. FIXED CHARGE RATE OR SALVAGE VALUE OPTION FOR COST
CALCULATIONS

ENTER THE NUMBER OF EXPANSION CANDIDATES YOU WILL CONSIDER IN THIS ITERATION
IT SHOULD NOT BE MORE THAN 8. IT IS RECOMMENDED TO BE 4 OR 5

?

3

ENTER THE 3 EXPANSION PLANT ID'S.
THE PLANT ID'S ARE DEFINED FROM THE PLANT ORDER IN THE INPUT MODULE.
ENTER -1 FOR HYDRO EXPANSION, 0 FOR PUMPED STORAGE EXPANSION.
THE NUMBERS ENTERED SHOULD BE SEPARATED BY BLANKS:

?

7 8 3

ENTER THE MAXIMUM NUMBER OF TUNNEL ITERATIONS.
IT SHOULD BE BETWEEN 10 AND 50

?

25

FOR SALVAGE VALUE OPTION ENTER -1.0
FOR FIXED CHARGE RATE ENTER THE FCR VALUE

?

-1.0

THE CRITICAL LOPP VALUE IN PREP MODULE IS 0.000274
IF YOU WANT TO INCREASE IT, ENTER THE NEW NUMBER. OTHERWISE ENTER -1.0

?

0.002

ENTER THE MAXIMUM RESERVE MARGIN IN %
IT SHOULD BE BETWEEN 20 AND 50%. DEFAULT IS 40%

?

35.0

ENTER THE DISCOUNT RATE IN %. DEFAULT IS 15%

?

15.0

PACKING BASE IS

1 16 256

LOWER AND UPPER CHANNEL BOUNDS PER YEAR

1981	0	0	0
	3	3	4
1982	0	0	0
	3	3	4
1983	0	0	0
	3	3	4
1984	1	1	0
	4	4	4
1985	2	1	0
	5	4	4
1986	3	1	0
	6	4	4
1987	3	2	1
	6	5	5
1988	6	3	2
	9	6	6
1989	7	4	3
	10	7	7
1990	8	5	4
	11	8	8
1991	8	5	4
	11	8	8
1992	8	5	4
	11	8	8
1993	11	8	7
	14	11	11
1994	12	9	8
	15	12	12
1995	13	10	9
	16	13	13

YEAR1981

9 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 1

YEAR1982

23 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 2

YEAR1983

44 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 3

YEAR1984

50 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 4

YEAR1985

62 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 5

YEAR1986

67 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 6

YEAR1987

69 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 7

YEAR1988
 57 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 8
 YEAR1989
 55 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 9
 YEAR1990
 48 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 10
 YEAR1991
 74 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 11
 YEAR1992
 48 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 12
 YEAR1993
 71 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 13
 YEAR1994
 78 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 14
 YEAR1995
 57 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

THE FINAL YEAR OBJECTIVE FUNCTION IS: 0.929775E+04
 THE NUMBER OF ITERATION IS: 1

LOLP	OP. COST	CAP. COST	OBJ FUNCTION	PLANT	MIX
0.0	617.392	247.930	741.357	0	0 1
0.0	588.608	232.839	1438.623	0	0 2
0.0	562.818	165.656	2073.325	2	0 2
0.0	548.808	225.581	2712.468	2	2 3
0.0	534.058	146.103	3300.094	4	2 3
0.0	527.886	137.210	3873.717	6	2 3
0.0	512.227	234.522	4456.257	7	2 4
0.0	527.690	167.605	5028.672	7	3 5
0.0	554.537	307.419	5854.834	7	4 7
0.0	582.415	201.187	6257.576	8	5 8
0.0	573.233	150.351	6855.863	11	5 8
0.0	616.193	0.0	7472.055	11	5 8
0.0	576.864	411.518	8080.066	12	8 11
0.0	583.575	238.518	8689.505	15	9 12
0.0	604.551	107.886	9287.750	15	10 13

NEW UPPER BOUND FOR FATHOMING IS 9287.75

*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	15	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	15	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	14	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	14	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	13	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	13	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	12	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	12	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	11	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	11	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	10	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	10	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	9	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	9	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	8	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	7	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	6	*****

THE NEW LOWER AND UPPER BOUNDS ARE:

YEAR 1981	0	0	0
	3	3	4
YEAR 1982	0	0	0
	3	3	4
YEAR 1983	0	0	0
	3	3	4
YEAR 1984	1	0	0
	4	3	4
YEAR 1985	N	0	0
	5	3	4
YEAR 1986	5	0	0
	5	0	4
YEAR 1987	5	0	3
	5	0	4
YEAR 1988	5	0	3
	5	0	4
YEAR 1989	5	2	5
	8	5	10
YEAR 1990	6	3	10
	9	6	10
YEAR 1991	10	3	7
	13	6	11

YEAR 1982	10	3	7
	13	6	11
YEAR 1983	11	6	10
	14	9	14
YEAR 1984	12	7	11
	15	10	15
YEAR 1985	13	8	11
	16	11	15

YEAR1986
 52 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 6
 YEAR1987
 62 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 7
 YEAR1988
 66 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 8
 YEAR1989
 31 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 9
 YEAR1990
 50 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 10
 YEAR1991
 22 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 11
 YEAR1992
 60 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 12
 YEAR1993
 30 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 13
 YEAR1994
 44 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 14
 YEAR1995
 56 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

THE FINAL YEAR OBJECTIVE FUNCTION IS: 0.815411E+04
 THE NUMBER OF ITERATION IS: 2

LOLP	OP. COST	CAP. COST	OBJ FUNCTION	PLANT	MIX
0.0	617.392	247.930	741.357	0	0
0.0	588.608	232.839	1438.623	0	0
0.0	552.919	165.656	2073.325	2	0
0.0	549.976	215.468	2709.489	2	1
0.0	534.832	146.103	3297.892	4	1
0.0	528.475	137.210	3872.104	6	1
0.0	512.615	234.522	4455.074	7	1
0.0	524.325	159.739	5021.892	7	1
0.0	553.416	307.418	5647.133	7	2
0.0	543.535	288.707	6248.398	7	3
0.0	554.811	150.351	6828.262	10	3
0.0	549.956	248.511	7411.344	10	3
0.0	561.562	178.132	7990.715	11	6
0.0	563.878	266.085	8572.324	12	7
0.0	575.593	185.555	9154.109	14	8

NEW UPPER BOUND FOR FATHOMING IS 9154.11

*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	15	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	14	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	13	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	12	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	12	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	11	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	10	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	9	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	8	*****

THE NEW LOWER AND UPPER BOUNDS ARE:

YEAR 1981	0	0	0
	3	3	4
YEAR 1982	0	0	0
	3	3	4
YEAR 1983	0	0	0
	3	3	4
YEAR 1984	1	0	0
	4	3	4
YEAR 1985	2	0	0
	5	3	4
YEAR 1986	3	0	0
	5	3	4
YEAR 1987	5	0	1
	5	3	5
YEAR 1988	6	0	2
	6	3	6

YEAR 1988	5	0	6
	8	3	10
YEAR 1989	6	1	8
	9	4	10
YEAR 1990	8	1	7
	11	4	11
YEAR 1991	8	1	10
	11	4	14
YEAR 1992	8	1	10
	11	4	14
YEAR 1993	9	4	10
	12	7	14
YEAR 1994	10	5	11
	13	8	15
YEAR 1995	13	8	11
	16	9	15

YEAR1988

66 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 8

YEAR1989

33 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 9

YEAR1990

55 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 10

YEAR1991

49 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 11

YEAR1992

34 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 12

YEAR1993

52 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 13

YEAR1994

66 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 14

YEAR1995

37 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

THE FINAL YEAR OBJECTIVE FUNCTION IS: 0.907446E+04

THE NUMBER OF ITERATION IS: 3

LOLP	OP. COST	CAP. COST	OBJ FUNCTION	PLANT	MIX
0.0	617.392	247.930	741.357	0	0 1
0.0	588.608	232.839	1438.623	0	0 2
0.0	562.918	165.856	2073.325	2	0 2
0.0	548.876	215.488	2709.489	2	1 3
0.0	534.832	145.103	3297.882	4	1 3
0.0	528.475	137.210	3872.104	6	1 3
0.0	528.158	170.093	4451.268	6	1 4
0.0	514.531	318.478	5051.008	6	1 6
0.0	533.393	300.032	5854.402	6	1 8
0.0	525.391	281.769	6236.141	6	1 10
0.0	537.051	150.351	6798.248	8	1 10
0.0	533.969	248.511	7365.344	9	1 12
0.0	546.752	178.132	7929.902	10	4 13
0.0	550.533	266.085	8488.168	11	5 15
0.0	572.232	122.021	8074.461	14	6 15

NEW UPPER BOUND FOR FATHOMING IS 8074.46

*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	15	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	14	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	13	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	12	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	11	*****
*****	SOLUTION AGAINST LOWER BOUNDARY IN YEAR	10	*****
*****	SOLUTION AGAINST UPPER BOUNDARY IN YEAR	8	*****

THE NEW LOWER AND UPPER BOUNDS ARE:

YEAR 1981	0	0	0
	3	3	4
YEAR 1982	0	0	0
	3	3	4
YEAR 1983	0	0	0
	3	3	4
YEAR 1984	1	0	0
	4	3	4
YEAR 1985	2	0	0
	5	3	4
YEAR 1986	4	0	0
	7	3	4

YEAR 1987	4	0	1
	7	3	5
YEAR 1988	4	0	5
	7	3	6
YEAR 1989	4	0	6
	7	3	10
YEAR 1990	4	0	6
	7	3	10
YEAR 1991	8	0	7
	11	3	11
YEAR 1992	8	0	10
	11	3	14
YEAR 1993	9	2	10
	12	5	14
YEAR 1994	10	3	11
	13	6	15
YEAR 1995	13	4	11
	16	7	15

YEAR1988

23 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 8

YEAR1989

32 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 8

YEAR1990

62 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 10

YEAR1991

47 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 11

YEAR1992

34 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 12

YEAR1993

57 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 13

YEAR1994

70 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 14

YEAR1995

13 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

THE FINAL YEAR OBJECTIVE FUNCTION IS: 0.905484E+04

THE NUMBER OF ITERATION IS: 4

LOLP	OP. COST	CAP. COST	OBJ FUNCTION	PLANT	MIX
0.0	617.392	247.930	741.357	0	0 1
0.0	588.608	232.839	1438.623	0	0 2
0.0	562.918	165.658	2073.325	2	0 2
0.0	549.976	215.468	2709.489	2	1 3
0.0	534.832	146.103	3297.892	4	1 3
0.0	528.475	137.210	3872.104	6	1 3
0.0	528.158	170.093	4451.289	6	1 4
0.0	514.531	318.478	5051.008	6	1 5
0.0	533.393	300.032	5654.402	6	1 6
0.0	525.391	281.769	6236.141	6	1 10
0.0	537.051	150.351	6798.246	6	1 10
0.0	533.969	248.511	7365.344	9	1 12
0.0	532.403	239.130	7921.656	9	2 14
0.0	546.828	198.008	8481.680	11	3 15
0.0	569.097	122.021	9054.836	14	4 15

NEW UPPER BOUND FOR FATHOMING IS 9054.84

***** SOLUTION AGAINST LOWER BOUNDARY IN YEAR 15 *****
 ***** SOLUTION AGAINST LOWER BOUNDARY IN YEAR 14 *****
 ***** SOLUTION AGAINST LOWER BOUNDARY IN YEAR 13 *****
 ++++++ SOLUTION AGAINST UPPER BOUNDARY IN YEAR 13 ++++++

THE NEW LOWER AND UPPER BOUNDS ARE:

YEAR 1981	0	0	0
	3	3	4
YEAR 1982	0	0	0
	3	3	4
YEAR 1983	0	0	0
	3	3	4
YEAR 1984	1	0	0
	4	3	4
YEAR 1985	2	0	0
	5	3	4
YEAR 1986	4	0	0
	7	3	4

YEAR 1987	4	0	1
	7	3	5
YEAR 1988	4	0	5
	7	3	9
YEAR 1989	4	0	6
	7	3	10
YEAR 1990	4	0	6
	7	3	10
YEAR 1991	7	0	7
	10	3	11
YEAR 1992	7	0	10
	10	3	14
YEAR 1993	7	0	11
	10	3	15
YEAR 1994	10	1	11
	13	4	15
YEAR 1995	13	2	11
	16	5	15

YEAR1993

52 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 13

YEAR1994

73 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 14

YEAR1995

10 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

THE FINAL YEAR OBJECTIVE FUNCTION IS: 0.903874E+04

THE NUMBER OF ITERATION IS: 5

LOLP	OP. COST	CAP. COST	OBJ FUNCTION	PLANT	MIX
0.0	617.382	247.830	741.357	0	0 1
0.0	588.608	232.838	1438.623	0	0 2
0.0	562.818	165.656	2073.325	2	0 2
0.0	549.976	215.468	2708.489	2	1 3
0.0	534.832	146.103	3297.882	4	1 3
0.0	528.475	137.210	3872.104	6	1 3
0.0	528.158	170.083	4451.288	6	1 4
0.0	514.531	319.478	5051.008	6	1 6
0.0	533.393	300.032	5654.402	6	1 6
0.0	525.381	281.768	6236.141	6	1 10
0.0	537.051	150.351	6798.248	6	1 10
0.0	533.868	248.511	7365.344	8	1 12
0.0	530.818	233.384	7918.293	8	1 14
0.0	538.554	234.122	8473.448	12	1 15
0.0	581.233	122.021	9038.742	15	2 15

NEW UPPER BOUND FOR FATHOMING IS 9038.74

***** SOLUTION AGAINST LOWER BOUNDARY IN YEAR 15 *****
 ***** SOLUTION AGAINST LOWER BOUNDARY IN YEAR 14 *****

THE NEW LOWER AND UPPER BOUNDS ARE:

YEAR 1981	0	0	0
	3	3	4
YEAR 1982	0	0	0
	3	3	4
YEAR 1983	0	0	0
	3	3	4
YEAR 1984	1	0	0
	4	3	4
YEAR 1985	2	0	0
	5	3	4
YEAR 1986	4	0	0
	7	3	4
YEAR 1987	4	0	5
	7	3	5
YEAR 1988	4	0	6
	7	3	9
YEAR 1989	4	0	6
	7	3	10
YEAR 1990	4	0	6
	7	3	10
YEAR 1991	7	0	7
	10	3	11
YEAR 1992	7	0	10
	10	3	14
YEAR 1993	7	0	11
	10	3	15
YEAR 1994	10	0	11
	13	3	15
YEAR 1995	13	0	11
	16	3	15

YEAR1994

73 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 14

YEAR1995

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL LOWER BOUND FOR YEAR 1995 IS 12 0 11

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL LOWER BOUND FOR YEAR 1995 IS 11 0 11

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL LOWER BOUND FOR YEAR 1995 IS 10 0 11

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 4 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 5 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 6 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 7 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 8 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 9 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 10 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 11 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 12 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 13 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 14 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

NEW TUNNEL UPPER BOUND FOR YEAR 1995 IS 16 15 15

2 STATES WERE ACCEPTED AFTER FATHOMING IN YEAR 15

ARTIFITIAL BOUNDARIES ELIMINATED IN YEAR 1995

THE FINAL YEAR OBJECTIVE FUNCTION IS: 0.803711E+04

THE NUMBER OF ITERATION IS: 6

LOLP	OP. COST	CAP. COST	OBJ FUNCTION	PLANT	MIX
0.0	617.392	247.930	741.357	0	0 1
0.0	588.608	232.839	1438.623	0	0 2
0.0	562.918	185.656	2073.325	2	0 2
0.0	549.976	215.468	2708.489	2	1 3
0.0	534.832	146.103	3297.892	4	1 3
0.0	528.475	137.210	3872.104	6	1 3
0.0	528.158	170.093	4451.289	6	1 4
0.0	514.531	319.478	5051.008	6	1 6
0.0	533.383	300.032	5654.402	6	1 8
0.0	525.391	281.769	6236.141	6	1 10
0.0	537.051	150.351	6798.246	9	1 10
0.0	533.969	248.511	7365.344	9	1 12
0.0	530.618	233.384	7919.293	9	1 14
0.0	538.554	234.122	8473.449	12	1 15
0.0	559.766	116.953	8037.109	15	1 15
NEW UPPER BOUND FOR FATHOMING IS 8037.11					

THE NUMBER OF MAIN ITERATIONS IS: 1

6 TUNNEL ITERATIONS WERE REQUIRED TO FIND THE OPTIMAL SOLUTION

INPUT THE NUMBER OF BEST SOLUTIONS YOU WANT REPORTED
THIS SHOULD BE LESS OR EQUAL TO 6

?

1

CHARACTERISTICS OF THE OPTIMAL OR SUBOPTIMAL SOLUTION

YEAR	PLANT TYPE			LOLP*	UNSERVED ENERGY (MWH)	TOTAL OPERATING COST*	CAPITAL COST**	SALVAGE VALUE*	OBJECTIVE FUNCTION*
	7	9	3						
1981	0	0	1	0.0	0.15360E+05	517.399	123.965	123.965	741.357
1982	0	0	2	0.0	0.19776E+05	588.615	108.658	124.181	1438.623
1983	2	0	2	0.0	0.26808E+05	562.924	71.784	83.872	2073.325
1984	2	1	3	0.0	0.30080E+05	549.983	66.187	129.281	2709.489
1985	4	1	3	0.0	0.34720E+05	534.839	53.571	92.532	3287.892
1986	6	1	3	0.0	0.33184E+05	528.483	45.737	91.473	3872.104
1987	6	1	4	0.0	0.37056E+05	528.167	51.028	118.065	4451.289
1988	6	1	6	0.0	0.28768E+05	514.539	65.184	234.284	5051.008
1989	6	1	8	0.0	0.24992E+05	533.401	70.007	230.025	5654.402
1990	6	1	10	0.0	0.12000E+05	525.399	56.354	225.415	6236.141
1991	9	1	10	0.0	0.19104E+05	537.059	25.059	125.293	6788.246
1992	9	1	12	0.0	0.0	533.978	33.135	215.376	7365.344
1993	9	1	14	0.0	0.0	530.624	23.338	210.046	7818.293
1994	12	1	15	0.0	0.0	538.564	15.608	218.514	8473.448
1995	15	1	15	0.0	0.0	559.775	3.898	113.054	9037.109

* ZERO MEANS LOLP VALUE SMALLER THAN 0.00001

** MILLIONS OF 1980 DOLLARS

** PLANT COSTS ARE GIVEN AS THE TOTAL WORTH OF THE PLANT AS IT COMES ON LINE LESS THE SALVAGE VALUE AT THE END OF THE STUDY PERIOD. IF THE FIXED CHARGE RATE OPTION IS USED, CAPITAL COSTS REPRESENT THE FIXED CHARGES FOR EACH PLANT.

COST OF OPERATION BY PLANT TYPE

PAGE 1

MILLIONS OF 1980 DOLLARS

PLANT TYPE	1981	1982	1983	1984	1985	1986	1987	1988
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	45.732	87.484	83.666	119.896	114.602	109.492	139.268	196.521
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	64.768	61.507	116.277	164.373	154.435	136.830
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	6.916	5.456	4.428	4.016	10.419
10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	49.242	46.821	43.823	40.800	37.198	33.362	30.664	25.055
12	48.126	45.969	43.485	40.840	37.760	34.364	31.853	26.399
13	65.628	56.053	44.689	37.226	27.579	21.488	19.725	13.466
14	47.027	40.397	29.522	25.351	19.834	15.808	14.752	10.499
15	40.551	33.450	27.094	22.970	18.126	14.969	13.815	9.882

COST OF OPERATION BY PLANT TYPE

PAGE 2

MILLIONS OF 1980 DOLLARS

PLANT TYPE	1981	1982	1983	1984	1985	1986	1987	1988
16	30.897	26.864	20.523	17.773	14.096	11.636	10.918	7.677
17	32.101	27.652	22.872	19.737	15.989	13.168	12.078	8.575
18	47.697	45.072	41.547	38.534	34.484	30.190	27.581	22.413
19	24.535	21.574	17.423	14.937	12.028	9.849	9.140	8.400
20	15.405	13.488	10.698	8.187	7.290	5.985	5.620	3.815
21	21.415	18.832	15.049	13.009	10.464	8.529	7.887	5.147
22	19.297	16.436	12.754	10.665	8.390	6.779	6.048	3.754
23	18.554	15.920	12.615	10.924	8.680	7.049	6.492	3.726
24	12.471	10.862	8.516	7.385	5.860	4.786	4.506	2.719
25	17.389	14.463	11.320	9.415	7.378	5.814	5.046	3.231
26	21.978	18.272	14.261	11.772	8.887	6.608	6.268	4.001
27	12.896	10.695	8.219	6.779	4.834	3.445	3.184	2.228
28	9.382	7.708	5.724	3.904	2.792	2.198	2.122	1.307
29	9.269	7.546	5.053	3.589	2.708	2.035	1.975	1.229
30	5.202	3.464	2.303	1.867	1.208	0.950	0.964	0.646

COST OF OPERATION BY PLANT TYPE

PAGE 3

MILLIONS OF 1980 DOLLARS

PLANT TYPE	1981	1982	1983	1984	1985	1986	1987	1988
31	0.209	0.193	0.136	0.127	0.097	0.083	0.118	0.080
32	0.003	0.006	0.004	0.006	0.005	0.005	0.011	0.008
33	0.007	0.009	0.007	0.008	0.007	0.006	0.014	0.010
34	0.004	0.006	0.004	0.005	0.005	0.004	0.008	0.006
35	0.000	0.000	0.000	0.001	0.001	0.001	0.003	0.002
36	0.000	0.000	0.000	0.000	0.000	0.000	0.001	0.001
37	0.002	0.004	0.003	0.004	0.004	0.004	0.008	0.008
38	0.000	0.001	0.001	0.002	0.002	0.002	0.005	0.004
39	0.000	0.000	0.000	0.000	0.000	0.000	0.001	0.001
40	0.000	0.000	0.000	0.000	0.000	0.000	0.002	0.002
41	21.534	18.725	16.283	14.159	12.312	10.706	8.310	6.095
42	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
43	0.748	0.651	0.566	0.492	0.428	0.372	0.324	0.281
TOTAL	617.398	588.815	562.924	549.883	534.838	528.483	520.167	514.539

COST OF OPERATION BY PLANT TYPE

PAGE 4

MILLIONS OF 1980 DOLLARS

PLANT TYPE	1989	1990	1991	1992	1993	1994	1995	TOTAL
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	244.510	284.481	270.940	302.517	329.524	331.190	320.521	2980.349
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	47.507	45.441	65.188	62.363	58.652	76.078	80.961	1145.389
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	8.866	8.533	8.118	8.722	8.343	7.980	7.633	92.528
10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	27.198	23.359	22.648	18.832	17.237	15.763	15.878	448.979
12	28.198	24.361	23.527	20.682	18.072	16.521	16.526	456.782
13	21.303	16.203	17.742	14.302	11.086	10.169	11.837	388.585
14	16.083	12.216	13.244	10.526	8.616	7.785	8.135	280.796
15	14.254	11.250	11.922	9.880	7.924	7.457	8.485	252.048

COST OF OPERATION BY PLANT TYPE

PAGE 5

MILLIONS OF 1980 DOLLARS

PLANT TYPE	1989	1990	1991	1992	1993	1994	1995	TOTAL
16	11.667	8.203	8.550	7.676	6.539	5.968	7.358	198.446
17	12.183	8.633	8.872	8.233	6.758	6.047	7.177	212.077
18	24.914	21.330	20.569	17.561	15.485	14.384	14.614	416.375
19	8.712	7.643	8.082	6.618	5.420	5.017	6.046	164.424
20	6.309	4.961	5.567	4.307	3.580	3.473	4.372	104.156
21	8.135	6.404	6.384	5.422	4.367	3.853	4.784	139.783
22	6.975	5.209	5.830	4.561	3.561	3.470	4.527	118.259
23	6.654	5.119	5.150	4.367	3.334	3.070	3.891	115.526
24	4.963	3.904	4.012	3.477	2.766	2.514	3.273	82.015
25	5.891	4.443	4.928	3.769	2.977	2.878	3.721	102.785
26	7.640	5.404	6.115	5.158	3.925	3.707	4.690	128.686
27	4.132	2.933	3.583	2.775	2.185	2.137	2.698	72.823
28	2.854	2.214	2.587	2.136	1.665	1.688	2.383	50.684
29	2.672	2.049	2.408	1.864	1.468	1.446	2.085	47.498
30	1.534	1.087	1.300	1.130	0.892	0.802	1.289	24.738

COST OF OPERATION BY PLANT TYPE

PAGE 6

MILLIONS OF 1980 DOLLARS

PLANT TYPE	1989	1990	1991	1992	1993	1994	1995	TOTAL
31	0.364	0.306	0.487	0.447	0.382	0.451	0.774	4.265
32	0.082	0.076	0.154	0.149	0.127	0.164	0.334	1.134
33	0.082	0.075	0.147	0.142	0.130	0.170	0.338	1.153
34	0.058	0.053	0.107	0.104	0.088	0.115	0.242	0.811
35	0.027	0.026	0.060	0.061	0.053	0.072	0.158	0.465
36	0.012	0.012	0.030	0.031	0.028	0.040	0.088	0.244
37	0.059	0.054	0.108	0.104	0.089	0.114	0.226	0.790
38	0.044	0.042	0.090	0.089	0.079	0.105	0.220	0.685
39	0.012	0.012	0.031	0.032	0.030	0.042	0.097	0.258
40	0.024	0.024	0.055	0.056	0.050	0.070	0.154	0.438
41	7.039	6.121	5.323	4.629	4.025	3.500	3.043	144.804
42	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
43	0.245	0.213	0.185	0.161	0.140	0.122	0.106	5.032
TOTAL	533.401	525.399	537.059	533.976	530.624	538.564	558.775	8183.738

CONSTRUCTION COSTS* BY PLANT TYPE
MILLIONS OF 1980 DOLLARS

PAGE 1

PLANT TYPE	1981	1982	1983	1984	1985	1986	1987	1988
1	0.0	0.0	71.784	0.0	53.571	45.737	0.0	0.0
2	0.0	0.0	0.0	4.045	0.0	0.0	0.0	0.0
3	123.965	108.658	0.0	82.142	0.0	0.0	51.028	85.194
TOTAL	123.965	108.658	71.784	86.187	53.571	45.737	51.028	85.194

*PLANT COSTS ARE GIVEN AS THE TOTAL WORTH OF THE PLANT AS IT COMES ON LINE LESS THE SALVAGE VALUE AT THE END OF THE STUDY PERIOD. IF THE FIXED CHARGE RATE OPTION IS USED, CONSTRUCTION COSTS REPRESENT THE FIXED CHARGES FOR EACH PLANT.

CONSTRUCTION COSTS* BY PLANT TYPE
MILLIONS OF 1980 DOLLARS

PAGE 2

PLANT TYPE	1989	1990	1991	1992	1993	1994	1995	TOTAL
1	0.0	0.0	25.059	0.0	0.0	8.302	3.898	208.351
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.045
3	70.007	56.354	0.0	33.135	23.338	7.306	0.0	841.127
TOTAL	70.007	56.354	25.059	33.135	23.338	15.608	3.898	853.524

*PLANT COSTS ARE GIVEN AS THE TOTAL WORTH OF THE PLANT AS IT COMES ON LINE LESS THE SALVAGE VALUE AT THE END OF THE STUDY PERIOD. IF THE FIXED CHARGE RATE OPTION IS USED, CONSTRUCTION COSTS REPRESENT THE FIXED CHARGES FOR EACH PLANT.

IF YOU WANT TO STOP THE MAIN ITERATIONS (SENSITIVITY ANALYSIS), TYPE 1
IF YOU DESIRE TO CONTINUE, TYPE 0

?
1
READY

BEGINNING OF FINAN MODULE

THIS ITERATION INCLUDES 2 UNIT TYPES.
THEY HAVE THE FOLLOWING UNIT CODE NUMBERS:
3 5

1 OPTIMAL SOLUTIONS ARE AVAILABLE FROM DYN0 OUTPUT FILE.

COMPUTATIONS TO BEGIN FOR SOLUTION 1
IF YOU WANT TO STOP ENTER 1
?
0

INPUT SESSION FOR THE FINANCIAL MODULE OF CERES BEGINS.
3 SETS OF DATA SHOULD BE ON FILE.
OTHERWISE ENTER THEM.

1 BEGINNING VALUES OF FINANCIAL DATA
2 YEARLY FINANCIAL DATA
3 RETIREMENT SCHEDULES OF SHORT AND LONG TERM DEBT

THE DATA TO BE ENTERED ARE LISTED BELOW

BEGINNING VALUES OF FINANCIAL DATA

- 01 BEGINNING GROSS PLANT IN SERVICE (MILLION DOLLARS)
- 02 BEGINNING ACCUM BOOK DEPR (MILLION DOLLARS)
- 03 BEGINNING ACCUM TAX DEPR (MILLION DOLLARS)
- 04 BEGINNING ACCUM CWIP SCHD PLANTS (MILLION DOLLARS)
- 05 BEGINNING OUTSYND S TERM DEBT (MILLION DOLLARS)
- 06 BEGINNING OUTSTND L TERM DEBT (MILLION DOLLARS)
- 07 BEGINNING INTEREST CHARGES ON S TERM DEBT (MILLION DOLLARS)
- 08 BEGINNING INTEREST CHARGES ON L TERM DEBT (MILLION DOLLARS)
- 09 BEGINNING S TERM DEBT RETIREMENT (MILLION DOLLARS)
- 10 BEGINNING L TERM DEBT RETIREMENT (MILLION DOLLARS)
- 11 BEGINNING OUTSTND COM STOCK (MILLION DOLLARS)
- 12 BEGINNING OUTSTND PR STOCK (MILLION DOLLARS)
- 13 BEGINNING RET EARNINGS (MILLION DOLLARS)
- 14 BEGINNING AFUDC IN CWIP ACCT (MILLION DOLLARS)
- 15 BEGINNING ACCUM TAX DEFER (MILLION DOLLARS)
- 16 BEGINNING ACCUM INV TAX CR (MILLION DOLLARS)
- 17 BEGINNING AFUDC IN PLANT INS ACCT (MILLION DOLLARS)

YEARLY FINANCIAL DATA

- 01 GROSS SCHEDULED PLANTS IN SERVICE*(MILLION DOLLARS)
- 02 BOOK DEPRECIATION OF SCHEDULED PLANTS (MILLION DOLLARS)
- 03 TAX DEPRECIATION OF SCHEDULED PLANTS (MILLION DOLLARS)
- 04 TOTAL CONST EXPENDITURE FOR SCHD PLANTS (MILLION DOLLARS)
- 05 SHORT TERM DEBT (PERCENT OF NEW CAPITAL)
- 06 UPPER LIMIT ON SHORT TERM DEBT (MILLION DOLLARS)
- 07 LONG TERM DEBT (PERCENT OF NEW CAPITAL)
- 08 UPPER LIMIT ON LONG TERM DEBT (MILLION DOLLARS)
- 09 COMMON STOCK (PERCENT OF NEW CAPITAL)
- 10 UPPER LIMIT ON COMMON STOCK (MILLION DOLLARS)
- 11 PREFERRED STOCK (PERCENT OF NEW CAPITAL)
- 12 UPPER LIMIT ON PREFERRED STOCK (MILLION DOLLARS)
- 13 INTEREST RATE ON SHORT TERM DEBT (PERCENT)
- 14 INTEREST RATE ON LONG TERM DEBT (PERCENT)
- 15 PREFERRED DIVIDEND RATE (PERCENT)
- 16 DIVIDEND PAYOUT RATIO (PERCENT)
- 17 FEDERAL INCOME TAX RATE (PERCENT)
- 18 STATE INCOME TAX RATE (PERCENT)
- 19 OPER REVENUE TAX RATE (PERCENT)
- 20 PROPERTY TAX RATE (PERCENT)
- 21 INVESTMENT TAX CREDIT RATE (PERCENT)
- 22 PERCENT OF CONST EXPND ELIGIBLE FOR INV TAX CR
- 23 PERCENT OF CWIP ALLOWED IN RATEBASE
- 24 AFUDC RATE (PERCENT)
- 25 ALLOWED RATE OF RETURN ON RATEBASE (PERCENT)
- 26 AFUDC FOR SCHD PLANTS IN SERVICE (MILLION DOLLARS)
- 27 RATIO OF CURRENT YR WK CAPTL/PREV YR OF REV (FRACTION)
- 28 AMOUNT OF BEGINNING DEBT TO BE REFUNDED (MILLION DOLLARS)

RETIREMENT SCHEDULES FOR SHORT AND LONG TERM DEBT

- 1 YEAR OF MATURITY FOR DEBT ISSUED IN EACH YEAR
- 2 PERCENT TO BE RETIRED EACH YEAR FOR EACH DEBT ISSUE

DO YOU HAVE A FILE ALREADY FOR FINANCIAL DATA?
ENTER YES OR NO.

?
y

ENTER CONSTRUCTION TIME IN YEARS (2 DIGITS)
FOR A SINGLE UNIT WITH UNIT CODE 3

55

THE TOTAL CONSTRUCTION COST FOR PLANT 3 IS 264,000 MILLION DOLLARS.
THE CONSTRUCTION TIME IS 5 YEARS.
ENTER CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS)
DURING EACH YEAR OF CONSTRUCTION.

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 1?

50.0

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 2?

50.0

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 3?

50.0

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 4?

50.0

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 5?

50.0

DO YOU WANT TO REINPUT CONSTRUCTION EXPENDITURE DATA
JUST ENTERRED FOR UNIT CODE 3 ?
ENTER YES OR NO.

n

ENTER CONSTRUCTION TIME IN YEARS (2 DIGITS)
FOR A SINGLE UNIT WITH UNIT CODE 5

05

THE TOTAL CONSTRUCTION COST FOR PLANT 5 IS 48,000 MILLION DOLLARS.
THE CONSTRUCTION TIME IS 5 YEARS.
ENTER CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS)
DURING EACH YEAR OF CONSTRUCTION.

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 1?

10.0

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 2?

10.0

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 3?

10.0

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 4?

10.0

CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS) FOR YEAR 5?

8.0

DO YOU WANT TO REINPUT CONSTRUCTION EXPENDITURE DATA
JUST ENTERRED FOR UNIT CODE 5?
ENTER YES OR NO.

?
NO

ENTER YES IF YOU WANT TO CHANGE,
ANY CONSTRUCTION EXPENDITURE DATA,
FOR EXPANSION UNITS.

?
N

ANY CHANGES IN FINANCIAL DATA?
ENTER YES OR NO.

?

9

ENTER 1 TO CHANGE BEGINNING VALUES OF DATA.
ENTER 2 TO CHANGE YEARLY FINANCIAL DATA.
ENTER 3 TO CHANGE DEBT RETIREMENT SCHEDULES.
ENTER 4 TO CHANGE CONSTRUCTION EXPENDITURE
DATA OF EXPANSION UNITS.
ENTER 5 TO REINPUT ALL OF THE ABOVE DATA.

?

1

DO YOU WANT TO CHANGE ALL THE BEGINNING DATA?
ENTER YES OR NO.

?

n

WHICH DATA YOU WANT TO CHANGE?
ENTER 2 DIGIT DATA CODE, FOR HELP, ENTER 99.

??

99
01 BEGINNING GROSS PLANT IN SERVICE (MILLION DOLLARS)
02 BEGINNING ACCUM BOOK DEPR (MILLION DOLLARS)
03 BEGINNING ACCUM TAX DEPR (MILLION DOLLARS)
04 BEGINNING ACCUM CWIP SCHD PLANTS (MILLION DOLLARS)
05 BEGINNING OUTSTND S TERM DEBT (MILLION DOLLARS)
06 BEGINNING OUTSTND L TERM DEBT (MILLION DOLLARS)
07 BEGINNING INTEREST CHARGES ON S TERM DEBT. (MILLION DOLLARS)
08 BEGINNING INTEREST CHARGES ON L TERM DEBT (MILLION DOLLARS)
09 BEGINNING S TERM DEBT RETIREMENT (MILLION DOLLARS)
10 BEGINNING L TERM DEBT RETIREMENT (MILLION DOLLARS)
11 BEGINNING OUTSTND COM STOCK (MILLION DOLLARS)
12 BEGINNING OUTSTND PR STOCK (MILLION DOLLARS)
13 BEGINNING RET EARNINGS (MILLION DOLLARS)
14 BEGINNING AFUDC IN CWIP ACCT (MILLION DOLLARS)
15 BEGINNING ACCUM TAX DEFER (MILLION DOLLARS)
16 BEGINNING ACCUM INV TAX CR (MILLION DOLLARS)
17 BEGINNING AFUDC IN PLANT INS ACCT (MILLION DOLLARS)

WHICH DATA YOU WANT TO CHANGE?
ENTER 2 DIGIT DATA CODE, FOR HELP, ENTER 99.

--
17
BEGINNING AFUDC IN PLANT IN SERVICE ACCOUNT (MILLION DOLLARS)?
PREVIOUS VALUE= 100.0000

95.0

ANY MORE CHANGES IN DATA?
ENTER YES OR NO.
?
no

ANY CHANGES IN FINANCIAL DATA?
ENTER YES OR NO.
?
y

ENTER 1 TO CHANGE BEGINNING VALUES OF DATA.
ENTER 2 TO CHANGE YEARLY FINANCIAL DATA.
ENTER 3 TO CHANGE DEBT RETIREMENT SCHEDULES.
ENTER 4 TO CHANGE CONSTRUCTION EXPENDITURE
DATA OF EXPANSION UNITS.
ENTER 5 TO REINPUT ALL OF THE ABOVE DATA.
?
2

ENTER 1 TO REINPUT ALL DATA FOR ALL YEARS.
ENTER 2 TO CHANGE DATA FOR A SINGLE YEAR.
2

ENTER YEAR(4 DIGITS) OF DATA CHANGE
?
1985

DO YOU WANT TO CHANGE ALL THE DATA FOR YEAR 1985?
ENTER YES OR NO.
?
n

WHICH DATA YOU WANT TO CHANGE?
ENTER 2 DIGIT DATA CODE.FOR HELP,ENTER 99.

- 99
01 GROSS SCHEDULED PLANTS IN SERVICE (MILLION DOLLARS)
02 BOOK DEPRECIATION OF SCHEDULED PLANTS (MILLION DOLLARS)
03 TAX DEPRECIATION OF SCHEDULED PLANTS (MILLION DOLLARS)
04 TOTAL CONSTR EXPENDITURE FOR SCHD PLANTS (MILLION DOLLARS)
05 SHORT TERM DEBT (PERCENT OF NEW CAPITAL)
06 UPPER LIMIT ON SHORT TERM DEBT (MILLION DOLLARS)
07 LONG TERM DEBT (PERCENT OF NEW CAPITAL)
08 UPPER LIMIT ON LONG TERM DEBT (MILLION DOLLARS)
09 COMMON STOCK (PERCENT OF NEW CAPITAL)
10 UPPER LIMIT ON COMMON STOCK (MILLION DOLLARS)
11 PREFERRED STOCK (PERCENT OF NEW CAPITAL)
12 UPPER LIMIT ON PREFERRED STOCK (MILLION DOLLARS)
13 INTEREST RATE ON SHORT TERM DEBT (PERCENT)
14 INTEREST RATE ON LONG TERM DEBT (PERCENT)
15 PREFERRED DIVIDEND RATE (PERCENT)
16 DIVIDEND PAYOUT RATIO (PERCENT)
17 FEDERAL INCOME TAX RATE (PERCENT)
18 STATE INCOME TAX RATE (PERCENT)
19 OPER REVENUE TAX RATE (PERCENT)
20 PROPERTY TAX RATE (PERCENT)
21 INVESTMENT TAX CREDIT RATE (PERCENT)
22 PERCENT OF CONSTR EXPND ELIGIBLE FOR INV TAX CR
23 PERCENT OF CWIP ALLOWED IN RATEBASE
24 AFUDC RATE (PERCENT)
25 ALLOWED RATE OF RETURN ON RATEBASE (PERCENT)
26 AFUDC FOR SCHD PLANTS IN SERVICE (MILLION DOLLARS)
27 RATIO OF CURRENT YR WK CAPTL/PREV YR OF REV (FRACTION)
28 AMOUNT OF BEGINNING DEBT TO BE REFUNDED (MILLION DOLLARS)

WHICH DATA YOU WANT TO CHANGE?
ENTER 2 DIGIT DATA CODE.FOR HELP,ENTER 99.

- 14
INTEREST RATE ON LONG TERM DEBT (PERCENT)?
PREVIOUS VALUE= 8,0000

--
7.5

ANY MORE CHANGES IN FINANCIAL DATA FOR 1985
ENTER YES OR NO.

?
0

DO YOU WANT TO CHANGE ANY YEARLY FINANCIAL DATA?
ENTER YES OR NO.
0

ANY CHANGES IN FINANCIAL DATA?
ENTER YES OR NO.

?
0

IS YOUR INPUT FILE FOR FINANCIAL DATA COMPLETE?
ENTER YES OR NO.

?
0

END OF INPUT PROCEDURE. INPUT DATA STORED ON UNIT 18.

ENTER ACCOUNTING METHOD FOR TREATING DEFERRED TAXES
1 : NORMALIZED ACCOUNTING
0 : FLOW THROUGH ACCOUNTING
?
1.

ENTER 1 IF AFUDC SHOULD BE TREATED AS INCOME
ENTER 0 OTHERWISE

?
1

ACCELERATED DEPRECIATION METHOD FOR TAX PURPOSES?

ENTER ACCORDING TO THE FOLLOWING.
2: SUM OF YEARS DIGITS
3: DOUBLE DECLINING
3

XYZ ELECTRIC COMPANY
BALANCE SHEET

MILLIONS OF CONSTANT DOLLARS

DECEMBER 31	1981	1982	1983	1984	1985	1986	1987	1988
ASSETS								
GR PLANT INS (INCL AFUDC)								
- ACCUM DEPRECIATION	1952.28	2216.28	2480.28	2797.08	3113.88	3430.68	3747.48	4117.07
NET PLANT VALUE	111.80	182.40	261.81	351.61	451.81	562.41	683.41	816.41
+ WIP	1840.48	2033.88	2218.47	2445.47	2662.07	2868.27	3064.06	3300.66
NET UTILITY PLANT	950.31	1058.50	1189.09	1266.48	1365.47	1510.06	1702.05	1859.04
NET WORKING CAPITAL	2790.79	3092.38	3407.56	3711.95	4027.54	4378.32	4766.11	5159.70
	371.69	173.66	134.54	140.22	196.19	152.32	158.52	164.97
TOTAL ASSETS	3162.48	3266.04	3542.11	3852.17	4173.73	4530.64	4924.63	5324.66
LIABILITIES								
COMMON STOCK	1105.67	1105.67	1172.06	1249.43	1328.97	1420.48	1524.87	1629.43
+ RETAINED EARNINGS	364.69	431.35	501.85	575.94	653.76	735.28	820.49	909.81
TOTAL COMMON EQUITY	1470.36	1537.01	1673.92	1825.37	1982.73	2155.75	2345.36	2539.25
+ PREFERRED STOCK	213.21	213.21	221.51	231.18	241.12	252.56	265.61	278.68
+ LONG TERM DEBT	1132.08	1132.08	1215.08	1311.79	1411.21	1525.60	1656.09	1786.79
TOTAL CAPITAL	2815.65	2882.30	3110.51	3368.34	3635.06	3933.91	4267.06	4604.71
+ SHORT TERM DEBT	163.21	163.21	171.51	181.18	191.12	202.56	215.61	228.68
+ DEFERRED INCOME TAXES	183.62	220.52	260.09	302.65	347.55	394.18	441.96	491.27
TOTAL LIABILITIES	3162.48	3266.04	3542.10	3852.17	4173.73	4530.64	4924.63	5324.66

XYZ ELECTRIC COMPANY
BALANCE SHEET

MILLIONS OF CONSTANT DOLLARS

DECEMBER 31	1989	1990	1991	1992	1993	1994	1995
<hr/>							
ASSETS							
GR PLANT INS (INCL AFUDC)	4433.87	4856.27	5384.27	5912.27	6334.67	6440.27	6440.27
- ACCUM DEPRECIATION	959.82	1116.82	1291.42	1483.62	1689.42	1898.43	2107.43
NET PLANT VALUE	3474.06	3739.46	4092.85	4428.65	4645.25	4541.85	4332.84
+CWIP	2031.03	2012.63	1755.82	1349.61	1057.30	1047.70	1047.70
NET UTILITY PLANT	5505.09	5752.08	5848.67	5778.25	5702.54	5589.55	5380.54
NET WORKING CAPITAL	171.23	177.30	232.20	456.24	798.21	1147.50	1480.22
TOTAL ASSETS	5676.31	5929.38	6080.87	6234.49	6500.75	6737.05	6860.76
<hr/>							
LIABILITIES							
COMMON STOCK	1713.53	1756.39	1756.39	1756.39	1756.39	1756.39	1756.39
+ RETAINED EARNINGS	1001.89	1097.15	1195.04	1293.07	1390.72	1483.92	1569.44
TOTAL COMMON EQUITY	2715.42	2853.54	2951.43	3049.46	3147.11	3240.31	3325.83
+ PREFERRED STOCK	289.19	294.55	294.55	294.55	294.55	294.55	294.55
+ LONG TERM DEBT	1891.92	1945.48	1945.48	1945.48	1945.48	1945.48	1945.48
TOTAL CAPITAL	4896.53	5093.57	5191.46	5289.49	5387.14	5480.34	5565.86
+ SHORT TERM DEBT	239.19	244.55	244.55	244.55	244.55	244.55	244.55
+ DEFERRED INCOME TAXES	540.59	591.26	644.85	700.45	754.96	802.06	840.25
TOTAL LIABILITIES	5676.30	5929.37	6080.86	6234.48	6386.65	6526.94	6650.66

XYZ ELECTRIC COMPANY
INCOME STATEMENT

MILLIONS OF CONSTANT DOLLARS

DECEMBER 31	1981	1982	1983	1984	1985	1986	1987	1988
OPERATING REVENUES	1263.20	1280.73	1322.69	1367.30	1416.14	1466.15	1512.95	1583.00
- OPER AND MAINT COSTS	22.68	19.78	17.26	15.08	13.18	11.53	10.09	8.85
- FUEL COSTS	594.72	568.84	556.88	532.45	516.03	508.60	503.00	501.91
TOTAL OPERATING EXPENSES	617.39	588.62	574.14	547.53	529.21	520.13	513.09	510.76
- DEPRECIATION	61.80	70.60	79.40	89.80	100.20	110.60	121.00	133.00
- OPER REV TAX	37.90	38.42	39.68	41.02	42.48	43.98	45.39	47.49
- PROPERTY TAX	29.28	33.24	37.20	41.96	46.71	51.46	56.21	61.76
- INCOME TAX PAID	184.73	184.66	195.28	212.98	227.09	233.67	237.38	252.15
- DEFERRED INCOME TAX	33.62	36.90	39.57	42.56	44.90	46.63	47.79	49.31
TOTAL EXPENSES	964.73	952.45	965.28	975.84	990.59	1006.47	1020.86	1054.46
OPERATING INCOME	276.07	305.08	332.77	366.82	399.31	430.24	459.61	495.10
+ AFUDC	79.59	82.19	86.59	86.19	87.79	93.39	102.79	108.59
INCOME BEFORE INTEREST	355.66	387.27	419.36	453.01	487.10	523.63	562.40	603.69
- INTRST ON SHORT TERM DEBT	16.32	32.64	33.47	35.27	37.23	39.37	41.82	44.43
- INTRST ON LONG TERM DEBT	14.57	20.02	31.21	44.21	56.62	71.44	87.94	104.79
NET INCOME	324.77	334.61	354.68	373.53	393.25	412.82	432.64	454.47
- PREFERRED DIVIDENDS	1.32	1.32	2.15	3.12	4.11	5.26	6.56	7.87
EARNINGS AVAIL TO COMMON	323.45	333.28	352.53	370.41	389.14	407.56	426.08	446.60
- COMMON DIVIDENDS	258.76	266.63	282.02	296.33	311.31	326.05	340.86	357.28
RETAINED EARNINGS	64.69	66.66	70.51	74.08	77.83	81.51	85.22	89.32

XYZ ELECTRIC COMPANY
INCOME STATEMENT

MILLIONS OF CONSTANT DOLLARS

DECEMBER 31	1989	1990	1991	1992	1993	1994	1995
OPERATING REVENUES							
- OPER AND MAINT COSTS	1640.79	1748.18	1891.89	2039.34	2144.18	2141.96	2101.01
- FUEL COSTS	7.77	6.83	6.02	5.31	4.70	4.17	3.71
	502.45	507.28	504.00	507.54	515.07	541.48	571.13
TOTAL OPERATING EXPENSES	510.22	514.11	510.01	512.85	519.77	545.65	574.84
- DEPRECIATION	143.40	157.00	174.60	192.20	205.80	209.00	209.00
- OPER REV TAX	49.22	52.45	56.76	61.18	64.33	64.26	63.03
- PROPERTY TAX	66.51	72.84	80.76	88.68	95.02	96.60	96.60
- INCOME TAX PAID	271.10	316.19	386.23	457.00	506.69	498.07	469.42
- DEFERRED INCOME TAX	49.32	50.67	53.59	55.60	54.51	47.10	38.20
TOTAL EXPENSES	1089.77	1163.27	1261.96	1367.52	1446.11	1460.68	1431.09
OPERATING INCOME	521.11	560.92	613.93	664.30	696.79	681.28	649.93
+ AFUDC	114.79	103.99	71.19	27.79	0.0	0.0	0.0
INCOME BEFORE INTEREST	635.90	664.91	685.12	692.09	696.79	681.28	649.93
- INTRST ON SHORT TERM DEBT	46.79	48.37	48.91	48.91	48.91	48.91	48.91
- INTRST ON LONG TERM DEBT	119.82	130.77	137.28	143.60	150.14	156.92	163.98
NET INCOME	469.29	485.77	498.93	499.58	497.74	475.44	437.04
- PREFERRED DIVIDENDS	8.92	9.45	9.45	9.45	9.45	9.45	9.45
EARNINGS AVAIL TO COMMON	460.38	476.31	489.48	490.13	488.28	465.99	427.58
- COMMON DIVIDENDS	368.30	381.05	391.58	392.10	390.63	372.79	342.07
RETAINED EARNINGS	92.07	95.26	97.90	98.03	97.66	93.20	85.52

XYZ ELECTRIC COMPANY
SOURCES AND USES OF FUNDS

MILLIONS OF CONSTANT DOLLARS

DECEMBER 31	1981	1982	1983	1984	1985	1986	1987	1988
SOURCES								
INTERNAL								
NET INCOME								
DEPRECIATION	324.77	334.61	354.68	373.53	393.25	412.82	432.64	454.47
DEFERRED INCOME TAXES	61.80	70.60	79.40	89.80	100.20	110.60	121.00	133.00
TOTAL INTERNAL	420.20	442.11	473.65	505.89	538.35	570.05	601.43	636.78
EXTERNAL								
COMMON STOCK								
COMMON STOCK	105.67	0.0	66.40	77.37	79.53	91.51	104.40	104.56
PREFERRED STOCK	13.21	0.0	8.30	9.67	9.94	11.44	13.05	13.07
LONG TERM DEBT	182.08	68.21	158.03	187.54	209.00	244.87	285.47	314.22
SHORT TERM DEBT	163.21	163.21	171.51	181.18	191.12	202.56	215.61	228.68
TOTAL EXTERNAL	464.17	231.42	404.23	455.76	489.60	550.38	618.52	660.53
TOTAL SOURCES	884.36	673.53	877.88	961.65	1027.95	1120.43	1219.95	1297.32
USES								
TOT CONS EXPN (INCL AFUDC)								
PREFERRED DIVIDENDS	359.59	372.19	394.59	394.19	415.79	461.39	508.79	526.59
COMMON DIVIDENDS	1.32	1.32	2.15	3.12	4.11	5.26	6.56	7.87
DEBT RETIREMENT	258.76	266.63	282.02	296.33	311.31	326.05	340.86	357.28
NET INCR WORKING CAPITAL	200.00	231.42	238.24	262.34	290.76	321.61	357.53	399.13
NET INCR WORKING CAPITAL	64.69	-198.03	-39.12	5.67	5.98	6.13	6.20	6.45
TOTAL USES	884.36	673.53	877.88	961.65	1027.95	1120.43	1219.95	1297.32

XYZ ELECTRIC COMPANY
SOURCES AND USES OF FUNDS

MILLIONS OF CONSTANT DOLLARS

DECEMBER 31	1989	1990	1991	1992	1993	1994	1995
-------------	------	------	------	------	------	------	------

SOURCES

INTERNAL

NET INCOME	469.29	485.77	498.93	499.58	497.74	475.44	437.04
DEPRECIATION	143.40	157.00	174.60	192.20	205.80	209.00	209.00
DEFERRED INCOME TAXES	49.32	50.67	53.59	55.60	54.51	47.10	38.20
TOTAL INTERNAL	662.02	693.44	727.13	747.38	758.05	731.54	684.24

313

EXTERNAL

COMMON STOCK	84.10	42.85	0.0	0.0	0.0	0.0	0.0
PREFERRED STOCK	10.51	5.36	0.0	0.0	0.0	0.0	0.0
LONG TERM DEBT	320.07	300.52	277.00	304.70	335.17	368.69	405.56
SHORT TERM DEBT	239.19	244.55	244.55	244.55	244.55	244.55	244.55
TOTAL EXTERNAL	653.87	593.28	521.55	549.25	579.72	613.24	650.11

TOTAL SOURCES

1315.89	1286.71	1248.68	1296.63	1337.77	1344.78	1334.34
---------	---------	---------	---------	---------	---------	---------

USES

TOT CONS EXPN (INCL AFUDC)	488.79	403.99	271.19	121.79	16.00	0.0	0.0
PREFERRED DIVIDENDS	8.92	9.45	9.45	9.45	9.45	9.45	9.45
COMMON DIVIDENDS	368.30	381.05	391.58	392.10	390.63	372.79	342.07
DEBT RETIREMENT	443.62	486.14	521.55	549.25	579.72	613.24	650.11
NET INCR WORKING CAPITAL	6.26	6.08	54.90	224.04	341.97	349.29	332.72

TOTAL USES

1315.89	1286.71	1248.68	1296.63	1337.77	1344.78	1334.34
---------	---------	---------	---------	---------	---------	---------

XYZ ELECTRIC COMPANY
RATIO ANALYSIS

DECEMBER 31	1981	1982	1983	1984	1985	1986	1987	1988
DEBT TO ASSET	0.41	0.40	0.39	0.39	0.38	0.38	0.38	0.38
DEBT TO EQUITY	0.77	0.74	0.73	0.73	0.72	0.72	0.72	0.72
INTEREST COVERAGE RATIO	11.51	7.35	6.48	5.70	5.19	4.73	4.33	4.05

XYZ ELECTRIC COMPANY
RATIO ANALYSIS

DECEMBER 31	1989	1990	1991	1992	1993	1994	1995
DEBT TO ASSET	0.38	0.37	0.36	0.35	0.34	0.33	0.32
DEBT TO EQUITY	0.71	0.70	0.67	0.65	0.64	0.62	0.60
INTEREST COVERAGE RATIO	3.82	3.71	3.68	3.60	3.50	3.31	3.05

END OF PROGRAM RUN WITH FINAN MODULE OF CERBS.
READY

REFERENCES

1. Jenkins, R. T., and Joy, D. S. Wien Automatic System Planning Package (WASP)--An Electric Utility Optimal Generation Expansion Planning Computer Code. ORNL-4945. Oak Ridge, Tenn: Oak Ridge National Laboratory, 1974.
2. Joy, D. S., and Jenkins, R. T. A Probabilistic Model for Estimating the Operation Cost of an Electric Power Generating System. ORNL-TM 3549. Oak Ridge, Tenn: Oak Ridge National Laboratory, 1974.
3. WASP-II (version 76-1), User's Manual. Vienna: International Atomic Energy Agency, 1976.
4. WASP-2, Generation System Expansion Model. Widsor, Ct.: Power Systems, Combustion Engineering, Inc., 1975.
5. Lubbers, R. H., Tzemos, S., and Nakamura, S. Sensitivity Analysis for Ten-Year Expansion of Generation Facilities of the CAPCO Group Using the WASP Program. Columbus, Ohio: The Ohio State University, Department of Mechanical Engineering, 1977.
6. Lubbers, R. H., Tzemos, S., and Nakamura, S. "An Investigation of Some Optimization Parameters." Proceedings of the 2nd WASP Conference, August 18-19, 1977. Columbus, Ohio: The Ohio State University, Department of Mechanical Engineering, 1977.
7. Nakamura, S., and Brown, J. "Accuracy of Probabilistic Simulation Using Fourier Expansions." Proceedings of the 2nd WASP Conference, August 18 19, 1977. Columbus, Ohio: The Ohio State University, Department of Mechanical Engineering, 1977.
8. Nakamura, S., and Tzemos, S. Modifications to the Wien Automatic System Planning Package (WASP) for Improving Power Pooling Analysis. Columbus, Ohio: The National Regulatory Research Institute, June 1979.
9. Stremel, J. P., Jenkins, R. T., Babb, R. A., and Bayless, W. D. "Production Costing Using the Cumulant Method of Representing the Equivalent Load Curve." IEEE-F79, 674-3.
10. Cramer, H. Mathematical Methods of Statistics. Princeton, N.J.: Princeton University Press, 1966.
11. Anderson, B. A. "Modifications to Augment Private Utility Use of the Wien Automatic System Planning Package." Proceedings of the 2nd WASP Conference, August 18-19, 1977. Columbus, Ohio: The Ohio State University, Department of Mechanical Engineering, 1977.

12. Regulatory Analysis Model, RAM, Descriptive Documentation,
Wellesley Hills, Mass.: Temple, Baker & Sloane, Inc., 1977.

13. Tzemos, S. Evaluation of Probabilistic Simulation Methods and
Development of Optimization Techniques for Capacity Expansion Planning
of Electric Power Generation Systems, Ph.D. Dissertation, Nuclear
Engineering Program, The Ohio State University, 1981.

Acknowledgements

Authors are grateful to Mr. M.R. Laux of Florida Public Service Commission and Mr. W. Pollard of National Regulatory Research Institute for their valuable discussions and comments on the FINAN module of the CERES program.

The administrative supports for developing the CERES program provided by Drs. D.N. Jones, K.A. Kelly, and R.W. Lawton, respectively, Director and Associate Directors of National Regulatory Research Institute, and their critical review of the present report are greatly appreciated.

