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

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

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 runnning 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

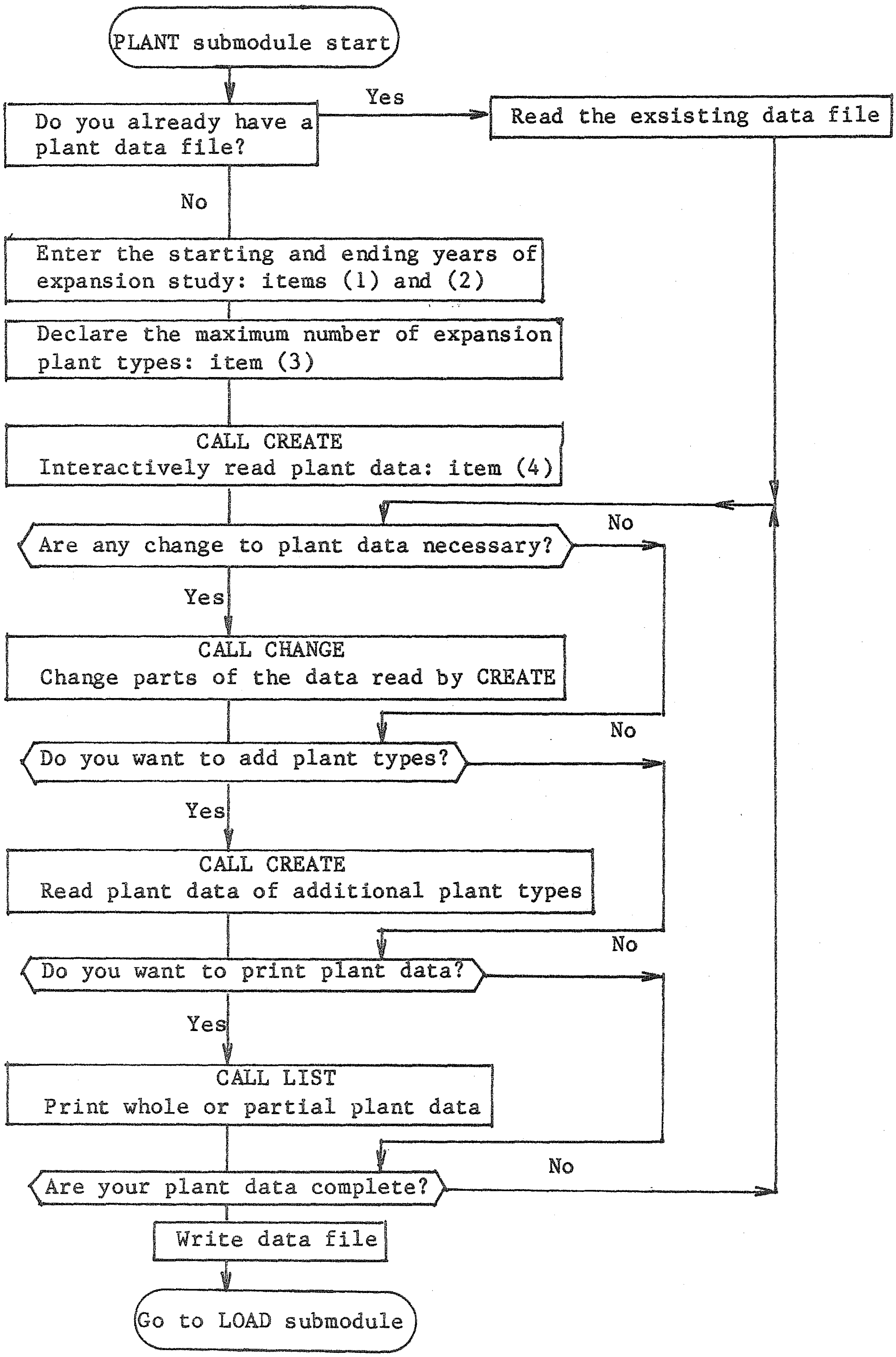


Figure 2-1 PLANT submodule flowchart

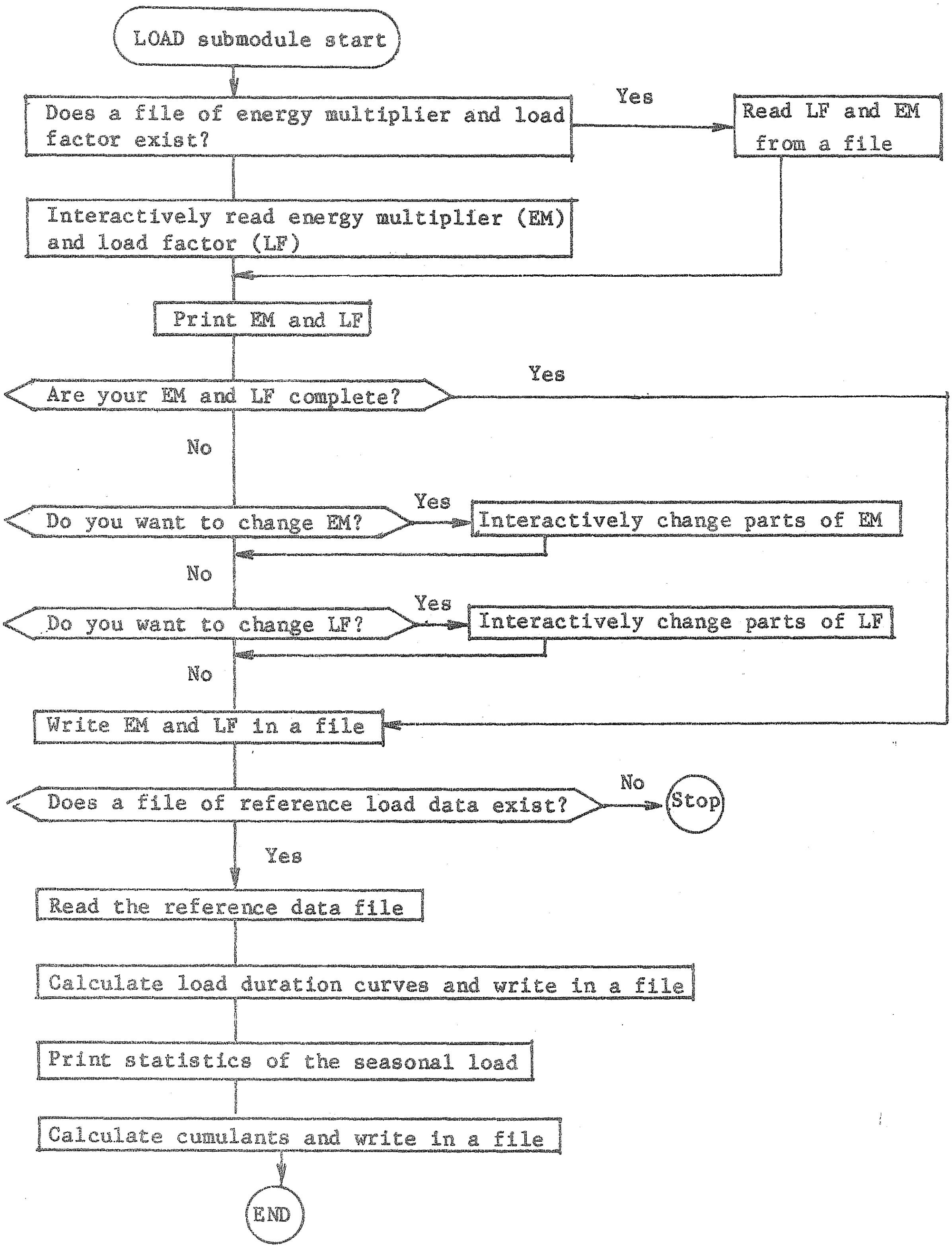


Figure 2-2 LOAD submodule flowchart

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

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

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

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

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

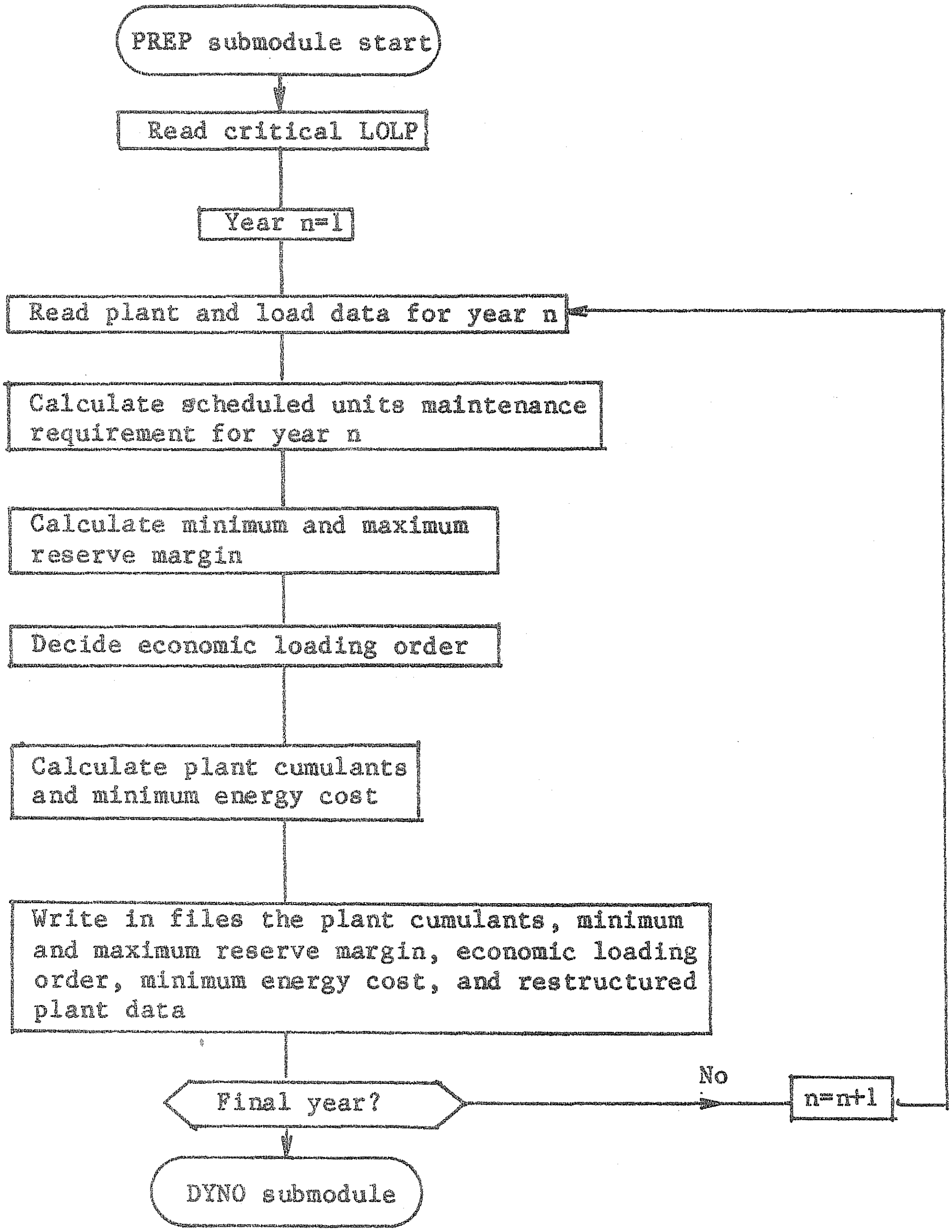


Figure 2-3 PREP submodule flowchart

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

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

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

$$RMIN_n = CS_n + CA_n - P_n \tag{4}$$

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

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

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

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

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,...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,

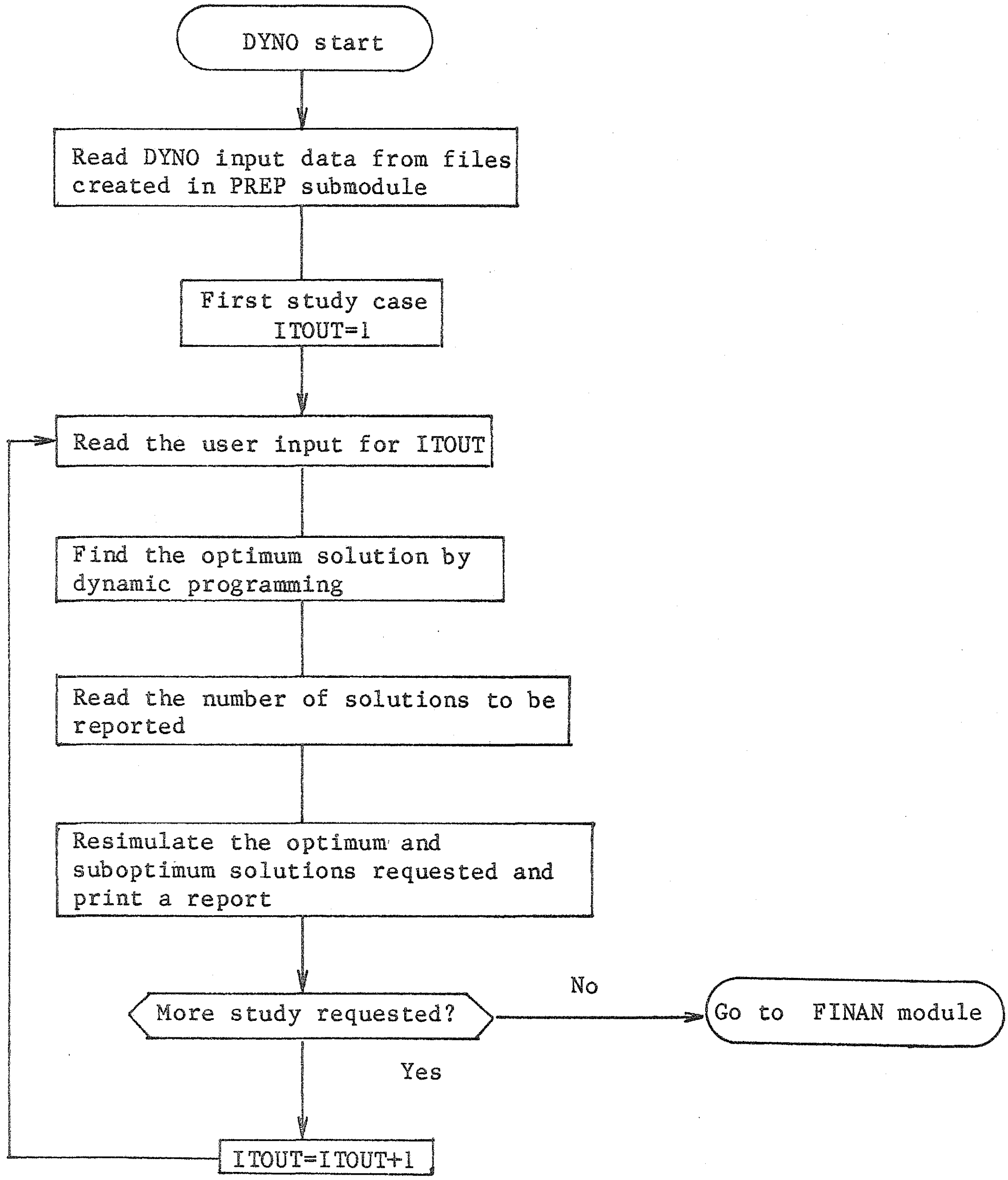


Figure 2-4 DYNO submodule flowchart

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 levelized 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) LOLP < 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 $OBJ_{g,n}$. Then, for the trajectories passing the state g, a lower bound for the objective function at the final stage N may be written as

$$LB_{g,n} = OBJ_{g,n} + \sum_{i=n+1}^{N} LOBC_i + CC_{g,n} \tag{7}$$

where

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

$OBJ_{g,n}$: 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

$CC_{g,n}$: 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,..., N

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

$LOBC_i$ is calculated in the PREP module. $CC_{g,n}$ 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

$$LB_{g,n} > OBJ_L \tag{8}$$

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(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

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

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

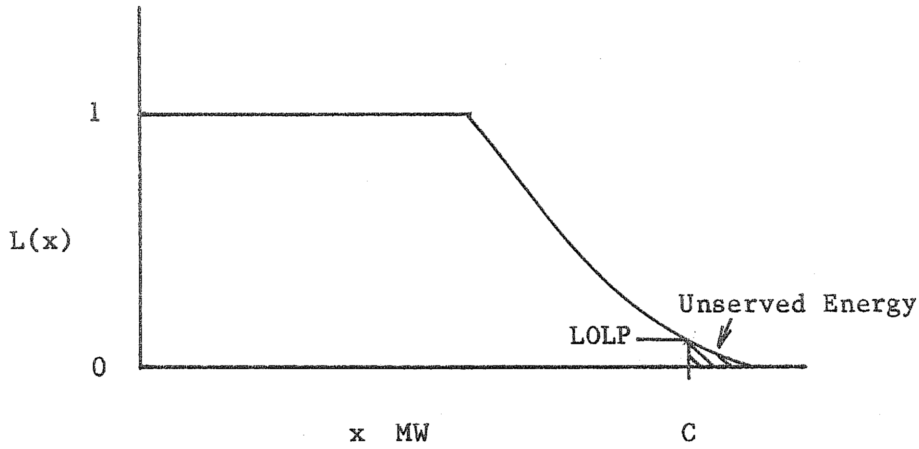
$$P(x<x'<x+dx) = f(x)dx \qquad (2)$$

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.

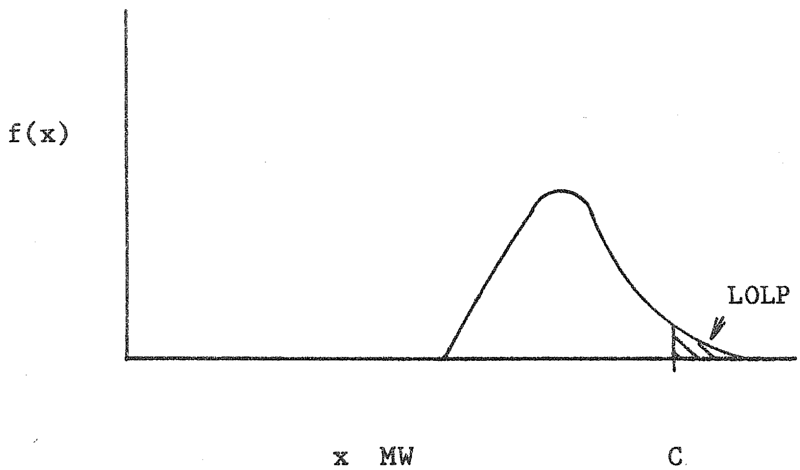
The hatched area under the curve of $L(x)$ in figure A-1(a) times T is the expected amount of demand that cannot be fulfilled by the system, and is called "unserved energy."

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

$$T \int_{C}^{C+G} L(x)\, dx \qquad (3)$$



(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

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

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

$$f_e(x) = (1-q)f(x) + qf(x-z) \qquad (5)$$

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

$$f_e(x) = (1 - \sum_k q_k)f(x) + \sum_k q_k f(x-z_k) \tag{6}$$

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

$$P(x<x'<x+dx) = g(x)dx \tag{7}$$

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

$$\int_0^C g(x)dx = 1 \tag{8}$$

where C is the total capacity of the system.

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

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

By defining the outage frequency function as

$$q(y) = g(C-y) \tag{10}$$

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, $\ldots c_n$, $\ldots c_N$, 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,...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

$$p_n = \int_0^{c_n} y g_n(y)dy/c_n \qquad (11)$$

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

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

where A_n is the capacity total of units 1,2,... 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

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

or equivalently,

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

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

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

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

$$L_o(x) = L(x)$$

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

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

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,...n. Notice that $f_n(x)$ is related to $f'(x)$ by

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

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, $f(x)$,

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

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

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

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

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

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

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

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

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

or using equation (21),

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

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

$$\log(1+z) = z/1 - z^2/2 + \ldots z^k/k + O(z^k) \tag{24}$$

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

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

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

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

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, $a_2, \ldots a_n$, and conversely a_n is a polynomial in k_1, $k_2, \ldots k_n$ as shown next:

$$\begin{aligned} k_1 &= a_1 = M \\ k_2 &= a_2 - a_1^2 = \sigma^2 \\ k_3 &= a_3 - 3a_1a_2 + 2a_1^3 \\ k_4 &= a_4 - 3a_2^2 - 4a_1a_3 + 12a_1^2a_2 - 6a_1^4 \end{aligned} \tag{27}$$

$$a_1 = k_1 \tag{28}$$

$$a_2 = k_2 + k_1^2$$

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

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

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

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

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

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

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

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

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

$$k_5 = u_5 - 10u_2u_3$$

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

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

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

The characteristic function of j(x) becomes

$$J(t) = \int_{-\infty}^{\infty} e^{itx} \; j(x)dx \tag{34}$$

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{itx} f(x-y)q(y)dydx$$

$$= \int_{-\infty}^{\infty} e^{itx'} f(x')dx' \int_{-\infty}^{\infty} e^{ity} q(y)dy$$

$$= F(t)Q(t)$$

Thus, referring to equation (26), we have

$$\log J(t) = \log F(t) + \log Q(t) \tag{35}$$

$$= \sum_{v=1}^{k} (a_v + b_v) (it)^v/v$$

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

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

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

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

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

$$f(z) = N(z) - g_1 N^{(3)}(z)/3! + g_2 N^{(4)}(z)/4! - g_3 N^{(5)}(z)/5! \qquad (38)$$

$$+ (g_4 + 10 g_1{}^2) N^{(6)}(z)/6! - (g_6 + 10 g_1 g_2) N^{(7)}(z)/7!$$

$$+ \; (g_6 + 56g_1g_3 + 35g_2^2)N^{(8)}(z)/8!$$

$$- \; (g_7 + 84g_1g_4 + 126g_2g_3 + 280g_1^3)N^{(9)}/9!$$

$$+ \; (g_8 + 120g_1g_5 + 210g_2g_4 + 126g_3^2 + 2100g_1^2g_2)\, N^{(10)}(z)/10! + \ldots$$

where z is the standardized variable defined by

$$z = (x - M)/\sigma \tag{39}$$

N(z) is the standard normal probability distribution

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

and g_i is related to the cumulants by

$$g_i = k_{i+2}/\sigma^{i+2} \tag{41}$$

The derivatives of N(z) are given by

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

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

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

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

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

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

$$x_n = [\, x_{1,n}, x_{2,n}, \;.\;.\;.\; x_{k,n}, \;.\;.\;]^T \tag{1}$$

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

The control vector is defined by

$$u_n = [\, u_{1,n}, u_{2,n}, \;.\;.\;.\;]^T \tag{2}$$

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

$$x_n = x_{n-1} + u_n \tag{3}$$

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

$$a_{k,n} \leq x_{k,n} \leq b_{k,n} \tag{4}$$

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

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

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

$$x_n = u_n + x_{n-1} \tag{6}$$

and the constraints

$$a_{n,k} \leq x_{n,k} \leq b_{n,k} \tag{7}$$

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

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

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

$$C_j = \sum_t QL_{t,j} \cdot Il_t \cdot MWC_t \cdot N_{t,j} \tag{9}$$

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

Il_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

$$R_j = \sum_t P_{t,nyr}\, S_{t,j} \tag{10}$$

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

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

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

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 i in year t

The operating cost in year j is given by

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

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

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

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

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

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.

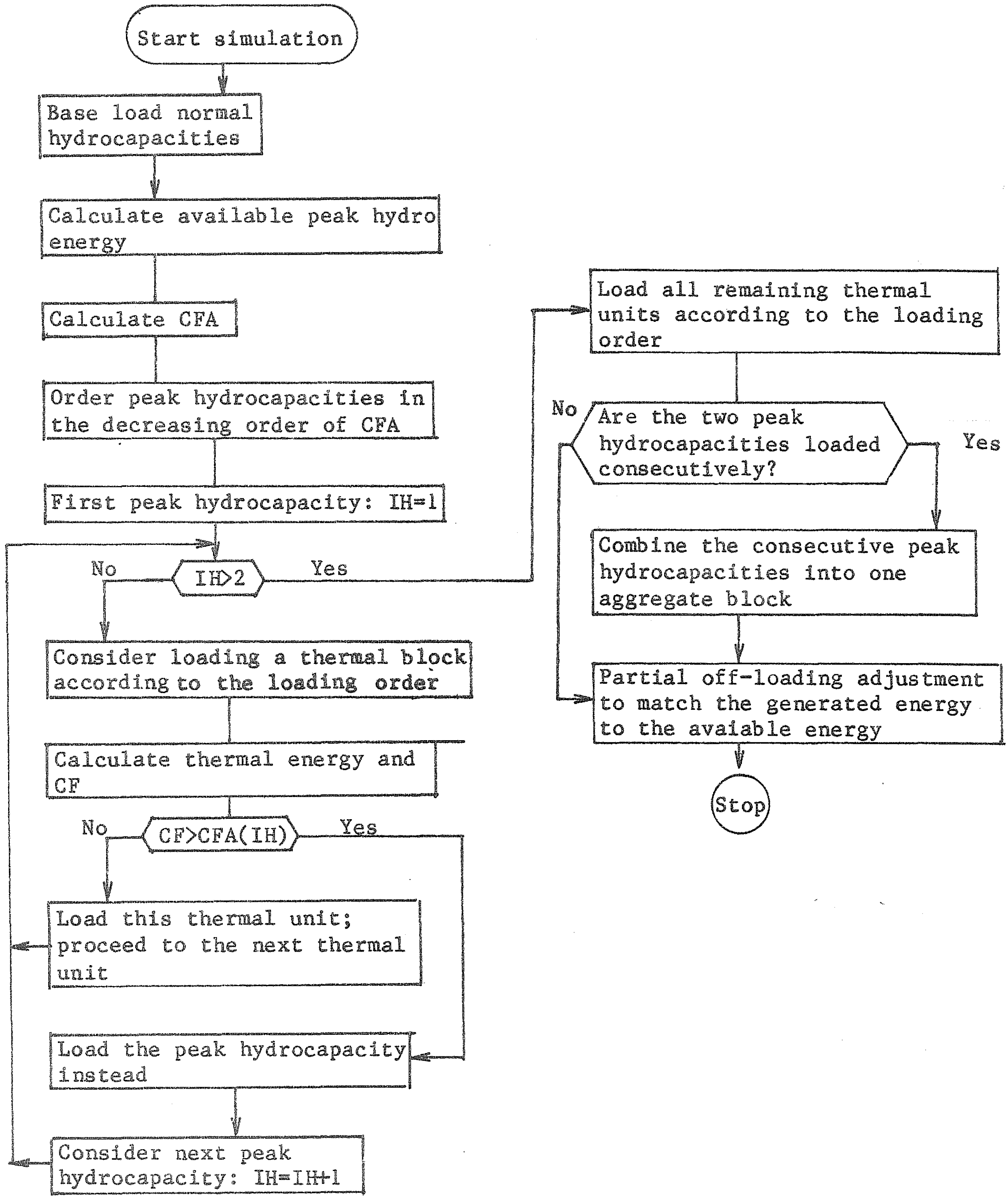


Figure D-1 Flowchart of probabilistic simulation with normal hydrocapacities

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

$$PE_i = p_i p_{ps} T \int [\, L_{i-1}(x-C_p) - L_{i-1}(x) \,]dx \qquad (2)$$

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.

APPEMDIX 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:

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

where

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

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

imax : maximum number of expansion unit types considered (DYNO)

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:

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

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

$$PLGA1_n = PLGS1_n - PLGS1_{n-1} \qquad (3)$$

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.

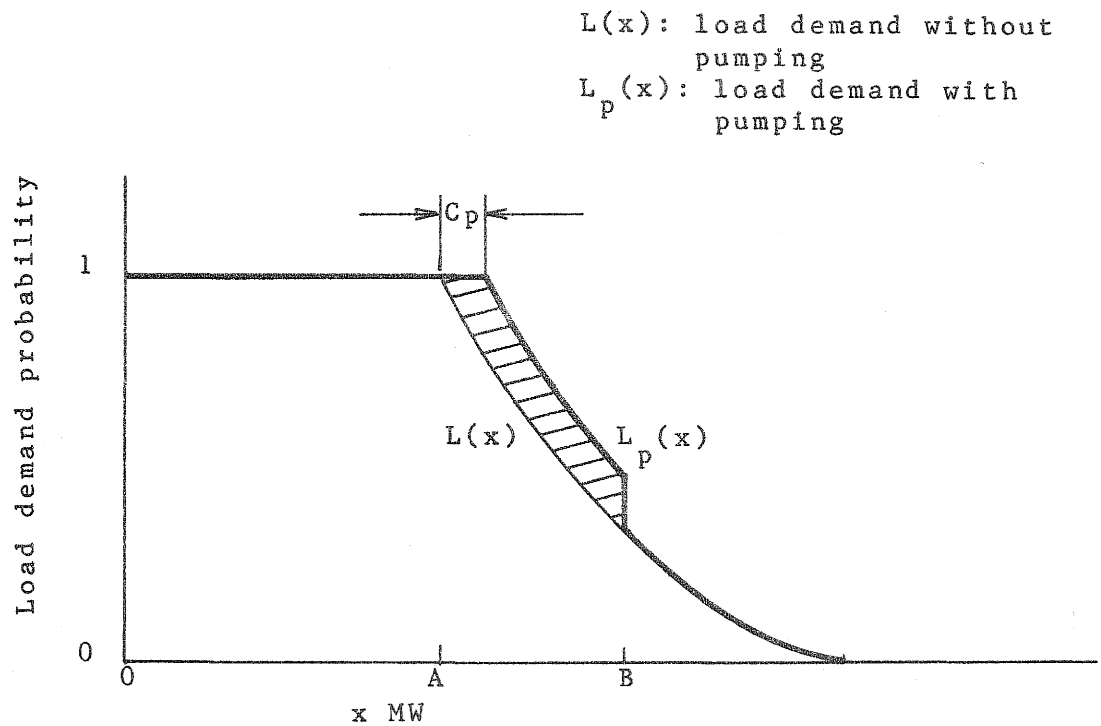


Figure D-2   Load demand probability functions with and without pumping requirement

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:

$$CEPX_{i,n} = \sum_{m=1}^{nmax} (CEPLX_{i,j})(IPLX_{i,m}) \tag{4}$$

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 DYNO output)

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

nmax : total number of study years (DYNO)

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

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

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:

$$CEX_n = \sum_{i=1}^{imax} CEPX_{i,n} \tag{6}$$

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

$$CE_n = CEF_n + CEX_n \tag{7}$$

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

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

$$CWIPB = CWIPFB + CWIPXB$$

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):

$$CWIPC_n = CEC_n - PLGS1_n \tag{9}$$

The CWIP increment is given by

$$CWIP_n = CWIPC_n - CWIPC_{n-1} \tag{10}$$

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

$$AFDC1_n = (CWIPC_n)(AFDCR_n)(1 - FCWIP_n) \tag{11}$$

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

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

$$CEPCX_{i,j} = \sum_{l=1}^{i} CEPXL_{i,l} \tag{13}$$

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

where

$CEPXC_{i,j}$	: cumulative construction expenditure for unit type i in the jth year of construction
$CEPLX_{i,j}$	: annual construction expenditure for unit type i in the jth year of construction (user)
$IPLX_{i,m}$	: no. of units of type i entering service in year m (DYNO)
$AFDCR_n$	: 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

$$AFCX2_m = \sum_{i=1}^{imax} AFDCX2_{i,m} \tag{14}$$

where

$AFCX2_m$	: 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

$$AFDC2_m = AFDCF2_m + AFDCX2_m \tag{15}$$

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:

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

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

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

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

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

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

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, $Lsum_i$, is given by

$$Lsum_i = \sum_{k=1}^{L_i} k \qquad (20)$$

where

L_i : life of a unit of type i

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

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

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

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

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

$$FDEP_i = 2/L_i \tag{23}$$

The declining balance in the beginning year of service is

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

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:

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

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

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

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

$$DEPX1_n = \sum_{i=1}^{imax} DEPLX1_{i,n} \tag{27}$$

$$DEPX2_n = \sum_{i=1}^{imax} DEPLX2_{i,n} \tag{28}$$

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

imax : the total number of expansion units

The accumulated book and tax depreciations are given by

$$DEP1C_n = \sum_{k=1}^{n} DEPX1_n + \sum_{k=1}^{n} DEPF1_k + DEP1B \tag{29}$$

$$DEP2C_n = \sum_{k=1}^{n} DEPX2_n + \sum_{k=1}^{n} DEPF2_k + DEP2B \tag{30}$$

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

$$PLNS1_n = PLGS1_n - DEP1C_n \tag{31}$$

The corresponding value of net plant added is

$$PLNA1_n = PLNS1_n - PLNS1_{n-1} \tag{32}$$

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

$$RATB_n = PLNS1_n + AFDC2C_n + FCWIPC_n\,(CWIPC_n + AFDC1C_n) \quad (33)$$

where

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

E-2 Capital Requirements Calculations

The total capital expenditure required for year n is given by

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

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 perferred and the common stocks are given by multiplying $CAPRQ_n$ by appropriate fractions:

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

$$DEBS_n = (CAPRQ_n)(FDEBS_n) \tag{36}$$

$$STCOM_n = (CAPRQ_n)(FSTCOM_n) \tag{37}$$

$$STPR_n = (CAPRQ_n)(FSTPR_n) \tag{38}$$

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

$$DEBN_n = DEBS_n + DEBL_n \tag{39}$$

where

$DEBN_n$: total new debt borrowed in year n

The total outstanding long-term debt and short-term debt are repectively 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:

$$CDEBT_n = CDEBS_n + CDEBL_n \tag{47}$$

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

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

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

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

$$BINT_n = OPINC_n + (AFDC1_n)(IAFC) \tag{50}$$

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:

$$ERNET_n = BINT_n - CDEBS_n - CDEBL_n \tag{51}$$

where

$ERNET_n$: net income

b. Dividends

The total preferred dividend is given by

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

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

$$ERNCOM_n = ERNET_n - DIVPR_n \tag{53}$$

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

$$DIVCOM_n = (ERNCOM_n)(FDIVC_n) \tag{54}$$

$$ERNRT_n = (ERNCOM_n)(FERN_n) \tag{55}$$

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

$$TXCR_n = (CE_n)(FCE_n)(RTXCR_n) \tag{56}$$

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

$$BINC_n = ERNET_n - (AFDC1_n)(IAFC) + TXSPD_n \tag{57}$$

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

$$TXDEFC_n = TXDEFB + \sum_{m=1}^{n} TXDEF_m \tag{64}$$

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

$$OPEXP_n = OMC_n + FUEL_n \tag{65}$$

where

$OPEXP_n$: total operating expense in year n

$DEP1C_n$: cummulative 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

$$GREV_n = OPEXP_n + TXSPD_n + TXDEF_n + OPINC_n \tag{66}$$

where

$GREV_n$: gross operating revenue in year n

The operating revenue tax is given by

$$TXREV_n = (GREV_n)(RTXREV_n) \tag{67}$$

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:

$$TOTEXP_n = OPEXP_n + TXSPD_n + TXDEF_n + TXCR_n + TXREV_n \tag{68}$$

E-4 Sources and Uses of Funds

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

$$SRCIN_n = ERNET_n + DEP1_n + TXDEF_n \tag{69}$$

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:

$$SRCEX_n = STCOM_n + STPR_n + DEBS_n + DEBL_n \tag{70}$$

where

$SRCEX_n$: external source of funds

The total source of funds is given by

$$SRCTOT_n = SRCIN_n + SRCEX_n \tag{71}$$

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:

$$PLNUT_n = PLNV_n + CWIPC_n \tag{76}$$

where

$PLNUT_n$: net utility plant

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

$$EQCOM_n = STCOMC_n + ERNRTC_n \tag{77}$$

where

$$STCOMC_n = \sum_{k=1}^{n} STCOM_k + STCOMB \tag{78}$$

$$ERNRTC_n = \sum_{k=1}^{n} ERNRT_k + ERNRTB \tag{79}$$

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:

$$CAPTOT_n = EQCOM_n + STPRC_n + DEBLT_n \tag{80}$$

where

$$STPRC_n = \sum_{k=1}^{n} STPR_k + STPRB \tag{81}$$

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:

$$TLIAB_n = CAPTOT_n + DEBST_n + TXDEFC_n \tag{82}$$

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

$$WKCAP_n = WKCAP_{n-1} + CAPIN_n \tag{83}$$

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

$$ASSET_n = PLNUT_n + WKCAP_n \tag{84}$$

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

$$DEBASS_n = DEBT_n/ASSET_n \tag{85}$$

$$DEBEQ_n = DEBT_n/EQTOT_n \tag{86}$$

$$COVRG_n = BINT_n/CDEBT_n \tag{87}$$

where EQTOT is the total equity given by

$$EQTOT_n = STCOM_n + STPR_n + ERNRT_n \tag{88}$$

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
      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 /CACA/ NPEXP,IUP(30,20), NBYR, NTYR
C
C
      DATA YES/3HYES/,YE/3HY  /
      DATA INP1,INP2/11,14/
      JNUM=14
      IPNUM=1
      IPMAX=250
      DO 2090 K=1,42
      READ (INP2,115) (KEY1(K,J),J=1,18)
 2090 CONTINUE
  115 FORMAT (18A4)
      WRITE (6,101)
  101 FORMAT(//'    ********** 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.',/2H ?)
      READ (5,103) RESP
  103 FORMAT (A3)
      IF(RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 11
C
C
      WRITE (6,311)
  311 FORMAT (5X,51H SINCE A FILE DOES NOT EXIST, YOU CAN CREATE A FILE)
      WRITE (6,411)
  411 FORMAT (5X,51H FROM SCRATCH. A SET OF INSTRUCTIONS WILL APPEAR ON)
      WRITE (6,511)
  511 FORMAT (5X,51H THE TERMINAL TO WHICH YOU SHOULD RESPOND PROPERLY.)
  520 PRINT 530
      READ *,NBYR
      PRINT 540
      READ *, IEND
      IF(IEND.LT.40) IEND=IEND+100
  530 FORMAT(/ 10X,' ENTER THE FIRST YEAR OF EXPANSION (2 DIGIT ).')
  540 FORMAT(/10X,' ENTER THE LAST YEAR OF EXPANSION')
      NTYR=IEND+1-NBYR
      IF(NTYR.GT.20) PRINT 550
  550 FORMAT(/10X,' THE NUMBER OF YEARS OF EXPANSION EXCEEDS 20'/
     + 10X,' REPEAT INPUT OF YEARS.')
      IF(NTYR.GT.20) GO TO 520
      PRINT 610
      PRINT 600
  600 FORMAT(10X,'DECLARE THE MAXIMUM NUMBER OF EXPANSION PLANTS'/
     +10X,'(NO MORE THAN 20).'/' --')
      READ 610, NPEXP
  610 FORMAT(I2)
  620 FORMAT(/10X,'PLANT CODE UP TO ',I2,' ARE ASSIGNED'/
     +10X,'FOR EXPANSION PLANTS.')
      IF(NPEXP.GT.1) PRINT 620, NPEXP
  645 FORMAT(I2)
      CALL CREATE
      GO TO 2020

   11 READ (INP1,1400) IPNUM,NPEXP,NBYR,IEND
 1400 FORMAT(4I10)
      NTYR=IEND+1-NBYR
      DO 1001 IP=1,IPNUM
      READ (INP1,1422)(NUF(IP,N),N=1,20)
```

```
1422 FORMAT(40I2)
1001 READ (INP1,8420) (PLANT(IP,J),J=1,JNUM)
     DO 1003 IP=1,NPEXP
1003 READ (INP1,1422) (IUP(IP,N),N=1,20)
     REWIND INP1
C
C
2020 WRITE (6,105)
 105 FORMAT (5X,30H ARE ANY CHANGES IN PLANT DATA,
    +28H NECESSARY? ENTER YES OR NO.)
     READ (5,103) RESP
     IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL CHANGE
     PRINT 107
 107 FORMAT (5X,27H DO YOU WANT TO ADD PLANTS?,
    +17H ENTER YES OR NO.)
     READ (5,103) RESP
     IF(RESP.EQ.YES.OR.RESP.EQ.YE) CALL CREATE
     WRITE (6,185)
 185 FORMAT(5X,'**** IF PRINT OF DATA IS NEEDED, ENTER YES.')
     READ (5,103) RESP
     IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL LIST
     PRINT 8000
8000 FORMAT(5X,' IS YOUR PLANT DATA COMPLETE ?   ENTER YES OR NO.')
     READ (5,103) RESP
     IF(RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 8300
     GO TO 2020
8300    REWIND INP1
     WRITE (INP1,1400) IPNUM,NPEXP,NBYR,IEND
     DO 8500 IP=1,IPNUM
     WRITE (INP1,1422) (NUF(IP,N),N=1,20)
     WRITE (INP1,8420) (PLANT(IP,J),J=1,JNUM)
8500 CONTINUE
     DO 8503 IP=1,NPEXP
8503 WRITE (INP1,1422) (IUP(IP,N),N=1,20)
     REWIND INP1
8420 FORMAT(5E15.5)
8590 FORMAT(//'  ATTN: THERMAL PLANT DATA FILE IS CREATED',/
    # '  OR REVISED ON UNIT 11')
     PRINT 8590
     PRINT 8600
8600 FORMAT(' ***** END OF  THERMAL PLANT INPUT PROCEDURE *****')
     PRINT 8700
8700 FORMAT (//,' DO YOU WANT TO INCLUDE HYDROELECTRIC',/
    + ' AND PUMPED STORAGE PLANTS',/' IN YOUR INPUT DATA FILE?',/
    + ' ENTER YES OR NO.'/2H ?)
     READ (5,103) RESP
     IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL HYDPS
C    CALL LOADI
     PRINT 8800
8800 FORMAT (//,'***** END OF PLANT DATA INPUT PROCEDURE *****')
     END
C
     SUBROUTINE CREATE
C    .....THIS SUBPROGRAM CREATES A PLANT DATA
C         FILE FROM SCRATCH IF NONE EXISTED BEFORE.
C         USER CAN INPUT DATA FOR EACH PLANT
C         INTERACTIVELY
C
C
     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 /CACA/ NPEXP,IUP(30,20), NBYR, NTYR
C
     DATA YES/3HYES/, YE/3HY  /
      IEND=NBYR+NTYR-1
  10 WRITE (6,102)
```

```
  102 FORMAT (//5X,' ENTER PLANT CODE NUMBER.')
      WRITE (6,103)
  103 FORMAT (4H ---)
      READ (5,104) IP
      IF(IP.LT.0.OR.IP.GT.200) PRINT 2515
 2515 FORMAT('          INPUT ERROR.    ')
      IF(IP.LT.0.OR.IP.GT.200) GO TO 10
  104 FORMAT (I3)
  204 FORMAT (1H I3)
C
C       .....ENTER ALL PLANT DATA

      CALL NUNIT (1)
      IF(IP.GT.IPNUM) IPNUM=IP
  199 FORMAT (3H --)
      DO 1002 K=2,JNUM
      IF(K.EQ.9) GO TO 1002
      WRITE (6,111) (KEY1(K+28,J),J=1,18)
  111 FORMAT (1H 18A4)
      WRITE (6,105)
  105 FORMAT (11H -----.-----)
      READ (5,106) PLANT(IP,K)
C           WRITE (6,206) PLANT(IP,K)
 1002 CONTINUE
      PLANT(IP,1)=FLOAT(IP)+0.00001
  106 FORMAT (F10.5)
  206 FORMAT (1H F10.5)
C
      INP1=11
 8300    REWIND INP1
      WRITE (INP1,1400) IPNUM,NPEXP,NBYR,IEND
 1400 FORMAT(4I10)
      DO 8500 IP=1,IPNUM
      WRITE (INP1,1422) (NUF(IP,N),N=1,20)
 1422 FORMAT(40I2)
      WRITE (INP1,8420) (PLANT(IP,J),J=1,JNUM)
 8420 FORMAT(5E15.5)
 8500 CONTINUE
      DO 8503 IP=1,NPEXP
 8503 WRITE (INP1,1422) (IUP(IP,N),N=1,20)
      REWIND INP1
C
C       .....PROMPT FOR ADDITION OF MORE PLANTS
      WRITE (6,107)
  107 FORMAT (5X,34H DO YOU WANT TO ADD ANOTHER PLANT?,
     +17H ENTER YES OR NO.)
  108 FORMAT (4H YYY)
      READ (5,112) RESP
  112 FORMAT (A3)
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 10
      RETURN
      END
C
C
      SUBROUTINE CHANGE
C     .....THIS SUBPROGRAM IS USED TO MODIFY EXISTING
C           PLANT DATA. ALL DATA MANIPULATIONS CAN
C           BE DONE INTERACTIVELY.
C
C
C
C
C
      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 /CACA/ NPEXP,IUP(30,20), NBYR, NTYR
      DATA YES/3HYES/
      DATA YE/3HY  /
```

```
    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
     +.')
      WRITE (6,102)
  102 FORMAT (4H ---)
      READ (5,103) IP
      IF(IP.LT.1.OR.IP.GT.IPNUM) GO TO 10
  103 FORMAT (I3)
   11 WRITE (6,104)
  104 FORMAT(5X, 'ENTER DATA CODE IN 2 DIGIT, OR 99 FOR HELP.')
      WRITE (6,105)
  105 FORMAT (3H --)
      READ (5,106) JP
  106 FORMAT (I2)
      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 (11H ----.-----)
      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
C     .....THIS SUBPROGRAM PRINTS A LEGEND WHICH ASSIGNS
C          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 /CACA/ NPEXP,IUP(30,20), NBYR, NTYR
      DATA YES/3HYES/
      DATA YE/3HY  /
      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.OR.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
C
      SUBROUTINE LIST
      DIMENSION JPR(200)
      INTEGER YE,IP,IPNUM,IPMAX,JNUM,YEAR,OPT
      INTEGER RESP, KEY1(42,18),  YES
      REAL PLANT(200,14)
      COMMON NUF(200,20), IP,PLANT,INS,KEY1,IPNUM,IPMAX,JNUM
      COMMON /CACA/ 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 (I1)
      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 IIXX=1,IPPP
      IP2=IP1+4
      IF(IPNUM.GT.IP2) GO TO 2813
      IF(IP2-IPNUM.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
```

```
      PRINT 3011, (KEY1(K+14,J),J=1,5),(PLANT(IP,K),IP=IP1,IP2)
 1001 CONTINUE
      PRINT 3011
      IF(IPNUM.EQ.IP2) GO TO 2830
      IP1=IP2+1
 2800 CONTINUE
 2830 CONTINUE
  111 FORMAT (1H 18A4)
  112 FORMAT (I3,1X,I2,3(1X,F7.2),1X,F4.3,1X,F5.2,
     +3(1X,F5.2),2(1X,F5.2),1X,F5.2,2(1X,F4.3))
      GO TO 55
C
   30 WRITE (6,113)
  113 FORMAT (5X,30H ENTER PLANT CODE IN 3 DIGITS.)
      WRITE (6,114)
  114 FORMAT (4H ---)
      READ (5,115) IP
  115 FORMAT (I3)
      CALL NUNIT(2)
      PRINT 5400,IP
      PRINT 5500,JPR(IP)
      DO 1002 K=2,JNUM
      IF(K.EQ.9) GO TO 1002
      WRITE (6,3011) (KEY1(K+14,J),J=1,5),(PLANT(IP,K))
 1002 CONTINUE
 3011 FORMAT(1X,5A4,6F10.3)
      PRINT 3011
      PRINT 116
  116 FORMAT (5X,38H DO YOU WANT TO PRINT DATA FOR ANOTHER,
     +' PLANT?     ENTER YES OR NO.')
      READ (5,136) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 30
       IF (RESP.EQ.YE) GO TO 30
      GO TO 55
C
   40 WRITE (6,119)
  119 FORMAT (5X,36H WHICH DATA YOU WANT TO PRINT? ENTER,
     +' A 2 DIGIT DATA CODE.  FOR HELP, ENTER 99.')
      READ (5,120) J
  120 FORMAT (I2)
      IF (J.EQ.99) GO TO 44
      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))
      GO TO 45
   42 DO 1005 IP=1,IPNUM
 1005 WRITE (6,122) PLANT (IP,J)
  122 FORMAT (20(1X,I3))
      GO TO 45
   44 CALL HELP
      GO TO 40
   45 WRITE (6,123)
  123 FORMAT (/5X,35H DO YOU WANT TO PRINT ANOTHER DATA?,
     +'  ENTER YES OR NO.' )
      READ (5,136) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 40
      IF(RESP.EQ.YE) GO TO 40
      GO TO 55
C
   50 WRITE (6,113)
      WRITE (6,114)
      READ (5,115) IP
   51 WRITE (6,119)
      READ (5,120) J
      IF (J.EQ.99) GO TO 54
      IF (J.EQ.1) GO TO 53
   52 WRITE (6,126) PLANT(IP,J)
```

```
126 FORMAT (1H ,F8.3)
    GO TO 55
 53 WRITE (6,127) PLANT (IP,J)
127 FORMAT (1H ,I2)
    GO TO 55
 54 CALL HELP
    GO TO 51
 55 WRITE (6,124)
124 FORMAT (/5X,32H DO YOU WANT TO PRINT MORE DATA?,
   +' ENTER YES OR NO.')
    READ (5,136) RESP
    IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 11
C
 60 WRITE (6,125)
125 FORMAT(5X,' END OF PRINTING'///)
136 FORMAT (A3)
    RETURN
    END
C
    SUBROUTINE NUNIT(IOPT)
C          THIS SUB READS AND PRINTS THE NUMBER OF UNITS
C           FOR EACH YEAR
C                    IOPT=1         READ
C                         2         PRINT FOR TH ENTIRE PERIOD
C                         3        CHANGE
C                         4        PRINT ONLY SPECIFIED YEAR
    DIMENSION JP(40)
    COMMON NUF(200,20),IP,PLANT,INS,KEY1,IPNUM,IPMAX,JNUM
    COMMON /CACA/ NPEXP,IUP(30,20), NBYR,NTYR
    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/' --'/
    K1=1+NBYR-80
    L1=1
    IF(IOPT.GT.2) GO TO 200
    IF(IOPT.EQ.1.AND.IP.GT.NPEXP) PRINT 420,IP
    IF(IOPT.EQ.1.AND.IP.LE.NPEXP) PRINT 520
520 FORMAT(5X,'ENTER MINIMUM NUMBER OF UNITS PERMITTED'/
   +5X,'EACH YEAR (LOWER BOUND OF EXPANSION PLANTS).'/
   +10X,'THEN ENTER THE MAX. NUMBER OF UNITS PERMITTED'/
   +10X,'EACH YEAR (UPPER BOUND) ON THE SECOND LINE.')
420 FORMAT(5X,'ENTER NUMBER OF UNITS OF PLANT CODE ',I2,
   +' WORKING IN EACH YEAR')
430 FORMAT(5X,'NUMBER OF UNITS OF PLANT CODE ',I2,
   +' WORKING IN EACH YEAR')
    IF(IOPT.EQ.2.AND.IP.GT.NPEXP) PRINT 430,IP
    IF(IOPT.EQ.2.AND.IP.LE.NPEXP) PRINT 440,IP
440 FORMAT(5X,' LOWER BOUND (1ST LINE) & UPPER BOUND (2ND LINE)'/
   + 5X,' OF PLANT CODE ',I3,' (EXPANSION PLANT).')
    DO 90 JJ=1,4
    IX=NTYR-L1
    L2=L1+19
    K2=K1+19
    IF(IX.GT.19) GO TO 40
    L2=L1+IX
    K2=K1+IX
 40 PRINT 10, (JP(K),K=K1,K2)
    PRINT 20,(IDSH,K=K1,K2)
    IF(IOPT.EQ.1) READ 30,  (NUF(IP,L),L=L1,L2)
    IF (IOPT.EQ.1.AND.IP.LE.NPEXP) READ 30,(IUP(IP,L),L=L1,L2)
    IF(IOPT.EQ.1) PRINT 10
    PRINT 31,  (NUF(IP,L),L=L1,L2)
     IF(IP.LE.NPEXP) PRINT 31, (IUP(IP,L),L=L1,L2)
    PRINT 10
    IF(IX.LE.19) GO TO 50
    L1=L2+1
```

```
      K1=K2+1
   90 CONTINUE
   10 FORMAT(1X,20(A2,1X))
   20 FORMAT(20A3)
   30 FORMAT(I2,1X,20(I2,1X))
   31 FORMAT(1X,20(I2,1X))
   50 CONTINUE
      RETURN
  200 PRINT 210
  210 FORMAT('      WHAT IS THE YEAR ?'/' --')
  220 FORMAT(2I2)
      READ 220,JYR
      LL=JYR
      IF(LL.LT.NBYR) LL=LL+100
      LL=LL+1-NBYR
      IF(IP.LE.NPEXP) PRINT 250
  250 FORMAT('      ENTER NIMIMUM AND MAXIMUM NUMBERS OF UNITS'/
     +'      OF THE EXPANSION PLANTS.'/' --',' --')
      IF(IP.GT.NPEXP) PRINT 240
  240 FORMAT('      ENTER THE NUMBER OF UNITS'/' --')
      IF(IOPT.EQ.3) READ 220, JNU,JUP
      NUF(IP,LL)=JNU
      IF(IP.LE.NPEXP) IUP(IP,LL)=JUP
      IF(IP.GT.NPEXP) PRINT 300, IP,JP(LL), NUF(IP,LL)
      IF(IP.LE.NPEXP) PRINT 327,IP,JP(LL),NUF(IP,LL),IUP(IP,LL)
  327 FORMAT('      PLANT CODE=',I2,' YR=',A2,' MIN.=',I2,' MAX=',I2/)
  300 FORMAT('      PLANT CODE=',I2,'  YEAR=',A2,'  NUMBER OF UNITS='
     +, I2//)
      RETURN
      END
C
C ------------------------------------------------------------
      SUBROUTINE HYDPS
C ------------------------------------------------------------
C
C     THIS SUBPROGRAM IS USED TO PROCESS AND STORE INPUT
C     DATA FOR HYDROELECTRIC AND PUMPED STORAGE PLANTS.
C
      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),
     + NYR(20)
C
C
C     PROCEDURE FOR HYDROELECTRIC INPUT
C
C
C
C
      IFLAG=1
C
      CALL HYDA
C
C     PROCEDURE FOR PUMPED STORAGE INPUT
C
C
C
C
      IFLAG=2
C
      CALL PSDA
```

```
C
C
      RETURN
      END
C
C
C
C
C
C     ----------------------------------
      SUBROUTINE HYDA
C     ----------------------------------
C
C
C
C     THIS PRUGRAM ACCEPTS INPUT DATA FOR OPERATING AND
C     COST PARAMETERS OF HYDROELECTRIC PLANTS.
C     THE ENTIRE PROCEDURE IS INTERACTIVE.
C
C
      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),
     + NYR(20)
      DOUBLE PRECISION ISSN(4)
      INTEGER RESP,KEY2,KEY3,YES,YE,NO
      INTEGER SEASON(4),BCAP,IENR
      DATA  ISSN/8H WINTER ,8H SPRIMG ,
     +8H SUMMER ,8H AUTUMN /
      DATA YES,YE/3HYES,3HY  /
      DATA NO/3HNO /
      IP=1
      ITYP=1
      KHU1=0
      JNUM=8
      KHMX=1
      KHM=3
      KHU=0
      KH1=1
      KX1=4
      KXM=6
      KXMX=4
      DO 100 K=1,34
      READ (31,110) (KEY2(K,J),J=1,18)
  100 CONTINUE
  110 FORMAT (18A4)
C
C     BEGINNING OF INPUT PROCEDURE
C
C
      WRITE (6,120)
  120 FORMAT (//,5X,34H INPUT PROCEDURE FOR HYDROELECTRIC,
     +21H PLANT SYSTEM BEGINS.)
      WRITE (6,123)
  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,I3)
  152 CONTINUE
      KHU=KHU+1
      IF (KHU.LE.KHU1) KHU=KHU1+1
      KHU1=KHU
```

```
  170 FORMAT (I2)
      IF (KHU.EQ.KHM) KHMX=KHM
  172 CONTINUE
      CALL HUNIT
  175 CONTINUE
      CALL HCREAT
      CALL HREG
C
C     CHANGES/CORRECTIONS OF INPUT DATA
C
  217 CONTINUE
      WRITE (6,220) KHU
  220 FORMAT (//,41H DO YOU WANT TO CHANGE ANY DATA FOR PLANT,
     +I3,1H?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 230
      IF (KHU.EQ.KH1.AND.KHMX.EQ.KH1) GO TO 250
      WRITE (6,257)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 417
  229 CONTINUE
      GO TO 250
  230 WRITE (6,240) KHU
  240 FORMAT (/,39H DO YOU WANT TO CHANGE ALL THE DATA FOR,
     +6H PLANT,I3,1H?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE)  GO TO 172
      CALL HCHNG
      IF (KHU.EQ.KH1.AND.KHMX.EQ.KH1) GO TO 250
  248 CONTINUE
      WRITE (6,257)
  257 FORMAT (/,31H DO YOU WANT TO CHANGE DATA FOR,
     +15H ANOTHER PLANT?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 417
  250 CONTINUE
      IF (KHU.GE.KHM.OR.KHMX.GE.KHM) GO TO 440
C
C     ADDITIONS OF MORE PLANTS TO THE SYSTEM
C
      WRITE (6,260)
  260 FORMAT (//,38H DO YOU WANT TO ADD MORE PLANTS TO THE,
     +8H SYSTEM?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 152
      KHMX=KHU
C
      GO TO 440
  306 CONTINUE
      READ (35,575,END=146) KHU
      READ (35,590) (IHUF(KHU,K),K=1,15)
  154 FORMAT (//,3I3)
      READ (35,587) (HPLR(KHU,J),J=1,JNUM)
      GO TO 306
  146 CONTINUE
      KHMX=KHU
      WRITE (6,158) KHMX
  158 FORMAT (//,40H THE TOTAL NO. OF EXISTING HYDROELECTRIC,
     +10H PLANTS IS,I3)
C     DO 161 KHU=1,KHMX
C     READ (37,575) KHU
C     READ (37,590) (IHUF(KHU,K),K=1,15)
C 161 CONTINUE
      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 HPRINT(2)
```

```
148 CONTINUE
159 FORMAT(//,I3)

265 WRITE (6,270)
270 FORMAT (//,44H DO YOU WANT TO CHANGE ANY DATA FOR EXISTING,
   +/22H HYDROELECTRIC 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 (I1)
    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
    IF (KHU1.GT.3) GO TO 750
    IF (ITYP.EQ.2) GO TO 750
    WRITE (6,450)
450 FORMAT ( //,38H IS YOUR INPUT FILE FOR EXISTING HYDRO,
   +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 HCUMP
575 FORMAT(/,3I3)
    REWIND 35
    DO 560 KHU=KH1,KHMX
    WRITE (35,575) KHU
    WRITE (35,590) (IHUF(KHU,K),K=1,15)
    WRITE (35,587) (HPLR(KHU,J),J=1,JNUM)
580 FORMAT (1H ,5F10.5)
560 CONTINUE
570 FORMAT (/,1H ,3I3)
585 FORMAT (1H ,5E13.5)
550 CONTINUE
590 FORMAT (20I3)
    REWIND 38
    DO 540 NY=1,15
    WRITE (38,575) NY
    WRITE (38,587) (HPLF(J,NY),J=1,JNUM)
587 FORMAT (5E12.5)
540 CONTINUE
600 CONTINUE
500 CONTINUE
602 ITYP=2
    JNUM=11
    KHU=KHM
    KHM=KXM
    KH1=KX1
    KHMX=KX1
    KX=KXM-KX1+1
    WRITE (6,605)
605 FORMAT (//,33H DO YOU HAVE A FILE FOR EXPANSION,
```

```
     +22H HYDROELECTRIC PLANTS?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 800
      WRITE (6,610)
  610 FORMAT (//,40H DO YOU WANT TO CONSIDER HYDRO PLANTS AS,
     +/22H EXPANSION CANDIDATES?,17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 710
      GO TO 900
  710 WRITE (6,715) KX,KX1,KXM
  715 FORMAT (//,30H NO. OF PLANT TYPES CONSIDERED,
     +17H FOR EXPANSION IS,I3/18H THEY ARE ASSIGNED,
     +19H PLANT CODE NUMBERS,I3,8H THROUGH,I3)
      GO TO 152
  750 CONTINUE
      WRITE (6,720)
  720 FORMAT (//,33H IS YOUR INPUT FILE FOR EXPANSION,
     +23H HYDRO PLANTS COMPLETE?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 850
  800 CONTINUE
      READ (39,575,END=830) KHU
      READ (39,590) (IHUL(KHU,K),K=1,15)
      READ (39,590) (IHUH(KHU,K),K=1,15)
      READ (39,587) (HPLR(KHU,J),J=1,JNUM)
      GO TO 800
  830 CONTINUE
      KHMX=KHU
      KHU1=KHU
      KX=KHMX-KX1+1
      WRITE (6,715) KX,KX1,KXM
      WRITE (6,832)
  832 FORMAT (//,31H ENTER YES IF YOU WANT TO PRINT,
     +23H EXPANSION SYSTEM DATA.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL HPRINT(3)
      WRITE (6,835)
  835 FORMAT (//,35H DO YOU WANT TO CHANGE ANY DATA FOR,
     +24H EXPANSION HYDRO PLANTS?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 400
      GO TO 750
  850 CONTINUE
      KHMX=KHU1
      REWIND 39
      DO 870 KHU=KH1,KHMX
      WRITE (39,575) KHU
      WRITE (39,590) (IHUL(KHU,K),K=1,15)
      WRITE (39,590) (IHUH(KHU,K),K=1,15)
      WRITE (39,587) (HPLR(KHU,J),J=1,JNUM)
  870 CONTINUE
  900 CONTINUE
      WRITE (6,902)
  902 FORMAT (//,31H DO YOU HAVE A FILE ALREADY FOR,
     +21H SEASONAL MULTIPLIERS/20H OF BASE CAPCITY AND,
     +21H TOTAL INFLOW ENERGY?/17H ENTER YES OR NO./2H ?)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 941
  908 WRITE (6,910)
  910 FORMAT (//,30H ENTER SEASONAL MULTIPLIERS OF,
     +/39H BASE CAPACITY AND TOTAL INFLOW ENERGY./
     +34H USE WINTER AS THE BASE SEASON AND/
     +40H SET MULTIPLIERS TO 1.0 FOR THIS SEASON.)
  915 FORMAT (F10.5)
  925 FORMAT (12H ----.------)
      WRITE (6,920)
  920 FORMAT (//,29H BASE CAPACITY MULTIPLIER FOR)
      DO 924 I=1,4
```

```
      WRITE (6,930) ISSN(I)
  930 FORMAT (/1H ,A8,2H ?)
      WRITE (6,925)
      READ (5,915) PBM(I)
  924 CONTINUE
      WRITE (6,940)
  940 FORMAT (//,29H INFLOW ENERGY MULTIPLIER FOR)
      DO 934 I=1,4
      WRITE (6,930) ISSN(I)
      WRITE (6,925)
      READ (5,915) EAM(I)
  934 CONTINUE
      GO TO 954
  941 READ (40,587) (PBM(JS),JS=1,4)
      READ (40,587) (EAM(JS),JS=1,4)
  954 CONTINUE
      WRITE (6,962)
  962 FORMAT (//,25H ENTER YES IF YOU WANT TO,
     +32H PRINT THE SEASONAL MULTIPLIERS./2H ?)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 970
      GO TO 981
  970 CONTINUE
      WRITE (6,1980)
      WRITE (6,1983)
      WRITE (6,1982) (PBM(JS),JS=1,4)
      WRITE (6,1985)
      WRITE (6,1983)
      WRITE (6,1982) (EAM(JS),JS=1,4)
  981 WRITE (6,985)
  985 FORMAT (//,25H ENTER YES IF YOU WANT TO,
     +29H RENTER SEASONAL MULTIPLIERS./2H ?)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 908
      REWIND 40
      WRITE (40,587) (PBM(JS),JS=1,4)
      WRITE (40,587) (EAM(JS),JS=1,4)
  950 CONTINUE
C
      WRITE (6,1965)
 1965 FORMAT (//,37H PRINTING OF THE ENTIRE HYDROELECTRIC,
     +21H SYSTEM DATA FOLLOWS./ 17H IF YOU WANT THIS,
     +20H PRINTING,ENTER YES.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 1975
      GO TO 2000
 1975 WRITE (6,1980)
 1980 FORMAT (//,27H BASE CAPACITY MULTIPLIERS:)
      WRITE (6,1983)
 1983 FORMAT (/,4X,7H WINTER,2X,7H SPRING,3X,
     +7H SUMMER,3X,7H AUTUMN)
      WRITE (5,1982) (PBM(JS),JS=1,4)
      WRITE (6,1985)
 1985 FORMAT (//,27H INFLOW ENERGY MULTIPLIERS:)
      WRITE (6,1983)
      WRITE (6,1982) (EAM(JS),JS=1,4)
 1982 FORMAT (4F10.5)
      CALL HPRINT(1)
 2000 CONTINUE
C     END OF INPUT PROCEDURE
C
      WRITE (6,2654)
 2654 FORMAT (//,41H END OF INPUT PROCEDURE FOR HYDROELECTRIC,
     +8H PLANTS./44H INPUT DATA STORED ON UNITS 35,38,39 AND 40.)
      RETURN
      END
C
C
```

```
C
C-----------------------------------------------------------------
      SUBROUTINE HREG
C-----------------------------------------------------------------
C
C
C     THIS SUBROUTINE  DETERMINES THE REGULATING AND
C     AND PEAKING CHARACTERISTICS OF HYDRO PLANTS.
C
      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,KHI,JNUM,ITYP,IFLAG
      COMMON /MULTPS/ PBM(4),EAM(4)
      COMMON/UNITS/ IHUF(6,20),IHUL(6,20),IHUH(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=P1-PB
      IF (PC.LE.PB) GO TO 220
      PMAX=PC-PB
      TV=1000.*V/PMAX
      V0=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
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(V0.GT.V1.AND.V0.LE.V2) GO TO 110
      IF(V0.GT.V2.AND.V0.LE.V3) GO TO 120
      IF (V0.GT.V4) GO TO 125
  101 CONTINUE
      WRITE (6,75)
   75 FORMAT (//,29H INFLOW ENERGY EXCEEDS PLANT ,
     +22H GENERATING CAPABILIY.)
      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 ,
     +19H 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,/30H AND NO REGULATING CAPABILITY.)
```

```
      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
      WRITE (6,126)
  126 FORMAT (//,34H THIS IS A WEEKLY REGULATING PLANT)
      GO TO 300
  125 CONTINUE
      WRITE (6,127)
  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
C
C-----------------------------------------------------------------
      SUBROUTINE HCREAT
C---------------------------------------------------------------
C
C      THIS SUBROUTINE CREATES A FILE OF HYDRO PARAMETERS
C
      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),
     + NYR(20)
      INTEGER RESP,KEY2,KEY3,YES,YE
      DATA YES,YE/3HYES,3HY  /
C
C
      GO TO (10,20),IFLAG
C
   10 DO 180 K=1,JNUM
      WRITE (6,112)
  110 FORMAT (18A4)
  112 FORMAT(/,1H )
      WRITE (6,110) (KEY2(K+23,J),J=1,18)
      WRITE (6,190)
  190 FORMAT (/,11H ----.-----)
      READ (5,210) HPLR(KHU,K)
  210 FORMAT (F10.5)
  180 CONTINUE
      GO TO 30
C
C
   20 DO 280 K=1,JNUM
      WRITE (6,112)
      WRITE (6,110) (KEY3(K+23,J),J=1,18)
```

```
      WRITE (6,190)
      READ (5,210) HPLR(KHU,K)
  280 CONTINUE
   30 RETURN
      END
      SUBROUTINE HUNIT
      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/UNITS/ IHUF(6,20),IHUL(6,20),IHUH(6,20),
     + NYR(20)
      INTEGER RESP,KEY2,KEY3,YES,YE
      DATA YES,YE/3HYES,3HY  /
      IC=1
      NBYR=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,/19H 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,/19H 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 ,20I3)
      WRITE (6,186)
  186 FORMAT (/,1H ,15(3H --))
      READ (5,190) (IHUH(KHU,K),K=1,15)
  190 FORMAT (20I3)
  187 FORMAT (//,3I3)
  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,I3,2H ?,
     +/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
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,KH1,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
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
```

```
      WT1(NY)=WT1(NY)+W1(KHU)*IHUF(KHU,NY)
      WT2(NY)=WT2(NY)+W2(KHU)*IHUF(KHU,NY)
  406 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
      KETURN
      END
C
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),
     + 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
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)
      WRITE (6,235)
  280 FORMAT (//,23H INDIVILUAL 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
```

```
      IF (IPRT.EQ.2) GO TO 500
  310 FORMAT (3X,6A4,3(1X,F10.5))
  400 CONTINUE
      WRITE (6,430)
  430 FORMAT (//,30H LOWER AND UPPER BOUNDS ON NO.,
     +12H OF UNITS OF/27H EACH PLANT CODE EACH YEAR:)
      WRITE (6,435)
  435 FORMAT (19H (EXPANSION PLANTS))
      WRITE (6,450) (KH(I),I=4,6)
  450 FORMAT (/,13H HYD PLT CODE,3I10)
      WRITE (6,455)
  455 FORMAT (/,15X,3(4X,1HL,4X,1HU))
      DO 460 NY=1,15
      KY=NY+1980
      WRITE (6,470) KY,(IHUL(JH,NY),IHUH(JH,NY),JH=4,6)
  460 CONTINUE
  470 FORMAT (/,5X,I5,5X,6I5)
      WRITE (6,280)
      WRITE (6,435)
      WRITE (6,290) (KH(I),I=4,6)
      DO 495 J=1,11
      WRITE (6,310) (KEY2(J,K),K=1,6),(HPLR(JH,J),JH=4,6)
  495 CONTINUE
  500 CONTINUE
      WRITE (6,520)
  520 FORMAT (//,34H *   K$ MEANS DOLLARS IN THOUSANDS)
      RETURN
      END
C
C
C
C
C     ---------------------------------
C     SUBROUTINE PSDA
C     ---------------------------------
C
C
C
C     THIS PROGRAM ACCEPTS INPUT DATA FOR OPERATING AND
C     COST PARAMETERS OF PUMPED STORAGE PLANTS.
C     THE ENTIRE PROCEDURE IS INTERACTIVE.
C
C
      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),
     + NYR(20)
      INTEGER RESP,KEY2,KEY3,YES,YE
      DATA YES,YE/3HYES,3HY  /
      IP=1
      ITYP=1
      KHU1=0
      JNUM=8
      KHMX=1
      KHM=3
      KH1=1
      DO 100 K=1,34
      READ (41,110) (KEY3(K,J),J=1,18)
  100 CONTINUE
  110 FORMAT (18A4)
C
C     BEGINNING OF INPUT PROCEDURE
C
C
      WRITE (6,120)
```

```
  120 FORMAT (//,5X,35H INPUT PROCEDURE FOR PUMPED STORAGE,
     +21H PLANT SYSTEM BEGINS.)
      WRITE (6,123)
  123 FORMAT (//,32H DO YOU HAVE A FILE FOR EXISTING,
     +23H PUMPED STORAGE 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 (//,38H BEGIN YOUR INPUT FOR EXISTING STORAGE,
     +8H PLANTS.,/38H THE MAX. NO. OF SUCH PLANTS SHOULD BE,I3)
  152 CONTINUE
      WRITE (6,160)
  160 FORMAT (//,36H ENTER PLANT CODE NUMBER (2 DIGITS).,
     +//3H --)
      READ(5,170) KHU
  170 FORMAT (I2)
      IF (KHU.EQ.KHM) KHMX=KHM
  172 CONTINUE
      CALL HUNIT
  175 CONTINUE
      CALL HCREAT
C
C     CHANGES/CORRECTIONS OF INPUT DATA
C
  217 CONTINUE
      WRITE (6,220) KHU
  220 FORMAT (//,41H DO YOU WANT TO CHANGE ANY DATA FOR PLANT,
     +I3,1H?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 230
      IF (ITYP.EQ.2) GO TO 750
      IF (KHU.LT.2.AND.KHMX.LT.2) GO TO 250
      WRITE (6,257)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 417
  229 CONTINUE
      GO TO 250
  230 WRITE (6,240) KHU
  240 FORMAT (/,39H DO YOU WANT TO CHANGE ALL THE DATA FOR,
     +6H PLANT,I3,1H?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE)  GO TO 172
      CALL HCHNG
      IF (ITYP.EQ.2) GO TO 750
      IF (KHU.LT.2.AND.KHMX.LT.2) GO TO 250
  248 CONTINUE
      WRITE (6,257)
  257 FORMAT (/,31H DO YOU WANT TO CHANGE DATA FOR,
     +15H ANOTHER PLANT?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 417
  250 CONTINUE
      IF (KHU.GE.KHM.OR.KHMX.GE.KHM) GO TO 440
C
C     ADDITIONS OF MORE PLANTS TO THE SYSTEM
C
      WRITE (6,260)
  260 FORMAT (//,38H DO YOU WANT TO ADD MORE PLANTS TO THE,
     +8H SYSTEM?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 152
      KHMX=KHU
C
      GO TO 440
  306 CONTINUE
      READ (45,575,END=146) KHU
  154 FORMAT (313)
```

```
      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,I3)
C     DO 161 KHU=1,KHMX
C     READ (47,575) KHU
C     READ (47,590) (IHUF(KHU,K),K=1,15)
C 161 CONTINUE
      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(//,I3)
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 (I1)
      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(/,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)
  580 FORMAT (1H ,5F10.5)
  560 CONTINUE
  570 FORMAT (/,1H ,3I3)
```

```
  585 FORMAT (1H ,5E13.5)
  550 CONTINUE
      REWIND 47
C     DO 554 KHU=1,KHMX
C     WRITE (47,575) KHU
C     WRITE (47,590) (IHUF(KHU,K),K=1,15)
C 554 CONTINUE
  590 FORMAT (20I3)
      DO 540 NY=1,15
      WRITE (48,575) NY
      WRITE (48,587) (HPLF(J,NY),J=1,JNUM)
  587 FORMAT (5E12.5)
  540 CONTINUE
  600 CONTINUE
  500 CONTINUE
  602 ITYP=2
      JNUM=11
      KHU=KHMX+KH1
      KHM1=KHU
      WRITE (6,605)
  605 FORMAT (//,33H DO YOU HAVE A FILE FOR EXPANSION,
     +23H PUMPED STORAGE PLANTS?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 800
      WRITE (6,610)
  610 FORMAT (//,42H DO YOU WANT TO CONSIDER STORAGE PLANTS AS,
     +/22H EXPANSION CANDIDATES?,17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 710
      GO TO 900
  710 WRITE (6,715) KH1,KHU
  715 FORMAT (//,5H ONLY,I2,14H PLANT TYPE IS,
     +26H CONSIDERED FOR EXPANSION./15H IT IS ASSIGNED,
     +15H PLANT CODE NO.,I3)
      GO TO 172
  750 CONTINUE
      WRITE (6,720)
  720 FORMAT (//,33H IS YOUR INPUT FILE FOR EXPANSION,
     +25H STORAGE PLANTS COMPLETE?/17H ENTER YES OR NO.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 850
      GO TO 217
  800 CONTINUE
      WRITE (6,715) KH1,KHU
      READ (49,590) (IHUL(KHU,K),K=1,15)
      READ (49,590) (IHUH(KHU,K),K=1,15)
      READ (49,587) (HPLR(KHU,J),J=1,JNUM)
      WRITE (6,832)
  832 FORMAT (//,31H ENTER YES IF YOU WANT TO PRINT,
     +23H EXPANSION SYSTEM DATA.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) CALL SPRINT(3)
      GO TO 217
  850 CONTINUE
      REWIND 49
      WRITE (49,590) (IHUL(KHU,K),K=1,15)
      WRITE (49,590) (IHUH(KHU,K),K=1,15)
      WRITE (49,587) (HPLR(KHU,J),J=1,JNUM)
  900 CONTINUE
      WRITE (6,965)
  965 FORMAT (//,38H PRINTING OF THE ENTIRE PUMPED STORAGE,
     +21H SYSTEM DATA FOLLOWS./ 17H IF YOU WANT THIS,
     +20H PRINTING,ENTER YES.)
      READ (5,130) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE)  CALL SPRINT(1)
 1000 CONTINUE
C
C     END OF INPUT PROCEDURE
```

```
C
      WRITE (6,654)
  654 FORMAT (//,42H END OF INPUT PROCEDURE FOR PUMPED STORAGE,
     +8H PLANTS.,/,41H INPUT DATA STORED ON UNITS 45,48 AND 49.)
      RETURN
      END
C
      SUBROUTINE SPRINT(IPRT)
      COMMON/TEN/HPLR(6,12),HPLI(6,12),HPLF(12,20),
     +      HPLRI(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),
     + 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 (6,235)
  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 (6,520)
      IF (IPRT.EQ.2) GO TO 500
  310 FORMAT (2X,6A4,3(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),IHUH(4,NY)
  460 CONTINUE
  470 FORMAT (/,5X,I5,5X,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
```

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

```
C*****-----*****-----*****-----*****-----*****-----*****-----*****-----*****
C-----*****-----*****-----*****-----*****-----*****-----*****-----*****-----
C      SUBROUTINE LOADI
C
C         THIS PROGRAM READ LOAD DATA AND CALCULATES LOAD DURATION
C         CURVES AND CUMULANTS FOR EACH SEASON EVERY EVERY YEAR.
C       F(FT20F001) DA(PR20.DATA)
C       F(FT21F001) DA(LF21.DATA)
C       F(FT25F001) DA(DUR25.DATA)
C       F(FT26F001) DA(CUM26.DATA)
C
C
      COMMON  /YEAR/ NBYR,NBY1,IEND,ICY,NY
      INTEGER YES,Y,IRESP
      DATA IYES,IY/'YES','Y'/
      REWIND 11
      READ (11,20) NDUM1,NDUM2 ,NBYR,IEND
      NBY1=NBYR-1
      REWIND 11
   20 FORMAT(4I10)
   99 FORMAT(//' ********** START OF LOAD INPUT PROCEDURE **********')
      PRINT 99
C
C C C C C C C C C C C C C C C C C   READ LOAD CHARACTERITICS
C
      CALL MULT
C
C C C C C C C C C C C C C C C C C C READ REFERENCE LOAD MULTIPLIER
C
      NPRD=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
C
C C C C C C C C C C C C C C C C C    CALCULATE LOAD DURATION CURVE
C                                   AND CUMULANT
      DO 200 I=NBYR,IEND
      NY=I-NBY1
      ICY=I
      CALL DURAT
      CALL CUMUL
  200 CONTINUE
   50 WRITE (6,106)
  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
C
C
C
C***********************************
C
       SUBROUTINE MULT
C       THIS SUBPROGRAM READ THE ENERGY MULTIPLIER AND LOAD FACCTOR
       COMMON  /YEAR/ NBYR,NBY1,IEND,ICY,NY
       COMMON /CHRLD/ EMLPY(4,30),FLD(4,30)
     +         , AVRB(4,5),PKAB(4,5), 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
      IF( IANS.EQ.YES.OR.IANS.EQ.Y) GO TO 201

C 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( 4F11.4)
   70 CONTINUE
  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 C        PRINT TO VARIFY

  201 CONTINUE
      REWIND 21
      READ (21,510)  EMLPY,FLD
      REWIND 21
  202 CONTINUE
      PRINT 200
  200 FORMAT('              ENERGY MULTIPLYER FOR EACH SEASON EACH YEAR')
      DO 270 I=NBYR,IEND
      NY=I-NBY1
      DO 231 J=1,4
      IF(EMLPY(J,NY).EQ.0) EMLPY(J,NY)=1.0
      IF(FLD(J,NY).EQ.0) FLD(J,NY)=0.6
  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 C        CHANGE DATA

      PRINT 290
  290 FORMAT('      IS YOUR INPUT COMPLETE ?    ENTER YES OR NO.')
      READ 99,IANS
      IF( IANS.EQ.YES.OR.IANS.EQ.Y) GO TO 500
  300 FORMAT('    YEAR',I5,' : ',4F10.5)
```

```
315 PRINT 330
330 FORMAT(9X,' DO YOU WANT TO CHANGE ENERGY MULTIPLIER?'/
   +,3X,'           ENTER YES OR NO.')
    READ 99,IANS
    IF(IANS.NE.YES.AND.IANS.NE.Y) GO TO 415
    PRINT 340
340 FORMAT('                 ENTER WHAT YEAR.')
    READ *, I
    NY=I-NBY1
    PRINT 40, I
    PRINT 50
    READ 80,(EMLPY(J,NY),J=1,4)
    GO TO 315

415 PRINT 430
430 FORMAT('               DO YOU WANT TO CHANGE LOAD FACTORS?'/
   +'          ENTER YES OR NO.')
    READ 99,IANS
 99 FORMAT(A4)
    IF(IANS.NE.YES.AND.IANS.NE.Y) GO TO 202
    PRINT 340
    READ *, I
    NY=I-NBY1
    PRINT 40, I
    PRINT 50
    READ 80, (FLD(J,NY),J=1,4)
    GO TO 415
500 WRITE (21,510) EMLPY,FLD
510 FORMAT(1P5E12.5)
    RETURN
    END
*****************************************************************************

    SUBROUTINE PANDA
    COMMON  /YEAR/ NBYR,NBY1,IEND,ILY,NY
    COMMON /CHRLD/ EMLPY(4,30),FLD(4,30)
   +          , AVKB(4,5),PKAB(4,5), HL(24), PK(4),BSL(4)
   +          , SUM(4),AVL(4),ELD(500),ELF(4)
    DIMENSION IHRL(24)
    INTEGER SSN(4)
    DATA SSN/'WI','SP','SU','AU'/
    PKMAX=0.0
    INT=20
    NPER=4

    DO   10 K=1,4
    BSL(K)=1.0E10
    ND=91
    IF (K.EQ.4) ND=92
    DO   20 ID=1,ND
    READ (INT,899,END=55) (IHRL(I),I=1,24)
 55 CONTINUE
    DO   30 I=1,24
    HL(I)=IHRL(I)
    SUM(K)=SUM(K)+HL(I)
899 FORMAT (20X,12I5)
    IF (HL(I).GT.PK(K)) PK(K)=HL(I)
    IF (HL(I).NE.0.AND.HL(I).LT.BSL(K)) BSL(K)=HL(I)
 30 CONTINUE
 20 CONTINUE
    AVL(K)=SUM(K)/ND/24
    ELF(K)=AVL(K)/PK(K)
 10 CONTINUE

    REWIND INT
    DO   40 K=1,NPER
    IF (PK(K).GT.PKMAX) PKMAX=PK(K)
 40 CONTINUE
```

```
107 FORMAT (9X ,'THE ANNUAL PEAK  OF THE REFERENCE LOAD DATA IS ',
   +1PE13.4,' MW')
    WRITE (6,108)
108 FORMAT(/'     SSN     PEAK LD(MW)   MIN LD(MW)   AVE LD(MW)',
   + '    TOT ENER(MWH)   LOAD FACTOR')
    DO   70 K=1,NPER
    WRITE (6,109) SSN(K),PK(K),BSL(K),AVL(K),SUM(K),ELF(K)
109 FORMAT (3X ,A4,1X,1P5E13.5)
 70 CONTINUE
    WRITE (6,107) PKMAX
    PRINT 109

    REWIND INT

    DO 140 K=1,4
    PTS=PK(K)-BSL(K)
    DO 166 NN=1,4
    AVRB(K,NN)=0.0
166 CONTINUE
    NPRD=91
    IF(K.EQ.4) NPRD=92
    DO 180 ND=1,NPRD
    READ (INT,899,END=155) (IHRL(I),I=1,24)
155 CONTINUE
    DO 160 I=1,24
    HL(I)=IHRL(I)
    AVRB(K,1)=AVRB(K,1) +HL(I)*(1.0-(HL(I)-BSL(K))/PTS*0.4)
    AVRB(K,2)=AVRB(K,2)+HL(I)*(1.0-(HL(I)-BSL(K))/PTS*0.2)
    AVRB(K,3)=AVRB(K,3)+HL(I)
    AVRB(K,4)=AVRB(K,4)+HL(I)*(1.0+(HL(I)-BSL(K))/PTS*0.2)
    AVRB(K,5)=AVRB(K,5)+HL(I)*(1.0+(HL(I)-BSL(K))/PTS*0.4)
160 CONTINUE
180 CONTINUE
    YY=-0.6
    DO 151 NN=1,5
    YY=YY+0.2
    BX=NPRD*24
    AVRB(K,NN)=AVRB(K,NN)/BX
    PKAB(K,NN)=PK(K)*(1.0+YY)/AVRB(K,NN)
151 CONTINUE
140 CONTINUE
    RETURN
    END

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

    SUBROUTINE DURAT
    COMMON /AVRAGE/ AVT(4)
    DOUBLE PRECISION AVT
    COMMON  /YEAR/ NBYR,NBY1,IEND,ICY,NY
    COMMON /CHRLD/ EMLPY(4,30),FLD(4,30)
   +          , AVRB(4,5),PKAB(4,5), HL(24), PK(4),BSL(4)
   +          , SUM(4),AVL(4),ELD(500),ELF(4)
    DIMENSION IHRL(24)
    DOUBLE PRECISION ETOT,AVLD
108 FORMAT('     SSN     PEAK LD(MW)   MIN LD(MW)   AVE LD(MW)',
   + '    TOT ENER(MWH)   LOAD FACTOR')
    PRINT 100, ICY
100 FORMAT(/'       YEAR OF ',I4)
    PRINT 108
    PKMAX=0.0
    PTS=500
    INT=20
    REWIND INT
    IPNT=500
    DO   214 K=1,4
    ETOT = 0.0
    CALL KONST(K,C1,C2,PKF)
```

```
      DO 209 J=1,IPNT
  209 ELD(J)=0.0
      NPRD=91
      IF (K.EQ.4) NPRD=92
      DX=(PK(K)*PKF-BSL(K)*C1)/FLOAT(IPNT-1)
      PRINT *,DX,C1,C2,PKF
      DO  218 ND=1,NPRD
      READ (INT,899,END=255) (IHRL(I),I=1,24)
  899 FORMAT(20X,12I5)
  255 CONTINUE

      DO  216 I=1,24
      HL(I)=IHRL(I)
      EKK= PK(K)-BSL(K)
      HL(I)=HL(I)*( C1+(HL(I)-BSL(K))/EKK*C2)
      IF( HL(I).EQ.0) GO TO 216
      LA=(HL(I)-BSL(K)*C1)/DX+1.0
      IF(LA.LT.1.OR.LA.GT.IPNT) GO TO 400
      ELD(LA)=ELD(LA)+1.0
      ETOT=ETOT+HL(I)
  216 CONTINUE
  300 FORMAT(10F7.0)
  218 CONTINUE
      DO 340 IX=2,IPNT
      I=IPNT+1-IX
      ELD(I)=ELD(I)+ELD(I+1)
  340 CONTINUE
C C C C C C C C C C C C          MAKE A FILE OF LOAD DURATION
      WRITE (25) ICY,K
  355 FORMAT(/'   YEAR=',I5,'    SEASON= ',I5)
      BSLD=BSL(K)*C1
      PKC=PK(K)*PKF
      AVLD=ETOT/(24*NPRD)
      WRITE (25) DX,BSLD,PKC,ETOT,AVLD
      WRITE (25) ELD
      AVT(K)=AVLD
      ELB=AVLD/PKC
      PRINT 109,K,   PKC,BSLD,AVLD,ETOT,ELB
  109 FORMAT (3X ,I4,1X,1P5E13.5)
  350 FORMAT(1P5E12.5)
  214 CONTINUE
      RETURN

  400 PRINT 410,LA,I,ND
  410 FORMAT('      ERROR WITH UNIT 20 :   SEE SUBROUTINE DURAT')
      REWIND INT
      RETURN
      END
*******************************************************************

      SUBROUTINE CUMUL
      COMMON /AVRAGE/ AVT(4)
      DOUBLE PRECISION AVT
      COMMON  /YEAR/ NBYR,NBY1,IEND,ICY,NY
      COMMON /CHRLD/ EMLPY(4,30),FLD(4,30)
     +          , AVRB(4,5),PKAB(4,5), HL(24), PK(4),BSL(4)
     +          , SUM(4),AVL(4),ELD(500),ELF(4)
       DIMENSION AVG(4),HOURC(24),HRL(24),EM(8),IHRL(24)
      DOUBLE PRECISION HOURC,EM
      INT=20
      REWIND INT
      DO 1008 J=1,8
 1008 EM(J)=0.0
      DO 1004 K=1,4
      PRINT 110
      CALL KONST(K,C1,C2,PKF)
      NPRD=91
```

```
      IF (K.EQ.4) NPRD=92

      DO 1020 ND=1,NPRD
      READ (INT,899,END=66) (IHRL(I),I=1,24)
   66 CONTINUE
      DO 1025 I=1,24
      HL(I)=IHRL(I)
      EKK= PK(K)-BSL(K)
      HL(I)=HL(I)*( C1+(HL(I)-BSL(K))/EKK*C2)
      HRL(I)=HL(I)-AVT(K)
      HOURL=1.0
      DO 1030 J=1,8
      HOURL=HOURL*HRL(I)
 1030 EM(J)=EM(J)+HOURL
 1025 CONTINUE
 1020 CONTINUE

      DO 1035 J=1,8
 1035 EM(J)=EM(J)/(NPRD*24)
      HOURC(1)=EM(1)
      HOURC(2)=EM(2)
      HOURC(3)=EM(3)
      HOURC(4)=EM(4)-3.*EM(2)*EM(2)
      HOURC(5)=EM(5)-10.*EM(3)*EM(2)
      HOURC(6)=EM(6)-15.*EM(4)*EM(2)-10.*EM(3)*EM(3)+
     + 30.*EM(2)*EM(2)*EM(2)
      HOURC(7)=EM(7)-21.*EM(5)*EM(2)-35.*EM(4)*EM(3)+
     + 210.*EM(3)*EM(2)*EM(2)
      HOURC(8)=EM(8)-28.*EM(6)*EM(2)-56.*EM(5)*EM(3)-
     + 35.*EM(4)*EM(4)+420.*EM(4)*EM(2)*EM(2)+
     + 560.*EM(3)*EM(3)*EM(2)-630.*EM(2)*EM(2)*EM(2)*EM(2)
  355 FORMAT(7X,'  YEAR=',I5,'    SEASON= ',I1,4X,'  AVERAGE=',1PE20.10)
      HOURC(1)=AVT(K)
      PRINT 110
      PRINT 134
      WRITE (6,355) ICY,K,AVT(K)
  134 FORMAT (1H ,'THE LOAD CUMULANTS FOR ',I5,1X,I4,' ARE')
      WRITE (6,110) (HOURC(J),J=1,8)
      WRITE (26,355) ICY,K,AVT(K)
      WRITE (26,110) (HOURC(J),J=1,8)
 1004 CONTINUE

  110 FORMAT(1P3E23.14)
  899 FORMAT(20X,12I5)
      REWIND INT
      RETURN
      END

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

      SUBROUTINE KONST(K,C1,C2,PKF)
      COMMON  /YEAR/ NBYR,NBY1,IEND,ICY,NY
      COMMON /CHRLD/ EMLPY(4,30),FLD(4,30)
     +         , AVRB(4,5),PKAB(4,5), HL(24), PK(4),BSL(4)
     +         , SUM(4),AVL(4),ELD(500),ELF(4)
      PTAA=1.0/FLD(K,NY)
      XA=-0.6
      DO 40 NN=2,5
      XA=XA+0.2
      XB=XA+0.2
      IF(PTAA.GT.PKAB(K,NN)) GO TO 30
   10 PTAB=PKAB(K,NN)
      PTAR=PKAB(K,NN-1)
      X=0.2*(PTAA-PTAR)/(PTAB-PTAR)+XA
      AVA=((XB-X)*AVRB(K,NN-1)+(X-XA)*AVRB(K,NN))/0.2
      C1=AVL(K)*EMLPY(K,NY)/AVA
      C2=C1*X
      PKF=C1+C2
```

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

```
C                       **** PREP SUBMODULE ****
C
C
C    THIS SUBMODULE DETERMINES, FOR EACH YEAR IN THE STUDY PERIOD
C
C      (1) MINIMUM AND MAXIMUM RESERVES
C      (2) ECONOMIC LOADING ORDER
C      (3) LOWER BOUND OF OPERATION COSTS
C
C
C    ON INPUT :
C      (1) PLANT DATA (UNIT 11)
C      (2) NORMAL DISTRIBUTION TABLE (UNIT 15)
C
C    ON OUTPUT :
C      (1) FINAL LDC AFTER THE CONVOLUTION OF
C          SCHEDULED SYSTEM FOR EACH YEAR(UNIT 22)
C      (2) ECONOMIC LOADING ORDER AND SCHEDULED
C          SYSTEM MAINTENANCE ALLOCATION FOR EACH
C          YEAR (UNIT 24)
C      (3) HOURLY LOAD CUMULANTS AND SCHEDULED
C          SYSTEM OUTAGE CUMULANTS FOR EACH
C          YEAR(UNIT 27)
C      (4) PLANT DATA (UNIT 23)
C      (5) OTHER INFORMATION REQUIRED BY
C          DYNO-SUBMODULE(UNIT 10).
C
C
C
C
C
C
C                       COMMON VARIABLE DESCRIPTION
C
C
C   NAME     TYPE    SIZE                    DEFINITION
C   ----     ----    ----     ------------------------------------------------
C
C   AVSPE    R*4     20,4     AVSPE(I,J)--AVAILABLE SPACE FOR MAINTENANCE
C                                        ALLOCATION
C                             I : YEAR INDEX
C                             J : SEASON INDEX
C   COSTOP   R*4     20       LOWER BOUND OF OPERATING COSTS FOR EACH YEAR
C   CMULT    R*4     4        SEASONAL CAPACITY MULTIPLIER
C   EAVAL    R*4     3        EAVAL(I)--MAX AVAILABLE HYDRO ENERGY IN MWH
C                             I=1   SCHLD HYDRO PLANT
C                             I=2   EXPN HYDRO PLANT
C                             I=3   PUMPED STORAGE
C   EC       R*4     20       SCHEDULED SYSTEM CAPACITY FOR EACH YEAR
C   EL       R*4     4,1250   EL(I,J)--NORMALIZED LOAD DURATION CURVE
C                             I : SEASON INDEX
C                             J : ABSCISSA INDEX
C   EMOR     R*4     220,4    EMOR(I,J)--THERMAL PLANT MAINTENANCE OUTAGE
C                             I : PLANT INDEX (EXPN&SCHLD)
C                             J : INDEX OF SEASON IN WHICH MAINTENANCE IS
C                                 SCHEDULED
C   EMORH    R*4     3,4      EMORH(I,J)--HYDRO PLANT MAINTENANCE OUTAGE
C                             I=1   SCHEDULED HYDRO
C                             I=2   EXPANSION HYDRO
C                             I=3   PUMPED STORAGE
C                             J :   SAME AS IN EMOR
C   EMULT    R*4     4        SEASONAL ENERGY MULTIPLIER
C   EXPLCU   R*4     4,8      EXPLCU(I,J)--EXPN PLANT OUTAGE CUMULANTS
C                             I : FIRST FOUR CUMULANTS
C                             J : PLANT TYPE
C   HRCUM    R*8     4,8      HRCUM(I,J)--LOAD CUMULANTS FOR EACH SEASON
C                             I : SEASON INDEX
C                             J : CUMULANT ORDER
```

```
C   HYDR     R*4    3,11     HYDR(I,J)--HYDRO PLANT CHARACTERISTICS
C                            I=1   SCHEDULED HYDRO
C                            I=2   EXPANSION HYDRO
C                            I=3   PUMPED STORAGE
C                            J=1   I=1,2 : BASE CAPACITY MW
C                                  I=3   : STORAGE CAPACITY MW
C                            J=2   I=1,2 : MAX CAPACITY MW
C                                  I=3   : ENERGY LIMIT GWH
C                            J=3   I=1,2 : STR ENERGY LIMIT GWH
C                                  I=3   : PUMPING EFFICIENCY
C                            J=4   I=1,2 : ENERGY LIMIT GWH
C                                  I=3   : GENERATING EFFICIENCY
C                            J=5         : FORCED OUTAGE RATE
C                            J=6         : MAINTENANCE OUTAGE D/YR
C                            J=7         : FIXED O&M COST K$/MW/YR
C                            J=8         : VARIABLE O&M COST $/MWH
C                            J=9         : CAPITAL COST $/KW
C                            J=10        : CAPITAL COST ESCALLATION RATE
C                            J=11        : SALVAGE VALUE K$
C   HYEXPN   R*4    11       HYEXPN(I)--HYDRO EXPN CHARACTERISTICS
C                            I : SAME AS IN J OF HYDR ARRAY
C   IBK      I      500,3    IBK(I,J)--ECONOMIC LOADING ORDER
C                            I : LOADING POSITION INDEX
C                            J=1 PLANT CODE NUMBER
C                            J=2 CAPACITY BLOCK IDENTIFIER
C                            J=3 NO OF UNITS (-1 DENOTES AN EXPN TYPE)
C   ID       I      -        FLAG USED IN MIN RESERVE CALCULATION
C                            ID=1  SCHEDULED SYSTEM CONVOLUTION
C                            ID=2  EXPANSION ADDITION
C                            ID=3  RESIMULATION WITH MIXED SYSTEM
C   IPMAX    I      -        TOTAL NO OF THERMAL PLANT TYPES CONSIDERED
C   IPT      I      -        NO OF POINTS DESIRED IN PW-LOLP CALCULATION
C   ITK      I      500,3    ITK(I,J)--EFFECTIVE LOADING ORDER FOR
C                                      OPERATING COST CALCULATION
C                            I : SAME AS IN IBK
C                            J : SAME AS IN IBK
C   IUP      I      20,20    IUP(I,J)--ALLOWED UPPER BOUND OF THERMAL EXPN
C                                      CANDIDATES
C                            I : PLANT INDEX
C                            J : YEAR INDEX
C   IX       I      20       THERMAL EXPN PLANT TYPES INDEX ARRAY
C   IYRB     I      -        BEGINNING YEAR OF STUDY PERIOD
C   JYR      I      -        STUDY YEAR COUNTER
C   KMAX     I      -        MAX NO ON LOADING POSITIONS
C   MAINS    I      200      SCHEDULED THERMAL SYSTEM MAINTENANCE
C                            ALLOCATION SPECIFICATION
C   MAXI     I      -        EXPECTED MAX NO OF POINTS IN X-ARRAY FOR LOLP
C                            CALCULATION
C   MX       I      -        MAX NO OF POINTS ALLOWED IN LOLP CALCULATION
C   MXELDC   I      20,4     MXELDC(I,J)--MAX NO OF POINTS IN THE FINAL EL
C                                         CURVE AFTER THE CONVOLUTION OF
C                                         SCHLD THERMAL SYSTEM
C                            I : YEAR INDEX
C                            J : SEASON INDEX
C   NHYMN    I      20       MIN NO OF HYDRO EXPN UNITS SPECIFIED FOR EACH
C                            STUDY YEAR
C   NHYMX    I      20       MAX NO OF HYDRO EXPN UNITS ALLOWED FOR EACH
C                            STUDY YEAR
C   NP       I      -        NO OF SIMULATION SEASONS(PERIODS)
C   NPEXP    I      -        NO OF THERMAL EXPN PLANT TYPES
C   NPSMN    I      20       MIN NO OF PUMPED STORAGE EXPN UNITS SPECIFIED
C                            FOR EACH STUDY YEAR
C   NPSMX    I      20       MAX NO OF PUMPED STORAGE EXPN UNITS ALLOWED
C                            FOR EACH YEAR
C   NSTPRE   I      20,10    NSTPRE(I,J)--EXPN ADDITION CONFIGURATION FOR
C                                         EACH YEAR
C                            I : YEAR INDEX
C                            J : PLANT TYPE
```

```
C     NT       I      20        CUMULATIVE NO OF UNITS OF THERMAL EXPANSION
C                               ADDITION FOR EACH PLANT TYPE
C     NTHY     I      20        CUMULATIVE NO OF UNITS OF HYDRO EXPANSION
C                               FOR EACH YEAR
C     NTPS     I      20        CUMULATIVE NO OF UNITS OF PUMPED STORAGE
C                               EXPANSION FOR EACH YEAR
C     NTYR     I      20        LENGTH OF STUDY PERIOD
C     NUF      I      220,20    NUF(I,J)--FOR THERMAL SCHLD SYSTEM : MAX NO OF
C                                         UNITS ALLOWED FOR EACH TYPE
C                                         FOR THERMAL EXPN SYSTEM : LOWER
C                                         BOUND FOR EACH TYPE
C                               I : PLANT INDEX
C                               J : YEAR INDEX
C     NX       I      20        NO OF THERMAL EXPN ADDITION FOR EACH TYPE
C     NXHY     I      20        NO FO HYDRO EXPN ADDITION FOR EACH YEAR
C     NXPS     I      20        NO OF PUMPED STORAGE EXPN ADDITION FOR
C                               EACH YEAR
C     PEAK     R*4    20,4      PEAK(I,J)--SEASONAL PEAK LOAD DEMAND FOR EACH
C                                          YEAR IN MW
C                               I : YEAR INDEX
C                               J : SEASON INDEX
C     PERENE   R*4    20,4      PERENE(I,J)--TOTAL ENERGY DEMAND IN MWH FOR
C                                            EACH SEASON EACH YEAR
C                               I : YEAR INDEX
C                               J : SEASON INDEX
C     PLANT    R*4    200,14    PLANT(I,J)--THERMAL PLANT CHARACTERISTICS
C                               I :   PLANT CODE NUMBER
C                               J=2   BASE CAPACITY MW
C                               J=3   MAX CAPACITY MW
C                               J=4   MAINT OUTAGE DAYS/YR
C                               J=5   FORCED OUTAGE RATE
C                               J=6   CAPITAL COST $/KW
C                               J=7   BASE FUEL COST $/MWH
C                               J=8   MAX FUEL COST $/MWH
C                               J=9   PLANT ECONOMIC LIFE YR
C                               J=10  FIXED O&M COST $/MW/YR
C                               J=11  VARIABLE O&M COST $/MWH
C                               J=12  SALVAGE VALUE K$
C                               J=13  CAPITAL ESCALLATION RATE
C                               J=14  FUEL ESCALLATION RATE
C     PLACA    R*4    20,14     PLACA(I,J)--THERMAL EXPN CHARACTERISTICS
C                               I : PLANT INDEX
C                               J : SAME AS IN PLANT ARRAY
C     PPMAX    R*4    -         LOLP VALUE ASSIGNED FOR MAX RESERVE
C                               CALCULATION
C     PPMIN    R*4    -         LOLP VALUE ASSIGNED FOR MIN RESERVE
C                               CALCULATION
C     PSEXP    R*4    11        PSEXP(I)--PUMPED STORAGE EXPN CHARACTERISTICS
C                               I : SAME AS IN HYDR ARRAY
C     PTCUM    R*8    211,16    PTCUM(I,J)--THERMAL PLANT OUTAGE CUMULANTS
C                               I : PLANT INDEX
C                               J=1  8  FOR BASE BLOCK
C                               J=9  16 FOR PEAK BLOCK
C     PTCUMH   R*8    3,16      PTCUMH(I,J)--HYDRO PLANT OUTAGE CUMULANTS
C                               I=1   SCHLD HYDRO PLANT
C                               I=2   EXPN HYDRO PLANT
C                               I=3   PUMPED STORAGE
C                               J :  SAME AS IN PTCUM ARRAY
C     RSMAX    R*4    20        MAX RESERVE FOR EACH YEAR IN MW
C     RSMIN    R*4    20        MIN RESERVE FOR EACH YEAR IN MW
C     SCHCUM   R*4    4,20      SCHCUM(I,J)--SUM OF LOAD&SCHLD PLANT CUMULANTS
C                                            FOR PEAK PERIOD FOR EACH YEAR
C                               I : FIRST FOUR CUMULANTS
C                               J : YEAR INDEX
C     TINT     R*8    120       CUMULATIVE NORMAL DISTRIBUTION
C                               CORRESPONDING TO TY
C     TY       R*8    120       NORMALIZED INDEP VARIABLE IN NORMAL DISTR.
C                               TABLE
```

```
C   X        R*4    1250     NORMALIZED MW ARRAY
C
C
C
C     --------------
C     MAIN PROGRAM
C     --------------
C
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)
C
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)
C
C
      COMMON/THREE/ AVSPE(20,4),COSTOP(20),EC(20),IBK(500,3),
     +              ITK(500,3),MXELOC(20,4),NSTPRE(20,10),
     +              PERENE(20,4)
C
C
      COMMON/FOUR/  ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +              NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/  HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/   TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
C
      DATA K11,K15/11,15/
C
C
C     UERS SPECIFICATION OF PPMIN & PPMAX
C
C
C
      WRITE(6,960)
  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)
  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.
C
C
C     *** S E C T I O N  1 ***
C
C     INITIALIZE PARAMETERS AND READ
C     YEAR INDEPENDENT INPUT
C
C
C
C
      NP=4
```

```
      NCUM=4
      IPT=500
      JNUM=14
      MX=1250
      DXNORM = 1.0/FLOAT(IPT)
      DO 25 I = 1,1250
   25 X(I) = FLOAT(I) * DXNORM
C
C
C     READ PLANT DATA
C
C
      READ(K11,1030) IPMAX,NPEXP,IYRB,IYRE
      NTYR=IYRE-IYRB+1
      DO 10 IP=1,IPMAX
      READ(K11,1040) (NUF(IP,N),N=1,20)
      READ(K11,1050) (PLANT(IP,J),J=1,JNUM)
      PLANT(IP,9)=30.
C
C
C     WRITE PLANT INFORMATION ON UNIT 23
C
C
      WRITE(23,904) (PLANT(IP,J),J=1,JNUM)
   10 CONTINUE
      I1=NPEXP+1
      DO 12 I=I1,IPMAX
      WRITE(23,1040) (NUF(I,N),N=1,NTYR)
   12 CONTINUE
C
C
C     STORE INFORMATION FOR EXP. SYSTEM
C
C
      DO 15 IP=1,NPEXP
      DO 15 J=1,JNUM
   15 PLACA(IP,J)=PLANT(IP,J)
C
      DO 20 IP=1,NPEXP
      READ(K11,1040) (IUP(IP,N),N=1,20)
   20 CONTINUE
C
C
C     READ NORMAL DISTRIBUTION TABLE
C
C
      READ(K15,1060) (TY(I),TINT(I),I=1,120)
C
C
C     HYDRO MULTIPLIERS
C
C
      READ(30,1080) (CMULT(I),I=1,4)
      READ(30,1080) (EMULT(I),I=1,4)
C
C
C     HYDRO EXPN SYSTEM
C
C
      READ(32,1090) NDUM
      READ(32,1100) (NHYMN(I),I=1,20)
      READ(32,1100) (NHYMX(I),I=1,20)
      READ(32,1080) (HYEXPN(J),J=1,11)
C
C
C     PUMPED STORAGE EXPN SYSTEM
C
C
```

```
      READ(34,1100) (NPSMN(I),I=1,20)
      READ(34,1100) (NPSMX(I),I=1,20)
      READ(34,1080) (PSEXP(J),J=1,11)
C
C
C     *** S E C T I O N  2 ***
C
C
C     START SIMULATION FOR EACH STUDY YEARS
C
C
      DO 30 J=1,NTYR
      JYR=J
      IYEAR=IYRB+JYR-1+1900
      WRITE(17,1070) IYEAR
C
C
C     READ YEAR DEPENDENT INPUTS
C
C
      CALL RINPUT
C
C
C     DETERMINE MIN. & MAX. RESERVES
C
C
      CALL RESERV
C
C
C     DETERMINE LOADING ORDER ACCORDING TO FUEL COST
C
C
      CALL LOADER
C
C
C     DETERMINE HYDRO COMPOSITE CHARACTERISTICS
C
C
      CALL COMPST
C
C
C     COMPUTE LOWER BOUND OF OPERATION COSTS
C
C
      CALL OPCOST(NCUM)
C
   30 CONTINUE
C
C
C     CREATE FILES FOR DP-MODULE
C
C
      CALL INOUT
C
C
      WRITE(6,1120)
C
  904 FORMAT(F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2)
 1030 FORMAT(4I10)
 1040 FORMAT(40I2)
 1050 FORMAT(5E15.5)
 1060 FORMAT(F10.4,F12.8)
 1070 FORMAT(//1H1,19('-')/' STUDY YEAR = ',I5/1X,19('-')//)
 1080 FORMAT(5E12.5)
 1090 FORMAT(/,3I3)
 1100 FORMAT(20I3)
 1120 FORMAT(//1H1,'*** END OF PREP SUBMODULE SIMULATION ***'//)
      STOP
```

```
      END
C
C
C
C     --------------------
      SUBROUTINE RESERV
C     --------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +                NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/    HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/     TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C.....................................................................
C
C     THIS ROUTINE COMPUTES THE MINIMUM RESERVE MARGIN FOR EACH
C     YEAR. IT IS OBTAINED BY REQUIRING THAT THE CRITICAL
C     SEASONAL LOLP IS LESS THAN OR EQUAL TO THE CRITICAL
C     VALUE ASSIGNED BY THE USER AT THE BEGINNING.
C
C     OBTAINED INFORMATION ARE EXPRESSED IN TERMS OF :
C
C          RSV(YEAR)    : AMOUNT IN MW
C          R            : IN PERCENT
C          NX(I)        : NUMBER OF UNIT OF TYPE IX(I)
C                         (EXPANSION SYSTEM ONLY)
C          NT(I)        : TOTAL NO OF ADDITION OF
C                         TYPE IX(I)
C                         (EXPANSION SYSTEM ONLY)
C          IX(I)        : TYPE OF UNIT
C                         (EXPANSION SYSTEM ONLY)
C     SUBROUTINE CALLED :
C          LPD
C
C.....................................................................
C
      DIMENSION IUT(20),SCAP(4)
C
C
C
C
C
```

```
C      FIND YEARLY PEAK
C
       PMAX(JYR)=0.0
       DO 10 I=1,4
       IF(PEAK(JYR,I).GT.PMAX(JYR)) PMAX(JYR)=PEAK(JYR,I)
   10  CONTINUE
C
C
C      *** S E C T I O N  1 ***
C
C      SET EXPN CANDIDATE INDEX ARRAY
C      DETERMINE UPPER BOUND OF EXPN CANDIDATE
C      INITIALIZE NX-ARRAY
C
C
C
       IF(JYR.GT.1) GO TO 30
       IX(1)=NPEXP
       IUT(1)=IUP(1,JYR)
       NX(1)=0
       NT(1)=0
       DO 20 I=2,NPEXP
       IX(I)=IX(I-1)-1
       NX(I)=0
       NT(I)=0
   20  IUT(I)=IUP(I,JYR)
       NHAVAL=NHYMX(JYR)
       NPAVAL=NPSMX(JYR)
       NXPS(JYR)=0
       NTPS(JYR)=0
       NXHY(JYR)=0
       NTHY(JYR)=0
       GO TO 50
C
C
   30  CONTINUE
C
C
C      DETERMINE NO OF UNITS AVAILABLE FOR THE PRESENT STUDY
C
C      YEAR AND INITIALIZE NX(I) ARRAY
C
       DO 40 I=1,NPEXP
       IUT(IX(I))=IUP(IX(I),JYR)-NT(I)
   40  NX(I)=0
       JPAST=JYR-1
       NHAVAL=NHYMX(JYR)-NTHY(JPAST)
       NPAVAL=NPSMX(JYR)-NTPS(JPAST)
       NXPS(JYR)=0
       NXHY(JYR)=0
C
C      *** S E C T I O N  2 ***
C
C
C      MINIMUN RESERVE CALCULATION
C      THIS IS OBTAINED BY REQUIRING THAT CRITICAL LOLP
C      IS LESS THAN OR EQUAL TO THE ASSIGNED PPMIN.
C
C
C
C      SCHEDULED SYSTEM CONVOLUTION
C
   50  ID=1
       IDL=-1
       ADD = 0.0
       CALL SYSLP(CLP,SCAP,ECP,IDL,NCRT,NHAVAL,NPAVAL)
       IF(CLP.LE.PPMIN) GO TO 1001
C
```

```
C
C     OTHERWISE PUT HYDRO EXPN UNITS ON LINE
C     ONE AT A TIME
C
C
      ID=2
      IDL=0
   60 IF(NHAVAL.LE.0) GO TO 65
      NXHY(JYR)=NXHY(JYR)+1
      ADD=ADD+HYEXPN(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
C     PUMP STORAGE EXPN ADDITION
C
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
C     OTHERWISE PUT THE SMALLEST THERMAL CANDIDATE ON LINE
C     ONE AT A TIME
C
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
C     DISTRIBUTE THIS AMOUNT BETWEEN OTHER AVAILABLE
C
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
      GO TO 160
  130 CONTINUE
  150 NX(J)=NX(J)+1
      IF(NX(J).GT.IUT(IX(J))) GO TO 110
      GO TO 170
  155 CALL SYSLP(CLP,SCAP,ECP,IDL,NCRT,NHAVAL,NPAVAL)
      IF(CLP.LE.PPMIN) GO TO 1001
C
C
C     START ADDING THE SMALLEST CANDIDATE AGAIN
C
C
      GO TO 90
  110 CONTINUE
C
      PRINT 1020
 1020 FORMAT(1X,' NO CONVERGED SOLUTION '/
     *1X,' CHECK YOUR INPUT FOR EXPANSION SYSTEM'//)
      GO TO 1100
C
C
 1001 RSMIN(JYR)=ECP-PMAX(JYR)
      IF(RSMIN(JYR).LE.0.0) RSMIN(JYR)=0.0
      RMIN=RSMIN(JYR)/PMAX(JYR)*100.0
C
C
C     STORE CURRENT EXPN ADDITION IN NSTPRE-ARRAY
C
C
      DO 900 I=1,NPEXP
      I1=NPEXP-I+1
  900 NSTPRE(JYR,I)=NX(I1)
C
C
C     DETERMINED PSEUDO-PLANT PARAMETERS
C
C
      II=IPMAX+1
      PLANT(II,2)=ADD
      PLANT(II,3)=ADD
      PLANT(II,4)=10.0
      PLANT(II,5)=0.02
      IP=IBK(1,1)
      DO 505 J=6,14
  505 PLANT(II,J)=PLANT(IP,J)
      DO 510 I=1,NPEXP
      J=NPEXP-I+1
      IF(NT(J).LE.0) GO TO 510
      IP=IX(J)
      PLANT(IP,2)=PLANT(IP,3)
      DO 520 K=4,14
  520 PLANT(IP,K)=PLANT(II,K)
  510 CONTINUE
C
C
C     REALLOCATE MAINTENANCE
C
C
      ID=3
      CALL MAINT1(ECP,IDL)
C
C
C     FIND TOTAL NO OF UNITS ADDED UP TO THE PRESENT
C     STUDY YEAR
C
C
      DO 500 I=1,NPEXP
```

```
  500 NT(I)=NT(I)+NX(I)
      IF(JYR.LE.1) GO TO 530
      NTHY(JYR)=NTHY(JPAST)+NXHY(JYR)
      NTPS(JYR)=NTPS(JPAST)+NXPS(JYR)
      GO TO 540
  530 NTHY(JYR)=NXHY(JYR)
      NTPS(JYR)=NXPS(JYR)
C
C
C
C     *** S E C T I O N   3 ***
C
C     MAX RESERVE CALCULATION
C     THIS IS OBTAINED BY REQUIRING THAT THE SMALLEST LOLP
C     IS LESS THAN OR EQUAL TO THE ASSIGNED PPMAX.
C
C
C
C
  540 IF(CLP.LE.PPMAX) GO TO 600
      CALL RESMX(ECP,SCAP,NCRL)
      RSMAX(JYR)=ECP-PMAX(JYR)
      RMAX=RSMAX(JYR)/PMAX(JYR)*100.0
      GO TO 610
  600 RSMAX(JYR)=RSMIN(JYR)*1.1
      RMAX=RMIN*1.1
C
C
C     PRINT RESULTS
C
C
C
  610 CONTINUE
C     WRITE(17,3) RSMIN(JYR),RMIN
C     WRITE(17,6) RSMAX(JYR),RMAX
C     WRITE(17,8) CLP
C
C
C     WRITE(17,4)(IX(I),I=1,NPEXP)
C     WRITE(17,5) NXHY(JYR),NXPS(JYR),(NX(I),I=1,NPEXP)
C     WRITE(17,7) NTHY(JYR),NTPS(JYR),(NT(I),I=1,NPEXP)
    3 FORMAT(/1H ,' MIN. RES. MARGIN = ',
     *F7.1,' MW ',' (',F5.2,' %)')
    6 FORMAT(/1H ,' MAX. RES. MARGIN = ',
     *F7.1,' MW ',' (',F5.2,' %)')
    4 FORMAT(/1H ,' PLANT TYPE',23X,'HY ','PS',10I3)
    5 FORMAT(/1H ,' NO OF UNITS ADDED THIS YEAR',5X,12I3)
    7 FORMAT(/1H ,' TOTAL UNITS ADDED',15X,12I3)
    8 FORMAT(/1H ,' CRITICAL LOLP = ',E14.7/)
C
 1100 RETURN
      END
C
C
C
C
      ----------------------------------------------------------
      SUBROUTINE SYSLP(CLP,SCAP,ECP,IDL,NCRT,NHAVAL,NPAVAL)
      ----------------------------------------------------------
C
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/   ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +               NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/   HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/    TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C.......................................................................
C
C     COMPUTES SEASONAL LOLP USING PIECE-WISE LINEAR
C     METHOD. MAINTENANCE OUTAGE IS CONSIDERED USING
C     EQUIVALENT CAPACITY ALGORITHM.
C
C     SUBROUTINE CALLED :
C
C        MAINT1,PWADD,PLOLP
C
C.......................................................................
C
C
      DIMENSION PLP(4),ELDC(4,1250),SUMY(4),SCAP(4)
      EQUIVALENCE (PTCUM(1),ELDC(1))
C
C
C
      K22=22
      ISTAR=(JYR-1)*4+1
C
      CALL MAINT1(ECP,IDL)
C
      PP=0.0
      CLP=0.0
C
C
C
C     LOOP FOR EACH SEASON
C
C
      DO 20 NS=1,4
      N=NS
      PK=PEAK(JYR,N)
      DX=PK/FLOAT(IPT)
      MAXI=(ECP/DX)+1+IPT
C
C
      GO TO (100,200),ID
C
  100 CONTINUE
C
C     *** S E C T I O N  1 ***
C
C     SCHEDULED SYSTEM CONVOLUTION
C     ID=1
C
C
      SUMY(N)=0.0
```

```
      SCAP(N)=0.0
C
C     HYDRO SYSTEM CONVOLUTION
C     CONVOLVE THE SCHEDULED SYSTEM FIRST
C     THEN CONVOLVE THE EXPN SYSTEM DETERMINED
C     FROM PREVIOUS STUDY YEARS
C
C     SCHEDULED HYDRO & PUMP STORAGE SYSTEM
C
      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)
C
C
      DO 110 I=1,IPMAX
      IL=IPMAX-I+1
      IF(IL.GT.NPEXP) GO TO 120
      IF(NT(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
C
C     STORE INFORMATION AFTER THE CONVOLUTION OF
C     THE ORIGINAL SCHEDULED SYSTEM
C
C
      MXELDC(JYR,N)=NMAXI
      WRITE(K22) (EL(N,K),K=1,MX)
C
  110 CONTINUE
C
C
C     EXPENSION HYDRO SYSTEM FROM PREVIOUS YEARS
C
C
      IF(JYR.LE.1) GO TO 90
      JPAST=JYR-1
      IF(NTHY(JPAST).LE.0) GO TO 80
      HMULT=FLOAT(NTHY(JPAST))
      Y=HMULT*HYEXPN(2)*(1.0-EMORH(2,N))/PK
      P=1.0-HYEXPN(5)
      SUMY(N)=SUMY(N)+Y
      SCAP(N)=SCAP(N)+Y
      CALL PWADD(Y,P,N,NMAXI)
```

```
C
C
C      PUMPED STORAGE SYSTEM FROM PREVIOUS YEARS
C
C
   80 IF(NTPS(JPAST).LE.0) GO TO 90
      HMULT=FLOAT(NTPS(JPAST))
      Y=HMULT*PSEXP(1)*(1.0-EMORH(3,N))/PK
      P=1.0-PSEXP(5)
      SUMY(N)=SUMY(N)+Y
      SCAP(N)=SCAP(N)+Y
      CALL PWADD(Y,P,N,NMAXI)
   90 S=SUMY(N)
      A=S
      IF(NHAVAL.LE.0.AND.NPAVAL.LE.0) GO TO 160
      GO TO 500
  160 DO 170 I=1,1250
  170 ELDC(N,I)=EL(N,I)
      GO TO 500
C
C
  200 CONTINUE
C
C      *** S E C T I O N  2 ***
C
C      START ADDING EXPANSION SYSTEM
C      ID=2
C      IDL=0 -- HYDRO EXPN ADDITION
C      IDL=1 -- SMALLEST THERMAL ADDITION
C      IDL=2 -- RESIMULATION
C

      IF(IDL.LE.0) GO TO 300
      IF(IDL.EQ.1) GO TO 205
C
C
C      RETRIVE FINAL LDC FROM ELDC ARRAY
C
C

      STH=SUMY(N)
      SCAP(N)=SUMY(N)
      DO 210 I=1,1250
  210 EL(N,I)=ELDC(N,I)
      GO TO 206
C
C
  300 IF(NHAVAL.LE.0) GO TO 310
      Y=HYEXPN(2)*(1.0-EMORH(2,N))/PK
      P=1.0-HYEXPN(5)
      SUMY(N)=SUMY(N)+Y
      SCAP(N)=SCAP(N)+Y
      CALL PWADD(Y,P,N,NMAXI)
      A=SUMY(N)
      IF(NHAVAL.GT.1) GO TO 500
      IF(NPAVAL.LE.0) GO TO 320
      GO TO 500
  310 Y=PSEXP(1)*(1.0-EMORH(3,N))/PK
      P=1.0-PSEXP(5)
      SUMY(N)=SUMY(N)+Y
      SCAP(N)=SCAP(N)+Y
      CALL PWADD(Y,P,N,NMAXI)
      A=SUMY(N)
      IF(NPAVAL.GT.1) GO TO 500
  320 DO 330 I=1,1250
  330 ELDC(N,I)=EL(N,I)
      GO TO 500
C
C
```

```
  205 IF(NX(1).GT.1) GO TO 206
      STH=SUMY(N)
      SCAP(N)=SUMY(N)
  206 DO 220 I=1,NPEXP
      IF(NX(I).LE.0) GO TO 220
      JE=NX(I)
      IP=IX(I)
      IN=2*IP
      JB=1
      IF(IDL.EQ.1) JE=1
      DO 230 J=JB,JE
      Y=PLANT(IP,3)*(1.0-EMOR(IN,N))/PK
      STH=STH+Y
      SCAP(N)=SCAP(N)+Y
      P=1.0-PLANT(IP,5)
  230 CALL PWADD(Y,P,N,NMAXI)
  220 CONTINUE
      A=STH
C
C
C     FIND LOLP
C
C
  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
C
C     FIND SYSTEM LOLP
C
      PLOP=PP/FLOAT(NP)
C
      RETURN
      END
C
C
C
C     ____________________________
      SUBROUTINE MAINT1(ECP,IDL)
C     ____________________________
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +                NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/  HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
```

```
      COMMON/SIX/   TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C.......................................................
C
C     ALLOCATES MAINTENANCE FOR SCHEDULED SYSTEM AND
C     EXPANSION SYSTEM ACCORDING TO FLAG CONTROL ID.
C     MAINTENANCE OUTAGE IS STORED IN EMOR(IP,NP),
C     WHERE IP=PLANT INDEX, NP=SEASON INDEX.
C
C     DEFINITION :
C
C        ID=1,     FOR 'SCHEDULED' SYSTEM (ORIGINAL SCHEDULED
C                  SYSTEM+EXP. CANDIDATES DETERMINED FROM
C                  PREVIOUS STUDY YEARS).
C        ID= 2,    FOR EXPANSION SYSTEM WITH ONE-BLOCK
C                  REPRESENTATION.
C        ID=3,     RESCHEDULE MAINTENANCE (ALLOCATE ORIGINAL
C                  SCHEDULED SYSTEM FIRST THEN NEW EXP.
C                  CANDIDATES).
C
C
C.......................................................
C
C
      DIMENSION AVSP(4),APT(4),APH(4),EP(20)
C
C
C
      RLT=365./FLOAT(NP)
      GO TO (100,200,400),ID
C
C
C     *** S E C T I O N  1 ***
C
C     ALLOCATE MAINTENANCE FOR SCHEDULED SYSTEM
C
C     ID=1
C
C
  100 EP(JYR)=0.0
      EC(JYR)=0.0
      DO 105 J=1,4
      DO 105 I=1,220
  105 EMOR(I,J)=0.0
      DO 106 I=1,3
      DO 106 J=1,4
  106 EMORH(I,J)=0.0
C
C
C     DETERMINE INSTALLED SYSTEM CAPACITY
C
C
      IB=1
      IE=IPMAX
      DO 110 I=IB,IE
      IF(I.GT.NPEXP) GO TO 111
      EP(JYR)=EP(JYR)+FLOAT(NT(I))*PLANT(IX(I),3)
      GO TO 110
  111 EP(JYR)=EP(JYR)+FLOAT(NUF(I,JYR))*PLANT(I,3)
      EC(JYR)=EC(JYR)+FLOAT(NUF(I,JYR))*PLANT(I,3)
  110 CONTINUE
      EC(JYR)=EC(JYR)+HYDR(1,2)+HYDR(3,1)
      EP(JYR)=EP(JYR)+HYDR(1,2)+HYDR(3,1)
```

```
      IF(JYR.LE.1) GO TO 112
      JPAST=JYR-1
      HMULT=FLOAT(NTHY(JPAST))
      EP(JYR)=EP(JYR)+HMULT*HYEXPN(2)
      HMULT=FLOAT(NTPS(JPAST))
      EP(JYR)=EP(JYR)+HMULT*PSEXP(1)
  112 ECP = EP(JYR)
C
C
C     HYDRO MAINTENANCE ALLOCATION
C
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=HYEXPN(6)
      Y=HYEXPN(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
C
C     THERMAL MAINTENNANCE ALLOCATION
C
C
      DO 130 I=IB,IE
      IF(I.GT.NPEXP) GO TO 131
      IF(NT(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
```

```
      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
C
C     *** S E C T I O N   2 ***
C
C     ALLOCATE MAINTENANCE FOR EXPANSION SYSTEM
C     USING 1-BLOCK REPRESENTATION
C
C     ID=2
C
C
      ECP=EP(JYR)
      HMULT=FLOAT(NXHY(JYR))
      ECP=ECP+HMULT*HYEXPN(2)
      DO 265 N=1,NP
      AVSP(N)=APT(N)
  265 AVSP(N)=AVSP(N)+HMULT*HYEXPN(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
C
```

```
C     *** S E C T I O N  2 ***
C
C
C     REALLOCATE MAINTENANCE WITH A SINGLE
C     PSEUDO PLANT REPRESENTING TOTAL CURRENT
C     THERMAL EXPANSION ADDITION
C
C
C
C     INITIALIZATION
C
      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*HYEXPN(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
      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
C
      AVSP(NN)=AVSP(NN)-PLANT(IP,3)*EMOR(IN,NN)
  435 CONTINUE
C
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(NT(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
      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))
     1*EMOR(IN,NN)
  440 CONTINUE
C
C
      IF(IB.EQ.1) GO TO 600
      DO 510 N=1,4
      AVSPE(JYR,N)=AVSPE(JYR,N)-HYDR(1,2)*EMORH(1,N)
      AVSPE(JYR,N)=AVSPE(JYR,N)-HYDR(3,1)*EMORH(3,N)
  510 CONTINUE
C
C
C
      IB=1
      IE=NPEXP
      GO TO 495
  600 CONTINUE
      EMXAS=0.0
      DO 610 L=1,NP
      IF(AVSP(L).GT.EMXAS) GO TO 620
      GO TO 610
  620 EMXAS=AVSP(L)
      LL=L
  610 CONTINUE
C
C
      EMOR(2,LL)=PLANT(1,4)/RLT
C
 1000 RETURN
      END
C
C
C
C     ----------------------------------
      SUBROUTINE PWADD(Y,P,N,NEWMAX)
C     ----------------------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +                NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/   HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/     TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
```

```
C
      DIMENSION ELF(1250)
C
C
C
      DATA PLPMAX/0.999/, ELMIN/0.000001/, ONE/1.0/
      IF(P.GE.PLPMAX) GO TO 100
      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(N,J)+
     *    (XY-X(J))/(X(J-1)-X(J))*EL(N,J-1)
      GO TO 45
   30 ELY=ONE
      GO TO 45
   40 ELY=EL(N,J)
   45 CONTINUE
      ELF( I )=P*EL(N,I)+Q*ELY
      MAXIMU=I
      IF(ELF( I ).LE.ELMIN) GO TO 55
   50 CONTINUE
   55 NEWMAX=MAXIMU
      IF (NEWMAX.GT.MAXI) NEWMAX=MAXI
      DO 60 I=1,NEWMAX
      EL(N,I)=ELF(I)
   60 CONTINUE
  100 CONTINUE
      RETURN
      END
C
C
C
C
      ------------------------
      FUNCTION PLOLP(Y,N)
      ------------------------
C
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/   ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +               NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/   HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/    TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
C
C
      YY=Y*IPT
      IY=YY
      MAXIM=MAXI-1
      IF(IY .GT. MAXIM) GO TO 100
      DY=IY+1-YY
      PLOLP=(EL(N,IY)-EL(N,IY+1))*DY + EL(N,IY+1)
      RETURN
  100 PLOLP=0.0
      RETURN
      END
C
C
C
C     -------------------
      SUBROUTINE LOADER
C     -------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +                NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/  HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/     TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
C
      DIMENSION M(3),B(500)
C
C
C
      DATA K24/24/
C
      II=1
      DO 30 I=1,IPMAX
      B(II)=PLANT(I,7)
      IBK(II,1)=I
      IBK(II,2)=1
      IBK(II,3)=-1
      IF(I.GT.NPEXP) IBK(II,3)=NUF(I,JYR)
      II=II+1
```

```
      C=PLANT(I,3)-PLANT(I,2)
      IF(C.EQ.0.0) GO TO 30
      B(II)=PLANT(I,8)
      IBK(II,1)=I
      IBK(II,2)=2
      IBK(II,3)=-1
      IF(I.GT.NPEXP) IBK(II,3)=NUF(I,JYR)
      II=II+1
   30 CONTINUE
      KMAX=II-1
C
C
C     ARRANGE 'IBK' ARRAY ACCORDING TO FUEL COST
C
C
      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
C
C
C
C     ---------------------------
      SUBROUTINE OPCOST(NCUM)
C     ---------------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +                NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/  HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/    TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
```

```
C
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)
C
C
      DIMENSION ENRT(220,2),PCUM(201,8,1),ENRH(3,2),CFH(3),
     *ENY(2,30),ENP(2,30),KLW1(3),KLW2(3),NORDER(4)
      DOUBLE PRECISION SUCUM(8)
C
C
C
      T=8760./NP
      IYR=IYRB+JYR-1+1900
      ISTAR=(JYR-1)*4+1
C
C
C     INITIALIZATION
C
C
      DO 5 J=1,16
      DO 5 I=1,211
    5 PTCUM(I,J)=0.0
C
      DO 50 J=1,2
      DO 50 I=1,220
   50 ENRT(I,J)=0.0
      DO 805 I=1,4
  805 NORDER(I)=0
C
      COSTOP(JYR)=0.0
      DTOT=0.0
      GENTOT=0.0
      HYTOT=0.0
C
C
C
C     DETERMINE EFFECTIVE LOADING ORDER
C
C
C
      KK=1
      IE=IPMAX+1
      IF(PLANT(IE,2).LE.0.0) GO TO 19
      ITK(KK,1)=IE
      ITK(KK,2)=1
      ITK(KK,3)=1
      KK=KK+1
   19 DO 10 I=1,NPEXP
      J=NPEXP-I+1
      IF((NT(J)-NX(J)).EQ.0) GO TO 10
      IP=IX(J)
      ITK(KK,1)=IP
      ITK(KK,2)=1
      ITK(KK,3)=NT(J)-NX(J)
      KK=KK+1
   10 CONTINUE
C
C
      DO 15 K=1,KMAX
      IF(IBK(K,3).LE.0) GO TO 15
      DO 16 J=1,3
   16 ITK(KK,J)=IBK(K,J)
      KK=KK+1
   15 CONTINUE
      NMAX=KK-1
C
```

```
      DO 2000 N=1,4
      IF(PEAK(JYR,N).EQ.PMAX(JYR)) GO TO 2010
 2000 CONTINUE
 2010 NCRT=N
C
C
C     *** S E C T I O N   1 ***
C
C
C     START ENERGY & COST CALCULATION FOR EACH SEASON
C
C     LOOP 60
C
C
      DO 60 NSA=1,NP
      N=NSA
C
C
C     INITIALIZATION
C
C
      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
      DO 55 I=1,3
   55 ENRH(I,J)=0.0
C
C
C
C     CUMULANTS OF SCHEDULED THERMAL PLANTS
C
C
C
      IFLAG=1
      CALL CUCAL(IFLAG,N)
C
C
C     CALCULATE CUMULANTS OF HYDRO PLANTS
C
      IFLAG=4
      CALL CUCAL(IFLAG,N)
      DO 3000 K=1,4
 3000 SUCUM(K)=HRCUM(N,K)+PTCUMH(3,K)
C
      IFLAG=3
      CALL CUCAL(IFLAG,N)
C
C
C     STORE & WRITE PTCUM,HRCUM
C
C
      DO 64 I=1,201
      DO 64 J=1,8
   64 PCUM(I,J,1)=0.0
C
C
      IF(N.NE.NCRT) GO TO 2020
      DO 56 K=1,4
      KK=K+8
   56 SUCUM(K)=SUCUM(K)+PTCUMH(1,KK)
      IB=NPEXP+1
      IE=IPMAX
      DO 51 I=IB,IE
      C=PLANT(I,3)-PLANT(I,2)
```

```
      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))
C
 2020 DO 65 K=1,8
      PCUM(1,K,1)=SNGL(HRCUM(N,K))
   65 CONTINUE
C
C
C
      IE=IPMAX-NPEXP+1
      DO 61 I=2,IE
      IP=I+NPEXP-1
      DO 62 K=1,8
      IF(K.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
C
      WRITE(27) PCUM
C
C
C     CALCULATE PLANT CUMULANTS FOR SINGLE 'PSEUDO'-PLANT
C
C
      IFLAG=2
      CALL CUCAL(IFLAG,N)
C
C
C     INITIALIZE SYSTEM CUMULANTS
C
C
      DO 30 I=1,8
   30 SYSCUM(I)=HRCUM(N,I)
C
C
      SUMP=0.0
      EPMP=0.0
C
C
C     *** S E C T I O N   2 ***
C
C     BASE LOAD RUN-OF-RIVER CAPACITY
C
C
      IH=4
      DO 20 K=1,2
      P=1.0-HYDR(K,5)
      Y=HYDR(K,1)*CMULT(N)*(1.0-EMORH(K,N))
      Y1=SUMP
      SUMP=SUMP+Y
      Y2=SUMP
      IF(K.EQ.1) GO TO 25
C
C
C     CONVOLVE THE FIRST HYDRO
C
```

```
C
      DO 24 J=1,NCUM
   24 SYSCUM(J)=SYSCUM(J)+PTCUMH(1,J)
C
C
C     EVALUATE AREA UNDER EL-CURVE
C
C
   25 CALL AREA(Y1,Y2,P,EH,CF,CFP,IH,N)
      ENRH(K,1)=EH
C
C
C     MAX. HYDRO ENERGY AVAILABLE
C
C
      EAVAL(K)=HYDR(K,4)*EMULT(N)*1000.0-EH
      C=HYDR(K,2)-HYDR(K,1)*CMULT(N)
      IF(C.LE.0.0) GO TO 20
      CFH(K)=EAVAL(K)/(C*P*T)
   20 CONTINUE
      EPS=HYDR(3,2)*1000.0
C
C
C     CONVOLVE THE SECOND HYDRO
C
C
      DO 27 J=1,NCUM
   27 SYSCUM(J)=SYSCUM(J)+PTCUMH(2,J)
C
C
C     ORDER OF HYDRO PLANT IN DECREASING CAPACITY FACTOR
C
C
      NUMBER=0
      IF(CFH(1).GE.CFH(2)) GO TO 800
      NORDER(1)=2
      IF(CFH(1).LE.0.0) GO TO 802
      NORDER(2)=1
      NUMBER=2
      GO TO 801
  800 IF(CFH(1).LE.0.0) GO TO 801
      NORDER(1)=1
      IF(CFH(2).LE.0.0) GO TO 802
      NORDER(2)=2
      NUMBER=2
      GO TO 801
  802 NUMBER=1
  801 IF(EPS.LE.0.0) GO TO 803
      NUMBER=NUMBER+1
      NORDER(NUMBER)=3
  803 NUMBER=NUMBER+1
      NORDER(NUMBER)=4
C
C
C     INITIALIZE INDEX FOR HYDRO LOADING ORDER
C
C
      DO 28 K=1,3
      KLW1(K)=0
   28 KLW2(K)=0
      NOHY=1
      IH=NORDER(NOHY)
      IPUMP=-1
      IF(EPS.LE.0.0) IPUMP=1
      KLOW1=0
      KLOW2=0
      SUEN2=0.0
C
```

```
C
C     *** S E C T I O N  3 ***
C
C     START LOADING THE THERMAL PLANTS
C
C     LOOP 70
C
      DO 70 KC=1,NMAX
      K=KC
      ID1=ITK(K,2)
      IP=ITK(K,1)
C
      DO 630 J=1,30
      ENP(2,J)=0.0
  630 ENY(2,J)=0.0
C
C
C     INDEX FOR EMOR ARRAY
C
C
      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))
C
C
C     LOOP FOR IDENTICAL UNITS
C
C     LOOP 75
C
C
C
   74 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)
C
C
C     PLANT CONVOLUTION
C
C
   79 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
C
C
C     PLANT DECONVOLUTION
C
```

```
C
      DO 100 J=1,NCUM
  100 SYSCUM(J)=SYSCUM(J)-PTCUM(IP,J)
C
C
  150 CONTINUE
C
C
C     EVALUATE AREA UNDER THE EL-CURVE
C
C
  152 CALL AREA(Y1,Y2,PT,ET,CF,CFP,IH,N)
      ENY(2,I)=ET
C
C
C     ANY MORE HYDRO LEFT
C
C
      IF(IH.GE.4) GO TO 75
C
C
C     CHECK HYDRO GENERATION FEASIBILITY
C
C
      IF(CF.LE.CFH(IH)) GO TO 200
C
C
C     CHECK PUMPING DUTY ASSIGNMENT
C
C
      IF(IPUMP.GT.0.OR.CFP.GE.1.0) GO TO 75
C
C
C     PUMPING ASSIGNMENT
C
C
      PP=1.0-HYDR(3,5)
      YP1=Y1-HYDR(3,1)*(1.0-EMORH(3,N))
      YP2=Y2-HYDR(3,1)*(1.0-EMORH(3,N))
      CALL AREA(YP1,YP2,PT,EP,CF,CFP,IH,N)
      ENP(2,I)=PP*(EP-ET)
      EPMP=EPMP+ENP(2,I)*HYDR(3,3)
      IF(EPMP.LT.EPS) GO TO 570
C
C
C     STOP PUMPING
C
C
      IPUMP=1
      ENP(2,I)=ENP(2,I)-(EPMP-EPS)/HYDR(3,3)
      EPMP=EPS
      EAVAL(3)=EPS*HYDR(3,4)
      CFH(3)=EAVAL(3)/(T*HYDR(3,1)*(1.0-HYDR(3,5)))
      GO TO 75
C
C
C
  570 EAVAL(3)=EPMP*HYDR(3,4)
      CFH(3)=EAVAL(3)/(T*HYDR(3,1)*(1.0-HYDR(3,5)))
      GO TO 75
C
C
C     HYDRO GENERATION
C
C
  200 CONTINUE
      ICNTL=-1
C
C
```

```
C
C      REPLACE THE THERMAL BY HYDRO
C
      SUMP=SUMP-YT
      ENY(2,I)=0.0
      ENP(2,I)=0.0
      IF(IDL.LE.1) GO TO 215
C
C
C      RETRIVE EL-CURVE INFO
C
C
      DO 210 J=1,NCUM
  210 SYSCUM(J)=SYSCUM(J)+PTCUM(IP,J)
  215 CONTINUE
C
C
C      DECONVE BASE LOADED HYDRO
C
C
  216 DO 220 J=1,NCUM
  220 SYSCUM(J)=SYSCUM(J)-PTCUMH(IH,J)
C
C
      PH=1.0-HYDR(IH,5)
      YH=(HYDR(IH,2)-HYDR(IH,1)*CMULT(N))*(1.0-EMORH(IH,N))
      IF(IH.EQ.3) YH=HYDR(IH,1)*(1.0-EMORH(IH,N))
      Y1=SUMP
      SUMP=SUMP+YH
      Y2=SUMP
      CALL AREA(Y1,Y2,PH,EH,CF,CFP,IH,N)
      ENRH(IH,2)=EH
      IF(ICNTL.GT.0) GO TO 240
      KLOW1=K
      KLOW2=I-1
      KLW1(IH)=K
      KLW2(IH)=I-1
C
C
C      CONVOLVE THIS HYDRO
C
C
      DO 230 J=1,NCUM
      J1=J+8
  230 SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J1)
C
C
C      NEXT HYDRO
C
C
      IF(IH.EQ.3) GO TO 620
C
      NOHY=NOHY+1
      IH=NORDER(NOHY)
      ICNTL=1
      GO TO 216
  240 IF(EH.GT.EAVAL(IH)) GO TO 250
      ENRH(IH,2)=EH
      KLOW1=K
      KLOW2=I-1
      KLW1(IH)=K
      KLW2(IH)=I-1
      GO TO 300
  250 CONTINUE
C
C
C      REPLACE THIS HYDRO BY THERMAL
C
C
```

```
C
      SUMP=SUMP-YH
      DO 270 J=1,NCUM
  270 SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J)
      NOHY=NOHY-1
      IH=NORDER(NOHY)
C
  620 IBLK=-1
      CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP,
     +NORDER)
      IPUMP=1
      GO TO 455
C
C
  300 CONTINUE
C
C
C
C     FORM CLUSTER AND TEST SWAP LOADING POSITION
C
      NPAST=NOHY-1
      IHPT=NORDER(NPAST)
      IBLK=1
      IF(KLOW2.LE.0) GO TO 301
      KIX=KLOW1
      ISWP=2
      ITRM=KLOW2
      GO TO 302
  301 KIX=KLOW1-1
      ISWP=1
      ITRM=ITK(KIX,3)
  302 II1=ITK(KIX,1)
      II2=ITK(KIX,2)
      DO 310 J=1,NCUM
      J1=J
      IF(II2.GT.1) J1=J+8
  310 SYSCUM(J)=SYSCUM(J)-PTCUM(II1,J1)
      DO 320 J=1,NCUM
      J1=J+8
  320 SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J1)
      P=1.0-PLANT(II1,5)
      IF(II1.GT.NPEXP) GO TO 321
      IMANT=2*II1-II2+1
      GO TO 322
  321 IMANT=II1+NPEXP
  322 R=1.0-EMOR(IMANT,N)
      Y=PLANT(II1,2)*R
      IF(II2.GT.1) Y=(PLANT(II1,3)-PLANT(II1,2))*R
      Y2=SUMP
      Y1=SUMP-Y
      CALL AREA(Y1,Y2,P,ET,CF,CFP,IH,N)
      SUEN1=ENY(ISWP,ITRM)+ENRH(IH,2)+ENRH(IHPT,2)
      SUEN2=SUEN1-ET
      ETOT=EAVAL(IH)+EAVAL(IHPT)
      IF(SUEN2.GT.ETOT) GO TO 400
C
C
C     ACCEPT SWAP TEST
C
      ENY(ISWP,ITRM)=ET
C
C
      KLOW1=K
      KLOW2=I-2
      KLW1(IH)=K
      KLW2(IH)=I-2
      KLW1(IHPT)=K
```

```
      KLW2(IHPT)=I-2
C
C
      COST=COST-CST1*ENP(ISWP,ITRM)
      EPMP=EPMP-ENP(ISWP,ITRM)
      ENP(ISWP,ITRM)=0.0
      CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP,
     +NORDER)
      IPUMP=1
      GO TO 450
C
C
  400 CONTINUE
C
C
C     REJECT SWAP TEST
C
C
      SUEN2=ENRH(IH,2)+ENRH(IHPT,2)
      CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP,
     +NORDER)
      IPUMP=1
  450 CONTINUE
C
C
C     LOAD THE CURRENT OFF-LOADED THERMAL PLANT
C
C
      DO 460 J=1,NCUM
      J1=J
      IF(II2.GT.1) J1=J+8
  460 SYSCUM(J)=SYSCUM(J)+PTCUM(III1,J1)
C
C
  455 NOHY=NOHY+1
      IH=NORDER(NOHY)
C
      Y1=SUMP
      SUMP=SUMP+YT
      Y2=SUMP
      IF(ID1.NE.2) GO TO 470
      DO 468 J=1,NCUM
  468 SYSCUM(J)=SYSCUM(J)-PTCUM(IP,J)
  470 CALL AREA(Y1,Y2,PT,ET,CF,CFP,IH,N)
      ENY(2,I)=ET
C
C
C     END OF IDENTICAL UNITS
C
C
   75 CONTINUE
      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=1,L3
      GENTOT=GENTOT+ENY(1,L)
   77 ENRT(L1,L2)=ENRT(L1,L2)+ENY(1,L)+ENP(1,L)
   80 DO 78 I=IUNITB,IUNITE
      ENP(1,I)=ENP(2,I)
   78 ENY(1,I)=ENY(2,I)
C
C
      IF(K.LT.NMAX) GO TO 70
      DO 81 I=IUNITB,IUNITE
      GENTOT=GENTOT+ENY(2,I)
   81 ENRT(IP,ID1)=ENRT(IP,ID1)+ENY(2,I)+ENP(2,I)
```

```
C
C
C      END OF SECTION 3
C
C
   70 CONTINUE
C
C
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
C
C
   66 DO 67 I=1,2
      DO 67 J=1,2
   67 HYTOT=HYTOT+ENRH(I,J)
      GENTOT=GENTOT+ENRH(3,2)
C     WRITE(17,905)
  905 FORMAT(//' ++++++ SEASONAL HYDRO OPERATIONAL SUMMARY ++++++'//)
C     WRITE(17,906)
  906 FORMAT(/3X,'PLANT',2X,'LOAD',2X,'BASE   MAX.',5X,'BASE ENERGY',5X,
     +'PEAK ENERGY')
C     WRITE(17,910)
  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
C
C      END OF SECTION 2
C
C
   60 CONTINUE
C
C
C      *** S E C T I O N   4 ***
C
```

```
C      COST CALCULATION
C
C
       IE=IPMAX+1
       DO 700  I=1,IE
       IF(I.GT.NPEXP) GO TO 711
       J=NPEXP-I+1
       MM=NT(J)-NX(J)
       IF(MM.LE.0) GO TO 700
   711 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))/1.2)**(FLOAT(JYR)+0.5)
       COSTOP(JYR)=COSTOP(JYR)+(ENRT(I,J)*(CFL+VAB)+C*FIX)*FAC/1.0E+06
   710 CONTINUE
   700 CONTINUE
C
C
C
C
C      REPORT RESULT
C
C
C
       WRITE(17,902)
   902 FORMAT(//' ++++++ ANNUAL THERMAL OPERATIONAL SUMMARY ++++++'//)
C      WRITE(17,903)
   903 FORMAT(/3X,'PLANT NO',3X,'BASE CAP',3X,'MAX. CAP',3X,
      +'BASE ENERGY(MWH)',3X,'PEAK ENERGY(MWH)'/)
  1230 FORMAT(//' ++++++ ANNUAL SYSTEM OPERATIONAL SUMMARY +++'//)
       GENTOT=GENTOT+HYTOT
C      WRITE(17,1240) DTOT,GENTOT
  1240 FORMAT(/2X,'ENERGY DEMAND = ',E14.7,' MWH'//
      +2X,'TOTAL GENERATION = ',E14.7,' MWH'//)
C      WRITE(17,1990) COSTOP(JYR)
  1990 FORMAT(/1H ,' LOWER BOUND OF OP. COST = ',E14.7,
      *' MILLION DOLLARS'/)
C
C
       DO 810  J=1,14
       DO 810  I=1,NPEXP
   810 PLANT(I,J)=PLACA(I,J)
       RETURN
       END
C
C
C      -----------------------------------------------------------------
       SUBROUTINE TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,
      +IPUMP,NORDER)
C      -----------------------------------------------------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +                NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/    HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/     TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
C
      DIMENSION CFH(3),ENY(2,30),ENP(2,30),ENRH(3,2),NORDER(4)
C
C
C
      IF(KLOW2.LE.0) GO TO 10
      ISWP=2
      ITRM=KLOW2
      L=KLOW1
      GO TO 15
   10 L=KLOW1-1
      ISWP=1
      ITRM=ITK(L,3)+KLOW2
   15 IP=ITK(L,1)
      ID=ITK(L,2)
      IF(IBLK.GT.0) GO TO 30
C
C
C     TRIM HYDRO BLOCK
C
C
      IH=NORDER(NOHY)
      DIFF=EAVAL(IH)-ENRH(IH,2)
      ENRH(IH,2)=EAVAL(IH)
      IF(NOHY.LE.1) GO TO 40
      ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF
      GO TO 60
C
C
C     TRIM HYDRO CLUSTER
C
C
   30 IH=NORDER(NOHY)
      NPAST=NOHY-1
      IHPT=NORDER(NPAST)
      DIFF=EAVAL(IH)+EAVAL(IHPT)-SUEN2
      ENRH(IH,2)=EAVAL(IH)
      ENRH(IHPT,2)=EAVAL(IHPT)
      IF(NPAST.LE.1) GO TO 40
      ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF
      GO TO 60
C
C
C     PUMPING ENERGY ADJUSTMENT
C
C
   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
C
C
C
C     ----------------------------------------
      SUBROUTINE AREA(Y1,Y2,P,E,CF,CFP,IH,N)
C     ----------------------------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +                NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/    HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/     TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
      DOUBLE PRECISION COEFF(8),SIGMA,SIGS,ZL1,ZL2,ZV1,ZV2
C
C
C
      NCUM=4
      COEFF(1)=SYSCUM(1)
      COEFF(2)=SYSCUM(2)
      SIGMA=DSQRT(SYSCUM(2))
      SIGS=SIGMA*SIGMA
      DO 10 J=3,NCUM
      SIGS=SIGS*SIGMA
   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
```

```
      E=X1*2190.0*P*C1
      IF(IH.GE.4) GO TO 60
      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+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
C
C
C
C     ------------------------------------------
      SUBROUTINE CUCAL(IFLAG,N)
C     ------------------------------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +                NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/  HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/    TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
C
```

```
      DOUBLE PRECISION R(8),C,Q
C
      GO TO (100,200,600,700),IFLAG
C
C
C     CUMULANTS OF THERMAL PLANTS
C
C
  100 LB=1
      LE=IPMAX
      GO TO 500
  200 LB=1
      LE=1
C
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
      IF(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))
      IF(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*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
      IF(J.GT.1) KK=K+8
  550 PTCUM(I,KK)=R(K)
  520 CONTINUE
  510 CONTINUE
C
C
      GO TO 1000
C
C
C     CUMULANTS OF HYDRO PLANTS
C
C
  600 CONTINUE
C
C
      DO 610 I=1,3
      Q=HYDR(I,5)
      DO 620 J=1,2
      IF(J.GT.1) GO TO 630
      C=HYDR(I,1)*CMULT(N)*(1.0-EMORH(I,N))
```

```
      IF(I.GE.3) C=0.0D00
      GO TO 640
  630 C=HYDR(I,2)*(1.0-EMORH(I,N))
      IF(I.GE.3) C=HYDR(3,1)*(1.0-EMORH(3,N))
  640 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*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 650 K=1,8
      KK=K
      IF(J.GT.1) KK=K+8
  650 PTCUMH(I,KK)=R(K)
  620 CONTINUE
  610 CONTINUE
C
C
      GO TO 1000
C
C
  700 CONTINUE
      Q=PSEXP(5)
      C=PSEXP(1)*(1.0-EMORH(3,N))
      PTCUMH(3,1)=C*Q
      PTCUMH(3,2)=C*C*Q*(1.0-Q)
      PTCUMH(3,3)=C*C*C*Q*(1.0-(3.0-2.0*Q)*Q)
      PTCUMH(3,4)=C*C*C*C*Q*(1.0-(7.0-(12.0-6.0*Q)*Q)*Q)
C
C
 1000 RETURN
      END
C
C
C
C
      ________________________________________________
      SUBROUTINE VALUE(Z,C,NCUM,FRE)
      ________________________________________________
C
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/   ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +               NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/   HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
```

```
      COMMON/SIX/    TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
C
      DOUBLE PRECISION Z,ZN(8),FRE,FAE,F3,F4,F5,F6,F7,F8,F9,E,
     *ZINTEG,ZSQ,C(8)
      IF ( Z.GT.5.90D00 ) GO TO 40
      ZSQ = Z*Z/2.0D00
      E = DEXP(-ZSQ)/2.506628275D00
      ZN(1) = -Z * E
      ZN(2) = E * (Z*Z - 1.0D00)
      ZN(3) = -Z*E*(Z*Z - 3.0D00)
      ZN(4) = E*(Z*Z*(Z*Z - 6.0D00) + 3.0D00)
      ZN(5) = -Z*E*(Z*Z*(Z*Z - 10.0D00) + 15.0D00)
      ZN(6) = E*(Z*Z*(Z*Z*(Z*Z - 15.0D00) + 45.0D00) - 15.0D00)
      ZN(7)=-Z*E*(Z*Z*(Z*Z*(Z*Z-21.0D00)+105.0D00)-105.0D00)
      ZN(8)=E*(Z*Z*(Z*Z*(Z*Z*(Z*Z-28.0D00)+210.0D00)-420.0D00)+
     +105.0D00)
      F3 = 6.0D00
      F4 = 24.0D00
      F5 = 120.0D00
      F6 = 720.0D00
      F7 = 5040.0D00
      F8 = 40320.0D00
      F9 = 362880.0D00
      NUMCU=NCUM-5
C
      CALL TALOOK(Z,ZINTEG)
C
      FRE = 1.0D00 - ZINTEG
      FRE = FRE + C(3)*ZN(2)/F3
     +          - C(4)*ZN(3)/F4
     +          -C(3)*C(3)*ZN(5)*10.0D00/F6
C
C
      IF(NUMCU) 5,10,15
C
C     FOUR CUMULANT EXPANSION
C
    5 IF(FRE.LT.0.0D00) FRE=0.0D00
      IF(FRE.GT.1.0D00) FRE=1.0D00
      RETURN
C
C     FIVE CUMULANT EXPANSION
C
   10 FRE = FRE + C(5)*ZN(4)/F5
     +          + C(3)*C(4)*ZN(6)*35.0D00/F7
     +          + C(3)*C(3)*ZN(8)*280.0D00/F9
      IF(FRE.LT.0.0D00) FRE=0.0D00
      IF(FRE.GT.1.0D00) FRE=1.0D00
      RETURN
C
C     EIGHT CUMULANT EXPANSION
C
   15 FRE = FRE + C(5)*ZN(4)/F5
     +          - C(6)*ZN(5)/F6
     +          +(C(7)+10.0D00*C(3)*C(4))*ZN(6)/F7
     +          -(C(8)+56.0D00*C(3)*C(5)+35.0D00*C(4)*C(4))*ZN(7)/F8
      IF(FRE.LT.0.0D00) FRE=0.0D00
      IF(FRE.GT.1.0D00) FRE=1.0D00
      RETURN
C
C
```

```
   40 FRE = 0.0D00
      RETURN
      END
C
C
C
C     ----------------------------
      SUBROUTINE TALOOK(Z,ZINTEG)
C     ----------------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/    ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +                NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/    HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/     YINT(120),Y(120)
      DOUBLE PRECISION YINT,Y
C
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)
      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
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 THEU.S. DEPARTMENT
C     OF COMMERCE.
C
C     DEFINITION OF KEY VARIABLES:
C     NAME      TYPE      SIZE          MEANING
C     Z         REAL      -             NORMALIZED INDEPENDENT VARIABLE
C     ZINTEG    REAL      -             INTEGRAL VALUE
C     Y         REAL      120           NORMALIZE INDEPENDENT VARIABLE
C                                       (INPUT DATA)
C     YINT      REAL      120           INTEGRAL VALUES CORRESPONDING
C                                       TO Y (INPUT DATA)
C     K         INT.      -             GRID POINT CORRESPONDING TO Z
C     DY        REAL      -             Y INCREMENT
C
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)
      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
C
C
C
C     -----------------
      SUBROUTINE RINPUT
C     -----------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/  ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +              NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/  HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/   TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
      DATA K25,K26/25,26/
C
C
C     READ LOAD CUMULANTS
C
C
      DO 10 I=1,4
      READ(K26,1000) (HRCUM(I,J),J=1,8)
 1000 FORMAT(1H /(3E23.14))
   10 CONTINUE
C
C     READ PEAK AND ORIGINAL LDC
C
C     INITIALIZATION
C
      DO 20 I=1,4
      DO 20 J=1,MX
   20 EL(I,J)=0.0
C
      DO 30 I=1,4
      READ(K25) ICY,KS
      READ(K25) DUM,DUM,PEAK(JYR,I),PERENE(JYR,I),DUM
      READ(K25) (EL(I,J),J=1,500)
   30 CONTINUE
C
C
C     NORMALIZATION
C
C
      DO 40 I=1,4
      FNOR=EL(I,1)
      DO 45 J=1,500
   45 EL(I,J)=EL(I,J)/FNOR
   40 CONTINUE
C
C
C     HYDRO SCHEDULED SYSTEM
C
C
      READ(31,1010) NDUM
      READ(31,1020) (HYDR(1,J),J=1,8)
C
C
C     PUMPED STORAGE SYSTEM
C
C
      READ(33,1010) NDUM
```

```
      READ(33,1020) (HYDR(3,J),J=1,8)
 1010 FORMAT(/,3I3)
 1020 FORMAT(5E12.5)
      RETURN
      END
C
C
C
C     -------------------
      SUBROUTINE INOUT
C     -------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/   ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +               NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/   HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/    TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
C
C
      DATA L10/10/
C
C
      WRITE(6,1000)
 1000 FORMAT(/////18X,7HMINIMUM,5X,7HMAXIMUM,5X,
     +12HLOWER  BOUND,5X,6HANNUAL)
      WRITE(6,1001)
 1001 FORMAT(18X,7HRESERVE,5X,7HRESERVE,5X,12HOF OPERATING,
     +5X,6H PEAK )
      WRITE(6,1002)
 1002 FORMAT(9X,4HYEAR,6X,5H(MW)*,7X,4H(MW),7X,7HCOST **,
     +11X,4H(MW))
      WRITE(6,1003)
 1003 FORMAT(9X,4(1H-),2(5X,7(1H-)),5X,12(1H-),5X,6(1H-))
      DO 100 NY=1,NTYR
      IY=IYRB+NY-1+1900
      WRITE(6,1004) IY,RSMIN(NY),RSMAX(NY),COSTOP(NY),PMAX(NY)
  100 CONTINUE
 1004 FORMAT(/9X,I4,2(5X,F7.2),6X,F9.3,6X,F7.1)
      WRITE(6,1005) PPMIN
 1005 FORMAT(////9X,29H*  CRITICAL LOLP ASSIGNED IS ,F9.6)
      IY=IYRB-1+1900
      WRITE(6,1006) IY
```

```
 1006 FORMAT(/9X,15H** MILLIONS OF ,I4,8H DOLLARS//)
      WRITE(L10,902) PPMIN,T
  902 FORMAT(5E16.8)
      IBASYR=IYRB-1+1900
      MAXPLA=IPMAX-NPEXP
      WRITE(L10,900) IBASYR,MX,NPEXP,MAXPLA,IPT,NTYR,KMAX
  900 FORMAT(20I4)
      DO 10 J=1,NTYR
   10 WRITE(L10,900) (MXELDC(J,I),I=1,4)
      DO 20 J=1,NTYR
   20 WRITE(L10,902) EC(J),RSMIN(J),RSMAX(J),COSTOP(J)
      DO 30 J=1,NTYR
   30 WRITE(L10,902) (PEAK(J,I),I=1,4)
      DO 35 J=1,NTYR
   35 WRITE(L10,902) (PERENE(J,I),I=1,4)
      WRITE(L10,911) (HYEXPN(J),J=1,11)
      WRITE(L10,911) (PSEXP(J),J=1,11)
  911 FORMAT(5E12.5)
      DO 40 IP=1,NPEXP
   40 WRITE(L10,904) (PLANT(IP,J),J=1,14)
  904 FORMAT(F3.0,2F5.0,F3.0,F8.6,3F7.2,F3.0,2F7.3,E10.3,2F4.2)
      DO 60 J=1,NTYR
      WRITE(L10,900) NHYMN(J),NPSMN(J),(NUF(IP,J),IP=1,NPEXP)
   60 WRITE(L10,900) NHYMX(J),NPSMX(J),(IUP(IP,J),IP=1,NPEXP)
      DO 70 J=1,NTYR
   70 WRITE(L10,900) NXHY(J),NXPS(J),(NSTPRE(J,I),I=1,NPEXP)
      DO 80 J=1,NTYR
   80 WRITE(L10,902) (AVSPE(J,N),N=1,4)
      DO 90 NY=1,NTYR
   90 WRITE(L10,909) (SCHCUM(J,NY),J=1,4)
  909 FORMAT(4E16.8)
      WRITE(L10,911) (CMULT(J),J=1,4)
      WRITE(L10,911) (EMULT(J),J=1,4)
C
      RETURN
      END
C
C
C     ------------------------------------------
      SUBROUTINE RESMX(ECP,SCAP,NCRL)
C     ------------------------------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/   ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +               NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/  HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/    TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
```

```
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)
C
C
C
      DIMENSION PLP(4),SCAP(4)
C
C
C
      ICNTL=-1
   10 ECP=ECP+PLANT(1,3)
      CLPMX=10.0
      DO 20 N=1,NP
      PK=PEAK(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
C     ----------------------------------
C
      SUBROUTINE COMPST
C     ----------------------------------
C
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)
C
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)
C
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)
C
C
      COMMON/FOUR/  ID,IPMAX,IPT,IYRB,JYR,KMAX,MAXI,MX,NP,NPEXP,
     +              NTYR,PPMAX,PPMIN,T,PMAX(20),EPMP
C
C
      COMMON/FIVE/  HRCUM(4,8),PTCUM(211,16),PTCUMH(3,16),SYSCUM(8)
      DOUBLE PRECISION HRCUM,PTCUM,PTCUMH,SYSCUM
C
C
      COMMON/SIX/   TINT(120),TY(120)
      DOUBLE PRECISION TINT,TY
C
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)
C
C
C     HYDRO EXPN PLANT
C
C
      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
   35 HYDR(2,J)=HYEXPN(J)
C
C
C     PUMPED STORAGE PLANT
C
C
      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)
C
C
      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)
C
C
      RETURN
      END
```

```
C                  ***     DYNO2.FORT      ***
C
C    DYNO IS THE OPTIMIZATION SUBMODULE OF THE  OPTIM
C    MODULE IN THE  CERES  CODE.
C  TITLE:    MAIN
C  FUNCTION:THE MAIN PROGRAM ROUTINE CONTAINS THE DYNAMIC
C           PROGRAMMING ALGORITHM.  THIS ROUTINE IS DIVIDED INTO
C           THE FOLLOWING SECTIONS:
C           1. READ INPUT DATA, INITIALIZE PROGRAM FLAG,
C              EXERCISE PROGRAM EXECUTION CONTROL (OUTER
C              ITERATIONS) THROUGH SUBROUTINE CHOICE, AND
C              DEFINE OUTER ITERATION VARIABLES.
C           2. GENERATE AND KEEP THE ACCEPTABLE STATES FOR THE
C              FIRST STAGE (YEAR) OF THE STUDY PERIOD.
C           3. FOR EACH STATE OF STAGE (YEAR) NY OF THE STUDY
C              PERIOD, GENERATE THE ACCEPTABLE STATES FOR STAGE
C              NYNEXT = NY + 1.  FOR ANY STATE IN STAGE NYNEXT,
C              THAT ORIGINATES FROM MORE THAN ONE STATES IN
C              STAGE NY, KEEP THE MINIMUM OBJECTIVE FUNCTION.
C           4. IDENTIFY THE MINIMUM OBJECTIVE FUNCTION IN THE LAST
C              YEAR AND TRACE BACK THE OPTIMUM SOLUTION HISTORY
C              (PATH) DURING THE STUDY PERIOD.
C           5. MODIFY THE ARTIFICIAL TUNNEL BOUNDARIES UNTIL THE
C              OPTIMUM SOLUTION DOES NOT VIOLATE THE TUNNEL
C              BOUNDARIES.  BRANCH BACK TO SECTION 2 OR 3 (INNER
C              OR TUNNEL ITERATIONS).
C
C                     DEFINITION OF COMMON VARIABLES
C
C  NAME    TYPE  SIZE             DEFINITION
C-----------------------------------------------------------------------
C AVSP     REAL  20,4    SPACE AVAILABLE FOR SCHEDULED MAINTENANCE
C CAPABS   REAL  20        SCHEDULED SYSTEM CAPACITY FOR EACH YEAR
C                          OF THE STUDY PERIOD.
C CLOLP    REAL   -        CRITICAL LOSS-OF-LOAD-PROBABILITY(LOLP)
C CPLOLP   REAL   -        CRITICAL LOSS-OF-LOAD-PROBABILITY(LOLP)
C                          AS DEFINED IN PREP SUBMODULE
C DISRAT   REAL   -        DISCOUNT RATE
C DX       REAL   -        NORMALIZED MW INCREMENT (DX=1./MAXPO)
C EL       REAL 1250       AUXILIARY ELDC FOR LOLP CALCULATION
C ELDC     REAL 4,1250     FINAL LOAD DURATION CURVE AFTER ALL
C                          THE UNITS OF THE SCHEDULED SYSTEM HAVE
C                          BEEN CONVOLVED (USED FOR LOLP CALCULATIONS).
C                          ELDC IS READ FROM THE DIRECT ACCESS FILE 22
C ELF      REAL  1260      AUXILIARY ELDC FOR LOLP CALCULATION
C ENEDEM   REAL  20,4      ENERGY DEMAND FOR EACH SEASON
C EXPLCU   REAL   4,8      (JCUM,I) EXPANSION PLANT CUMULANTS
C                           JCUM=1,4 IS THE JCUM-TH CUMULANT
C                           I=1,8     IS THE I-TH PLANT TYPE IN THE
C                                    RESTRICTED EX.PLANT LIST.
C FATOPE   REAL  20        LOWER BOUND FOR ENERGY COST FOR EACH
C                          YEAR (IN MILLION DOLLARS. USED FOR FATHOMING)
C FCR      REAL   -        FIXED CHRAGE RATE
C HOURS    REAL   -        NUMBER OF HOURS PER SIMULATION PERIOD
C IBASYR   INT.   -        BASE YEAR. CALENDAR YEAR NEXT TO
C                          WHICH THE STUDY PERIOD BEGINS.
C IDEXP    INT.    8       IDEXP(I)=IP MEANS THAT THE I-TH PLACA
C                          PLANT IS THE IP-TH INPUT MODULE PLANT.
C IPCH     INT,    8       NUMBER OF UNITS BY WHICH THE ARTIFICIAL
C ISOL     INT.    20,8    IISOL(N,J): OPTIMUM OF SUBOPTIMUM SOLUTION
C                          N: YEAR OF STUDY PERIOD
C                          I: NUMBER OF UNITS OF I-TH PLANT TYPE
C                          TUNNEL CHANGE  FOR EACH EXP. CANDIDATE
C ITIN     INT.   -        NUMBER OF TUNNEL ITERATIONS
C ITMAX    INT.   -        MAXIMUM ITIN
C ITOUT    INT.   -        NUMBER OF DYNO SENSITIVITY ANALYSES
C LBOLD    INT.    8       (I) ARTIFICIAL LOWER TUNNEL BOUNDARY BEFORE
C                          THE CURRENT CALL TO THE ADJUST ROUTINE.
```

```
C                            I=1,8 I-TH PLANT IN RESTRICTED EXP.PLANT LIST
C LIST      INT. 8,1000      LIST OF STATES GENERATED WITHIN TUNNELS
C                            FROM ORIGIN STATE
C LOWB      INT.  20,8       ARTIFICIAL YEARLY PLANTS TUNNEL LOWER BOUND
C MAINS     INT.  200        MAINS(I) DEFINES THE SEASON IN WHICH SCHEDULED
C                            PLANT I IS SUT DOWN FOR MAINTENANCE.
C MAXADD    INT.  8          MAXIMUM NUMBER OF UNITS THAT CAN BE ADDED
C                            EACH YEAR.
C MAXALL    INT.  -          MAXPLA + MXPL
C MAXINP    INT.  -          MAXIMUM NUMBER OF EXPANSION CANDIDATES
C                            DEFINED BY INPUT MODULE
C MAXI      INT.  -          MAXIMUM NUMBER OF POINTS IN THE ELDC ARRAYS
C MAXPLA    INT.  -          MAXIMUM NUMBER OF PLANTS IN THE
C                            SCHEDULED SYSTEM.
C MAXPO     INT.  -          MAXIMUM NUMBER OF POINTS IN THE NORMALIZED
C                            ORIGINAL ELDC.
C MAXOR     INT.  -          NUMBER OF PLANT BLOCKS IN THE LOADING ORDER
C MAXTUN    INT.   8         (I) ARTIFICIAL UPPER TUNNEL BOUNDARY FOR
C                             CURRENT ORIGIN STATE.
C                            I=1,8 I-TH PLANT IN RESTRICTED EXP.PLANT LIST
C MAXUN     INT.   20,8      MAXIMUM ALLOWED NUMBER OF UNITS FOR EACH
C                            YEAR AND EXP. CANDIDATE
C MINTUN    INT.   8         (I) ARTIFICIAL LOWER TUNNEL BOUNDARY FOR
C                             CURRENT ORIGIN STATE.
C                            I=1,8 I-TH PLANT IN RESTRICTED EXP.PLANT LIST
C MINUN     INT.   20,8      MINIMUM ALLOWED NUMBER OF UNITS FOR EACH
C                            YEAR AND EXP. CANDIDATE
C MXELDC    INT.  20,4       MAXIMUM NUMBER OF POINTS IN EACH OF THE
C                            SCHEDULED SYSTEM ELDC'S FOR THE STUDY PERIOD
C MXPL      INT.  -          MAXIMUM NUMBER OF NEW CANDIDATES.
C MXYEAR    INT.  -          MAXIMUM NUMBER OF YEARS IN STUDY PERIOD
C NEXPID    INT.   8         NEXPID(IP)=I MEANS THAT THE IP-TH INPUT
C                            MODULE PLANT IS THE I-TH PLACA PLANT.
C                            IF I<0 THE PLANT IS NOT USED IN THE
C                            CURRENT SENSITIVITY ANALYSIS
C NORDER    INT.  420,3      LORDER(I,J): PLANT LOADING ORDER.
C                            I=1,420: PLANT BLOCK LOADING ORDER
C                            J=1: INDICATES THE POSITION OF THE PLANT
C                                 IN THE SCHEDULED OR NEW CANDIDATE
C                                 FILES (PLANTS & PLACA RESPECTIVELY)
C                            J=2: PLANT BLOCK
C                            J=3: IF >=0, IT INDICATES THE NUMBER OF
C                                         UNITS OF PLANT (I,J=1) IN
C                                         THE SCHEDULED SYSTEM.
C                                 IF < 0  (USUALLY -1), INDICATES THAT
C                                         THE CORRESPONDING PLANT
C                                         (I,J=1) IS A NEW CANDIDATE.
C                                         THE NUMBER OF UNITS FOR THIS
C                                         PLANT IS FOUND FROM THE STATE
C                                         UNDER EXAMINATION.
C NUBOLD    INT.   8         (I) ARTIFICIAL UPPER TUNNEL BOUNDARY BEFORE
C                            THE CURRENT CALL TO THE ADJUST ROUTINE.
C                            I=1,8 I-TH PLANT IN RESTRICTED EXP.PLANT LIST
C NSTPRE    INT.  20,8       YEARLY STATES USED FOR THE MINIMUM RESERVE
C                            MARGIN CALCULATION IN PREP SUBMODULE. THEY
C                            USED HERE FOR THE FIRST ESTIMATE OF UBOUND.
C NUPB      INT.  20,8       ARTIFICIAL YEARLY PLANTS TUNNEL UPPER BOUND
C PEAKS     REAL  20         YEARLY PEAKS FOR THE 20 YEARS OF THE STUDY PERIOD
C PLACA     REAL  8,14       PLACA(I,J): PLANT CANDIDATES.
C                            I:   PLANT NUMBER IN THE SAME ORDER AS IN
C                                 THE NEW CANDIDATE PLANTS FILE.
C                            J=1  NUMBER OF UNITS BELONGING TO THIS
C                                 PLANT CODE
C                            J=2  BASE CAPACITY IN MW
C                            J=3  MAX CAPACITY IN MW
C                            J=4  MAINTENANCE REQUIREMENT IN DAYS PER YEAR
C                            J=5  FORCED OUTAGE RATE
C                            J=6  CAPITAL COST IN $/KW
```

```
C                          J=7  BASE FUEL COST IN $/MWH
C                          J=8  MAX OPERATING FUEL COST IN $/MWH
C                          J=9  ECONOMIC PLANT LIFE IN YEARS
C                          J=10 FIXED OPERATION AND MAINTENANCE
C                               COSTS IN $/MW.YEAR
C                          J=11 VARIABLE OPERATION AND MAINTENANCE
C                               COSTS IN $/MWH
C                          J=12 SALVAGE VALUE IN THOUSAND DOLLARS
C                          J=13 ANNUAL CAPITAL COST ESCALATION RATE
C                          J=14 FUEL COST ESCALATION RATE
C PLANTS  REAL   200,14  PLANTS(I,J): SCHEDULED SYSTEM PLANTS
C                          I  : PLANT NUMBER IN THE SAME ORDER AS IN
C                               THE SCHEDULED SYSTEM FILE.
C                          J=1  NUMBER OF UNITS BELONGING TO THIS
C                               PLANT CODE
C                          J=2  BASE CAPACITY IN MW
C                          J=3  MAX CAPACITY IN MW
C                          J=4  MAINTENANCE REQUIREMENT IN DAYS PER YEAR
C                          J=5  FORCED OUTAGE RATE
C                          J=6  CAPITAL COST IN $/KW
C                          J=7  BASE FUEL COST IN $/MWH
C                          J=8  MAX OPERATING FUEL COST IN $/MWH
C                          J=9  ECONOMIC PLANT LIFE IN YEARS
C                          J=10 FIXED OPERATION AND MAINTENANCE
C                               COSTS IN $/MW.YEAR
C                          J=11 VARIABLE OPERATION AND MAINTENANCE
C                               COSTS IN $/MWH
C                          J=12 SALVAGE VALUE IN THOUSAND DOLLARS
C                          J=13 ANNUAL CAPITAL COST ESCALATION RATE
C                          J=14 FUEL COST ESCALATION RATE
C POPEC   REAL   225,20  POPEC(IP,NY) OPERATING COST
C                          IP: PLANT ID
C                          NY: YEAR OF STUDY PERIOD
C PTCUM   REAL 200,8,4  PTCUM(IP,I,J) SCHEDULED SYSTEM PLANT CUMULANTS
C                          IP:   PLANT ID AS DEFINED IN INPUT MODULE
C                          I=1,4 BASE BLOCK CUMULANTS
C                          I=5,8 PLANT CUMULANTS
C                          J=1,4 YEARLY SEASON
C PTCUMX  REAL   8,8,4   EXP. CANDIDATE CUMULANTS DEFINED AS IN PTCUM
C                          ARRAY MOR(I,J)
C PESMAR  REAL     -     MAXIMUM RESERVE MARGIN (DEFINED IN % AND
C                          USED AS DECIMAL)
C RESMAX  REAL    20     MAXIMUM RESERVE MARGIN EACH YEAR
C RESMIN  REAL    20     MINIMUM RESERVE MARGIN EACH YEAR
C ROM     REAL    20,4   EXP. CANDIDATE MAINTENANCE OUTAGE RATE
C                          I:    EXP. CANDIDATE NUMBER
C                          J=1,4 SEASON NUMBER
C SOL     REAL    20,2   SOL(N,J): OPTIMUM OR SUBOPTIMUM SOLUTION
C                          CHARACTERISTICS.
C                          N:  YEAR OF STUDY PERIOD
C                          J=1 LOLP
C                          J=2 OBJECTIVE FUNCTION
C SSCUM   REAL    4,20   (JCUM,NY) SCHEDULED SYSTEM CUMULANTS
C                                   FOR THE PEAK SEASON IN EACH YEAR
C                          JCUM=1,4  IS THE JCUM-TH CUMULANT
C                          NY=1,20   IS THE YEAR IN THE STUDY DERIOD.
C UBOUND  REAL     -     UPPER BOUND OF OBJECTIVE FUNCTION.  IN
C                          MILLION DOLLARS. (USED FOR FATHOMING)
C UNE     REAL    20     UNSERVED ENERGY FOR EACH YEAR
C X       REAL   1250    NORMALIZED MW ARRAY
C
C
C
C                 FILE DESCRIPTION
C______________________________________________________________________
C
C NAME     UNIT     TYPE    DEFINITION
C INOUT     8      SEQUEN.   INPUT DATE READ BY READIN SUBROUTINE
```

```
C DPOUT       11      SEQUEN.   DEBUG OUTPUT
C STATES      14      DIR.AC.   YEARLY ACCEPTED STATES
C LDFIN       22      SEQUEN.   SCHEDULED SYSTEM FINAL ELDC'S
C PLSCH       23      DIR.AC    SCHEDULED PLANT DATA
C LORD        24      SEQUEN.   ECONOMIC LOADING ORDER
C PTCUM       27      SEQUEN.   SCHEDULED PLANT CUMULANTS
C SOL         28   DIR.AC    FINAL SOLUTIONS AFTER EACH TUNNEL ITERATION
C
C
C
      DEFINE FILE 14(20,8004,L,IDAF),15(20,16004,L,IDAF),
     +            28(400,60,L,I28)
C
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),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             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
C
      DIMENSION CUMLOL(1000), MINSNU(20), NSTCUR(20), NSTATE(20),
     +NOPTIM(20,2), NUMACC(20), OPTIMA(20,4), RESCAP(20,20),
     +NST2(2,1000),NST1(1000),STN2(4,1000),STN1(1000)
C
C           ***** S E C T I O N  1 *****
C
C=====================================================================
C
      DATA MINSNU/20*5/
C     READ INPUT DATA
      CALL READIN
C     INITIALIZE PROGRAM FLAGS
      IRES = 0
      ITOUT = 0
      ICFLAG = 0
C     INITIALIZE OBJECTIVE FUNCTION UPPER BOUND
      CALL UFIRST
      RESET=UBOUND
```

```
C    DEFINE INPUT PARAMETERS FOR CURRENT OUTER ITERATION
  10 CALL CHOICE(ISTOP)
     IF (ISTOP .GT. 0) GO TO 600
     UBOUND=RESET
C
C
C       CALCULATE PLANT CUMULANTS OF EXPANSION CANDIDATES
C       WITHOUT MAINTENANCE CONSIDERATION
C
C
C
        N=1
        ICUM=-1
        DO 16 I=1,MXPL
  16    NSTCUR(I)=1
        CALL CUCALI(NSTCUR,N,ICUM)
        DO 17 I=1,MXPL
        IP=IDEXP(I)+2
        DO 18 K=1,4
  18    EXPLCU(K,I)=PTCUMX(IP,K)
  17    CONTINUE
C
C               DEFINE THE PACKING BASE
        CALL BASE
C
C    INITIALIZE INNER (TUNNEL) ITERATION COUNTER AND ORIGINAL
C    ATIFICIAL TUNNEL BOUNDARIES
        ITIN = 1
        CALL CHANEL(MAXSTA, ICFLAG)
        IF (ICFLAG .GT. 0) GO TO 10
C
C    START INNER ITERATION CALCULATIONS
  20 CONTINUE
C    INITIALIZE AUXILIARY ARRAYS
        DO 30 NYF = 1,MXYEAR
        UNE(NYF) = 0.0
        NUMACC(NYF) = 0
        DO 30 I=1,8
        RESCAP(NYF,I) = 0.0
  30    ISOL(NYF,I) = 0
C
C
        WRITE(6,926)
 926 FORMAT(1H1///)
C
C             ***** S E C T I O N 2 *****
C
C
C=================================================================
C=================================================================
C
C
C
C             SET BASE YEAR ORIGIN STATE
        DO 40 I=1,MXPL
        NSTATE(I) = 0
  40    CONTINUE
        OBJORI = 0.0
        NORI = 1
C             SET BEGINNING YEAR
        NYCR = 1
        NYOR = 1
        KYEAR = IBASYR + NYCR
        WRITE(6,926)
        WRITE(6,900) KYEAR
C             READ SEQUENCIAL FILES FOR THIS YEAR
        REWIND 22
        REWIND 24
        REWIND 27
        DO 50 NS = 1,4
```

```
      MAXK = MXELDC(NYCR,NS)
      READ(22) (ELDC(NS,KPOINT),KPOINT=1,1250)
      READ(27) ((PTCUM(IP,JCUM,NS),IP=1,201),JCUM=1,8)
   50 CONTINUE
      READ(24,902) ((LORDER(JBLOCK,KATTR),KATTR=1,3),JBLOCK=1,420)
      READ(24,921) (MAINS(IP),IP=1,200)
C          DEFINE THE FIRST YEAR ORIGINAL TUNNEL. SET TUNNEL FLAG
C          TO INDICATE THAT THIS IS THE ORIGINAL TUNNEL DEFINITION.
      IFLTUN=-1
C          SET ACCEPTED STATE COUNTER
      NACCEP = 0
C           DEFINE TUNNEL
   60 CALL TUNNEL(NSTATE,NYCR,IFLTUN)
C
C          GENERATE THE STATES FOR THE FIRST YEAR. STORE THEM IN THE
C          "LIST" ARRAY IN UNPACKED FORM.
      CALL STAGEN(NSTCNT,CUMLOL,NYCR)
C
C          INITIALIZE AUXILIARY VARIABLES
      NSTOLD = 0
      OLOLP = 1.0
C          CHECK THE STATES IN "LIST" ARRAY FOR PW-LOLP (IF NEEDED)
C          AND FATHOMING.CALCULATE THE OBJECTIVE FUNCTION FOR THE
C          ACCEPTED STATES.
      DO 100 NL=1,NSTCNT
      WRITE(11,903) NYCR,NL
C
C          COPY CURRENT STATE  FROM THE "LIST" ARRAY INTO "NSTCUR"
      DO 70 I=1,MXPL
   70 NSTCUR(I) = LIST(I,NL)
C
C           RETIEVE THE CUMULANT LOLP FROM THE CUMLOL ARRAY.
      CULOLP = CUMLOL(NL)
C
C             IF CRITICAL LOLP IS LESS THAN 0.001, CALCULATE LOLP
C          WITH THE PW-LINEAR METHOD. REJECT UNACCEPTABLE STATES.
      IF(CLOLP.GT.0.001) GO TO 80
      CALL FLOLP(NSTCUR, CULOLP, NSTOLD, OLOLP, NYCR)
      IF (CULOLP .GT. CLOLP) GO TO 100
C
C          CURRENT STATE IS ACCEPTED FOR LOLP. CALCULATE ITS
C          OPERATING COST.
   80 CALL OPERCO(NSTCUR, NYCR, OPEC,IRES)
C
C          CALCULATE THE OBJECTIVE FUNCTION
      CALL OBJFUN(NSTCUR, NYCR, OPEC, NSTATE, CAPCO, OBJORI, OBJ,
     + SALV,RESCAP,IRES)
C
      WRITE(11,*) (NSTCUR(I),I=1,MXPL)
      WRITE(11,*) NYCR,OPEC,CAPCO,OBJORI,OBJ
C
C          EXERCISE FATHOMING
      CALL FATHOM(NSTCUR, NYCR, OBJ, IFATH)
      IF (IFATH .EQ. 0) GO TO 100
C
C          REACHING THIS POINT MEANS THAT THE STATE CONSIDERED
C          IS ACCEPTABLE.
C          INCREMENT THE ACCEPTED STATES COUNTER.
      NACCEP = NACCEP+1
C
C          PACK ACCEPTED STATE.
      CALL PACK(NSTCUR, NST)
C          STORE ACCEPTED STATE ATTRIBUTES IN CORE.
      WRITE(11,*) NST,NORI,CULOLP,OPEC,CAPCO,OBJ
      NST2(1,NACCEP) = NST
      NST1(NACCEP) = NST
      NST2(2,NACCEP) = NORI
      STN2(1,NACCEP) = CULOLP
```

```
      STN2(2,NACCEP) = OPEC
      STN2(3,NACCEP) = CAPCO
      STN2(4,NACCEP) = OBJ
      STN1(NACCEP) = OBJ
C
C           STORE THE CURRENT STATE LOLP FOR OLD LOLP USAGE
      NSTOLD = NST
      OLOLP = CULOLP
C
  100 CONTINUE
C           CHECK IF THE NUMBER OF ACCEPTED STATES IS ADEQUATE.
      IF(NACCEP.GT.MINSNU(NYCR)) GO TO 150
C           THE NUMBER OF STATES IS TOO SMALL. INCREASE THE ARTIFITIAL
C           TUNNEL WIDTH
      CALL ADJUST(NYCR,IFLTUN)
      IF(IFLTUN.GT.0) GO TO 60
C           REACHING THIS POINT MEANS THAT THE ARTIFITIAL TUNNELS
C           DO NOT CONSTRAIN STATE GENERATION.
      WRITE(6,914) KYEAR
C           IF NO STATES WERE ACCEPTED IN THE FIRST YEAR, REDIFINE
C           THE INPUT DATA FOR THIS ITERATION.
      IF(NACCEP.LT.1) GO TO 480
C
C
C           STORE THE NUMBER OF ACCEPTED STATES IN NUMACC ARRAY FOR
C           CURRENT YEAR
  150 NUMACC(NYCR) = NACCEP
C           STORE ACCEPTED STATES ATTRIBUTES ON DISK.
      IDAF = 1
      WRITE(14'IDAF) NST2
      IDAF=1
      WRITE(15'IDAF) STN2
C
      WRITE(6,909) NACCEP,NYCR
      WRITE(11,904) NYCR,NACCEP
C
C           FIRST YEAR CALCULATIONS ARE FINISHED. LOOP THROUGH ALL
C           OTHER YEARS.
C
C            ***** S E C T I O N 3 *****
C
C======================================================================
C======================================================================
C
  200 NYCR = NYOR+1
      KYEAR = IBASYR + NYCR
      WRITE(6,900) KYEAR
C           READ SEQUENTIAL FILES FOR YEAR NYCR.
      DO 210 NS =1,4
      MAXK = MXELDC(NYCR,NS)
      READ(22) (ELDC(NS,KPOINT),KPOINT=1,1250)
      READ(27) ((PTCUM(IP,JCUM,NS),IP=1,201),JCUM=1,8)
  210 CONTINUE
      READ(24,902) ((LORDER(JBLOCK,KATTR),KATTR=1,3),JBLOCK=1,420)
      READ(24,921) (MAINS(IP),IP=1,200)
C           SET TUNNEL FLAG FOR ORIGINAL TUNNEL DEFFINITION
      IFLTUN=-1
C
  220 MAXNST = NUMACC(NYOR)
C           SET STATE COUNTER
      NACCEP = 0
C
      DO 300 N=1,MAXNST
C             IDENTIFY THE CURRENT ORIGIN STATE AND ITS OBJECTIVE
C             FUNCTION.
      NSTORI = NST1(N)
      OBJORI = STN1(N)
C             INITIALIZE ORIGIN LOLP FOR OLD LOLP USAGE.
      OLOLP = 1.0
```

```
      NSTOLD = NSTORI
      CALL UNPACK(NSTORI,NSTATE)
      WRITE(11,*) (NSTATE(I),I=1,MXPL)
C            SET TUNNELS
      CALL TUNNEL(NSTATE, NYCR,IFLTUN)
C
C            GENERATE STATES IN STAGE NYCR, ORIGINATING FROM NSTATE
C            AND WITHIN THE DEFINED TUNNELS.
C
      CALL STAGEN(NSTCNT,CUMLOL,NYCR)
      IF(NSTCNT.LT.1) GO TO 300
C
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 "NORI" (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 DISCARTED 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
      DO 280 NL=1,NSTCNT
C              DEFINE THE CURRENT STATE.
      DO 230 I=1,MXPL
  230 NSTCUR(I) = LIST(I,NL)
C             PACK CURRENT STATE.
      CALL PACK(NSTCUR,NST)
      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.
      IF (N.EQ.1) GO TO 240
C             SEARCH PREVIOUSLY GENERATED STATES IN NYCR STAGE
      CALL SEARCH(NST,NST2,NUMACC,NYCR,NACCEP,NTH)
      IF(NTH.EQ.0) GO TO 240
C             STATE WAS PREVIOUSLY CONSIDERED. RECALCULATE OBJ.
C             CHECK FOR LOLP
      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
      OPEC = STN2(2,NTH)
      CALL OBJFUN(NSTCUR, NYCR, OPEC, NSTATE, CAPCO, OBJORI,
     +OBJ,SALV,RESCAP,IRES)
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.
      NST2(2,NTH) = N
      STN2(3,NTH) = CAPCO
      STN2(4,NTH) = OBJ
      GO TO 280
C
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.
      IF(CLOLP.GT.0.001) GO TO 250
      CALL FLOLP(NSTCUR, CULOLP, NSTOLD, OLOLP, NYCR)
      IF (CULOLP .GT. CLOLP) GO TO 260
C          CALCULATE OBJECTIVE FUNCTION
  250 CALL OPERCO(NSTCUR, NYCR, OPEC,IRES )
      CALL OBJFUN(NSTCUR, NYCR, OPEC, NSTATE, CAPCO, OBJORI,
     +OBJ,SALV,RESCAP,IRES)
      GO TO 270
  260 OPEC = UBOUND*2.0
      CAPCO = OPEC
      OBJ = OPEC
  270 CONTINUE
C
C          REACHING THIS POINT MEANS THAT THE STATE CONSIDERED
C          IS ACCEPTABLE EXEPT FOR FATHOMING. STATES WITH BAD
C          RELIABILITY WILL BE FATHOMED.
C
      NACCEP = NACCEP+1
      NTH = NUMACC(NYCR)+NACCEP
      NST2(1,NTH) = NST
      NST2(2,NTH) = N
      STN2(1,NTH) = CULOLP
      STN2(2,NTH) = OPEC
      STN2(3,NTH) = CAPCO
      STN2(4,NTH) = OBJ
C          STORE LOLP FOR OLD LOLP USAGE.
      NSTOLD = NST
      OLOLP = CULOLP
C
  280 CONTINUE
  300 CONTINUE
C          EXERCISE FATHOMING FOR THE STATES IN YEAR NYCR
      WRITE(11,912) NYCR,NACCEP
      IF (NACCEP.LT.1) GO TO 355
C          SET THE RANGE OF STATES TO BE CHECKED FOR FATHOMING.
      NTH1 = NUMACC(NYCR) + 1
      NTH2 = NUMACC(NYCR) + NACCEP
      NACCEP = 0
      NPOSTN=NUMACC(NYCR)
      DO 350 NTH = NTH1,NTH2
      NST = NST2(1,NTH)
      OBJ = STN2(4,NTH)
      CALL UNPACK(NST, NSTCUR)
      CALL FATHOM(NSTCUR, NYCR, OBJ, IFATH)
      IF (IFATH .EQ. 0) GO TO 350
C
C          STORE THE ACCEPTED STATES IN THE NST2 AND STN2 ARRAYS.
C          NOTE THAT THE INDEX FOR THE ACCEPTED STATES IS LESS OR
C          EQUAL TO THE INDEX OF ALL STATES,(I.E.NACCEP.LE.NTH).
      NACCEP = NACCEP + 1
      NPOSTN=NPOSTN+1
      NST2(1,NPOSTN) = NST
      NST2(2,NPOSTN) = NST2(2,NTH)
      STN2(1,NPOSTN) = STN2(1,NTH)
      STN2(2,NPOSTN) = STN2(2,NTH)
      STN2(3,NPOSTN) = STN2(3,NTH)
      STN2(4,NPOSTN) = OBJ
  350 CONTINUE
C          RECORD THE NUMBER OF STATES ACCEPTED AFTER FATHOMING
  355 NUMACC(NYCR) = NUMACC(NYCR) + NACCEP
      NACCEP = NUMACC(NYCR)
      WRITE(6,909) NACCEP,NYCR
      WRITE(11,909)NACCEP,NYCR
C          CHECK IF THE NUMBER OF STATES ACCEPTED IN YEAR NYCR
C          IS ADEQUATE.
      IF(NACCEP.GT.MINSNU(NYCR)) GO TO 360
C          THE NUMBER OF STATES IS TOO SMALL.ADJUST ARTIFICIAL TUNNEL
```

```
      CALL ADJUST(NYCR,IFLTUN)
      IF(IFLTUN.GT.0) GO TO 220
C           REACHING THIS POINT MEANS THAT THE ARTIFITIAL TUNNELS
C           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
C
C
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
C
C           GO TO NEXT STAGE
      NYOR = NYOR+1
      IF (NYOR .LT. MXYEAR) GO TO 200
C
C
C             ***** S E C T I O N 4 *****
C
C==========================================================================
C==========================================================================
C
C
C           REACHING THIS POINT MEANS ALL STAGES(YEARS) WERE EXAMINED.
C           FIND THE MINIMUM OBJECTIVE FUNCTION AND TRACE BACK THE
C           OPTIMUM SOLUTION
C
      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
C
```

```
C           PRINT ITERATION NUMBER
      WRITE(6,917) ITIN
      WRITE(11,917) ITIN
C
C           WRITE THE OPTIMUM SOLUTION
      I28 = (ITIN-1) * 20 + 1
      WRITE(6,898)
  898 FORMAT(/5X,4HLOLP,5X,8HOP. COST,3X,9HCAP. COST,2X,
     112HOBJ FUNCTION,3X,10HPLANT   MIX/)
      DO 400 NYF=1,MXYEAR
      NST = NOPTIM(NYF,1)
      CALL UNPACK(NST, NSTCUR)
      DO 395 I = 1,MXPL
  395 ISOL(NYF,I) = NSTCUR(I)
      SOL(NYF,1) = OPTIMA(NYF,1)
      SOL(NYF,2) = OPTIMA(NYF,4)
      WRITE(6,899) (OPTIMA(NYF,IATTR),IATTR=1,4),(NSTCUR(I),I=1,MXPL)
  899 FORMAT(5X,F7.5,2X,F8.3,3X,F8.3,3X,F9.3,5X,8I3)
C
      WRITE(28'I28,913)(ISOL(NYF,I),I=1,8),SOL(NYF,1),SOL(NYF,2)
C
  400 CONTINUE
C
C
C
C
C           ADJUST THE OBJECTIVE FUNCTION UPPER BOUND
      UBOUND = OPTIMA(MXYEAR,4)
      WRITE(6,920) UBOUND
      WRITE(11,920)UBOUND
C
C
C
C             ***** S E C T I O N 5 *****
C
C=====================================================================
C=====================================================================
C
C
C
C           CHECK WHETHER THE OPTIMUM SOLUTION IS CONSTRAINED BY THE
C           ARTIFICIAL LOWER OR UPPER TUNNEL BOUNDARIES FOR EACH
C           YEAR IN THE STUDY PERIOD. ADJUST BOUNDARIES AS NEEDED.
C           INITIALIZE THE TUNNEL VIOLATION FLAG.
      NYFLAG = 21
C           THE YEAR LOOP GOES BACKWARDS (I.E. FROM MXYEAR TO 1)
C           SO THAT THE LOWEST YEAR OF TUNNEL VIOLATIONS CAN BE
C           IDENTIFIED.
C
      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 = IDEXP(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
C           REACHING HERE MEANS THE SOLUTION IS CONSTRAINED.
C           LOWER THE LOWER BOUNDARY.
      LOWB(NYB,I) = LOWB(NYB,I) - IPCH(I)
C           ADJUST THE ARTIFICIAL LOWER BOUNDARY TO COMPLY WITH THE
C           REAL CONSTRAINTS.
      IF (LOWB(NYB,I).LT.MINUN(NYB,IDP)) LOWB(NYB,I) = MINUN(NYB,IDP)
C           ACTIVATE THE BOUNDARY VIOLATION CONSTRAINT.
      LOFLAG = I
C           ADJUST THE ARTIFICIAL UPPER BOUNDARY TO COMPLY WITH THE
```

```
C           MAXIMUM WIDTH CONSTRAINT.
      NUPB(NYB,I) = LOWB(NYB,I) + IWIDTH(I) -1
      IF (NUPB(NYB,I).GT.MAXUN(NYB,IDP)) NUPB(NYB,I) = MAXUN(NYB,IDP)
C
      GO TO 420
C
  410 CONTINUE
C           CHECK AND IF NEEDED ADJUST THE ARTIFICIAL UPPER BOUNDARY
      IF ( NUPB(NYB,I).GE.MAXUN(NYB,IDP)) GO TO 420
      IF (NSTCUR(I).GE.MAXUN(NYB,IDP)) GO TO 420
      IF ( NSTCUR(I).NE.NUPB(NYB,I)) GO TO 420
C           REACHING HERE MEANS THE SOLUTION IS CONSTRAINED.
C           INCREASE THE UPPER BOUNDARY.
      NUPB(NYB,I) = NUPB(NYB,I) + IPCH(I)
C           ADJUST THE ARTIFICIAL UPPER BOUNDARY TO COMPLY WITH THE
C           REAL CONSTRAINTS.
      IF(NUPB(NYB,I).GT.MAXUN(NYB,IDP)) NUPB(NYB,I) = MAXUN(NYB,IDP)
C           ACTIVATE THE BOUNDARY VIOLATION CONSTRAINT.
      NUFLAG = I
C           ADJUST THE ARTIFICIAL LOWER BOUNDARY TO COMPLY WITH THE
C           MAXIMUM WIDTH CONSTRAINT.
      LOWB(NYB,I) = NUPB(NYB,I) - IWIDTH(I)+1
      IF (LOWB(NYB,I).LT.MINUN(NYB,IDP)) LOWB(NYB,I) = MINUN(NYB,IDP)
C
  420 CONTINUE
      IF (LOFLAG.GT.0) WRITE(6,915) NYB
      IF (NUFLAG.GT.0) WRITE(6,916) NYB
C
C           RECORD THE FIRST YEAR THAT ARTIFICIAL BOUNDS CONSTRAINED
C           THE SOLUTION
      IF (LOFLAG.LT.1.AND.NUFLAG.LT.1 ) GO TO 430
      NYFLAG = NYB
  430 CONTINUE
C
C           ADJUST THE ARTIFICIAL TUNNEL BOUNDARIES SO THAT THEY ARE
C           AN INCREASING FUNCTION OF THE YEARS.
      DO 450 I = 1,MXPL
      IDP = IDEXP(I)+2
      DO 440 NYF=2,MXYEAR
      NYB = MXYEAR - NYF + 1
      IF(LOWB(NYB+1,I).GE.LOWB(NYB,I))  GO TO 440
      LOWB(NYB,I) = LOWB(NYB+1,I)
C     NUPPER = LOWB(NYB,I) + IWIDTH(I) -1
C     NUPB(NYB,I) = MIN0(NUPPER,MAXUN(NYB,IDP))
      NUPB(NYB,I)=NUPB(NYB+1,I)
  440 CONTINUE
  450 CONTINUE
C           ITERATE THROUGH DP AT MOST ITMAX TIMES
      IF (NYFLAG .GT. MXYEAR) GO TO 500
      ITIN = ITIN + 1
      IF (ITIN.GE.ITMAX) GO TO 470
C
      WRITE(6,923)
      DO 455 NYF=1,MXYEAR
      KYEAR = IBASYR + NYF
      WRITE(6,924) KYEAR,(LOWB(NYF,I),I=1,MXPL)
  455 WRITE(6,925) (NUPB(NYF,I),I=1,MXPL)
      WRITE(6,926)
C
      IF(NYFLAG.EQ.1) GO TO 20
C           SET THE ORIGIN YEAR FOR NEW INNER ITERATION.
      NYOR = NYFLAG - 1
C           REWIND AND RE-READ THE FIRST NYOR RECORDS OF THE
C           SEQUENCIAL FILES.
      REWIND 22
      REWIND 24
      REWIND 27
      DO 465 NYF=1,NYOR
```

```
      DO 460 NS = 1,4
      MAXK = MXELDC(NYF,NS)
      READ(22) (ELDC(NS,KPOINT),KPOINT=1,1250)
      READ(27) ((PTCUM(IP,JCUM,NS),IP=1,201),JCUM=1,8)
  460 CONTINUE
      READ(24,902) ((LORDER(JBLOCK,KATTR),KATTR=1,3),JBLOCK=1,420)
      READ(24,921) (MAINS(IP),IP=1,200)
  465 CONTINUE
      IDAF=NYOR
      READ(14'IDAF) NST2
      IDAF=NYOR
      READ(15'IDAF) STN2
      NACCEP=NUMACC(NYOR)
      DO 475 N=1,NACCEP
      NST1(N)=NST2(1,N)
  475 STN1(N)=STN2(4,N)
      NYCR=NYOR+1
      DO 476 NYF=NYCR,MXYEAR
  476 NUMACC(NYF)=0
C
      GO TO 200
C
  470 WRITE(6,918) ITMAX
      GO TO 500
  480 WRITE(6,922) KYEAR,NACCEP
      ITOUT = 0
      GO TO 10
  500 CONTINUE
C
      IF (ISTOP.LE.0 ) GO TO 10
C
  600 STOP
  900 FORMAT(6H  YEAR ,I4)
  902 FORMAT(125(I3,I2,I3))
  903 FORMAT(5H YEAR ,I4,7H  STATE ,I4)
  904 FORMAT(5H YEAR ,I4,27H  NUMBER OF ACCEPTED STATES ,I4)
  906 FORMAT(4H NL=,I4,9H  NSTCUR=,10I2)
  908 FORMAT(6H NYCR=,I4,17H FINAL NACCEP IS=,  I4)
  909 FORMAT(1H ,I4,37H STATES WERE ACCEPTED AFTER FATHOMING   ,
     + 8H IN YEAR ,I4)
  910 FORMAT(1H ,2I10, F11.8, 3F13.2)
  911 FORMAT(1H ///40H  THE FINAL YEAR OBJECTIVE FUNCTION IS:
     + ,E13.6)
  912 FORMAT( '  THE UNFATHOMED STATES FOR YEAR ',I4,' ARE',I4)
  913 FORMAT(8I4,2E14.7)
  914 FORMAT(42H  ARTIFITIAL BOUNDARIES ELIMINATED IN YEAR ,I5)
  915 FORMAT(40H  *****  SOLUTION AGAINST LOWER BOUNDARY ,
     + 9H IN YEAR ,I4,7H  ***** )
  916 FORMAT(40H  +++++  SOLUTION AGAINST UPPER BOUNDARY ,
     + 9H IN YEAR ,I4,7H  +++++  )
  917 FORMAT(29H  THE NUMBER OF ITERATION IS:,I4)
  918 FORMAT(10H MORE THAN ,I4,
     + 41H  ITERATIONS HAVE BEEN RUN. PROGRAM STOPS )
  919 FORMAT(29H  THE NEW BOUNDARIES FOR YEAR,I4,7H ARE : ,20I3)
  920 FORMAT(35H  NEW UPPER BOUND FOR FATHOMING IS  ,F13.2/)
  921 FORMAT(20012)
  922 FORMAT(38H  THE NUMBER OF STATES DEFINED IN YEAR ,I5,
     +4H IS ,I3//19H  PROBABLE REASONS:  /
     +34H  A. SMALL MAXIMUM RESERVE MARGIN. /
     +55H  B. THE SIZE OF THE EXPANSION PLANT TYPES SELECTED IS  /
     +47H      TOO SMALL TO PROVIDE ADEQUATE RELIABILITY. /
     +41H  **** RERUN WITH PROPER ADJUSTMENTS **** )
  923 FORMAT(1H //36H THE NEW LOWER AND UPPER BOUNDS ARE: /)
  924 FORMAT(7H  YEAR   ,I4,2X,20I4)
  925 FORMAT(7H         ,6X,20I4)
      END
C
C________________________________________________________________
```

```
      SUBROUTINE ADJUST(NYCR, IFLTUN)
C-----------------------------------------------------------------------
C
C        TITLE:      ARTIFICIAL BOUNDARY ADJUSTMENTS
C
C        FUNCTION:   THIS SUBROUTINE ADJUSTS THE ARTIFICIAL
C                    BOUNDARIES SO THAT MORE STATES CAN BE
C                    GENERATED IN THE CURRENT YEAR.  THIS
C                    ROUTINE IS CALLED ONLY WHEN THE NUMBER OF
C                    ACCEPTED STATES IS LESS THEN THE MINIMUM
C                    ALLOWED FOR THE CURRENT YEAR.
C                    THE ARTIFICIAL BOUNDARIES ARE ADJUSTED IN
C                    INCREMENTS OF 1 UNIT FOR ALL EXPANSION
C                    CANDIDATES IN CURRENT YEAR (NYCR).  THE
C                    LOWER BOUND IS ADJUSTED FIRST, UNTIL THE
C                    MINIMUM UNIT NUMBER FOR EACH CANDIDATE IS
C                    REACHED.  IF NO LOWER BOUND ADJUSTMENTS
C                    ARE POSSIBLE, THE UPPER BOUND IS INCREASED
C                    UNTIL THE MAXIMUM UNIT NUMBER FOR EACH
C                    CANDIDATE IS REACHED.  IF NO ADJUSTMENTS
C                    ARE POSSIBLE FLAG IFLTUN IS SET TO 0.
C
C        CALLED BY: MAIN
C        CALLS TO:  NONE
C
C        SUBROUTINE ARGUMENTS
C        NAME        DESCRIPTION
C        IFLTUN      FLAG <0   SPECIFIES ORIGINAL TUNNEL
C                         =0   SPECIFIES NO ADJUSTMENTS POSSIBLE
C                         =1   LOWER BOUNDARY ADJUSTED
C                         =2   UPPER BOUNDARY ADJUSTED
C        NYCR        DESTINATION (CURRENT) STAGE (YEAR)
C
C        COMMON VARIABLES USED: IBASYR, IDEXP, LBOLD, LOWB,
C                               MAXUN, MINUN, MXPL, NUBOLD, NUPB
C
C        LOCAL VARIABLES
C        NAME        DESCRIPTION
C        I           I-TH EXPANSION PLANT TYPE
C        IDP         PLANT TYPE INDEX IN ORIGINAL EXPANSION PLANT
C                    TYPE LIST
C        ITEMP       TEMPORARY VALUE OF PLANT TYPE BOUNDARY THAT
C                    MAY BE ALTERED
C        KYEAR       DESTINATION (CURRENT) CALENDAR YEAR
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),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
C
      COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
C
C           IDENTIFY THE CURRENT CALENDAR YEAR
      KYEAR = IBASYR + NYCR
C           CHECK IF LOWER BOUND HAS REACHED THE MINIMUM
      IFLTUN=0
      DO 15 I=1,MXPL
      LBCR=LOWB(NYCR,I)
      LBOLD(I)=LBCR
      IDP=IDEXP(I)+2
      MN=MINUN(NYCR,IDP)
      IF(NYCR.NE.1) GO TO 10
      LB=MN
      GO TO 11
   10 LB=LOWB(NYCR-1,I)
   11 IF(LBCR.GT.LB) GO TO 12
      GO TO 15
   12 ITEMP=LBCR-1
      IF(ITEMP.GE.MN) GO TO 13
      GO TO 15
   13 IFLTUN=1
      LOWB(NYCR,I)=ITEMP
   15 CONTINUE
C
C
      IF(IFLTUN.LE.0) GO TO 20
      WRITE(6,900) KYEAR, (LOWB(NYCR,I), I=1,MXPL)
  900 FORMAT(33H  NEW TUNNEL LOWER BOUND FOR YEAR,I5,
     +4H  IS,8I3/)
      RETURN
C
C
   20 DO 25 I=1,MXPL
      NBCR=NUPB(NYCR,I)
      IDP=IDEXP(I)+2
      MX=MAXUN(NYCR,IDP)
      NUBOLD(I)=NBCR
      IF(NBCR.GT.MX) NUBOLD(I)=MX
      IF(NYCR.NE.MXYEAR) GO TO 21
      NB=MX
      GO TO 22
   21 NB=NUPB(NYCR+1,I)
   22 IF(NB.GT.NBCR) GO TO 23
      GO TO 25
   23 ITEMP=NBCR+1
      IF(ITEMP.LE.MX) GO TO 24
      GO TO 25
   24 IFLTUN=2
      NUPB(NYCR,I)=ITEMP
   25 CONTINUE
      IF(IFLTUN.LE.0) GO TO 30
C
C
      WRITE(6,901) KYEAR, (NUPB(NYCR,I), I=1,MXPL)
  901 FORMAT(33H  NEW TUNNEL UPPER BOUND FOR YEAR,I5,
     +4H  IS,8I3/)
C
   30 RETURN
```

```
      END
C
C
C
C
C
C     ---------------------------------------------------
      SUBROUTINE AREA(Y1,Y2,P,E,CF,CFP,IH,N)
C     ---------------------------------------------------
C
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),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             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
      DOUBLE PRECISION ZZL1,ZZL2
      DIMENSION COEFF(4)
C
C
C
      NCUM=4
      COEFF(1)=SYSCUM(1)
      COEFF(2)=SYSCUM(2)
      SIGMA=SQRT(SYSCUM(2))
      SIGS=SIGMA*SIGMA
      DO 10 J=3,NCUM
      SIGS=SIGS*SIGMA
   10 COEFF(J)=SYSCUM(J)/SIGS
      ZL1=(Y1-COEFF(1))/SIGMA
      ZL2=(Y2-COEFF(1))/SIGMA
      ZZL1=DBLE(ZL1)
      ZZL2=DBLE(ZL2)
      CF3=COEFF(3)
      CF4=COEFF(4)
      CALL VALUE(ZZL1,CF3,CF4,ZV1)
      CALL VALUE(ZZL2,CF3,CF4,ZV2)
      C1=(ZV1+ZV2)/2.0
      CFP=C1
      X1=Y2-Y1
      E=X1*2190.0*P*C1
      IF(IH.GE.4) GO TO 60
```

```
      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+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
C
C
C
C-----------------------------------------------------------------------
      SUBROUTINE BASE
C-----------------------------------------------------------------------
C
C          TITLE:      PACKING BASE
C
C          FUNCTION:   THIS ROUTINE DETERMINES THE PACKING BASE
C                      VECTOR (DENOTED BY IPABA) USED FOR PACKING
C                      THE STATES IN ONEWORD.
C                      IPABA(I), I=1,MXPL, IS SET TO 10**(I-1),IF THE
C                      MAXIMUM NUMBER OF UNITS FOR ALL EXPANSION
C                      PLANT TYPES (MAXUN) IS LESS THAN 10.
C                      OTHERWISE, IPABA(I) IS IDENTICAL TO THE
C                      PRODUCT OF MAXUN(I) II=1-I.  IF IN THIS CASE
C                      THE HIGHEST IPABA ENTRY IS GREATER THAN THE
C                      MAXIMUM INTEGER SUPPORTED BY THE HOST
C                      COMPUTER (HERE TAKEN AS 2.1 * 1000000000),
C                      THE USER IS PROMPED TO REDUCE THE ENTRIES OF
C                      MAXUN(I).
C           THE PACKING BASE VECTOR AND THE "PACK" SUBROUTINE
C           ARE CONSTRUCTED IN SUCH A MANNER SO THAT , WHEN
C           THE NEW STATES GENERATED BY THE "STAGEN" ROUTINE
C           ARE PACKED THEY WILL BE IN ASCENDING ORDER.
C
C
C          CALLED BY: MAIN
C          CALLS TO:  NONE
C
C          COMMON VARIABLES USED: IBASYR, IDEXP, IPABA, MAXUN,
C                      MXPL, MXYEAR
C
C          LOCAL VARIABLES
C          NAME        DESCRIPTION
C          I           PLANT TYPE INDEX
C          IDP         PLANTY TYPE INDEX IN ORIGINAL LIST
C          ITEMP       TEMPORARY ARRAY WITH MAXIMUM UNIT NUMBERS FOR
C                      EACH PLANT TYPE IN CURRENT YEAR
C          KYEAR       CURRENT CALENDAR YEAR
```

```
C       LIMIT       MAXIMUM INTEGER SIZE ALLOWED FOR CURRENT VALUE
C                   OF BASE VECTOR ENTRY.
C       NYF         YEAR INDEX
C       NY1         FIRST YEAR FOR WHICH MAXIMUM UNIT LIMITS SHOULD
C
      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
     1             FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
     2             ITOUT,MAXALL,MAX1,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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
      DIMENSION ITEMP(8)
C
C           THE MAXIMUM NUMBER OF UNITS ALLOWED FOR EACH EXPANSION
C           CANDIDATE IS STORED IN THE MAXUN ARRAY ENTRIES FOR THE
C           LAST YEAR (MXYEAR) OF THE PLANNING HORIZON.
C
C            CHECK IF ANY OF THE VALUES OF MAXUN EXCEED 9.
      DO 10 I = 1,MXPL
      IDP=IDEXP(I)+2
      IF (MAXUN(MXYEAR,IDP) .GE. 10) GO TO 30
   10 CONTINUE
C           DROPPING THROUGH THE LOOP MEANS THAT BASE 10 PACKING
C           IS ACCEPTABLE
C           SET PACKING VECTOR IPABA WITH BASE 10
      IPABA(1) = 10
      DO 20 I=2,MXPL
   20 IPABA(I) = 10 * IPABA(I-1) + 1
      RETURN
C
C
C           REACHING 30 MEANS PACKING BASE SHOULD BE IDENTICAL TO
C           MAXIMUM UNIT NUMBER.
C           SET DEFAULT BASE TO 1 SO THAT IF MXPL<8 THE IPABA VECTOR
C           IS VALID
   30 DO 40 I=1,8
   40 IPABA(I) = 1
C
C           SET BASE VECTOR EQUAL TO THE PRODUCT OF MAXIMUM UNIT
C           NUMBER FOR SUCCESSIVE PLANT TYPES
      IDP = IDEXP(1)+2
```

```
C     IPABA(1) = MAXUN(MXYEAR,IDP)
      DO 50 I=2,MXPL
      IDP=IDEXP(I-1)+2
C           CHECK BASE VECTOR SIZE
      LIMIT = 2000000000/MAXUN(MXYEAR,IDP)
      IF (IPABA(I-1) .GT. LIMIT) GO TO 60
   50 IPABA(I) = IPABA(I-1) * (MAXUN(MXYEAR,IDP) + 1)
      WRITE(6,900) (IPABA(I), I=1,MXPL)
  900 FORMAT (17H  PACKING BASE IS/8I10/)
      RETURN
C
C
C           REACHING 60 MEANS BASE IS TOO LARGE.  THE USER SHOULD
C           REDUCE THE MAXIMUM UNIT NUMBER FOR SOME EXPANSION
C           CANDIDATES SO THAT THEIR PRODUCT IS <2.0*10**9.
   60 WRITE(6,901)
  901 FORMAT (48H  PACKING BASE CONTAINS NUMBERS LARGER THAN 2E09 /
     +         44H  WHICH IS NOT ALLOWED BY THE HOST COMBUTER.  /
     2         47H  PLEASE REDUSE THE MAXIMUM NUMBER OF UNITS FOR  /
     3         50H  SOME EXPANSION PLANT TYPES SO THAT THEIR PRODUCT /
     4         30H  BECOMES SMALLER THAN 2.0E09.  /
     5         48H  ENTER THE NEW LIMITS FOLLOWING THE LIST OF THE /
     6         50H  OLD MAXIMUM UNIT NUMBER FOR EACH EXPANSION PLANT /
     7         40H  TYPE IN EACH YEAR OF THE STUDY PERIOD. )
C             LIST AND RE-ENTER THE MAXIMUM UNIT NUMBERS FOR EACH
C             EXPANSION PLANT TYPE IN EACH YEAR
C
C             SET DEFAULT BEGINING YEAR OF CHANGES.
      NY1=1
C             BEGIN LISTINGS AND CHANGES.
   70 DO 100 NYF=NY1,MXYEAR
C             DEFINE CURRENT CALENDAR YEAR.
      KYEAR=IBASYR + NYF
C             LOOP FOR ALL PLANT TYPES IN CURRENT YEAR.
      DO 75 I=1,MXPL
      IDP=IDEXP(I)+2
   75 ITEMP(I) = MAXUN(NYF,IDP)
      WRITE(6,902) KYEAR,(ITEMP(I),I=1,MXPL)
  902 FORMAT(6H  YEAR,I5/16H  OLD LIMITS ARE /1H ,8I3,
     +       18H  ENTER NEW LIMITS  )
   80 READ(5,903) (ITEMP(I),I=1,MXPL)
  903 FORMAT(8I3)
C
C             CREATE AND CHECK NEW PACKING BASE VECTOR.
C
C
      IPABA(1) = ITEMP(1)
      DO 85 I=2,MXPL
      LIMIT = 2000000000/ITEMP(I)
      IF(IPABA(I-1).GT.LIMIT) GO TO 95
      IPABA(I) = IPABA(I-1) * (ITEMP(I-1) + 1)
   85 CONTINUE
C
C            DROPPING THOUGH THE LOOP MEANS THAT THE UNIT LIMITS
C            FOR THE CURRENT YEAR ARE ACCEPTABLE.ASSIGN THEM TO
C            THE MAXUN ARRAY.
      DO 90 I=1,MXPL
      IDP=IDEXP(I)+2
   90 MAXUN(NYF,IDP) = ITEMP(I)
C
      GO TO 100
C
   95 WRITE(6,904)
  904 FORMAT(34H  UNIT LIMITS TOO BIG. ENTER AGAIN )
      GO TO 80
  100 CONTINUE
C
C            LEAVE ROOM FOR ERROR. PROVIDE THE MEANS TO RESPECIFY
C            THE UNIT NUMBERS. REQUEST THE FIRST YEAR OF CHANGES.
```

```
C          CHANGES SHOULD ALWAYS PROCCED UP TO AND INCLUDING THE
C          LAST YEAR OF THE STUDY PERIOD, SINCE IT IS THE LAST
C          YEARS LIMITS THAT DEFINE THE PACKING BASE VECTOR.
C
      WRITE(6,900) (IPABA(I),I=1,MXPL)
      WRITE(6,905)
  905 FORMAT(50H  IF YOU ARE SATISFIED WITH THE CHANGES, ENTER 0,   /
     +       50H  OTHERWISE TYPE THE YEAR AT AND AFTER WHICH YOU    /
     +       50H  WANT TO MODIFY THE EXPANSION PLANT TYPE UNIT      /
     +       50H  LIMITS,IE TYPE 1985 FOR CHANGES BEGINING AT 1985 /
     +       50H  AND ENDING AT THE END OF THE STUDY PERIOD.        /
     +       50H  PLEASE NOTE THAT THE UNIT LIMIT CHANGES ARE       /
     +       50H  FINAL,AND IF YOU ENTER 0, YOU WILL NOT BE ABLE    /
     +       50H  TO RE-ADJUST THESE LIMITS UNLESS YOU RERUN THE    /
     +       50H  DYNO MODULE.                                      )
C
      READ(5,906) KYEAR
  906 FORMAT(I4)
      IF (KYEAR.LE.0) GO TO 120
C          SET BEGINING YEAR OF CHANGES
      NY1 = KYEAR -IBASYR
      GO TO 70
C
  120 RETURN
      END
C
C
C
C
C
C
C_____________________________________________________________________
      SUBROUTINE CHANEL (MAXSTA,ICFLAG)
C_____________________________________________________________________
C
C
C
C     THIS SUBROUTINE DEFINES THE ARTIFICIAL CHANNEL WIDTH,
C     THE MAXIMUM NUMBER OF STATES PER YEAR, THE INCREMENT BY
C     WHICH THE ARTIFICIAL CHANNELS WILL BE MOFDIFIED AFTER EACH
C     INNER ITERATION, THE MAXIMUM PLANT ADDITION PER YEAR FOR
C     EXPANSION CANDIDATE AND THE ORIGINAL ARTIFICIAL CHANNELS.
C
C
      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
     1             FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
     2             ITOUT,MAXALL,MAX1,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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
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
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,IDP)) 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 N = NYY,MXYEAR
      DO 126 I = 1,MXPL
      LOWB(N,I) = LOWB(NYY,I)
  126 NUPB(N,I) = LOWB(N,I) + IWIDTH(I) - 1
  128 CONTINUE
C
  130 CONTINUE
C
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)
      WRITE(11,900)
  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
      WRITE(11,901) IWIDTH
      WRITE(11,901)  IPCH
      WRITE(11,901) MAXSTA
  901 FORMAT(1H ,20I4)
C
      WRITE(6,902)
  902 FORMAT(1H //' LOWER AND UPPER CHANNEL BOUNDS PER YEAR')
      DO 140 NY = 1,MXYEAR
      NYY = IBASYR + NY
      WRITE(6,901) NYY,(LOWB(NY,I), I=1,MXPL)
      WRITE(6,903) (NUPB(NY,I), I=1,MXPL)
      WRITE(11,901) NYY, (LOWB(NY,I), I=1,MXPL)
      WRITE(11,903) (NUPB(NY,I), I=1,MXPL)
  140 CONTINUE
  903 FORMAT(1H ,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 RESRVE MARGIN.'/
     +' ***** RERUN WITH MORE OR LARGER PLANTS *****')
      ITOUT = 0
  300 RETURN
      END
C
C
C______________________________________________________________________
      SUBROUTINE CHOICE(ISTOP)
C______________________________________________________________________
C
C
C     USER INTERACTION SUBROUTINE. THE USER CHOOSES UP TO SIX
C     EXPANSION PLANT CANDIDATES THAT WILL BE USED IN OPTIMIZA-
C     TION. THIS PLANT COMBINATION CAN BE CHANGED AFTER EACH
C     OPTIMIZATION.  THE MAXIMUM NUMBER OF ITERATIONS IS ALSO
C     DEFINED BY THE USER.  AT THE END OF EACH OPTIMIZATION THE UP
```

```
C     TO TEN BEST OPTIMAL SOLUTIONS WILL BE PRINTED DEPENDING ON
C     USER REQUEST.
C
C     DEFINITION OF LOCAL VARIABLES
C
C     NAME       TYPE    SIZE      DEFINITION
C     ISTOP      INT.     -        PROGRAM STOP FLAG.
C                                  IF ISTOP = 0 PROGRAM CONTINUES
C                                  IF ISTOP = 1 PROGRAM STOPS
C     NBEST      INT.     -        NUMBER OF BEST SOLUTIONS REQUESTED
C                                  FOR REPORT
C
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),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             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(20,6),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
C
C
      DIMENSION KEEP(10)
      IF (ITOUT .GT. 0) GO TO 10
C     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('     FOR EACH MAIN ITERATION THE USER MUST SPECITY:'/
     +' 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 THEN 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'/)
C
      GO TO 100
   10 CONTINUE
C
C
      WRITE(6,902) ITOUT
  902 FORMAT('  THE NUMBER OF MAIN ITERATIONS IS:',I3/)
C
      WRITE(6,903) ITIN
  903 FORMAT(1H ,I4,' TUNNEL ITERATIONS WERE REQUIRED',
     +' TO FIND THE OPTIMAL SOLUTION'/)
C
      NNIT = 10
      NIT = MIN0(NNIT,ITIN)
      WRITE(6,904) NIT
  904 FORMAT(' INPUT THE NUMBER OF BEST SOLUTIONS YOU WANT',
     +' REPORTED'/' THIS SHOULD BE LESS OR EQUAL TO',I4)
      READ(5,*)NBEST
C
C     CHECK NBEST AND SET TO 1 IF OUT OF RANGE
      IF (NBEST .LT. 1 .OR. NBEST .GT. NIT) NBEST = 1
      WRITE(40,920) MXPL
      WRITE(40,930) (IDEXP(I),I=1,MXPL)
  920 FORMAT(I2)
  930 FORMAT(8I2)
      WRITE(40,920) NBEST
C     CALL RESIMULATION AND REPORT ROUTINES NBEST TIMES.
      IRES = 1
      DO 20 N = 1,NBEST
      NN=N
      NB=NBEST
      CALL RESIM(NN,NB,IRES,KEEP)
   20 CONTINUE
      IRES = 0
C
      WRITE(6,905)
  905 FORMAT(' IF YOU WANT TO STOP THE MAIN ITERATIONS',
     +' (SENSITIVITY ANALYSIS), TYPE 1 '/
     +' IF YOU DESIRE TO CONTINUE, TYPE 0 '/)
      READ(5,*) ISTOP
      IF (ISTOP .LT. 1) GO TO 100
      RETURN
C
  100 ITOUT = ITOUT+1
C     ASK FOR EXPANSION CANDIDATE COMBINATION
      WRITE(6,906)
  906 FORMAT(1H /'   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')
      READ(5,*) MXPL
C     CALCULATE NUMBER OF ALL PLANTS
      MAXALL = MAXPLA + MXPL
      WRITE(6,907) MXPL
  907 FORMAT(1H /'   ENTER THE',I2,' 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:')
      READ(5,*) (IDEXP(I),I=1,MXPL)
```

```
C
C     ASK FOR ITMAX
      WRITE(6,908)
  908 FORMAT(1H /'  ENTER THE MAXIMUM NUMBER OF TUNNEL',
     +' 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('   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('    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(' 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
C
      DO 130 NY = 1,MXYEAR
      PYR=AUPEAK(NY)
C
  130 RESMAX(NY)= PYR*RESMAR
C
C
  140 CONTINUE
C
C
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=HYEXPN(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
C
C
C     RESTRUCTURE AND PRINT THE EXPANSION CANDIDATE FILE.
```

```
C
      DO 104 IP=1,8
  104 NEXPID(IP) = -2
      DO 107 IP=1,MXPL
      IDP=IDEXP(IP)
      IF(IDP.GT.0) GO TO 105
      IF(IDP.GT.-1) GO TO 106
      PLACA(IP,2)=HYEXPN(1)
      PLACA(IP,3)=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 REWIND 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
C
      RETURN
      END
C
C
C
C
C
C
C     ------------------------------------------------------------
      SUBROUTINE COMPST(NSTCUR,NYNEXT)
C     ------------------------------------------------------------
C
C
C
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),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             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
      DIMENSION NSTCUR(20)
C
C
C
C     INITIALIZATION
C
      DO 10 J=1,11
      DO 10 I=1,3
   10 HYDR(I,J)=0.0
C
C     HYDRO SCHEDULED PLANT
C
      DO 15 J=1,8
   15 HYDR(1,J)=HYSCHD(NYNEXT,J)
C
C
C     PUMPED STORAGE SCHEDULED PLANT
C
C
      DO 16 J=1,8
   16 HYDR(3,J)=PSSCHD(NYNEXT,J)
C
C
C
C     EXPANSION PLANTS
C
C
      DO 20 I=1,MXPL
      IP=IDEXP(I)
      IF(IP.GT.0) GO TO 20
      IH=IP+2
      HMULT=FLOAT(NSTCUR(I))
      IF(IH.GT.1) GO TO 40
C
C
C     HYDRO PLANT
C
C
      DO 30 J=1,4
   30 HYDR(2,J)=HMULT*HYEXPN(J)
      DO 35 J=5,11
   35 HYDR(2,J)=HYEXPN(J)
      GO TO 20
C
C
C     PUMPED STORAGE
C
C
   40 W1=PSSCHD(NYNEXT,1)
      W2=PSSCHD(NYNEXT,2)
      W3=HMULT*PSEXPN(1)
      W4=HMULT*PSEXPN(2)
      HYDR(3,1)=W1+W3
      HYDR(3,2)=W2+W4
      DO 45 J=3,7
   45 HYDR(3,J)=(W1*PSSCHD(NYNEXT,J)+W3*PSEXPN(J))/HYDR(3,1)
      HYDR(3,8)=(W2*PSSCHD(NYNEXT,8)+W4*PSEXPN(8))/HYDR(3,2)
      DO 50 J=9,11
   50 HYDR(3,J)=PSEXPN(J)
```

```
   20 CONTINUE
C
C
      RETURN
      END
C_____________________________________________________________________
      SUBROUTINE CONSTR (NSTATE, NY, CULOLP,IFLCON)
C_____________________________________________________________________
C
C           TITLE  CONSTRAINTS CHECK
C           FUNCTION
C           THIS SUBROUTINE CHECKS THE ACCEPTABILITY OF STATE
C           "NSTATE" WITH RESPECT TO MINIMUM AND MAXIMUM RESERVE
C           MARGINS AND LOLP.
C           THE LOLP IS CALCULATED WITH THE
C           CUMULANT METHOD.  THE LOLP CHECK IS FINAL ONLY WHEN
C           EITHER THE STATE LOLP (DENOTED BY CULOLP) IS GREATER
C           THAN 0.001 OR THE CRITICAL LOLP (DENOTED BY CLOLP) IS
C           GREATER THAN 0.001.  IF THESE CONDITIONS ARE NOT MET
C           THE STATE IS CONDITIONALLY ACCEPTED AND LOLP IS
C           RECALCULATED IN MAIN THROUGH THE PIECEWISE LINEAR
C           APPROXIMATION.  THE 0.001 LIMIT IS SET BECAUSE THE
C           CUMULANT METHOD IS NOT ACCURATE FOR LOLP VALUES LESS
C           THAN 0.001.  CULOLP MUST ALSO BE SMALLER THAN THE
C           CRITICAL LOLP UPPER BOUND (DENOTED BY CLOLPU).  IT
C           IS ASSUMED THAT CLOLPU IS ALWAYS GREATER THAN 0.001.
C           THE CUMULANT LOLP CALCULATION OMMITS MAINTENANCE
C           CONSIDERATIONS FOR THE EXPANSION CANDIDATES.  THEREFORE,
C           WHEN CLOLP>0.001, AND A STATE IS ACCEPTED LOLP WILL BE
C           CALCULATED AND CHECKED AGAIN WHEN OPERATING COSTS ARE
C           ESTIMATED AT THE OPERCO SUBROUTINE.
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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C           SUBROUTINE ARGUMENTS
C           NAME       DESCRIPTION
C           CULOLP     CURRENT STATE LOLP
```

```
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
C           LOCAL VARIABLES
C           NAME        DESCRIPTION
C           I           I-TH PLANT TYPE
C           SYSCAP      SYSTEM CAPACITY IN MW
C
      DIMENSION NSTATE(20)
C           TOTAL SYSTEM CAPACITY
      SYSCAP = CAPABS(NY)
      DO 10 I=1,MXPL
   10 SYSCAP = SYSCAP + PLACA(I,3) * NSTATE(I)
      SYSDIF=SYSCAP-AUPEAK(NY)
C
C           CHECK RESERVE MARGINS
      IF (SYSDIF .LT. RESMIN(NY)) GO TO 30
      IF (SYSDIF .GT. RESMAX(NY)) GO TO 30
C
C           CALCULATE LOLP
      CALL FLOLPC(CULOLP,NSTATE,NY,SYSCAP)
C
C           CHECK CULOLP AGAINST CLOLP AND 0.001 LIMIT
      IF (CLOLP .GE. 0.001 .AND. CULOLP .GT. CLOLP) GO TO 30
C     IF (CLOLP .GE. 0.001 .AND. CULOLP .LE. CLOLP .AND.
C    *CULOLP .LT. CLOLPU) GO TO 20
C           NOTE THAT IT IS ASSUMED THAT CLOLPU>0.001
      IF (CLOLP .LT. 0.001 .AND. CULOLP .GE. 0.001) GO TO 30
   20 IFLCON = 0
      RETURN
   30 IFLCON = 1
      RETURN
      END
C
C
C
C
C_____________________________________________________________________
      SUBROUTINE CUCALI(NSTCUR,N,ICUM)
C_____________________________________________________________________
C
C
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),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             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
      DIMENSION R(4), NSTCUR(20)
C
C     INITIALIZATION
C
      DO 10 I=1,20
      DO 10 J=1,8
   10 PTCUMX(I,J)=0.0
      DO 15 J=1,8
      DO 15 I=1,3
   15 PTCUMH(I,J)=0.0
      IF(ICUM.GT.0) GO TO 50
      DO 20 I=1,MXPL
      IP=IDEXP(I)+2
      Q=PLACA(I,5)
      C=PLACA(I,3)
      PTCUMX(IP,1)=C*Q
      PTCUMX(IP,2)=C*C*Q*(1.0-Q)
      PTCUMX(IP,3)=C*C*C*Q*(1.0-(3.0-2.0*Q)*Q)
      PTCUMX(IP,4)=C*C*C*C*Q*(1.0-(7.0-(12.0-6.0*Q)*Q)*Q)
   20 CONTINUE
      GO TO 200
   50 CONTINUE
C
C     HYDRO CUMULANTS
C
      DO 55 I=1,3
      Q=HYDR(I,5)
      DO 60 J=1,2
      IF(J.GT.1) GO TO 65
      C=HYDR(I,1)*CMULT(N)*(1.0-ROM(2,N))
      IF(I.GE.3) C=0.0
      GO TO 70
   65 C=HYDR(I,2)*(1.0-ROM(1,N))
      IF(I.GE.3) C=HYDR(I,1)*(1.0-ROM(3,N))
   70 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)
      DO 75 K=1,4
      KK=K
      IF(J.GT.1) KK=K+4
   75 PTCUMH(I,KK)=R(K)
   60 CONTINUE
   55 CONTINUE
C
C     THERMAL EXPN CUMULANTS
C
      DO 80 I=1,MXPL
      IP=IDEXP(I)
      IF(IP.LE.0) GO TO 80
      Q=PLACA(I,5)
```

```
      DO 85 J=1,2
      IF(J.GT.1) GO TO 90
      IN=2*(IP+2)
      C=PLACA(I,2)*(1.0-ROM(IN,N))
      GO TO 95
   90 IN=2*IP+3
      C=PLACA(I,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-(3.0-2.0*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
C
C
  200 RETURN
      END
C
C
C
C
C______________________________________________________________________
      SUBROUTINE FATHOM(NSTCUR,NY,OBJ,IFATH)
C______________________________________________________________________
C
C
C     CALCULATES THE MINIMUM CONSTRUCTION COST FOR REACHING THE
C     FINAL YEARS MINIMUM RESERVE MARGIN. THE MINIMUM YEARLY
C     OPERATING COST FOR ALL FUTURE STUDY YEAR PERIODS IS ADDED
C     TO THE ABOVE CONSTRUCTION COST. BOTH THESE ARE ADDED TO THE
C     STATE'S OBJECTIVE FUNCTION AND COMPARED WITH THE UPPER BOUND
C     FOR THE OBJECTIVE FUNCTION. THE STATE "NSTCUR" IS ACCEPTABLE
C     IF THE UPPER BOUND IS NOT EXCEEDED BY THE ABOVE COSTS.  THEN
C     FLAG IFATH = 1.  OTHERWISE IFATH = 0.
C
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),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             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
      DIMENSION NSTCUR(20)
C
      IF (OBJ .LE. UBOUND) GO TO 5
      IFATH = 0
      RETURN
    5 CONTINUE
      IF (NY .EQ. MXYEAR) GO TO 100
C     CALCULATE ADDITIONAL CAPACITY REQUIREMENTS
      C = 0.0
      DO 10 I=1,MXPL
   10 C = C+PLACA(I,3)*NSTCUR(I)
C
      ADDCAP = RESMIN(MXYEAR)-C
      IF (ADDCAP .LE. 0.0) GO TO 100
C     THE MINIMUM UNDISCOUNTED CONSTRUCTION COST IS:
      COST = ADDCAP*PLACA(IMXPL,6)*(1.0+PLACA(IMXPL,13))/1000.0
C     THE ADDITIONAL CAPACITY IS ASSUMED ADDED AT THE MIDLE
C     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
C
C
C
C     THE DISCOUNTED SUM OF THE YEARLY OPERATING COSTS IN THE
C     REMAINING STUDY PERIOD IS:
   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
C
C     THE TOTAL COST IS:
      COST = COST+COP+OBJ
C
C     CHECK THE TOTAL COST AGAINST THE OBJECTIVE FUNCTION UPPER
C     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
C
C
C
C-----------------------------------------------------------------
      SUBROUTINE FLOLP(NSTCUR, COLOLP, NSTOLD, OLOLP, NYNEXT)
C-----------------------------------------------------------------
C
C
```

```
C
C     FLOLP FINDS THE LOLP FOR THE STATE UNDER EXAMINATION
C     (NSTCUR). IF THE LAST STATE EXAMINED (NSTOLD) WAS RELIABLE
C     AND THE CURRENT STATE HAS THE SAME OR MORE UNITS FOR EACH
C     PLANT CANDIDATE, THEN THIS STATE IS ALSO RELIABLE AND THE
C     CURRENT STATE LOLP(CULOLP) IS EQUATED TO THE OLD STATE LOLP
C     (OLOLP). IF THE ABOVE CONDITIONS ARE NOT SATISFIED THE LOLP
C     IS CALCULATED STARTING FROM THE ELDC OF THE SCHEDULED
C     SYSTEM.
C
C     NOTE THAT LOLP SERVES ONLY AS A CONSTRAINTED.  OPTIMIZATION
C     FOR RELIABILITY IS BASED ON UNSERVED ENERGY AND NOT LOLP.
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),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             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
      DIMENSION NSTCUR(20), ISTOLD(20)
C
C      *** DEFIINITION OF LOCAL VARIABLES ***
C     NAME      TYPE   SIZE        DEFINITION
C
C     ISTOLD    INT.    10       UNPACKED NSTOLD STATE
C
C
C     CHECK LOLP OF THE LAST EXAMINED STATE
      IF (OLOLP .GT. CLOLP) GO TO 20
C
C     COMPARE THE NUMBER OF UNITS IN CURRENT AND OLD STATE
C     UNPACK NSTOLD
      CALL UNPACK(NSTOLD, ISTOLD)
      DO 10 I=1,MXPL
      IF (NSTCUR(I) .LT. ISTOLD(I)) GO TO 20
   10 CONTINUE
C     DROPPING THROUGH THE LOOP MEANS THAT THE PREREQUISITS FOR
C     EQUATING THE OLD STATE LOLP TO THE CURRENT ONE ARE SATISFIED
      CULOLP = OLOLP
```

```
      RETURN
   20 CONTINUE
C
C
C
      CALL MAINT2(NSTCUR,NYNEXT)
      CULOLP=0.0
      DO 40 N=1,4
      PK=PEAKS(NYNEXT,N)
      MAX=MXELDC(NYNEXT,N)
      DO 50 I=1,MAX
   50 EL(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-ROM(IN,N))/PK
      P=1.0-PLACA(IP,5)
      SUMY=SUMY+PLACA(IP,3)/PK
   70 CALL PWADD(Y,P,MAXEL)
   60 CONTINUE
      Y=CAPABS(NYNEXT)/PK +SUMY
      PLP=PLOLP(Y)
      IF (PLP.GT.CULOLP) CULOLP = PLP
   40 CONTINUE
      RETURN
      END
C
C
C______________________________________________________________________
      SUBROUTINE FLOLPC(CULOLP, NSTCUR, NYCR, SYSCAP)
C----------------------------------------------------------------------
C
C             CUMULANT LOLP
C
C            FUNCTION:  CUMULANT BASED CALCULATION OF LOLP.  THIS
C                       CALCULATION DOES NOT CONSIDER MAINTENANCE
C                       OF THE EXPANSION PLANT TYPES, AND IS DONE
C                       ONLY FOR THE SEASON WITH THE SMALLEST RESERVE
C                       MARGIN.
C
C            CALLED BY: CONSTR
C            CALLS TO:  VALUE
C
C            SUBROUTINE ARGUMENTS
C            NAME       DESCRIPTION
C            CULOLP     LOLP FOR CURRENT STATE
C            NSTCUR     CURRENT STATE (UNPACKED)
C            NYCR       CURRENT YEAR
C            SYSCAP     SYSTEM CAPACITY
C
C            COMMON VARIABLES USED: EXPLCU, MXPL, SSCUM, SSCUM
C
C
C            LOCAL  VARIABLES
C            NAME        DESCRIPTION
C            CF3         THIRD EXPANSION COEFFICIENT
C            CF4         FORTH EXPANSION COEFFICIENT
C            JCUM        JCUM-TH CUMULANT
C            SIGMA       STANDARD DEVIATION
C            SYSCUM      CURRENT SYSTEM CUMULANTS
C            Z           NORMALIZED SYSTEM CAPACITY
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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
      DIMENSION NSTCUR(20), SYCUM(4)
      DOUBLE PRECISION Z
C
C
C           CALCULATE THE CURRENT SYSTEM CUMULANTS BY ADDING THE
C           CURRENT STATE PLANT CUMULANTS TO THOSE OF THE SCHEDULED
C           SYSTEM
C
      DO 20 JCUM = 1,4
      SYCUM(JCUM) = SSCUM(JCUM,NYCR)
      DO 10 I = 1,MXPL
   10 SYCUM(JCUM) = SYCUM(JCUM) + NSTCUR(I) * EXPLCU(JCUM,I)
   20 CONTINUE
C
C
C           CALCULATE THE CRAM-CHARLIER SERIES COEFFICIENTS THE
C           "MEAN" IS IDENTICAL WITH THE FIRST CUMULANT I.E.
C           MEAN = SYCUM(1)
C
C           STANDARD DEVIATION
      SIGMA = SQRT(SYCUM(2))
C
C           THIRD AND FORTH COEFFICIENTS
      CF3 = SYCUM(3)/(SIGMA**3)
      CF4 = SYCUM(4)/(SIGMA**4)
C
C           NORMALIZED SYSTEM CAPACITY
      Z = DBLE((SYSCAP - SYCUM(1))/SIGMA)
C
C           FIND THE GRAM-CHARLIER SERIES VALUE THAT CORRESPONDS
C           TO Z
      CALL VALUE (Z, CF3, CF4, CULOLP)
      RETURN
      END
C
C--------------------------------------------------------------------
      SUBROUTINE FORM(IPAGE,K,N,LN,I,FMT1,FMT2,FMT3,
     +                FMT4,FMT5,FMT6,FMT7,C,IYR,NP,MX1)
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 /LCS/  EL(1250),ELDC(4,1250),ELF(1260),
     +              MXELDC(20,4),X(1250)
C
      COMMON /HYS/  CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
     +              HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +              PSSCHD(20,8),SYSCUM(4)
      DIMENSION A1(250),A2(20),FMT1(18),FMT2(18),FMT3(18),
     +FMT4(18),FMT5(18),FMT6(18),FMT7(18),C(60),IYR(20)
      LMX=51
      M=N-K+1
      N1=(M*11-45)/2+11
      N2=N1-11
      N3=N1+4
      FMT1(3)=C(N1)
      FMT2(3)=C(N1)
      FMT1(14)=C(N2)
      FMT2(14)=C(N2)
      FMT3(3)=C(N3)
      FMT4(5)=C(M)
      FMT5(5)=C(M)
      IF (LN.EQ.0) GO TO 350
      LK=LMX-LN*2-9
      IF (I.EQ.NP) LK=LK-3
      IF (NP.EQ.MXPL) GO TO 355
      GO TO 360
  355 WRITE (6,365)
  365 FORMAT (1H //)
      WRITE (6,366)
  366 FORMAT (36H *PLANT COSTS ARE GIVEN AS THE TOTAL,
     +48H WORTH OF THE PLANT AS IT COMES ON LINE LESS THE //
     +46H SALVAGE VALUE AT THE END OF THE STUDY PERIOD.,
     +41H IF THE FIXED CHARGE RATE OPTION IS USED, //
     +39H CONSTRUCTION COSTS REPRESENT THE FIXED,
     +24H CHARGES FOR EACH PLANT.)
      LK=LK-8
  360 CONTINUE
      DO 320 IK=1,LK
      WRITE (6,330)
  330 FORMAT (1H )
  320 CONTINUE
  350 CONTINUE
      MN1=MXALL+3
      IF (NP.EQ.MXPL) WRITE(6,FMT1) IPAGE
      IF (NP.EQ.MN1) WRITE(6,FMT2) IPAGE
      WRITE (6,FMT3) IBASYR
      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
C
C______________________________________________________________________
      SUBROUTINE MAINT2(NSTCUR,NYNEXT)
C______________________________________________________________________
C
C
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),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                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),
     +                HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +                PSSCHD(20,8),SYSCUM(4)
C
C
C
C
      DIMENSION NSTCUR(20),ASP(4)
      RLT=365./4.
      DO 50 J=1,4
      DO 50 I=1,40
   50 ROM(I,J)=0.0
      DO 100 NP=1,4
      N=NP
      ASP(N)=AVSP(NYNEXT,N)
      DO 110 I=1,MXPL
      IF(NSTCUR(I).LE.0) GO TO 110
      NEXP=NSTCUR(I)
      ASP(N)=ASP(N)+FLOAT(NEXP)*PLACA(I,3)
  110 CONTINUE
  100 CONTINUE
C
C
      DO 120 I=1,MXPL
      IF(NSTCUR(I).LE.0) GO TO 120
      NEXP=NSTCUR(I)
      IP=IDEXP(I)+2
      DO 130 J=1,2
      IF(J.GT.1) GO TO 140
      C=PLACA(I,2)
      IF(C.LE.0.0) GO TO 120
      IN=2*IP
      GO TO 150
  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
```

```
  120 CONTINUE
      RETURN
      END
C
C
C-----------------------------------------------------------------------
      SUBROUTINE OBJFUN(NSTCUR,NY,OPEC,NSTATE,CAPCO,OBJORI,OBJ,
     +S,RESCAP,IRES)
C-----------------------------------------------------------------------
C
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
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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
      DIMENSION NSTCUR(20),NSTATE(20),NADD(20),CONCO(20),RESCAP(20,20)
C     CALCULATE THE PLANT ADDITIONS
      DO 10 I=1,MXPL
   10 NADD(I) = NSTCUR(I)-NSTATE(I)
C
C     CALCULATE THE UNDISCOUNTED CONSTRUCTION COST FOR EACH
C     CANDIDATE TYPE IN MILLION OF DOLLARS.  ALSO CALCULATE
C     THE TOTAL CONSTRUCTION COST
      C = 0.0
      DO 20 I=1,MXPL
      CONCO(I) = NADD(I)*PLACA(I,3)*PLACA(I,6)/1000.0*
     + ((1.0+PLACA(I,13))**NY)
   20 C = C+CONCO(I)
C     CALCULATE THE DISCOUNT FACTOR
      DISFAC = 1.0/(1.0+DISRAT)**NY
C
C     CALCULATE THE SALVAGE VALUE FOR EACH CANDIDATE USING
C     STRAIGHT LINE DEPRECIATION.  ALSO CALCULATE TOTAL SALVAGE
```

```
C       VALUE.
        S = 0.0
        DO 30 I=1,MXPL
C       NUMBER OF YEARS OF THE PLANT'S LIFE IN STUDY PERIOD.
        NYLEFT = MXYEAR-NY+1
C       PLANT SALVAGE VALUE
        SALV = CONCO(I)*(1.0-FLOAT(NYLEFT)/PLACA(I,9))
        RESCAP(NY,I) = (CONCO(I) - SALV)*DISFAC
   30   S = S+SALV*DISFAC
C
C
        IF (FCR .GT. 0.0) GO TO 40
C       CALCULATE DISCOUNTED CAPITAL COST AND OBJECTIVE FUNCTION
        CAPCO = C*DISFAC
        GO TO 50
   40   NYLEFT = MXYEAR-NY+1
        PWFC = (1.0-(1.0+DISRAT)**(-NYLEFT))/DISRAT
        DISFAC = 1.0/(1.0+DISRAT)**(NY+0.5)
        C = C*FCR*PWFC
        CAPCO = C*DISFAC
        S = 0.0
C
   50   OBJ = OBJORI+CAPCO-S+OPEC
        IF (FCR .LE. 0.0 .OR. IRES .LT. 1) GO TO 100
        DISFAC = 1.0/(1.0+DISRAT)**NY
        DO 60 I=1,MXPL
        CNEW = CONCO(I)*FCR*DISFAC
   60   RESCAP(NY,I)=RESCAP(NY,I)+CNEW
  100   RETURN
        END
C
C
C
C_____________________________________________________________________
        SUBROUTINE OPERCO(NSTCUR, NYNEXT, OPEC, IRES)
C_____________________________________________________________________
C
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
C                   *** DEFINITION OF LOCAL VARIABLES ***
C       FAC         REAL    -       DISCOUNT FACTOR
C       IDP         INT.    -       PLANT ID IN SCHEDULED OR CANDIDATE
C                                   PLANT FILES
C       P           REAL    -       PLANT AVAILABILITY
C       Y           REAL    -       NORMALIZED AND DERATED PLANT CAPACITY
C                                   FOR ALL MEMBER UNITS
C       COST        REAL    -       PLANT OPERATING COST IN $/MWH
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),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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
C
      DIMENSION NSTCUR(20),ITK(420,3),POPEC(250,20),CFH(3),ENRT(220,2),
     1   ENRH(3,2),ENY(2,30),ENP(2,30),NORDER(4)
      EQUIVALENCE (ELF(1),ITK(1)), (ELDC(1),POPEC(1))
C
C
      NCUM=4
      T=HOURS
C
C
C     DETERMINE EFFECTIVE LOADING ORDER
C
C
      DO 10 I=1,MXPL
      IP=IDEXP(I)
      IF(IP.LE.0) GO TO 10
      DO 11 K=1,MAXOR
      IF(LORDER(K,1).NE.IP) GO TO 11
      LORDER(K,3)=NSTCUR(I)
      J1=K+1
      DO 12 J=J1,MAXOR
      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
      IF(LORDER(K,3).LE.0) GO TO 13
      K1=K1+1
      DO 14 J=1,3
   14 ITK(K1,J)=LORDER(K,J)
   13 CONTINUE
      KNMAX=K1
C
C
C     EXPN MAINTENANCE ALLOCATION
C
C
      CALL MAINT2(NSTCUR,NYNEXT)
C
C
```

```
C      HYDRO COMPOSITE CHARACTERISTICS
C
C
       CALL COMPST(NSTCUR,NYNEXT)
C
C
       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
C
C      *** S E C T I O N  1 ***
C
C      ENERGY & COST CALCULATION FOR EACH SEASON
C
C      LOOP 60
C
C
       DO 60 NS=1,4
       N=NS
       ICUM=1
       CALL CUCAL1(NSTCUR,N,ICUM)
       DO 16 I=1,4
       NORDER(I)=0
    16 SYSCUM(I)=PTCUM(1,I,N)
       SUMP=0.0
       EPMP=0.0
       DO 18 K=1,3
       EAVAL(K)=0.0
    18 CFH(K)=0.0
       DO 19 J=1,2
       DO 19 I=1,3
    19 ENRH(I,J)=0.0
       DTOT=DTOT+ENEDEM(NYNEXT,N)
C
C
C
C      *** S E C T I O N  2 ***
C
C
C      BASE LOAD RUN-OF-RIVER CAPACITY
C
       IH=4
       DO 20 K=1,2
       P=1.0-HYDR(K,5)
       Y=HYDR(K,1)*CMULT(N)*(1.0-ROM(2,N))
       IF(Y.LE.0.0) GO TO 20
       Y1=SUMP
       SUMP=SUMP+Y
       Y2=SUMP
       IF(K.EQ.1) GO TO 25
C
C
C      CONVOLVE THE FIRST HYDRO
C
       DO 24 J=1,NCUM
    24 SYSCUM(J)=SYSCUM(J)+PTCUMH(1,J)
C
C
C      EVALUATE AREA UNDER EL-CURVE
C
```

```
C
   25 CALL AREA(Y1,Y2,P,EH,CF,CFP,IH,N)
      ENRH(K,1)=EH
C
C
C     MAX. HYDRO ENERGY AVAILABLE
C
C
      EAVAL(K)=HYDR(K,4)*EMULT(N)*1000.0-EH
      C=HYDR(K,2)-HYDR(K,1)*CMULT(N)
      IF(C.LE.0.0) GO TO 20
      CFH(K)=EAVAL(K)/(C*P*T)
   20 CONTINUE
      EPS=HYDR(3,2)*1000.0
C
C
C     CONVOLVE THE SECOND HYDRO
C
C
      DO 27 J=1,NCUM
   27 SYSCUM(J)=SYSCUM(J)+PTCUMH(2,J)
C
C
C     ORDER OF HYDRO PLANT IN DECREASING CAPACITY FACTOR
C
C
      NUMBER=0
      IF(CFH(1).GE.CFH(2)) GO TO 800
      NORDER(1)=2
      IF(CFH(1).LE.0.0) GO TO 802
      NORDER(2)=1
      NUMBER=2
      GO TO 801
  800 IF(CFH(1).LE.0.0) GO TO 801
      NORDER(1)=1
      IF(CFH(2).LE.0.0) GO TO 802
      NORDER(2)=2
      NUMBER=2
      GO TO 801
  802 NUMBER=1
  801 IF(EPS.LE.0.0) GO TO 803
      NUMBER=NUMBER+1
      NORDER(NUMBER)=3
  803 NUMBER=NUMBER+1
      NORDER(NUMBER)=4
C
C
C     INITIALIZE INDEX FOR HYDRO LOADING ORDER
C
C
      NOHY=1
      IH=NORDER(NOHY)
      IPUMP=-1
      IF(EPS.LE.0.0) IPUMP=1
      KLOW1=0
      KLOW2=0
      SUEN2=0.0
C
C
C     *** S E C T I O N   3 ***
C
C     START LOADING THE THERMAL PLANTS ,
C
C     LOOP 70
C
C
      DO 70 KC=1,KNMAX
      K=KC
```

```
      ID1=ITK(K,2)
      IP=ITK(K,1)
      IUNITB=1
      IUNITE=ITK(K,3)
C
      DO 630 J=1,30
      ENP(2,J)=0.0
  630 ENY(2,J)=0.0
C
C
C     INDEX FOR EMOR ARRAY
C
C
      IF(IP.GT.MAXINP) GO TO 55
      IF(ID1.GT.1) GO TO 50
      IMANT=2*(IP+2)
      R=1.0-ROM(IMANT,N)
      GO TO 56
   50 IMANT=2*IP+3
      R=1.0-ROM(IMANT,N)
      GO TO 56
   55 JP=IP-MAXINP
      R=1.0
      IF(MAINS(JP).EQ.N) R=1.0-PLANT(IP,4)/(365.0/4.0)
   56 PT=1.0-PLANT(IP,5)
      IF(ID1.GT.1) GO TO 57
      YT=PLANT(IP,2)*R
      GO TO 74
   57 YT=(PLANT(IP,3)-PLANT(IP,2))*R
C
C
C     LOOP FOR IDENTICAL UNITS
C
C     LOOP 75
C
C
   74 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
      K1X=K-1
      IQ=ITK(K1X,1)
      ID2=ITK(K1X,2)
C
C
C     PLANT CONVOLUTION
C
C
   79 DO 90 J=1,NCUM
      J1=J
      IF(ID2.GT.1) J1=J+4
      IF(IQ.GT.MAXINP) GO TO 92
      SYSCUM(J)=SYSCUM(J)+PTCUMX(IQ,J1)
      GO TO 90
   92 JC=IQ-MAXINP+1
      SYSCUM(J)=SYSCUM(J)+PTCUM(JC,J1,N)
   90 CONTINUE
      IF(ID1.LE.1) GO TO 150
C
C
C     PLANT DECONVOLUTION
C
C
```

```
C
      DO 100 J=1,NCUM
      IF(IP.GT.MAXINP) GO TO 101
      SYSCUM(J)=SYSCUM(J)-PTCUMX(IP,J)
      GO TO 100
  101 JC=IP-MAXINP+1
      SYSCUM(J)=SYSCUM(J)-PTCUM(JC,J,N)
  100 CONTINUE
C
C
  150 CONTINUE
C
C
C     EVALUATE AREA UNDER THE EL-CURVE
C
C
  152 CALL AREA(Y1,Y2,PT,ET,CF,CFP,IH,N)
      ENY(2,I)=ET
C
C
C     ANY MORE HYDRO LEFT
C
C
      IF(IH.GE.4) GO TO 75
C
C
C     CHECK HYDRO GENERATION FEASIBILITY
C
C
      IF(CF.LE.CFH(IH)) GO TO 200
C
C
C     CHECK PUMPING DUTY ASSIGNMENT
C
C
      IF(IPUMP.GT.0.OR.CFP.GE.1.0) GO TO 75
C
C
C     PUMPING ASSIGNMENT
C
C
      PP=1.0-HYDR(3,5)
      YP1=Y1-HYDR(3,1)*(1.0-ROM(3,N))
      YP2=Y2-HYDR(3,1)*(1.0-ROM(3,N))
      CALL AREA(YP1,YP2,PT,EP,CF,CFP,IH,N)
      ENP(2,I)=PP*(EP-ET)
      EPMP=EPMP+ENP(2,I)*HYDR(3,3)
      IF(EPMP.LT.EPS) GO TO 570
C
C
C     STOP PUMPING
C
C
      IPUMP=1
      ENP(2,I)=ENP(2,I)-(EPMP-EPS)/HYDR(3,3)
      EPMP=EPS
      EAVAL(3)=EPS*HYDR(3,4)
      CFH(3)=EAVAL(3)/(T*HYDR(3,1)*(1.0-HYDR(3,5)))
      GO TO 75
C
C
C
  570 EAVAL(3)=EPMP*HYDR(3,4)
      CFH(3)=EAVAL(3)/(T*HYDR(3,1)*(1.0-HYDR(3,5)))
      GO TO 75
C
C
C     HYDRO GENERATION
```

```
C
C
  200 CONTINUE
      ICNTL=-1
C
C
C     REPLACE THE THERMAL BY HYDRO
C
C
      SUMP=SUMP-YT
      ENY(2,I)=0.0
      ENP(2,I)=0.0
      IF(IDI.LE.1) GO TO 215
C
C
C     RETRIVE EL-CURVE INFO
C
C
      DO 210 J=1,NCUM
      IF(IP.GT.MAXINP) GO TO 211
      SYSCUM(J)=SYSCUM(J)+PTCUMX(IP,J)
      GO TO 210
  211 JC=IP-MAXINP+1
      SYSCUM(J)=SYSCUM(J)+PTCUM(JC,J,N)
  210 CONTINUE
  215 CONTINUE
C
C
C     DECONVE BASE LOADED HYDRO
C
C
  216 DO 220 J=1,NCUM
  220 SYSCUM(J)=SYSCUM(J)-PTCUMH(IH,J)
C
C
      PH=1.0-HYDR(IH,5)
      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
      SUMP=SUMP+YH
      Y2=SUMP
      CALL AREA(Y1,Y2,PH,EH,CF,CFP,IH,N)
      ENRH(IH,2)=EH
      IF(ICNTL.GT.0) GO TO 240
      KLOW1=K
      KLOW2=I-1
C
C
C     CONVOLVE THIS HYDRO
C
C
      DO 230 J=1,NCUM
      J1=J+4
  230 SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J1)
C
C
C     NEXT HYDRO
C
C
      IF(IH.EQ.3) GO TO 620
C
      NOHY=NOHY+1
      IH=NORDER(NOHY)
      ICNTL=1
      GO TO 216
  240 IF(EH.GT.EAVAL(IH)) GO TO 250
      ENRH(IH,2)=EH
      KLOW1=K
```

```
      KLOW2=I-1
      GO TO 300
  250 CONTINUE
C
C
C     REPLACE THIS HYDRO BY THERMAL
C
      SUMP=SUMP-YH
      DO 270 J=1,NCUM
  270 SYSCUM(J)=SYSCUM(J)+PTCUMH(IH,J)
      NOHY=NOHY-1
      IH=NORDER(NOHY)
C
  620 IBLK=-1
      CALL TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,IPUMP,
     +NORDER,EPMP)
      IPUMP=1
      GO TO 455
C
C
  300 CONTINUE
C
C
C     FORM CLUSTER AND TEST SWAP LOADING POSITION
C
C
      NPAST=NOHY-1
      IHPT=NORDER(NPAST)
      IBLK=1
      IF(KLOW2.LE.0) GO TO 301
      KIX=KLOW1
      ISWP=2
      ITRM=KLOW2
      GO TO 302
  301 KIX=KLOW1-1
      ISWP=1
      ITRM=ITK(KIX,3)
  302 II1=ITK(KIX,1)
      II2=ITK(KIX,2)
      DO 310 J=1,NCUM
      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
      SYSCUM(J)=SYSCUM(J)-PTCUM(JC,J1,N)
  310 CONTINUE
      DO 320 J=1,NCUM
      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-ROM(IMANT,N)
      GO TO 325
  321 IMANT=2*II1+3
      R=1.0-ROM(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
      Y1=SUMP-Y
      CALL AREA(Y1,Y2,P,ET,CF,CFP,IH,N)
      SUEN1=ENY(ISWP,ITRM)+ENRH(IH,2)+ENRH(IHPT,2)
      SUEN2=SUEN1-ET
      ETOT=EAVAL(IH)+EAVAL(IHPT)
      IF(SUEN2.GT.ETOT) GO TO 400
C
C
C     ACCEPT SWAP TEST
C
C
      ENY(ISWP,ITRM)=ET
C
C
      KLOW1=K
      KLOW2=I-2
C
C
      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
C
C
  400 CONTINUE
C
C
C     REJECT SWAP TEST
C
C
      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
C
C
C     LOAD THE CURRENT OFF-LOADED THERMAL PLANT
C
C
      DO 460 J=1,NCUM
      J1=J
      IF(II2.GT.1) J1=J+4
      IF(II1.GT.MAXINP) GO TO 461
      SYSCUM(J)=SYSCUM(J)+PTCUMX(II1,J1)
      GO TO 460
  461 JC=II1-MAXINP+1
      SYSCUM(J)=SYSCUM(J)+PTCUM(JC,J1,N)
  460 CONTINUE
C
C
  455 NOHY=NOHY+1
      IH=NORDER(NOHY)
C
      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,IH,N)
      ENY(2,I)=ET
C
C
   75 CONTINUE
C
C
C     END OF IDENTICAL UNITS
C
C
      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=1,L3
      GENTOT=GENTOT+ENY(1,L)
   77 ENRT(L1,L2)=ENRT(L1,L2)+ENY(1,L)+ENP(1,L)
   80 DO 78 I=IUNITB,IUNITE
      ENP(1,I)=ENP(2,I)
   78 ENY(1,I)=ENY(2,I)
C
C
      IF(K.LT.KNMAX) GO TO 70
      DO 81 I=IUNITB,IUNITE
      GENTOT=GENTOT+ENY(2,I)
   81 ENRT(IP,ID1)=ENRT(IP,ID1)+ENY(2,I)+ENP(2,I)
C
C
   70 CONTINUE
C
C
C     END OF SECTION 3
C
C
      IF(IH.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
      YH=HYDR(3,1)*(1.0-ROM(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)
C
C
C
   66 DO 67 J=1,2
      DO 67 I=1,2
   67 HYTOT=HYTOT+ENRH(I,J)
      GENTOT=GENTOT+ENRH(3,2)
C
C
C
      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
C
C     *** S E C T I O N  4 ***
C
C     COST CALCULATION
C
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
C
      GENTOT=GENTOT+HYTOT
C
C
      UNE(NYNEXT)=DTOT-GENTOT
```

```
      IF(UNE(NYNEXT).LT.0.0) UNE(NYNEXT)=0.0
C
C
      DO 500 I=1,MXPL
      IP=IDEXP(I)
      IF(IP.LE.0) GO TO 500
      DO 510 K=1,MAXOR
      IF (LORDER(K,1).NE.IP) GO TO 510
      LORDER(K,3)=-1
      JJ=K+1
      DO 520 J=JJ,MAXOR
      IF(LORDER(J,1).NE.IP) GO TO 520
      LORDER(J,3)=-1
      GO TO 500
  520 CONTINUE
      GO TO 500
  510 CONTINUE
  500 CONTINUE
C
C
      IF(IRES.NE.1) GO TO 1500
      WRITE(40,1400) CTFUL
      WRITE(40,1400) CTOM
 1400 FORMAT(E12.5)
C
C
 1500 RETURN
C
C
      END
C
C
C_____________________________________________________________________
      SUBROUTINE PACK(NUNP, NPA)
C_____________________________________________________________________
C
C
C
C           TITLE:      STATE PACKING
C           FUNCTION:   THIS ROUTINE PACKS CURRENT STATE "NSTCUR" INTO
C                       ONE WORD "NST" USING PACKING BASE VECTOR IPABA
C
C           CALLED BY:  MAIN
C           CALLS TO:   NONE
C
C           SUBROUTINE ARGUMENTS
C           NAME        DESCRIPTION
C           NPA         PACKED STATE
C           NUNP      UNPACKED STATE
C
C           COMMON VARIABLES USED: IPABA, MXPL
C
C           LOCAL VARIABLES
C           NAME DESCRIPTION
C           I           I-TH EXPANSION PLANT TYPE
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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
      DIMENSION NUNP(20)
C           INITIALIZE NST
      NPA = 0
C           PACK NSTCUR
      DO 10 I=1,MXPL
   10 NPA = NPA+NUNP(I)*IPABA(I)
      RETURN
      END
C
C
C_______________________________________________________________
      FUNCTION PLOLP(Y)
C_______________________________________________________________
C
C     FIND LOLP.  I.E. THE EL(I) POINT THAT CORRESPONDS TO Y
C     Y: NORMALIZED MW CAPACITY
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(6),IPCH(8),ISOL(20,3),
     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)
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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
```

```
     +                PSSCHD(20,8),SYSCUM(4)
C
C
C
C
C
      YY = Y*MAXPO
      IY = YY
      MAXIM = MAXI-1
      IF (IY .GT. MAXIM) GO TO 100
      DY = IY+1-YY
      PLOLP = (EL(IY)-EL(IY+1))*DY+EL(IY+1)
      RETURN
  100 PLOLP = 0.0
      RETURN
      END
C
C
C
C______________________________________________________________________
      SUBROUTINE PWADD(Y,P,NEWMAX)
C______________________________________________________________________
C
C
C     DETERMINES THE NEW ELDC AFTER THE ADDITION OF THE UNIT WITH
C     NORMALIZED MW CAPACITY Y AND RELIABILITY P.
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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C           DEFINITION OF TERMS:
C
C           X: X-AXIS I.E. NORMALIZED MW AXIS
C           EL: LOAD PROBABILITY MATRIX.
C           ELF: TEMPORARY LOAD PROBABILITY MATRIX. STORES THE NEW
C                EL TEMPORARILY.
C           Y: NORMALIZED MW CAPACITY OF ADDED UNIT: Y=MWC/PERPK
```

```
C            P=1-FOR, LOAD PROBABILITY
C            Q=FOR ,  FORCED AOUTAGE RATE
C            MAXI: EXPECTED MAXIMUM DIMENSION OF EL MATRIX.
C            PMAX: =0.999 IS CONSIDERED A REALISTIC LIMIT OF UNIT
C                     AVAILABILITY ESTIMATE. I.E. IF P>PMAX
C                     IT IS ASSUMED THAT P=1.0 AND THE ELD
C                     CURVE DOES NOT CHANGE
C            ELMIN=0.0000001, IS THE LIMIT OF LOLP ACCURACY.
C                     IF LOLP < ELMIN THEN LOLP = 0.0
C
      IF(P.GE.PMAX) GO TO 100
C
C            FORM THE NEW ELD CURVE. STORE IN  ELF
C
      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
C
C            ALL ELD VALUES FOR WHICH I SATISFIED THE :
C                     NEWMAX<I<MAXI,HAVE EL(I)<ELMIN AND
C                     ARE LEFT TO BE ZERO.(NOTE THAT THE EL MATRIX
C                     IS INITIALIZED TO ZERO IN LOADSY)
C
C            REDEFINE THE EL MATRIX AS THE NEW ELD CURVE
C                     THAT IS TEMPORARILY STORED IN ELF.
C
      DO 60 I=1,NEWMAX
      EL(I)=ELF(I)
   60 CONTINUE
C
  100 CONTINUE
      RETURN
      END
C
C
C
C
C_______________________________________________________________________
      SUBROUTINE READIN
C_______________________________________________________________________
C
C
C
      COMMON /VRS/ CLOLP,CLOLPU,CPLOLP,DISRAT,DX,
     1             FCR,HOURS,IBASYR,IMXPL,ITIN,ITMAX,
     2             ITOUT,MAXALL,MAX1,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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
C     CRITICAL LOLP, HOURS PER SEASON
      READ(10,902)  CPLOLP, HOURS
      WRITE(11,902) CPLOLP, HOURS
C     BASE YEAR (CALENDAR), MAX.ELDC DIMENSION, NUMBER OF NEW PLANTS
C     NUMBER OF SCHEDULED PLANTS, ORIGINAL ELDC DIMENSION,
C     NUMBER OF YEARS IN THE STUDY PERIOD
      READ(10,900) IBASYR,MAXI,MAXINP,MAXPLA,MAXPO,MXYEAR,MAXOR
      WRITE(11,900)IBASYR,MAXI,MAXINP,MAXPLA,MAXPO,MXYEAR,MAXOR
C     MAXIMUM NUMBER OF POINTS OF SCHEDULED SYSTEM FINAL
C     ELDC'S. (BY SEASON PER YEAR)
C
      DO 10 NY = 1, MXYEAR
      READ(10,900)  (MXELDC(NY,I),I = 1,4)
   10 WRITE(11,900) (MXELDC(NY,I),I = 1,4)
C     READ FOR EACH YEAR:  SCHEDULED SYSTEM CAPACITY,
C     MIN. &MAX. RES. MARGINS, MINIMUM POSIBLE YEARLY
C     PRODUCTION COST
      DO 20 NY = 1, MXYEAR
      READ(10,902)  CAPABS(NY),RESMIN(NY),RESMAX(NY),FATOPE(NY)
   20 WRITE(11,902) CAPABS(NY),RESMIN(NY),RESMAX(NY),FATOPE(NY)
C     DEFINE AND PRINT THE NORMALIZED X-AXIS.
      DX = 1.0 / MAXPO
      DO 30 I = 1,MAXI
   30 X(I) = I * DX
      WRITE(11,906)  (X(I),I = 1,MAXI)
C     PEAK LOAD DEMAND (BY SEASON PER YEAR).
C
      DO 40 NY = 1,MXYEAR
      READ(10,902)  (PEAKS(NY,I),I = 1,4)
   40 WRITE(11,902) (PEAKS(NY,I),I = 1,4)
C     SCCHEDULED SYSTEM TOTAL ENERGY DEMAND (BY SEASON PER YEAR)
      DO 50 NY = 1,MXYEAR
      READ(10,902)  (ENEDEM(NY,I),I =1,4)
   50 WRITE(11,902) (ENEDEM(NY,I),I =1,4)
C     EXPANSION CANDIDATE DATA
      MXEXPN=MAXINP+2
      CLOLPU=0.01
      READ(10,911) (HYEXPN(J),J=1,11)
      READ(10,911) (PSEXPN(J),J=1,11)
C
C
      DO 140 I=1,2
      IF(I.GT.1) GO TO 130
```

```
      PLACA(I,2)=HYEXPN(1)
      PLACA(I,3)=HYEXPN(2)
      PLACA(I,4)=HYEXPN(6)
      PLACA(I,5)=HYEXPN(5)
      PLACA(I,6)=HYEXPN(9)
      PLACA(I,9)=30.0
      PLACA(I,13)=HYEXPN(10)
      GO TO 140
  130 PLACA(I,2)=0.0
      PLACA(I,3)=PSEXPN(1)
      PLACA(I,4)=PSEXPN(6)
      PLACA(I,5)=PSEXPN(5)
      PLACA(I,6)=PSEXPN(9)
      PLACA(I,9)=30.0
      PLACA(I,13)=PSEXPN(10)
  140 CONTINUE
      DO 60 IP=3,MXEXPN
      READ(10,904) (PLACA(IP,J),J=1,14)
C     WRITE(11,904) (PLACA(IP,J),J=1,14)
   60 CONTINUE
C     MIN. & MAX. NUMBER OF UNITS ALLOWED FOR EACH CANDIDATE PER YEAR.
      DO 80 NY = 1,MXYEAR
      READ(10,900)  (MINUN(NY,IP),IP = 1,MXEXPN)
      WRITE(11,900) (MINUN(NY,IP),IP = 1,MXEXPN)
      READ(10,900)  (MAXUN(NY,IP), IP=1,MXEXPN)
   80 WRITE(11,900) (MAXUN(NY,IP) ,IP =1,MXEXPN)
C
C     READ PREP MODULE SOLUTION FOR CALCULATION OF OBJ UPPER BOUND
      DO 90 NY=1,MXYEAR
      READ(10,900) (NSTPRE(NY,IP),IP=1,MXEXPN)
   90 WRITE(11,901) (NSTPRE(NY,IP),IP=1,MXEXPN)
C     READ SPACE AVAILABLE FOR MAINTENANCE
      DO 100 NY=1,MXYEAR
      READ(10,902) (AVSP(NY,J), J=1,4)
  100 WRITE(11,903) (AVSP(NY,J), J=1,4)
      DO 110 NY=1,MXYEAR
      READ(10,909) (SSCUM(J,NY),J=1,4)
  110 WRITE(11,910) (SSCUM(J,NY),J=1,4)
C
C     READ NORMAL DISTRIBUTION TABLE
      READ(9,907) (TY(K),TINT(K),K =1,120)
C     WRITE(11,908) (TY(K),TINT(K),K =1,120)
C
      DO 120 NY=1,MXYEAR
      READ(31,1010) NDUM
      READ(31,911) (HYSCHD(NY,J),J=1,8)
      READ(33,1010) NDUM
      READ(33,911) (PSSCHD(NY,J),J=1,8)
  120 CONTINUE
      READ(10,911) (CMULT(J),J=1,4)
      READ(10,911) (EMULT(J),J=1,4)
  900 FORMAT(20I4)
  901 FORMAT(1H ,20I4)
  902 FORMAT(5E16.8)
  903 FORMAT(1H ,4E16.8)
  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)
  906 FORMAT(10F11.6)
  907 FORMAT(F10.4,F12.8)
  908 FORMAT(1H ,5(F10.4,F12.8))
  909 FORMAT(4E16.8)
  910 FORMAT(1H ,4E16.8)
  911 FORMAT(5E12.5)
 1010 FORMAT(/,3I3)
C
      RETURN
      END
C
```

```
C-----------------------------------------------------------------------
      SUBROUTINE REPORT(SALV,RESCAP)
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),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             FATOPE(20),SOL(20,2),AUPEAK(20)
C
      COMMON /LCS/ EL(1250),POPEC(250,20),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
      DIMENSION SALV(20),RESCAP(20,20),OPTOT(20),CAP(20),
     +          ETABLE(20,10),RSCAP1(20,10),POPEC1(10,20),
     +          PL(10),ALPHA(20)
      DIMENSION A1(250),A2(20),FMT1(18),FMT2(18),FMT3(18),
     +FMT4(18),FMT5(18),FMT6(18),FMT7(18),C(60),IYR(20)
      EQUIVALENCE (EL(1),A1(1)),(EL(251),ETABLE(1)),
     +            (EL(451),POPEC1(1)),(EL(651),RSCAP1(1)),
     +            (EL(851),OPTOT(1)),(EL(871),CAP(1)),
     +            (EL(891),IYR(1)),(EL(911),PL(1)),(EL(921),A2(1))
      EQUIVALENCE (EL(941),C(1)),(EL(1001),FMT1(1)),
     +            (EL(1021),FMT2(1)),(EL(1041),FMT3(1)),
     +            (EL(1061),FMT4(1)),(EL(1081),FMT5(1)),
     +            (EL(1101),FMT6(1)),(EL(1121),FMT7(1))
      DATA ALPHA/3H  1,3H  2,3H  3,3H  4,3H  5,3H  6,3H  7,3H  8,
     +3H  9,3H 10,3H 11,3H 12,3H 13,3H 14,3H 15,3H 16,3H 17,
     +3H 18,3H 19,3H 20/
      DATA HY/3H HY/, PS/3H PS/, ZERO/3H  0/, BLNK/3H   /
      REWIND 19
      MX1=MXYEAR-MXYEAR/2
C
C
C
      DO 10 I=1,8
   10 PL(I)=BLNK
C
C
C
      DO 14 I=1,MXPL
      IP=IDEXP(I)
      IF(IP.GT.0) GO TO 12
      IF(IP.GT.-1) GO TO 11
      PL(I)=HY
      GO TO 14
   11 PL(I)=PS
```

```
      GO TO 14
   12 DO 13 J=1,20
      IF(IP.NE.J) GO TO 13
      PL(I)=ALPHA(J)
      GO TO 14
   13 CONTINUE
   14 CONTINUE
C
C
C
      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
C
C
C
      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(IS.NE.J) GO TO 18
      ETABLE(NY,I)=ALPHA(J)
      GO TO 19
   18 CONTINUE
   19 CONTINUE
   20 CONTINUE
C
C
C
      MN1=MXALL+3
      DO 23 NY=1,MXYEAR
      DO 21 I=1,MN1
   21 OPTOT(NY)=OPTOT(NY)+POPEC(I,NY)
      DO 22 I=1,MXPL
   22 CAP(NY)=CAP(NY)+RESCAP(NY,I)
   23 CONTINUE
C
C
C
      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 ,I5,1X,8A3,1X,F11.6,1X,E12.5,2F10.3,1X,F10.3,
     +2X,F10.3)
   60 CONTINUE
```

```
      WRITE(6,66)
   66 FORMAT(//1H ,46H *  ZERO MEANS LOLP VALUE SMALLER THAN 0.00001)
      WRITE (6,67) IBASYR
   67 FORMAT (//1H ,16H *  MILLIONS OF ,I4,8H DOLLARS)
      WRITE (6,69)
   69 FORMAT (//1H ,38H ** PLANT COSTS ARE GIVEN AS THE TOTAL,
     +48H WORTH OF THE PLANT AS IT COMES ON LINE LESS THE //
     +46H SALVAGE VALUE AT THE END OF THE STUDY PERIOD.,
     +41H IF THE FIXED CHARGE RATE OPTION IS USED, //
     +34H CAPITAL COSTS REPRESENT THE FIXED,
     +24H CHARGES FOR EACH PLANT.)
      WRITE (6,41)
      DO 74 I=1,10
      WRITE (6,71)
   71 FORMAT (1H /)
   74 CONTINUE
      READ (19,99) C
   99 FORMAT (30A2)
      WRITE (6,310)
  310 FORMAT (1H1)
      READ (19,100) FMT1
      READ (19,100) FMT2
      READ (19,105) FMT3
      READ (19,110) FMT4
      READ (19,110) FMT5
      READ (19,115) FMT6
      READ (19,116) FMT7
  100 FORMAT (2A4,A2,10A4,A2,4A4)
  105 FORMAT (A3,A2,A2,15A4)
  110 FORMAT (4A4,A2,13A4)
  115 FORMAT (3A4,A1,A2,13A4)
  116 FORMAT (5A4,A2,12A4)
      NP=MN1
      DO 122 NY=1,MXYEAR
  122 A2(NY)=OPTOT(NY)
      CALL TABLE(FMT1,FMT2,FMT3,FMT4,FMT5,FMT6,
     +           FMT7,C,A1,A2,IYR,NP,MX1)
      NP=MXPL
      DO 130 I=1,NP
      DO 140 J=1,MXYEAR
  140 POPEC(I,J)=RESCAP(J,I)
  130 CONTINUE
      DO 125 NY=1,MXYEAR
  125 A2(NY)=CAP(NY)
      CALL TABLE(FMT1,FMT2,FMT3,FMT4,FMT5,FMT6,
     +           FMT7,C,A1,A2,IYR,NP,MX1)
      RETURN
      END
C
C_______________________________________________________________________
      SUBROUTINE RESIM(N, NBEST, IRES, KEEP)
C_______________________________________________________________________
C
C     RESIMULATES THE NBEST SOLUTIONS
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),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                 FATOPE(20),SOL(20,2),AUPEAK(20)
C
      COMMON /LCS/ EL(1250),POPEC(250,20),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
C
C
      DIMENSION KEEP(10),SALV(20),NSTATE(20),NSTCUR(20),RESCAP(20,20)
     + ,REFIND(50)
C
C     FIND THE NBEST SOLUTIONS THE FIRST TIME CALLED FOR EACH MAIN
C     ITERATION
      KEEP(1) = ITIN
C     WRITE(6,*) ITIN,N,NBEST,KEEP(1)
C     WRITE(11,*)ITIN,N,NBEST,KEEP(1)
      DO 5 NY = 1,MXYEAR
      I28 = NY
    5 READ(28'I28,900) (ISOL(NY,I),I=1,8),SOL(NY,1),SOL(NY,2)
C     WRITE(6,*) (ISOL(NY,I),I=1,6),SOL(NY,1),SOL(NY,2)
C   5 WRITE(11,*) (ISOL(NY,I),I=1,6),SOL(NY,1),SOL(NY,2)
      IF (NBEST .EQ. 1 ) GO TO 100
      IF (N.NE.1) GO TO 100
C
      FFIND = UBOUND*100.0
      KSTOP = ITIN-1
      IF (KSTOP .LT. 1) GO TO 100
      DO 10 K=1,KSTOP
      I28 = (K-1)*20 + MXYEAR
      READ(28'I28,900) (ISOL(MXYEAR,I),I=1,8),(SOL(MXYEAR,J),J=1,2)
   10 REFIND(K) = SOL(MXYEAR,2)
C
  900 FORMAT(8I4,2E14.7)
C
      DO 30 M=2,NBEST
      FIND = 0.99*FFIND
      DO 20 K = 1,KSTOP
      IF (REFIND(K).GE.FIND) GO TO 20
      KK = K
      FIND = REFIND(K)
   20 CONTINUE
      REFIND(KK) = FFIND
   30 KEEP(M) = KK
C
C
C     RESIMULATE THE N-TH BEST SOLUTION
C
  100 ISTAR = (KEEP(N)-1)*20 + 1
C     WRITE(6,*) N,KEEP(N),I28
C     WRITE(11,*) N,KEEP(N),I28
      OBJ = 0.0
C     INITIALIZE RESCAP & POPEC.
```

```
      REWIND 24
      REWIND 27
      DO 200 NY=1,MXYEAR
      I28 = ISTAR + NY - 1
      READ(28'I28,900) (ISOL(NY,I),I=1,8),SOL(NY,1),SOL(NY,2)
      WRITE(40,910) (ISOL(NY,I),I=1,MXPL)
  910 FORMAT(8I4)
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) ((LORDER(J,K),K=1,3),J=1,420)
      READ(24,921) (MAINS(J),J=1,200)
  902 FORMAT(125(I3,I2,I3))
  921 FORMAT(200I2)
      DO 230 I=1,MXPL
      NSTATE(I) = 0
      NSTCUR(I) = ISOL(NY,I)
      IF (NY .NE. 1) NSTATE(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
C     WRITE(6,*) NSTCUR,NSTATE
C     WRITE(6,*) OPEC,CAPCO,OBJ,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
C     DO 300 NY = 1,MXYEAR
C     WRITE(11,*) (RESCAP(NY,K),K=1,MAXIPN)
C     WRITE(11,*) (POPEC(I,NY),I=1,40)
C 300 CONTINUE
      CALL REPORT(SALV,RESCAP)
C
      RETURN
      END
C
C_______________________________________________________________________
      SUBROUTINE SEARCH (NST, NST2, NUMACC, NYCR, NACCEP, NTH)
C_______________________________________________________________________
C         TITLE:        STATE SEARCH SUBROUTINE
C         FUNCTION:     THIS SUBROUTINE SEARCHES THE FIRST SLOT
C                       OF ARRAY NST2 TO FIND WEATHER STATE "NST" IS
C                       ALREADY CONTAINED IN NST2.  IF SO, NTH IS
C                       THE STATE INDEX NUMBER, IF NOT, NTH IS 0.
C
C         CALLED BY: MAIN
C         CALLS TO:  NONE
C
C         SUBROUTINE ARGUMENT
C         NAME          DESCRIPTION
C         NACCEP        NUMBER OF ACCEPTED STATES
C         NST           CURRENT STATE PACKED
C         NST2          ARRAY WITH ACCEPTED STATES (TO BE SEARCHED)
C         NTH           INDEX OF STATE FOUND IN NST ARRAY.
C                       IF 0, STATE WAS NOT FOUND
C         NUMACC        (NYCR) NUMBER OF STATES ACCEPTED IN YEAR
C                       "NYCR" AFTER LAST FATHOMING
C         NYCR          CURRENT YEAR
C
```

```
C         COMMON VARIABLES USED:  NONE
C         LOCAL VARIABLES
C         N          TEMPORARY INDEX OF SEARCH POINTER
C         NBEGIN     BEGINNING INDEX OF SEARCH RANGE
C         NEND       ENDING INDEX OF SEARCH RANGE
C         NRANGE     SEARCH RANGE
      DIMENSION NUMACC(20), NST2(2,1000)
C
C
C         DEFINE SEARCH RANGE
      NBEGIN = NUMACC(NYCR) + 1
      NEND = NBEGIN - 1 + NACCEP
C
C
      DO 10 N=NBEGIN, NEND
      IF (NST .NE. NST2(1,N)) GO TO 10
C         STATE "NST" IS FOUND IN "NST2" ARRAY IDENTIFY THE
C         SEQUENCE NUMBER AND RETURN
      NTH = N
      GO TO 20
   10 CONTINUE
C         DROPPING THROUGH THE LOOP MEANS THAT STATE "NST" WAS
C         NOT FOUND IN THE "NST2" ARRAY.  THIS FACT IS INDICATED
C         BY SETTIN NTH = 0
      NTH = 0
   20 RETURN
      END
C
C
C
C_______________________________________________________________________
      SUBROUTINE STAGEN(NSTCNT, CUMLOL,NYCR)
C_______________________________________________________________________
C
C
C          TITLE     STAGE GENERATION
C
C          FUNCTION    THIS SUBROUTINE GENERATES ALL THE ACCEPTABLE
C                      STATES WITHIN THE TUNNEL DEFINED BY THE
C                      "TUNNEL" SUBROUTINE.  STATE ACCEPTABILITY
C                      IS CHECKED THROUGH CALLS TO THE "CONSTR"
C                      SUBROUTINE.  THE ACCEPTED STATES ARE STORED
C                      IN THE "LIST" ARRAY.
C
C          CALLED BY   MAIN
C          CALLS TO    CONSTR
C
C          SUBROUTINE ARGUMENTS
C          NAME        DESCRIPTION
C          NSTCNT      COUNTER OF THE NUMBER OF ACCEPTED
C                                STATES
C          COMMON VARIABLES USED.  LIST, MAXTUN, MINTUN, MXPL
C          LOCAL VARIABLES
C          NAME     TYPE  SIZE  DESCRIPTION
C          I           DUMMY PLANT TYPE INDEX
C          IFLCON      CONSTRAINT FLAG.
C                                IF =0, STATE IS ACCEPTABLE
C                                IF =1, STATE IS UNACCEPTABLE
C          NSTCUR      NSTCUR(I) DENOTES THE NUMBER OF
C                                UNITS OF THE I-TH PLANT TYPE
C                                CONTAINT IN STATE "NSTCUR".
C
C
C
C          ALGORITHM LOGIC
C          THE FIRST STATE IS IDENTICAL TO THE TUNNEL LOWER BOUND.
C          A NEW STATE IS GENERATED RECURSIVELY FROM THE PREVIOUS-
C          LY GENERATED STATE BY INCREASING THE NUMBER OF UNITS
```

```
C          IN THE FIRST PLANT TYPE WHOSE UNIT NUMBER HAS NOT YET
C          REACHED THE TUNNEL UPPER LIMIT.  EACH TIME THE UPPER
C          LIMIT FOR A PLANT TYPE (I) IS REACHED, RESET UNIT
C          NUMBER FOR PLANT TYPES 1 THROUGH I TO THE LOWER TUNNEL
C          BOUND.
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),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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
      DIMENSION NSTCUR(20), CUMLOL(1000)
C
C
C          INITIALIZE STATE COUNTER
      NSTCNT = 0
C
C          INITIALIZE NSTCUR TO THE LOWER TUNNEL BOUNDARY
      DO 10 I=1,MXPL
   10 NSTCUR(I) = MINTUN(I)
C          CHECK IF NEW STATE IS ACCEPTABLE
   20 CALL CONSTR(NSTCUR, NYCR, CULOLP, IFLCON)
      IF (IFLCON .GE. 1) GO TO 40
C          NOT BRANCHING TO 40 MEANS NSTCUR IS ACCEPTABLE
C          INCREMENT STATE COUNTER AND STORE IN LIST ARRAY
      NSTCNT = NSTCNT + 1
      DO 30 I=1,MXPL
   30 LIST(I,NSTCNT) = NSTCUR(I)
C          STORE CUMULANT LOLP IN CORE
      CUMLOL(NSTCNT) = CULOLP
C          FIND THE FIRST PLANT TYPE WHOSE UNIT NUMBER IN THE LAST
C          STATE GENERATED HAS NOT REACHED THE TUNNEL UPPER BOUND.
C          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
C          DROPPING THROUGH THE LOOP MEANS THAT THE TUNNEL UPPER
C          BOUND IS REACHED FOR ALL PLANT TYPES, I.E. ALL
C          POSSIBLE STATES WERE GENERATED.
```

```
C     WRITE(11,*) ((LIST(I,N),I=1,MXPL),N=1,NSTCNT)
      RETURN
C          GENERATE NEW STATE
   60 NSTCUR(I) = NSTCUR(I) + 1
      GO TO 20
      END
C
C
C
C
C
C______________________________________________________________________
      SUBROUTINE TABLE(FMT1,FMT2,FMT3,FMT4,FMT5,FMT6,
     +                 FMT7,C,A1,A2,IYR,NP,MX1)
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 /LCS/ EL(1250),POPEC(250,20),ELF(1260),
     +             MXELDC(20,4),X(1250)
C
      COMMON /HYS/ CMULT(4),EAVAL(3),EMULT(4),HYDR(3,11),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
      DIMENSION A1(250),A2(20),FMT1(18),FMT2(18),FMT3(18),
     +FMT4(18),FMT5(18),FMT6(18),FMT7(18),C(60),IYR(20)
      LMX=51
      LN=0
      K=1
      N=MX1
      TOT=0.0
      IF (MXYEAR.LE.10) N=MXYEAR
      IPAGE=1
      DO 170  I=1,NP
  170 A1(I)=0.0
      DO 190  I=1,NP
      DO 200  J=1,MXYEAR
  200 A1(I)=A1(I)+POPEC(I,J)
  190 CONTINUE
      DO 195  I=1,NP
  195 TOT=TOT+A1(I)
  240 CALL FORM(IPAGE,K,N,LN,I,FMT1,FMT2,FMT3,
     +           FMT4,FMT5,FMT6,FMT7,C,IYR,NP,MX1)
      LN=0
      M=N-K+1
      DO 250  I=1,NP
      LN=LN+1
      FMT6(5)=C(M)
      IF (N.EQ.MX1) WRITE (6,FMT6) I,(POPEC(I,J),J=K,N)
      IF (N.EQ.MXYEAR) WRITE (6,FMT6) I,(POPEC(I,J),J=K,N),A1(I)
      IF (LN.EQ.15.AND.I.NE.NP) GO TO 280
      GO TO 250
  280 IPAGE=IPAGE+1
      CALL FORM(IPAGE,K,N,LN,I,FMT1,FMT2,FMT3,
     +           FMT4,FMT5,FMT6,FMT7,C,IYR,NP,MX1)
      LN=0
  250 CONTINUE
      FMT7(6)=C(M)
      WRITE (6,FMT7) (A2(J),J=K,N),TOT
      IF (N.EQ.MXYEAR) GO TO 300
      K=MX1+1
      N=MXYEAR
      IPAGE=IPAGE+1
      GO TO 240
  300 CONTINUE
```

```
      LK=LMX-LN*2-9
      IF (I.EQ.NP) LK=LK-3
      IF (NP.EQ.MXPL) GO TO 355
      GO TO 360
  355 WRITE (6,365)
  365 FORMAT (1H //)
      WRITE (6,366)
  366 FORMAT (36H *PLANT COSTS ARE GIVEN AS THE TOTAL,
     +48H WORTH OF THE PLANT AS IT COMES ON LINE LESS THE //
     +46H SALVAGE VALUE AT THE END OF THE STUDY PERIOD.,
     +41H IF THE FIXED CHARGE RATE OPTION IS USED, //
     +39H CONSTRUCTION COSTS REPRESENT THE FIXED,
     +24H CHARGES FOR EACH PLANT.)
      LK=LK-8
  360 CONTINUE
      DO 302 IK=1,LK
      WRITE (6,304)
  304 FORMAT (1H )
  302 CONTINUE
      RETURN
      END
C
C
C______________________________________________________________________
      SUBROUTINE YALOOK(Z,ZINTEG)
C______________________________________________________________________
C
C
      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 THEU.S. DEPARTMENT
C     OF COMMERCE.
C
C     DEFINITION OF KEY VARIABLES:
C     NAME      TYPE      SIZE          MEANING
C     Z         REAL      -             NORMALIZED INDEPENDENT VARIABLE
C     ZINTEG    REAL      -             INTEGRAL VALUE
C     Y         REAL      120           NORMALIZE INDEPENDENT VARIABLE
C                                       (INPUT DATA)
C     YINT      REAL      120           INTEGRAL VALUES CORRESPONDING
C                                       TO Y (INPUT DATA)
C     K         INT.      -             GRID POINT CORRESPONDING TO Z
C     DY        REAL      -             Y INCREMENT
C
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)
      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
C
C
C     ---------------------------------------------------------------
      SUBROUTINE TRIM(ENY,ENP,ENRH,KLOW1,KLOW2,SUEN2,IBLK,NOHY,CFH,
     +IPUMP,NORDER,EPMP)
C     ---------------------------------------------------------------
C
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),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             FATOPE(20),SOL(20,2),AUPEAK(20)
C
      COMMON /LCS/ EL(1250),ELDC(4,1250),ITK(420,3),
     +             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
      DIMENSION CFH(3),ENY(2,30),ENP(2,30),ENRH(3,2),NORDER(4)
C
C
C
      IF(KLOW2.LE.0) GO TO 10
      ISWP=2
      ITRM=KLOW2
      L=KLOW1
      GO TO 15
   10 L=KLOW1-1
      ISWP=1
      ITRM=ITK(L,3)+KLOW2
   15 IP=ITK(L,1)
      ID=ITK(L,2)
      IF(IBLK.GT.0) GO TO 30
C
C
C     TRIM HYDRO BLOCK
C
C
      IH=NORDER(NOHY)
      DIFF=EAVAL(IH)-ENRH(IH,2)
      ENRH(IH,2)=EAVAL(IH)
      IF(NOHY.LE.1) GO TO 40
      ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF
      GO TO 60
C
C
C     TRIM HYDRO CLUSTER
C
C
   30 IH=NORDER(NOHY)
      NPAST=NOHY-1
      IHPT=NORDER(NPAST)
      DIFF=EAVAL(IH)+EAVAL(IHPT)-SUEN2
      ENRH(IH,2)=EAVAL(IH)
      ENRH(IHPT,2)=EAVAL(IHPT)
      IF(NPAST.LE.1) GO TO 40
      ENY(ISWP,ITRM)=ENY(ISWP,ITRM)-DIFF
      GO TO 60
C
C
C     PUMPING ENERGY ADJUSTMENT
C
C
   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
C
C
C
C___________________________________________________________________
      SUBROUTINE TUNNEL(NSTATE, NYNEXT, IFLTUN)
C
C___________________________________________________________________
C
C
C           TITLE:     TUNNEL
C           FUNCTION:  THIS SUBROUTINE DEFINES THE RANGE (CALLED
C                      TUNNEL) IN THE NUMBER OF UNITS THAT EXPANSION
C                      PLANT TYPES CAN HAVE IN STATE NYNEXT WHEN
C                       THEY ORIGINATE FROM STATE "NSTATE" IN STAGE
C                      "NY".  THE CONSTRAINTS THAT ARE CONSIDERED IN
C                      THE "TUNNEL" DEFINITION ARE BOTH REAL AND
C                      ARTIFICIAL.
C                      1.  REAL CONSTRAINTS
C                          A. MINIMUM NUMBER OF UNITS REQUIRED
C                             (MINUN)
C                          B. MAXIMUM NUMBER OF UNITS ALLOWED (MAXUN)
C                          C. MAXIMUM NUMBER OF UNITS THAT CAN BE
C                             CONSTRUCTED IN A YEAR FOR EACH EXPANSION
C                             PLANT TYPE (MAXADD)
C                      2.  ARTIFICIAL CONSTRAINTS
C                          A. ARTIFICIAL UPPER BOUNDARY (NUPB)
C                          B. ARTIFICIAL LOWER BOUNDARY (LOWB)
C
C                      IF IFLTUN>=0 THE TUNNEL FOR STATE "NSTATE"
C                      HAS ALREADY BEEN DEFINED BUT THE NUMBER OF
C                      STATES ACCEPTED IN STAGE "NYNEXT" IS SMALLER
C                      THAN THE MINIMUM NUMBER ALLOWED.  THEREFORE,
C                      ADDITIONAL STATES ARE GENERATED BY RELAXING
C                      THE ARTIFICIAL CONSTRAINTS (2A AND 2B).
C                      IN THIS CASE ONLY THE REGION NOT INCLUDED IN
C                      THE OLD TUNNELS WILL CONSTITUTE THE SUPPLE-
C                      MENTAL TUNNEL.
C
C
C
C           SUBROUTINE ARGUMENTS
C           NAME      DESCRIPTION
C           IFLTUN    FLAG -1 = ORIGINAL TUNNELS
C                           0 = NO TUNNEL ADJUSTMENTS POSSIBLE
C                           1 = LOWER BOUNDARY DECREASED
C                           2 = UPPER BOUNDARY INCREASED
C           NSTATE    UNPACKED ORIGIN STATE IN STATE NY=NYNEXT-1
C           NYNEXT    CURRENT STAGE
C
C
C           CALLED BY :  MAIN
C           CALLS     :  NONE
C
C           COMMON VARIABLES USED: IDEXP, LBOLD, MAXTUN, MINTUN,
C           MXPL, NUBOLD
C           LOCAL VARIABLES
C           NAME      DESCRIPTION
C           I         EXPANSION PLANT TYPE INDEX
C           IADD      MAXIMUM NUMBER OF UNITS ALLOWED FOR EXPANSION
C                     PLANT TYPE I DUE TO LIMIT OF UNIT ADDITION
C                     EACH YEAR
C           IP        I-TH EXPANSION PLANT TYPE ID IN THE ORIGINAL
C                     EXPANSION PLANT TYPES LIST
C
C
```

```
C
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),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             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
      DIMENSION NSTATE(20)
C
C          IF THIS IS NOT THE FIRST TIME THIS ROUTINE IS CALLED
C          FOR STATE "NSTATE", BRANCH TO SUPPLEMENTAL TUNNEL
C          DEFINITION
      IF (IFLTUN .GT. 0) GO TO 20
C
C          SET ORIGINAL TUNNEL
      DO 10 I=1,MXPL
      IP=IDEXP(I)+2
C          LOWER BOUND
      MINTUN(I) = MAXO(NSTATE(I), LOWB(NYNEXT,I), MINUN(NYNEXT,IP))
C          UPPER BOUND
      IADD = MAXADD(I) + NSTATE(I)
   10 MAXTUN(I) =MINO(IADD, NUPB(NYNEXT,I), MAXUN(NYNEXT,IP))
C
      RETURN
   20 CONTINUE
C          SET SUPPLEMENTAL TUNNEL
C
C          IF THE UPPER BOUNDARY WAS CHANGED BRANCH TO 40
      IF (IFLTUN .GT. 1) GO TO 40
C          LOWER BOUNDARY WAS CHANGED
      DO 30 I=1,MXPL
      IP=IDEXP(I)+2
C          LOWER BOUND
      MINTUN(I) = MAXO(NSTATE(I), LOWB(NYNEXT,I), MINUN(NYNEXT,IP))
C          UPPER BOUND
   30 MAXTUN(I) = LBOLD(I)
C
      RETURN
C          UPPER BOUNDARY WAS CHANGED
```

```
   40 DO 50 I=1,MXPL
      IP=IDEXP(I)+2
C           LOWER BOUND
      MINTUN(I) = NUBOLD(I)
C           UPPER BOUND
      IADD = MAXADD(I) + NSTATE(I)
   50 MAXTUN(I) = MIN0(IADD, NUPB(NYNEXT,I), MAXUN(NYNEXT,IP))
C
      RETURN
      END
C
C
C______________________________________________________________________
      SUBROUTINE UFIRST
C______________________________________________________________________
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),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             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
C
C
C
      DIMENSION NSTATE(20), NSTCUR(20), RESCAP(20,20),NUF(20)
C
      IF (ITOUT .LE. 0) GO TO 10
      RETURN
   10 CONTINUE
      OBJ = 0.0
C
C
C     ANNUAL PEAK DEMAND
C
C
      DO 130 NY=1,MXYEAR
```

```
      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
C
C
C
C
      MXPL=MAXINP+2
      IDEXP(1)=-1
      IDEXP(2)=0
      DO 20 I=3,MXPL
      I1=I-2
      IDEXP(I)=I1
   20 NEXPID(I)=I1
      MXALL = MAXPLA + MAXINP
      REWIND 23
      DO 40  IP = 1,MXALL
      READ(23,900) (PLANT(IP,J),J=1,14)
   40 CONTINUE
C
      WRITE(40,940) MAXPLA,MAXINP,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
      WRITE(40,960) EHY,EPS,(PLANT(I,9),I=I1,MXALL)
  960 FORMAT(20F3.0)
C
      WRITE(40,960) EHY,EPS,(PLANT(I,9),I=1,MAXINP)
C
      WRITE(40,970) (PLANT(I,6),I=I1,MXALL)
  970 FORMAT(10F7.2)
      WRITE(40,970) HYEXPN(9),PSEXPN(9),(PLANT(I,6),I=1,MAXINP)
      WRITE(40,980) (HYSCHD(N,2),N=1,MXYEAR)
      WRITE(40,980) (PSSCHD(N,1),N=1,MXYEAR)
      WRITE(40,980) (PLANT(I,3),I=I1,MXALL)
      WRITE(40,980) HYEXPN(2),PSEXPN(1),(PLANT(I,3),I=1,MAXINP)
  980 FORMAT(16F5.0)
  900 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)
C     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(I3,I2,I3))
```

```
C     WRITE(11,*) ((LORDER(J,K),K=1,3),J=1,100)
      READ(24,921) (MAINS(J),J=1,200)
  921 FORMAT(200I2)
C     WRITE(11,*) (MAINS(J),J=1,MAXPLA)
      CALL OPERCO(NSTCUR, NY, OPEC,IRES)
      CALL OBJFUN(NSTCUR, NY, OPEC, NSTATE, CAPCO, OBJ, OBJ1, S,
     +RESCAP, IRES)
C     WRITE(6,*) (NSTCUR(I),I=1,MXPL),NY,NSTATE,IRES
C     WRITE(6,*) OPEC,CAPCO,OBJ1,S
C
C     WRITE(11,*) (NSTCUR(I),I=1,MXPL),NY,NSTATE,IRES
C     WRITE(11,*) OPEC,CAPCO,OBJ1,S
   50 OBJ = OBJ+OBJ1
      UBOUND = OBJ*1.5
C     WRITE(6,*) UBOUND
      WRITE(11,*)UBOUND
  200 RETURN
      END
C
C
C_____________________________________________________________________
      SUBROUTINE UNPACK(NPA, NUNP)
C_____________________________________________________________________
C
C
C           TITLE:      STATE UNPACKING
C
C           FUNCTION:   UNPACKS STATE "NST" INTO "NSTCUR" USING
C                       THE PACKING BASE IPABA.
C
C           CALLED BY: MAIN
C           CALLS TO:  NONE
C
C           SUBROUTINE ARGUMENTS
C           NPA         PACKED STATE
C           NUNP      UNPACKED STATE
C
C     COMMON VARIABLES USED: IPABA, MXPL
C
C           LOCAL VARIABLES
C           NAME        DESCRIPTION
C           I           I-TH PLANT TYPE
C           II          BACKWARD PLANT TYPE INDEX
C           NSTEMP      PACKED STATE REMAINDER
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),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             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),
     +             HYEXPN(11),HYSCHD(20,8),PSEXPN(11),
     +             PSSCHD(20,8),SYSCUM(4)
C
C
C
      DIMENSION NUNP(20)
C
      NSTEMP = NPA
      DO 10 I=1,MXPL
      II = MXPL - I + 1
      NUNP(II) = NSTEMP/IPABA(II)
   10 NSTEMP = NSTEMP - NUNP(II) * IPABA(II)
      RETURN
      END
C
C-----------------------------------------------------------------------
      SUBROUTINE VALUE(Z,C3,C4,FRESGL)
C-----------------------------------------------------------------------
C
C
      COMMON /TBL/ TINT(120),TY(120)
      DOUBLE PRECISION TY,TINT
C
C
      DOUBLE PRECISION Z,ZN(8),FRE,F3,F4,F6,E,ZINTEG,ZSQ
      IF ( Z.GT.5.90D00 ) GO TO 40
      ZSQ = Z*Z/2.0D00
      E = DEXP(-ZSQ)/2.506628275D00
      ZN(1) = -Z * E
      ZN(2) = E * (Z*Z - 1.0D00)
      ZN(3) = -Z*E*(Z*Z - 3.0D00)
      ZN(4) = E*(Z*Z*(Z*Z - 6.0D00) + 3.0D00)
      ZN(5) = -Z*E*(Z*Z*(Z*Z - 10.0D00) + 15.0D00)
      F3 = 6.0D00
      F4 = 24.0D00
      F6 = 720.0D00
C
      CALL TALOOK(Z,ZINTEG)
C
      FRE = 1.0D00 - ZINTEG
      FRE = FRE + DBLE(C3)*ZN(2)/F3
     +          - DBLE(C4)*ZN(3)/F4
     +          -DBLE(C3)*DBLE(C3)*ZN(5)*10.0D00/F6
      IF(FRE.LT.0.0D00) FRE=0.0D00
      IF(FRE.GT.1.0D00) FRE=1.0D00
C
      FRESGL = SNGL(FRE)
C
      RETURN
   40 FRESGL = 0.0
      RETURN
      END
```

```
C
C-----------------------------------------------------------
C                               MAIN PROGRAM
C-----------------------------------------------------------
C
C        THIS PROGRAM PERFORMS FINANCIAL CALCULATIONS FOR
C        THE OPTIMAL OR SUBOPTIMAL CAPACITY EXPANSION
C        PLAN OBTAINED FROM THE DYNO MODULE.
C
C
C                       DESCRIPTION OF SUBROUTINES
C
C        SUBROUTINE NAME         FUNCTION
C        ---------------         --------
C
C
C        BALANC                  PERFORMS  CALCULATIONS OF ASSETS
C                                AND LIABILITIES. ALSO PERFORMS
C                                RATIO ANALYSIS.
C
C        CAPTL                   PERFORMS CAPITAL BUDGET CALCULATIONS
C                                FROM THE USER SPECIFIED CAPITAL
C                                STRUCTURE FOR EACH YEAR IN THE
C                                STUDY PERIOD.
C
C        DEPR                    CALCULATES DEPRECIATION OF PLANTS.
C
C        FBCHNG                  CHANGES/CORRECTS BEGINNING VALUES
C                                OF FINANCIAL DATA.
C
C        FUND                    CALCULATES THE USES AND SOURCES
C                                FUNDS.
C
C        FYCHNG                  CHANGES/CORRECTS YEARLY
C                                FINANCIAL DATA.
C
C        INCOM                   CALCULATES EARNINGS AND EXPENSES
C                                OF THE UTILITY INCLUDING OPERATING
C                                REVENUES,TAXES, AND OPERATING AND
C                                MAINTENANCE COSTS.
C
C        INFIN                   ACCEPTS AND STORES USER SUPPLIED
C                                EXOGENEOUS FINANCIAL PARAMETERS.
C
C        RATBAS                  CALCULATES THE RATEBASE ON WHICH
C                                THE ALLOWED OPERATING INCOME OF
C                                THE UTILITY WILL BE BASED. THE
C                                RATEBASE IS FOUND BY COMPUTING THE
C                                DOLLAR VALUE OF ALL PLANT ADDITIONS,
C                                RETIREMENTS INCLUDING THE EFFECT
C                                OF CONSTRUCTION WORK IN PROGRESS
C                                (CWIP) AND DEPRECIATION.
C
C        REPORT                  PRINTS THE BALANCE SHEET,THE
C                                INCOME STATEMENT AND THE FINANCIAL
C                                STATEMENT.
C
C
C
C
C
C
C
C
      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),DIVPR(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),OMC(20),FERN(20),
+                     ANINC(20),ERNET(20),ERNRT(20),
+                     OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
+                     CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
+                     CEPXC(20,20),CE1(20)
      COMMON /FUNDS/  SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
+                     USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
+                     ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20),
+                     TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
+                     OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
+                     PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
+                     DEBSR(20),FCE(20),LCK(20),TXCRC(20),
+                     OPCOST(20),PLGSX(20),TXDEFC(20)
      COMMON /AUX2/   DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
      COMMON /BVAL/   PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
+                     DEBSB,DEBLB,DEBSRB,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),IDEXP(20),NEXP,
+                     NBEST,HXC,SXC,ISOL(20,20)
      COMMON /PARM/   NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
      COMMON /FLAGS/  ISTOP
      INTEGER RESP,YES,YE,NEG,NO
C
C
C
      DATA YES,YE/3HYES,1HY/
      DATA NEG,NE/2HNO,1HN/
      ISTOP=0
C
C     READ DATA FROM DYNO OUTPUT FILE
C
      WRITE (6,11)
   11 FORMAT (//,27H BEGINNING  OF FINAN MODULE)
      READ (21,15) MXPLF,MXPLX,MXYR
   15 FORMAT (3I5)
      MXALL=MXPLF+MXPLX
      NF1=MXPLX+1
      DO 17 I=NF1,MXALL
```

```
      READ (21,20) (IPLW(I,NY),NY=1,MXYR)
   17 CONTINUE
   20 FORMAT (20I2)
      READ (21,22) ELHF,ELSF,(ELPL(I),I=NF1,MXALL)
      READ (21,22) ELHX,ELSX,(ELPL(I),I=1,MXPLX)
   22 FORMAT (20F3.0)
      READ (21,25) (PLCC(I),I=NF1,MXALL)
      READ (21,25) HXCC,SXCC,(PLCC(I),I=1,MXPLX)
   25 FORMAT (10F7.2)
      READ (21,27) (HFCAP(NY),NY=1,MXYR)
      READ (21,27) (SFCAP(NY),NY=1,MXYR)
      READ (21,27) (PLCAP(I),I=NF1,MXALL)
      READ (21,27) HXCAP,SXCAP,(PLCAP(I),I=1,MXPLX)
   27 FORMAT (16F5.0)
      READ (21,30) MXPL
   30 FORMAT (I2)
      READ (21,32) (IDEXP(I),I=1,MXPL)
      WRITE (6,31) MXPL,(IDEXP(I),I=1,MXPL)
   31 FORMAT (//,24H THIS ITERATION INCLUDES,I3,
     +12H UNIT TYPES./24H THEY HAVE THE FOLLOWING,
     +19H UNIT CODE NUMBERS:/8I4)
   32 FORMAT (8I2)
      READ (21,30) NBEST
      WRITE (6,33) NBEST
   33 FORMAT (//,1H ,I3,23H  OPTIMAL SOLUTIONS ARE,
     +33H AVAILABLE FROM DYNO OUTPUT FILE.)
      DO 35 NSOL=1,NBEST
      WRITE (6,37) NSOL
   37 FORMAT (//,26H COMPUTATIONS TO BEGIN FOR,
     +9H SOLUTION,I2,/20H IF YOU WANT TO STOP,
     +8H ENTER 1/2H ?)
      READ (5,38) ISTOP
   38 FORMAT (I1)
      IF (ISTOP.EQ.1) GO TO 99
      DO 40 NY=1,MXYR
      READ (21,47) (ISOL(I,NY),I=1,MXPL)
   47 FORMAT (8I4)
      READ (21,42) FUEL(NY)
      READ (21,42) OMC(NY)
      READ (21,46) (PLX(I,NY),I=1,MXPL)
   45 CONTINUE
   40 CONTINUE
   42 FORMAT (E12.5)
   46 FORMAT (5E12.5)
      HXC=HXCC*HXCAP
      SXC=SXCC*SXCAP
      DO 50 I=1,MXPL
      J=IDEXP(I)
      IF (J) 52,54,56
   52 PLC(I)=HXCC*HXCAP*0.001
      LPL(I)=ELHX
      GO TO 58
   54 PLC(I)=SXC*SXCAP*0.001
      LPL(I)=ELSX
      GO TO 58
   56 PLC(I)=PLCC(J)*PLCAP(J)*0.001
      LPL(I)=ELPL(J)
   58 IPLX(I,1)=ISOL(I,1)
      PLX(I,1)=IPLX(I,1)*PLC(I)
      DO 60 NY=2,MXYR
      IPLX(I,NY)=ISOL(I,NY)-ISOL(I,NY-1)
      PLX(I,NY)=IPLX(I,NY)*PLC(I)
   60 CONTINUE
   50 CONTINUE
      IF (NSOL.GT.1) GO TO 70
      CALL INFIN
   70 CALL RATBAS
      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
C
C
C
C-----------------------------------------------------------------
       SUBROUTINE INFIN
C-----------------------------------------------------------------
C
C
C
C      THIS SUBROUTINE ACCEPTS USER SUPPLIED VALUES OF
C      THE INPUT PROCEDURE IS ENTIRELY INTERACTIVE.
C
C
C
C
      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),DIVPR(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),OMC(20),FERN(20),
     +                 ANINC(20),ERNET(20),ERNRT(20),
     +                 OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
     +                 CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
     +                 CEPXC(20,20),CE1(20)
      COMMON /FUNDS/   SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
     +                 USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
     +                 ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20),
     +                 TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
     +                 OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
     +                 PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
     +                 DEBSR(20),FCE(20),LCR(20),TXCRC(20),
     +                 OPCOST(20),PLGSX(20),TXDEFC(20)
      COMMON /AUX2/    DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
```

```
      COMMON /BVAL/    PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
     +                 DEBSB,DEBLB,DEBSRB,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),IDEXP(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
C
      DATA YES,YE/3HYES,1HY/
      DATA NEG,NE/2HNO,1HN/
C
      NBYR=1980
      NY=1
      NYDT=28
      NBDT=17
      NYRD=NYDT*3
      NBRD=NBDT*3
      NRD=NYRD+NBRD
      NBYR=1980
      IFILE=1
      NY=1
  105 CONTINUE
C
C       READ THE KEY OF  VARIABLE DEFINITIONS
C
      DO 110 I=1,NRD
      READ (16,120) (KEYFIN(I,J),J=1,18)
  110 CONTINUE
  120 FORMAT (18A4)
C
  140 FORMAT (A3)
C
      IFILE=1
  149 CONTINUE
C     BEGINNING OF INPUT SESSION
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
      WRITE (6,215)
  215 FORMAT (//,40H THE DATA TO BE ENTERED ARE LISTED BELOW)
      WRITE (6,220)
  220 FORMAT (//,35H BEGINNING VALUES OF FINANCIAL DATA,
     +/35H -----------------------------------)
      DO 225 I=1,NBDT
      WRITE (6,120) (KEYFIN(I+17,J),J=1,18)
  225 CONTINUE
      WRITE (6,230)
  230 FORMAT (//,22H YEARLY FINANCIAL DATA,
     +/22H ----------------------)
      DO 235 I=1,NYDT
      WRITE (6,120) (KEYFIN(I+79,J),J=1,18)
  235 CONTINUE
      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,
     +/38H 2 PERCENT TO BE RETIRED EACH YEAR FOR,
     +16H EACH DEBT ISSUE)

 9000 CONTINUE
      WRITE (6,258)
  258 FORMAT (//,31H DO YOU HAVE A FILE ALREADY FOR,
     +16H 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,
     +5H YEAR,I5)
      DO 423 I=1,NYDT
      WRITE (6,434)
  434 FORMAT (1H ,//)
      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
C
      WRITE (6,475) KY
```

```
  475 FORMAT (//,30H ANY CHANGES IN FINANCIAL DATA,
     +13H FOR THE YEAR,I5,1H?/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.MXYR.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,I5,1H?/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
C
      IYCH=2
      NY=1
      WRITE (6,560)
  560 FORMAT (//,33H DO YOU WANT TO CHANGE ANY YEARLY,
     +16H 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 (I4)
      NY=KY-NBYR
      GO TO 490
  590 CONTINUE
      IF (IFILE.EQ.2) GO TO 1200
C
C      ENTER DEBT RETIREMENT SCHEDULES
C
      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,I5/5H ----)
      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 (6,445)
    READ (5,470) FDEBR(NY,K,IDEBT)
635 CONTINUE
    IF (IDCH.EQ.2) GO TO 710
    IF (NY.EQ.MXYR) GO TO 700
    NY=NY+1
    GO TO 610
700 CONTINUE
    IF (IDEBT.EQ.2) GO TO 710
    IDEBT=2
    GO TO 600
710 CONTINUE
    WRITE (6,720)
720 FORMAT (//,33H DO YOU WANT TO CHANGE ANY OF THE,
   +27H 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,
   +/38H SHORT TERM DEBT RETIREMENT SCHEDULES.,
   +/30H ENTER 2 IF YOU WANT TO CHANGE,
   +/37H LONG TERM DEBT RETIREMENT SCHEDULES./2H ?)
    READ (5,726) IDEBT
726 FORMAT(I1)
    WRITE (6,730)
730 FORMAT(//,32H ENTER YES IF YOU WANT TO CHANGE,
   +/33H ALL THE RETIREMENT SCHEDULES FOR)
    IF (IDEBT.EQ.1) WRITE (6,608)
    IF (IDEBT.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
C   ENTER CONSTRUCTION EXPENDITURE FOR EXPANSION PLANTS
C
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(I),JY
845 FORMAT (//,32H THE TOTAL CONSTRUCTION COST FOR,
   +6H PLANT,I3,3H IS,F8.3,17H MILLION DOLLARS.,
   +/26H THE CONSTRUCTION TIME IS ,I2,7H YEARS.,
   +/53H ENTER CONSTRUCTION EXPENDITURE (MILLIONS OF DOLLARS),
   +/34H DURING EACH YEAR OF CONSTRUCTION.)
    DO 850 IY=1,JY
```

```
      WRITE (6,848) IY
  848 FORMAT (//,25H CONSTRUCTION EXPENDITURE,
     +31H (MILLIONS OF DOLLARS) FOR YEAR,I3,1H?,
     +/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 ENTERRED FOR UNIT,
     +5H CODE,I4,2H ?/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 (/,28H 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 (5,987)
  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
C
C     READ DATA FROM EXISTING DATA FILE
C
      IFILE=2
      READ (18,1225) (BDAT(I),I=1,NBDT)
      READ (18,1215) MXYR
      DO 1120 NY=1,MXYR
```

```
      READ (18,1225) (FINDAT(NY,I),I=1,NYDT)
      READ (18,1225) ((FDEBR(NY,K,IDEBT),IDEBT=1,2),K=1,MXYR)
 1120 CONTINUE
 1215 FORMAT (I2)
 1225 FORMAT (5E12.5)
      REWIND 18
C
C     CHANGES/CORRECTIONS OF DATA ALREADY
C     ON FILE.
C
 1200 CONTINUE
      WRITE (6,1250)
 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
      IFILE=2
      REWIND 18
 1260 CONTINUE
 1255 CONTINUE
      GO TO 1450
 1400 CONTINUE
      WRITE (6,1430)
 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 REINPUT ALL OF THE ABOVE DATA./2H ?)
      READ (5,726) ICHNG
      GO TO (358,570,722,950,149),ICHNG
 1450 CONTINUE
      WRITE (6,1480)
 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
C     TRANSFER THE INPUT DATA INTO APPROPRIATE
C     ARRAYS OF FINANCIAL VARIABLES.
C
 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(20)
      AFDC2B=BDAT(17)
      DO 1490 NY=1,MXYR
```

```
      PLGSF(NY)=FINDAT(NY,1)
      DEPF1(NY)=FINDAT(NY,2)
      DEPF2(NY)=FINDAT(NY,3)
      CEF(NY)=FINDAT(NY,4)
      FDEBS(NY)=FINDAT(NY,5)*0.01
      DSLIM(NY)=FINDAT(NY,6)
      FDEBL(NY)=FINDAT(NY,7)*0.01
      DLLIM(NY)=FINDAT(NY,8)
      FSTCOM(NY)=FINDAT(NY,9)*0.01
      COMLIM(NY)=FINDAT(NY,10)
      FSTPR(NY)=FINDAT(NY,11)*0.01
      PRLIM(NY)=FINDAT(NY,12)
      RDEBS(NY)=FINDAT(NY,13)*0.01
      RDEBL(NY)=FINDAT(NY,14)*0.01
      RDIVP(NY)=FINDAT(NY,15)*0.01
      FDIVC(NY)=FINDAT(NY,16)*0.01
      RTXFD(NY)=FINDAT(NY,17)*0.01
      RTXST(NY)=FINDAT(NY,18)*0.01
      RTXREV(NY)=FINDAT(NY,19)*0.01
      RTXP(NY)=FINDAT(NY,20)*0.01
      RTXCR(NY)=FINDAT(NY,21)*0.01
      FCE(NY)=FINDAT(NY,22)*0.01
      FCWIP(NY)=FINDAT(NY,23)*0.01
      AFDCR(NY)=FINDAT(NY,24)*0.01
      ARATE(NY)=FINDAT(NY,25)*0.01
      AFDCF2(NY)=FINDAT(NY,26)
      RWKCAP(NY)=FINDAT(NY,27)*0.01
      RFDEBL(NY)=FINDAT(NY,28)
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
C     END OF INPUT PROCEDURE
C
 1900 CONTINUE
      WRITE (6,1930)
 1930 FORMAT (//,34H END OF INPUT PROCEDURE.INPUT DATA,
     +19H 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
      WRITE (6,2600)
 2600 FORMAT (//,35H ENTER 1 IF AFUDC SHOULD BE TREATED,
     +10H AS INCOME/18H ENTER 0 OTHERWISE/2H ?)
      READ (5,726) IAFC
      IF (IDEF.EQ.0) TXDEFB=0
C
 4000 CONTINUE
      RETURN
      END
C
C
C-----------------------------------------------------------
      SUBROUTINE FYCHNG(NY,NYDT)
C-----------------------------------------------------------
C
C
```

```
C
C     THIS SUBPROGRAM IS USED TO CHANGE OR CORRECT
C     FINANCIAL PARAMETERS ENTERED DURING THE CURRENT
C     SESSION OR ALREADY EXISTING ON A DATA FILE.
C
      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),DIVPR(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),OMC(20),FERN(20),
     +                ANINC(20),ERNET(20),ERNRT(20),
     +                OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
     +                CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
     +                CEPXC(20,20),CE1(20)
      COMMON /FUNDS/  SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
     +                USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
     +                ASSET(20),EQCUM(20),ERNCUM(20),CAPTOT(20),
     +                TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
     +                OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
     +                PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
     +                DEBSR(20),FCE(20),LCR(20),TXCRC(20),
     +                OPCOST(20),PLGSX(20),TXDEFC(20)
      COMMON /AUX2/   DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
      COMMON /BVAL/   PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
     +                DEBSB,DEBLB,DEBSRB,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),IDEXP(20),NEXP,
     +                NBEST,HXC,SXC,ISOL(20,20)
      COMMON /PARM/   NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
      COMMON /FLAGS/  ISTOP
      INTEGER RESP,YES,YE,NEG,NO
C
C
C
      DATA YES,YE/3HYES,1HY/
      DATA NEG,NE/2HNO,1HN/
C
C
      KY=NBYR+NY
```

```
  105 FORMAT (A3)
  106 CONTINUE
      WRITE (6,110)
  110 FORMAT (//,31H WHICH DATA YOU WANT TO CHANGE?,
     +/43H ENTER 2 DIGIT DATA CODE.FOR HELP,ENTER 99.,
     +/3H --)
      READ (5,120) IP
  120 FORMAT (I2)
      IF (IP.EQ.99) GO TO 160
C
C     ENTER CHANGED/CORRECTED PARAMETER
C
  125 CONTINUE
      WRITE (6,135) (KEYFIN(IP+107,J),J=1,18)
  135 FORMAT (18A4)
      WRITE (6,142) FINDAT(NY,IP)
  142 FORMAT (16H PREVIOUS VALUE=,F10.4)
      WRITE (6,145)
  145 FORMAT (11H -----.-----)
      READ (5,155) FINDAT(NY,IP)
  155 FORMAT (F10.5)
      WRITE (6,159) KY
  159 FORMAT (//,30H ANY MORE CHANGES IN FINANCIAL,
     +9H DATA FOR,I5,
     +/17H ENTER YES OR NO./2H ?)
      READ (5,105) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 106
      GO TO 190
  160 CONTINUE
C
C     PRINT 2 DIGIT  DATA CODES AND
C     DEFINITONS.
C
      DO 180 IP=1,NYDT
      WRITE (6,135) (KEYFIN(IP+79,J),J=1,18)
  180 CONTINUE
      GO TO 106
  190 CONTINUE
C
C
      RETURN
      END
C
C
C---------------------------------------------------------
      SUBROUTINE INCOM(NY)
C---------------------------------------------------------
C
C
C     THIS SUBPROGRAM  PERFORMS ALL CALCULATIONS OF
C     EARNINGS AND EXPENSES. THESE INCLUDE OPERATING
C     REVENUES AND COSTS,FEDRAL,STATE AND OTHER TAXES,
C     AND INTEREST CHARGES ON DEBT.
C
C
      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),FSTCUM(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),
     +                AINC(20),BINC(20),OMC(20),FERN(20),
     +                ANINC(20),ERNET(20),ERNRT(20),
     +                OMCF(20),OMCX(20)
      COMMON /INPUTS/ FINDAT(20,30),BDAT(20),KEYFIN(140,18)
      COMMON /PLANTS/ PLGSI(20),PLGAI(20),PLNSI(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),
+                   CW1PC(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),RATB(20),CE(20),AFDCX2(20),
+                   CEPXC(20,20),CE1(20)
 COMMON /FUNDS/     SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
+                   USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
+                   ASSET(20),EQCOM(20),ERNCUM(20),CAPTOT(20),
+                   TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
+                   OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
+                   PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
+                   DEBSR(20),FCE(20),LCR(20),TXCRC(20),
+                   OPCOST(20),PLGSX(20),TXDEFC(20)
 COMMON /AUX2/      DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
 COMMON /BVAL/      PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
+                   DEBSB,DEBLB,DEBSRB,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),IDEXP(20),NEXP,
+                   NBEST,HXC,SXC,ISOL(20,20)
 COMMON /PARM/      NBYR,MXYK,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
 COMMON /FLAGS/     ISTOP
 INTEGER RESP,YES,YE,NEG,NO


 DATA YES,YE/3HYES,1HY/
 DATA NEG,NE/2HNO,1HN/

 PRFAC=0.5

 DIVPR(NY)=0
 RFD=0
 FACT1=0
 FACT2=0
 NY1=NY-1
 IF (NY.GT.1) GO TO 102
 OPINC(1)=RATB(1)*ARATE(1)
 CDEBS(1)=DEBS(1)*RDEBS(1)+CDEBSB
 CDEBL(1)=DEBL(1)*RDEBL(1)+CDEBLB
 CDEBT(1)=CDEBS(1)+CDEBL(1)
 STCOMC(1)=STCOM(1)+STCOMB
 DIVPR(1)=STPR(1)*RDIVP(1)
 TXDEFC(1)=TXDEFB+TXDEF(1)*IDEF
 STPRC(1)=STPR(1)+STPRB
 GO TO 115
102 CONTINUE
 OPINC(NY)=RATB(NY)*ARATE(NY)
 STCOMC(NY)=STCOMC(NY-1)+STCOM(NY)
```

```
      STPRC(NY)=STPRC(NY-1)+STPR(NY)
      TXDEFC(NY)=TXDEFC(NY-1)+TXDEF(NY)
      DO 110 M=1,NY
      DIVPR(NY)=DIVPR(NY)+STPR(M)*RDIVP(M)
      RFD=RFD+RFDEBL(M)
      FACT1=0
      FACT2=0
      IF (M.EQ.NY) GO TO 122
      DO 120 K=M,NY1
      FACT1=FACT1+FDEBSR(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)


  115 CONTINUE
      BINT(NY)=OPINC(NY)+AFDC1(NY)*IAFC
      ERNET(NY)=BINT(NY)-CDEBT(NY)
      ERNCOM(NY)=ERNET(NY)-DIVPR(NY)
      IF (ERNCOM(NY).LT.0.) ERNCOM(NY)=0.0
      DIVCOM(NY)=ERNCOM(NY)*FDIVC(NY)
      ERNRT(NY)=ERNCOM(NY)*(1.-FDIVC(NY))
      ERNRTC(1)=ERNRT(1)+ERNRTB
      IF (NY.GT.1) ERNRTC(NY)=ERNRTC(NY-1)+ERNRT(NY)
      TXSPD(NY)=(ERNET(NY)-AFDC1(NY)*IAFC)*(RTXST(NY)+RTXFD(NY))/
     1          (1.-RTXST(NY)-RTXFD(NY))-(TXCR(NY))/
     2          (1.-RTXST(NY)-RTXFD(NY))-(DEP2(NY)-DEP1(NY))*
     3         (RTXFD(NY)+RTXST(NY))/(1.-RTXFD(NY)-RTXST(NY))
       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+TXSPD(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)=(PLGS1(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)+OPINC(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(20,20,2)
      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),OPINC(20),TXINC(20),ARATE(20),
     +                AINC(20),BINC(20),OMC(20),FERN(20),
     +                ANINC(20),ERNET(20),ERNRT(20),
     +                OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
     +                CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
     +                CEPXC(20,20),CE1(20)
      COMMON /FUNDS/  SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
     +                USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
     +                ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20),
     +                TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
     +                OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
     +                PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
     +                DEBSR(20),FCE(20),LCR(20),TXCRC(20),
     +                OPCOST(20),PLGSX(20),TXDEFC(20)
      COMMON /AUX2/   DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
      COMMON /BVAL/   PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
     +                DEBSB,DEBLB,DEBSRB,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),IDEXP(20),NEXP,
     +                NBEST,HXC,SXC,ISOL(20,20)
      COMMON /PARM/   NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
      COMMON /FLAGS/  ISTOP
      INTEGER RESP,YES,YE,NEG,NO


      DATA YES,YE/3HYES,1HY/
      DATA NEG,NE/2HNO,1HN/

      INITIALIZE VARIABLES

      DO 110 NY=1,MXYR
      PLGSX(NY)=0
      CEX(NY)=0
      CEC(NY)=0
      OPCOST(NY)=OMC(NY)+FUEL(NY)
  110 CONTINUE

      COMPUTE GROSS PLANT VALUE OF EXPANSION PLANTS

      DO 125 NY=1,MXYR
      DO 130 I=1,MXPL
```

```
      PLGSX(NY)=PLGSX(NY)+PLC(I)*ISOL(I,NY)
  130 CONTINUE
  125 CONTINUE

       CALCULATE GROSS PLANT VALUE OF ALL PLANTS

      DO 135 NY=1,MXYR
  135 PLGS1(NY)=PLGSF(NY)+PLGSX(NY)

      CALCULATE GROSS PLANT ADDITIONS

      PLGA1(1)=PLGS1(1)-PLGSB
      DO 140 NY=2,MXYR
  140 PLGA1(NY)=PLGS1(NY)-PLGS1(NY-1)

      CALCULATE DEPRECIATION FOR BOOK PURPOSES

  255 FORMAT (//,34H ENTER ACCORDING TO THE FOLLOWING./,
     +23H 2: SUM OF YEARS DIGITS/
     +20H 3: DOUBLE DECLINING)
  257 FORMAT (I1)
      CALL DEPR(1)
      DO 260 I=1,MXPL
      DO 270 NY=1,MXYR
      DEPLX1(I,NY)=DEPLX(I,NY)
  270 CONTINUE
  260 CONTINUE

      CALCULATE DEPRECIATION FOR TAX PURPOSES

      WRITE (6,275)
  275 FORMAT (//,32H ACCELERATED DEPRECIATION METHOD,
     +18H FOR TAX PURPOSES?)
      WRITE (6,255)
      READ (5,257) IDEP
      CALL DEPR(IDEP)
      DO 265 I=1,MXPL
      DO 277 NY=1,MXYR
      DEPLX2(I,NY)=DEPLX(I,NY)
  277 CONTINUE
  265 CONTINUE

      CALCULATE TOTAL DEPRECIATION FOR
      BOTH BOOK AND TAX PURPOSES

      DO 290 NY=1,MXYR
      DEP1(NY)=DEPF1(NY)
      DEP2(NY)=DEPF2(NY)
      DO 300 I=1,MXPL
      DEP1(NY)=DEP1(NY)+DEPLX1(I,NY)
      DEP2(NY)=DEP2(NY)+DEPLX2(I,NY)
  300 CONTINUE
  290 CONTINUE

      CALCULATE ACCUMULATED DEPRECIATION

      DEP1C(1)=DEP1B+DEP1(1)
      DEP2C(1)=DEP2B+DEP2(1)
      DO 315 NY=2,MXYR
      DEP1C(NY)=DEP1C(NY-1)+DEP1(NY)
      DEP2C(NY)=DEP2C(NY-1)+DEP2(NY)
  315 CONTINUE


      CALCULATE NET PLANT IN SERVICE
      AND NET PLANT ADDED

      PLNS1(1)=PLGS1(1)-DEP1C(1)
```

```
      PLNA1(1)=PLNS1(1)-(PLGSB-DEP1B)
      PLNS2(1)=PLGS1(1)-DEP2C(1)
      PLNA2(1)=PLNS1(1)-(PLGSB-DEP2B)
      DO 360 NY=2,MXYR
      PLNS1(NY)=PLGS1(NY)-DEP1C(NY)
      PLNA1(NY)=PLNS1(NY)-PLNS1(NY-1)
      PLNS2(NY)=PLGS1(NY)-DEP2C(NY)
      PLNA2(NY)=PLNS2(NY)-PLNS2(NY-1)
  360 CONTINUE

      FIND CWIP FOR EXPANSION UNITS

      DO 370 I=1,MXPL
      DO 375 M=1,MXYR
      L=ITPL(I)-M+1
      IF (L.LT.1) GO TO 377
      DO 376 K=1,L
  376 CWIPXB=CWIPXB+CEPLX(I,K)*IPLX(I,M)
  377 CONTINUE
      DO 380 NY=1,MXYR
      J=ITPL(I)-M+1+NY
      IF (J.LT.1.OR.J.GT.ITPL(I)) GO TO 380
      CEPX(I,NY)=CEPX(I,NY)+CEPLX(I,J)*IPLX(I,M)
  380 CONTINUE
  375 CONTINUE
  370 CONTINUE

      DO 385 I=1,MXPL
      NMAX=ITPL(I)
      DO 390 J=1,N
  390 CEPXC(I,J)=0
  385 CONTINUE
      DO 410  I=1,MXPL
      DO 420  J=1,NMAX
  420 CEPXC(I,J)=CEPXC(I,J)+CEPLX(I,J)
  410 CONTINUE
      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(1)*(1.-FCWIP(1))
      IF (NY.LE.0) AFDC1B=AFDC1B+CEP*AFCR

      IF (NY.GT.0) AFCR=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,MXPLX
      CEX(NY)=CEX(NY)+CEPX(I,NY)
  500 CONTINUE
  490 CONTINUE

      ADD CE FOR  SCHEDULED AND EXPANSION
      PLANTS TO FIND TOTAL CE

      DO 510 NY=1,MXYR
  510 CE(NY)=CEF(NY)+CEX(NY)

      CALCULATE ACCUMULATED CONSTRUCTION
      EXPENDITURES

      CWIPB=CWIPFB+CWIPXB
      CEB=PLGSB+CWIPB
```

```
      CEC(1)=CEB+CE(1)
      DO  520 NY=2,MXYR
  520 CEC(NY)=CEC(NY-1)+CE(NY)

       CALCULATE ACCUMULATED AND
       INCREMENTAL CWIP

      DO 530 NY=1,MXYR
      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,MXYR
      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,MXYR
      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,MXYR
      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,MXYR
      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)+RTXFD(NY))*IDEF
  600 CONTINUE

      RETURN
      END

---------------------------------------------------
      SUBROUTINE DEPR(IDEP)
---------------------------------------------------

      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),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),RFDEBL(20)
      COMMON /INCOME/  GREV(20),OPINC(20),TXINC(20),ARATE(20),
     +                 AINC(20),BINC(20),OMC(20),FERN(20),
```

```
+                   ANINC(20),ERNET(20),ERNRT(20),
+                   OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
+                   CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
+                   CEPXC(20,20),CE1(20)
 COMMON /FUNDS/     SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
+                   USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
+                   ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20),
+                   TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
+                   OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
+                   PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
+                   DEBSR(20),FCE(20),LCR(20),TXCRC(20),
+                   OPCOST(20),PLGSX(20),TXDEFC(20)
 COMMON /AUX2/      DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
 COMMON /BVAL/      PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
+                   DEBSB,DEBLB,DEBSRB,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),PLLAP(200),IDEXP(20),NEXP,
+                   NBEST,HXC,SXC,ISOL(20,20)
 COMMON /PARM/      NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
 COMMON /FLAGS/     ISTOP
 DIMENSION LSUM(200),BPL(200,20)
 INTEGER RESP,YES,YE,NEG,NO


 DATA YES,YE/3HYES,1HY/
 DATA NEG,NE/2HNO,1HN/


 MXPL1=MXPLX+1
 DO 80 NY=1,MXYR
 DO 85 I=1,MXPL1
 DEPLX(I,NY)=0
85 CONTINUE
80 CONTINUE
 GO TO (90,135,185),IDEP

90 CONTINUE
 CALCULATE DEPRECIATION BY USING THE
 LINEAR METHOD

 DO 110 I=1,MXPL
 FDEP(I)=1./FLOAT(LPL(I))
```

```
      DO 120 NY=1,MXYR
      DO 125 M=1,NY
      DEPLX(I,NY)=DEPLX(I,NY)+PLX(I,M)*FDEP(I)
  125 CONTINUE
  120 CONTINUE
  110 CONTINUE
      GO TO 999

      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,1)=PLX(I,1)*FDEP(I)
      BPL(1,1)=PLX(I,1)-DEPLX(I,1)
      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 /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),DIVPR(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),OMC(20),FERN(20),
+                 ANINC(20),ERNET(20),ERNRT(20),
+                 OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
+                 CAPKQ(20),RATB(20),CE(20),AFDCX2(20),
+                 CEPXC(20,20),CE1(20)
 COMMON /FUNDS/   SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
+                 USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
+                 ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20),
+                 TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
+                 OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
+                 PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
+                 DEBSR(20),FCE(20),LCR(20),TXCRC(20),
+                 OPCOST(20),PLGSX(20),TXDEFC(20)
 COMMON /AUX2/    DSLIM(20),DLLIM(20),COMLIM(20),
+                 PRLIM(20),FCWIP(20),BINT(20),CWIPC1(20),
+                 TXSPD(20),FUEL(20),IIPL(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)
 COMMON /BVAL/    PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
+                 DEBSB,DEBLB,DEBSRB,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),IDEXP(20),NEXP,
+                 NBEST,HXC,SXC,ISOL(20,20)
 COMMON /PARM/    NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
 COMMON /FLAGS/   ISTOP
 INTEGER RESP,YES,YE,NEG,NO


 DATA YES,YE/3HYES,1HY/
 DATA NEG,NE/2HNO,1HN/


 FIND CAPITAL REQUIREMENTS AND CAPITAL
 BUDGET FOR THE BASE YEAR

 A1=STCOMB+ERNRTB+STPRB+DEBLB+DEBSB+TXDEFB+TXCRB
 A2=PLGSB-DEP1B+CWIPB+AFDC1B+AFDC2B
 WKCAPB=A1-A2
```

```
      NY=1
      KY=NBYR+NY
      IY=1
      DEBSR(1)=DEBSRB
      DEBLR(1)=RFDEBL(1)
      DEBR(1)=DEBLR(1)+DEBSR(1)
      WKCAP1(1)=GREVB*RWKCAP(1)
      CAPRQ(1)=CE(1)+AFDC1(1)-
     +         DEP1(1)-TXDEF(1)
      IF (CAPRQ(1).LT.0.) CAPRQ(1)=0.0
      DEBS(1)=CAPRQ(1)*FDEBS(1)+DEBSR(1)
   80 IF (DEBS(1).GT.DSLIM(1)) GO TO 201
      DEBL(1)=CAPRQ(1)*FDEBL(1)+DEBLR(1)
   85 IF (DEBL(1).GT.DLLIM(1)) GO TO 211
      DEBN(1)=DEBS(1)+DEBL(1)
      STCOM(1)=CAPRQ(1)*FSTCOM(1)
   90 IF (STCOM(1).GT.COMLIM(1)) GO TO 221
      STPR(1)=CAPRQ(1)*FSTPR(1)
   95 IF (STPR(1).GT.PRLIM(1)) GO TO 231
      DEBST(1)=DEBSB-DEBSR(1)+DEBS(1)
      DEBLT(1)=DEBLB-DEBLR(1)+DEBL(1)
      DEBT(1)=DEBST(1)+DEBLT(1)
      CALL INCOM(NY)
      CALL FUND(NY)
      WKCAP(1)=WKCAPB+CAPIN(1)
      CALL BALANC(NY)
      IF (ISTOP.EQ.1) GO TO 400

      CALCULATE CAPITAL BUDGET FOR OTHER
      YEARS IN THE STUDY PERIOD

  110 CONTINUE
      NY=NY+1
      KY=NY+NBYR
      NY1=NY-1
      RFD1=0
      RFD2=0
      DO 120 M=1,NY1
      RFD1=RFD1+DEBS(M)*FDEBSR(M,NY)
  120 RFD2=RFD2+DEBL(M)*FDEBLR(M,NY)
      DEBSR(NY)=RFD1
      DEBLR(NY)=RFD2+RFDEBL(NY)
      DEBR(NY)=DEBSR(NY)+DEBLR(NY)
      WKCAP1(NY)=GREV(NY-1)*RWKCAP(NY)
      CAPRQ(NY)=CE(NY)+AFDC1(NY)-
     +         DEP1(NY)-TXDEF(NY)+
     +           (WKCAP1(NY)-WKCAP(NY-1))
      IF (CAPRQ(NY).LT.0.) CAPRQ(NY)=0.0
      DEBS(NY)=CAPRQ(NY)*FDEBS(NY)+DEBSR(NY)
  130 IF (DEBS(NY).GT.DSLIM(NY)) GO TO 201
      DEBL(NY)=CAPRQ(NY)*FDEBL(NY)+DEBLR(NY)
  135 IF (DEBL(NY).GT.DLLIM(NY)) GO TO 211
      DEBN(NY)=DEBS(NY)+DEBL(NY)
      DEBST(NY)=DEBST(NY-1)+DEBS(NY)-DEBSR(NY)
      DEBLT(NY)=DEBLT(NY-1)+DEBL(NY)-DEBLR(NY)
      DEBT(NY)=DEBST(NY)+DEBLT(NY)
      STCOM(NY)=CAPRQ(NY)*FSTCOM(NY)
  140 IF (STCOM(NY).GT.COMLIM(NY)) GO TO 221
      STPR(NY)=CAPRQ(NY)*FSTPR(NY)
  145 IF (STPR(NY).GT.PRLIM(NY)) GO TO 231
      CALL INCOM(NY)
      CALL FUND(NY)
      WKCAP(NY)=WKCAP(NY-1)+CAPIN(NY)
      CALL BALANC(NY)
      IF (ISTOP.EQ.1) GO TO 400
      IF (NY.EQ.MAXYR) GO TO 400
      GO TO 110
```

```
      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 1S,F10.2,17H MILLION DOLLARS./14H EXCEEDS UPPER,
     +9H LIMIT OF,F10.2,17H 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),PRLIM(NY)
      READ (5,300) PRLIM(NY)
      IF (PRLIM(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 SUBROGRAM 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),DIVPR(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),OMC(20),FERN(20),
     +                ANINC(20),ERNET(20),ERNRT(20),
     +                OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
     +                  CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
     +                  CEPXC(20,20),CE1(20)
      COMMON /FUNDS/    SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
     +                  USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
     +                  ASSET(20),EQCUM(20),ERNCOM(20),CAPTOT(20),
     +                  TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
     +                  OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
     +                  PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
     +                  DEBSR(20),FCE(20),LCR(20),TXCRC(20),
     +                  OPCOST(20),PLGSX(20),TXDEFC(20)
      COMMON /AUX2/     DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
      COMMON /BVAL/     PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
     +                  DEBSB,DEBLB,DEBSRB,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),IDEXP(20),NEXP,
     +                  NBEST,HXC,SXC,ISOL(20,20)
      COMMON /PARM/     NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
      COMMON /FLAGS/    ISTOP
      DIMENSION NYR(20),HYPH(20)
      INTEGER RESP,YES,YE,NEG,NO
C
C
C
      DATA YES,YE/3HYES,1HY/
      DATA NEG,NE/2HNO,1HN/
      DATA DASH/4H----/
      MAX=8
      N1=1
      DO 110 NY=1,MXYR
      HYPH(NY)=DASH
  110 NYR(NY)=NBYR+NY
C
C
C     PRINT THE BALANCE SHEET
C
C
  112 CONTINUE
      WRITE (6,99)
      WRITE (6,115)
  115 FORMAT (//,40X,21H XYZ ELECTRIC COMPANY)
      WRITE (6,125)
  125 FORMAT (40X,14H BALANCE SHEET)
      WRITE (6,145)
  145 FORMAT (//,40X,29H MILLIONS OF CONSTANT DOLLARS)
      WRITE (6,155) (NYR(I),I=N1,MAX)
```

```
155 FORMAT (//,9X,12H DECEMBER 31,14X,8I10)
    WRITE (6,165) (HYPH(I),I=N1,MAX)
165 FORMAT (35X,8(6X,A4))
    WRITE (6,170)
170 FORMAT(//,7H ASSETS)
    WRITE (6,178) (PLGS2(NY),NY=N1,MAX)
178 FORMAT (//,26H GR PLANT INS ( INCL AFUDC),9X,8(1X,F9.2))
    WRITE (6,180) (DEP1C(NY),NY=N1,MAX)
180 FORMAT (21H - ACCUM DEPRECIATION,14X,8(1X,F9.2))
    WRITE (6,182) (PLNV(NY),NY=N1,MAX)
182 FORMAT (16H NET PLANT VALUE,19X,8(1X,F9.2))
    WRITE (6,185) (CWIPC1(NY),NY=N1,MAX)
185 FORMAT (6H +CWIP,29X,8(1X,F9.2))
    WRITE (6,195) (PLNUT(NY),NY=N1,MAX)
195 FORMAT (18H NET UTILITY PLANT,17X,8(1X,F9.2))
    WRITE (6,190) (WKCAP(NY),NY=N1,MAX)
190 FORMAT (20H NET WORKING CAPITAL,15X,8(1X,F9.2))
    WRITE (6,200) (ASSET(NY),NY=N1,MAX)
200 FORMAT (//,13H TOTAL ASSETS,22X,8(1X,F9.2))
    WRITE (6,210)
210 FORMAT (///,12H LIABILITIES)
    WRITE (6,220) (STCOMC(NY),NY=N1,MAX)
220 FORMAT (//,13H COMMON STOCK,22X,8(1X,F9.2))
    WRITE (6,230) (ERNRTC(NY),NY=N1,MAX)
230 FORMAT (20H + RETAINED EARNINGS,15X,8(1X,F9.2))
    WRITE (6,240) (EQCOM(NY),NY=N1,MAX)
240 FORMAT (20H TOTAL COMMON EQUITY,15X,8(1X,F9.2))
    WRITE (6,250) (STPRC(NY),NY=N1,MAX)
250 FORMAT (18H + PREFERRED STOCK,17X,8(1X,F9.2))
    WRITE (6,260) (DEBLT(NY),NY=N1,MAX)
260 FORMAT (17H + LONG TERM DEBT,18X,8(1X,F9.2))
    WRITE (6,270) (CAPTOT(NY),NY=N1,MAX)
270 FORMAT (14H TOTAL CAPITAL,21X,8(1X,F9.2))
    WRITE (6,280) (DEBST(NY),NY=N1,MAX)
280 FORMAT (18H + SHORT TERM DEBT,17X,8(1X,F9.2))
    WRITE (6,290) (TXDEFC(NY),NY=N1,MAX)
290 FORMAT (24H + DEFERRED INCOME TAXES,11X,8(1X,F9.2))
    WRITE (6,295) (TXCRC(NY),NY=N1,MAX)
295 FORMAT (26H + DEFERRED INV TAX CREDIT,9X,8(1X,F9.2))
    WRITE (6,300) (TLIAB(NY),NY=N1,MAX)
300 FORMAT (//,18H TOTAL LIABILITIES,17X,8(1X,F9.2))
    IF (MAX.EQ.MXYR) GO TO 302
    N1=MAX+1
    MAX=MXYR
    GO TO 112
302 N1=1
    MAX=8
305 CONTINUE

    PRINT THE INCOME STATEMENT

    WRITE (6,99)
 99 FORMAT (1H1)
    WRITE (6,115)
    WRITE (6,310)
310 FORMAT (40X,17H INCOME STATEMENT)
    WRITE (6,145)
    WRITE (6,155) (NYR(I),I=N1,MAX)
    WRITE (6,165) (HYPH(I),I=N1,MAX)
    WRITE (6,320) (GREV(NY),NY=N1,MAX)
320 FORMAT (//,19H OPERATING REVENUES,16X,8(1X,F9.2))
    WRITE (6,330) (OMC(NY),NY=N1,MAX)
330 FORMAT (23H - OPER AND MAINT COSTS,12X,8(1X,F9.2))
    WRITE (6,335) (FUEL(NY),NY=N1,MAX)
335 FORMAT (13H - FUEL COSTS,22X,8(1X,F9.2))
    WRITE (6,350) (OPEXP(NY),NY=N1,MAX)
```

```
  350 FORMAT (//,25H TOTAL OPERATING EXPENSES,10X,8(1X,F9.2))
      WRITE (6,340) (DEP1(NY),NY=N1,MAX)
  340 FORMAT (//,15H - DEPRECIATION,20X,8(1X,F9.2))
      WRITE (6,360) (TXREV(NY),NY=N1,MAX)
  360 FORMAT (15H - OPER REV TAX,20X,8(1X,F9.2))
      WRITE (6,363) (TXP(NY),NY=N1,MAX)
  363 FORMAT (15H - PROPERTY TAX,20X,8(1X,F9.2))
      WRITE (6,370) (TXSPD(NY),NY=N1,MAX)
  370 FORMAT (18H - INCOME TAX PAID,17X,8(1X,F9.2))
      WRITE (6,373) (TXDEF(NY),NY=N1,MAX)
  373 FORMAT (22H - DEFFERED INCOME TAX,13X,8(1X,F9.2))
C     WRITE (6,377) (TXCR(NY),NY=N1,MAX)
  377 FORMAT (22H - DEFFERED INV TAX CR,13X,8(1X,F9.2))
      WRITE (6,380) (TOTEXP(NY),NY=N1,MAX)
  380 FORMAT (//,15H TOTAL EXPENSES,20X,8(1X,F9.2))
      WRITE (6,383) (OPINC(NY),NY=N1,MAX)
  383 FORMAT (//,17H OPERATING INCOME,18X,8(1X,F9.2))
      IF (IAFC.EQ.1) WRITE (6,387) (AFDC1(NY),NY=N1,MAX)
  387 FORMAT (8H + AFUDC,27X,8(1X,F9.2))
      WRITE (6,390) (BINT(NY),NY=N1,MAX)
  390 FORMAT (//,23H INCOME BEFORE INTEREST,12X,8(1X,F9.2))
      WRITE (6,400) (CDEBS(NY),NY=N1,MAX)
  400 FORMAT (28H - INTRST ON SHORT TERM DEBT,7X,8(1X,F9.2))
      WRITE (6,410) (CDEBL(NY),NY=N1,MAX)
  410 FORMAT (27H - INTRST ON LONG TERM DEBT,8X,8(1X,F9.2))
      WRITE (6,420) (ERNET(NY),NY=N1,MAX)
  420 FORMAT (//,11H NET INCOME,24X,8(1X,F9.2))
      WRITE (6,430) (DIVPR(NY),NY=N1,MAX)
  430 FORMAT (//,22H - PREFERRED DIVIDENDS,13X,8(1X,F9.2))
      WRITE (6,440)  (ERNCOM(NY),NY=N1,MAX)
  440 FORMAT (25H EARNINGS AVAIL TO COMMON,10X,8(1X,F9.2))
      WRITE (6,450) (DIVCOM(NY),NY=N1,MAX)
  450 FORMAT (//,19H - COMMON DIVIDENDS,16X,8(1X,F9.2))
      WRITE (6,460) (ERNRT(NY),NY=N1,MAX)
  460 FORMAT (18H RETAINED EARNINGS,17X,8(1X,F9.2))
      IF (MAX.EQ.MXYR) GO TO  462
      N1=MAX+1
      MAX=MXYR
      GO TO 305
  462 N1=1
      MAX=8
  465 CONTINUE
C
C
C     PRINT THE SOURCES AND USES OF FUNDS STATEMENT
C
C
      WRITE (6,99)
      WRITE (6,115)
      WRITE (6,470)
  470 FORMAT (40X,26H SOURCES AND USES OF FUNDS)
      WRITE (6,145)
      WRITE (6,155) (NYR(I),I=N1,MAX)
      WRITE (6,165) (HYPH(I),I=N1,MAX)
      WRITE (6,480)
  480 FORMAT (//,8H SOURCES)
      WRITE (6,490)
  490 FORMAT (//,9H INTERNAL)
      WRITE (6,500) (ERNET(NY),NY=N1,MAX)
  500 FORMAT (12H  NET INCOME,23X,8(1X,F9.2))
      WRITE (6,510) (DEP1(NY),NY=N1,MAX)
  510 FORMAT (14H  DEPRECIATION,21X,8(1X,F9.2))
      WRITE (6,520) (TXDEF(NY),NY=N1,MAX)
  520 FORMAT (23H  DEFERRED INCOME TAXES,12X,8(1X,F9.2))
C     WRITE (6,530) (TXCR(NY),NY=N1,MAX)
  530 FORMAT (20H  DEF INV TAX CREDIT,15X,8(1X,F9.2))
      WRITE (6,540) (SRCIN(NY),NY=N1,MAX)
  540 FORMAT (15H TOTAL INTERNAL,20X,8(1X,F9.2))
```

```
      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) (DEBR(NY),NY=N1,MAX)
  670 FORMAT (17H   DEBT RETIREMENT,18X,8(1X,F9.2))
      WRITE (6,680) (CAPIN(NY),NY=N1,MAX)
  680 FORMAT (25H NET INCR WORKING CAPITAL,10X,8(1X,F9.2))
      WRITE (6,690) (USETOT(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
C       PRINT THE RATIO ANALYSIS
C
      N1=1
      MAX=8
  720 CONTINUE
      WRITE (6,99)
      WRITE (6,115)
      WRITE (6,725)
  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 (1H ,/)
      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 RATIO,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
C-----------------------------------------------------------------
      SUBROUTINE BALANC(NY)
C-----------------------------------------------------------------
C
C
C     FIND CUMULATIVE VALUES OF COMMON STOCKS,RETAINED EARNINGS
C     AND PREFERRED STOCKS.
```

```
C
      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),DIVPR(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),OMC(20),FERN(20),
     +                 ANINC(20),ERNET(20),ERNRT(20),
     +                 OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
     +                 CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
     +                 CEPXC(20,20),CE1(20)
      COMMON /FUNDS/   SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
     +                 USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
     +                 ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20),
     +                 TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
     +                 OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
     +                 PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
     +                 DEBSR(20),FCE(20),LCR(20),TXCRC(20),
     +                 UPCOST(20),PLGSX(20),TXDEFC(20)
      COMMON /AUX2/    DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
      COMMON /BVAL/    PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
     +                 DEBSB,DEBLB,DEBSRB,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),IDEXP(20),NEXP,
     +                 NBEST,HXC,SXC,ISOL(20,20)
      COMMON /PARM/    NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
      COMMON /FLAGS/   ISTOP
      INTEGER RESP,YES,YE,NEG,NO
C
C
C
      DATA YES,YE/3HYES,1HY/
      DATA NEG,NE/2HNO,1HN/
C
C      FIND TOTAL ASSETS AND LIABILITIES
C
      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)
      EQTOT(NY)=EQCOM(NY)+STPRC(NY)
      CAPTOT(NY)=EQTOT(NY)+DEBLT(NY)
      TLIAB(NY)=CAPTOT(NY)+DEBST(NY)+TXDEFC(NY)
     +          + TXCRC(NY)
      BAL(NY)=TLIAB(NY)-PLNUT(NY)
      IF (NY.EQ.1) WKCAP(NY)=BAL(NY)
      ASSET(NY)=PLNUT(NY)+WKCAP(NY)
      DEBASS(NY)=DEBT(NY)/ASSET(NY)
      DEBEQ(NY)=DEBT(NY)/EQTOT(NY)
      COVRG(NY)=BINT(NY)/CDEBT(NY)
C
C
      RETURN
      END
C
C
C
C
C-----------------------------------------------------------------
      SUBROUTINE FUND(NY)
C-----------------------------------------------------------------
C
      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),DIVPR(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),OMC(20),FERN(20),
     +                ANINC(20),ERNET(20),ERNRT(20),
     +                OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
     +                CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
     +                CEPXC(20,20),CE1(20)
      COMMON /FUNDS/  SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
     +                USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
     +                ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20),
     +                TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
     +                OPEXP(20),TOTEXP(20),PLGA2(20),PLNV(20),
     +                PLNUT(20),STCOMC(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),DEBST(20),DEBLR(20),
     +                DEBSR(20),FCE(20),LCR(20),TXCRC(20),
     +                OPCOST(20),PLGSX(20),TXDEFC(20)
      COMMON /AUX2/   DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
      COMMON /BVAL/   PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
```

```
     +                DEBSB,DEBLB,DEBSRB,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),IDEXP(20),NEXP,
     +                NBEST,HXC,SXC,ISOL(20,20)
      COMMON /PARM/   NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
      COMMON /FLAGS/  ISTOP
      INTEGER RESP,YES,YE,NEG,NO
C
C
C
      DATA YES,YE/3HYES,1HY/
      DATA NEG,NE/2HNO,1HN/
C
      SRCIN(NY)=ERNET(NY)+DEP1(NY)+TXDEF(NY)
      SRCEX(NY)=STCOM(NY)+STPR(NY)+ DEBS(NY)+
     1          DEBL(NY)
      CE1(NY)=CE(NY)+AFDC1(NY)
      SRCTOT(NY)=SRCIN(NY)+SRCEX(NY)
      USE(NY)=CE1(NY)+DIVPR(NY)+DIVCOM(NY)+DEBR(NY)
      CAPIN(NY)=SRCTOT(NY)-USE(NY)
      USETOT(NY)=USE(NY)+CAPIN(NY)
C
C
      RETURN
      END
C
C
C----------------------------------------------------
      SUBROUTINE FBCHNG(NBDT)
C----------------------------------------------------
C
C
C     THIS SUBPROGRAM  IS USED TO CORRECT
C     OR CHANGE BEGINNING VALUES OF FINANCIAL DATA
C
C
      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),DIVPR(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),OMC(20),FERN(20),
     +                ANINC(20),ERNET(20),ERNRT(20),
     +                OMCF(20),OMCX(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),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),AFDC1(20),AFDC2(20),AFCX2(22,20),
     +                CAPRQ(20),RATB(20),CE(20),AFDCX2(20),
     +                CEPXC(20,20),CE1(20)
      COMMON /FUNDS/  SRCIN(20),SRCEX(20),SRC(20),RWKCAP(20),
     +                USE(20),CAPIN(20),SRCTOT(20),USETOT(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),DEP1C(20),DEP2C(20),BAL(20),
     +                ASSET(20),EQCOM(20),ERNCOM(20),CAPTOT(20),
     +                TXDEF(20),DEFCR(20),TLIAB(20),FTXDEF(20),
```

```
     +                 OPEXP(20),TOTEXP(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),DEBST(20),DEBLR(20),
     +                 DEBSR(20),FCE(20),LCR(20),TXCRC(20),
     +                 OPCOST(20),PLGSX(20),TXDEFC(20)
      COMMON /AUX2/    DSLIM(20),DLLIM(20),COMLIM(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),COVRG(20),KYR(20,2)
      COMMON /BVAL/    PLGSB,DEP1B,DEP2B,CWIPFB,CWIPXB,CWIPB,
     +                 DEBSB,DEBLB,DEBSRB,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),IDEXP(20),NEXP,
     +                 NBEST,HXC,SXC,ISOL(20,20)
      COMMON /PARM/    NBYR,MXYR,MXPLF,MXPLX,MXPL,MXALL,IDEF,IAFC
      COMMON /FLAGS/   ISTOP
C
      INTEGER RESP,YES,YE,NEG,NO
C
C
C
      DATA YES,YE/3HYES,1HY/
      DATA NEG,NE/2HNO,1HN/
C
  105 FORMAT (A3)
  106 CONTINUE
      WRITE (6,110)
  110 FORMAT (//,31H WHICH DATA YOU WANT TO CHANGE?,
     +/43H ENTER 2 DIGIT DATA CODE.FOR HELP,ENTER 99.,
     +/3H --)
      READ (5,120) IP
  120 FORMAT (I2)
      IF (IP.EQ.99) GO TO 160
C
C     ENTER CHANGED/CORRECTED PARAMETER
C
  125 CONTINUE
      WRITE (6,135) (KEYFIN(IP+34,J),J=1,18)
  135 FORMAT (18A4)
      WRITE (6,142) BDAT(IP)
  142 FORMAT (16H PREVIOUS VALUE=,F10.4)
      WRITE (6,145)
  145 FORMAT (11H -----.----)
      READ (5,155) BDAT(IP)
  155 FORMAT (F10.5)
      WRITE (6,159)
  159 FORMAT (//,26H ANY MORE CHANGES IN DATA?,
     +/17H ENTER YES OR NO./2H ?)
      READ (5,105) RESP
      IF (RESP.EQ.YES.OR.RESP.EQ.YE) GO TO 106
      GO TO 190
  160 CONTINUE
C
C     PRINT 2 DIGIT  DATA CODES AND
C     DEFINITONS.
C
      DO 180 IP=1,NBDT
      WRITE (6,135) (KEYFIN(IP+17,J),J=1,18)
  180 CONTINUE
      GO TO 106
```

```
  190 CONTINUE
C
C
      RETURN
      END
C
```

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.
?
y
     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.
y
     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
5	5	5	5	5	10	10	10	10	10	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
5	5	5	5	5	10	10	10	10	10	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
5	5	5	5	5	10	10	10	10	10	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
5	5	5	5	5	10	10	10	10	10	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
5	5	5	5	5	10	10	10	10	10	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
5	5	5	5	5	10	10	10	10	10	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
5	5	5	5	5	10	10	10	10	10	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
5	5	5	5	5	10	10	10	10	10	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
5	5	5	5	5	10	10	10	10	10	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
5	5	5	5	5	10	10	10	10	10	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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

PLANT CODE	1	2	3	4	5
	EXPN	EXPN	EXPN	EXPN	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 OTG DYS/YR	40.000	30.000	25.000	25.000	15.000
05 FORCED OTG.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	7	8	9	10
	EXPN	EXPN	EXPN	EXPN	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 OTG DYS/YR	10.000	25.000	0.800	5.000	2.000
05 FORCED OTG.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	12	13	14	15
	SCHD	SCHD	SCHD	SCHD	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 OTG DYS/YR	26.000	26.000	21.000	25.000	24.000
05 FORCED OTG.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	17	18	19	20
	SCHD	SCHD	SCHD	SCHD	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 OTG DYS/YR	20.000	26.000	25.000	14.000	21.000
05 FORCED OTG.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	[illegible]		

PLANT CODE	21 SCHD	22 SCHD	23 SCHD	24 SCHD	25 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. $/MWH	0.009	0.009	0.009	0.009	0.009
12 SALVG VALUE K$	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	26 SCHD	27 SCHD	28 SCHD	29 SCHD	30 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 SCHD	32 SCHD	33 SCHD	34 SCHD	35 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 SCHD	37 SCHD	38 SCHD	39 SCHD	40 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.

N

END OF PRINTING

```
     IS YOUR PLANT DATA COMPLETE ?   ENTER YES OR NO.
n
     ARE ANY CHANGES IN PLANT DATA NECESSARY? ENTER YES OR NO.
y
    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.
y


 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.

y
```

```
     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. AVILABLE 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 CAPCITY 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
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
MAINT OUTG(D/YR)	2.50000	2.00000	2.00000
*FIXED O.M.C.(K$/MW/YR)	12.50000	10.50000	10.50000
VAR O.M.C($/MWH)	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.
INPUT DATA STORED ON 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 4

	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.
y
           ENERGY MULTIPLYER FOR EACH SEASON EACH YEAR
  YEAR     81 :      1.07000     1.07000     1.07000     1.07000
  YEAR     82 :      1.14000     1.14000     1.14000     1.14000
  YEAR     83 :      1.22500     1.22500     1.22500     1.22500
  YEAR     84 :      1.31000     1.31000     1.31000     1.31000
  YEAR     85 :      1.40000     1.40000     1.40000     1.40000
  YEAR     86 :      1.50000     1.50000     1.50000     1.50000
  YEAR     87 :      1.60000     1.60000     1.60000     1.60000
  YEAR     88 :      1.72000     1.72000     1.72000     1.72000
  YEAR     89 :      1.83000     1.83000     1.83000     1.83000
  YEAR     90 :      1.97000     1.97000     1.97000     1.97000
  YEAR     91 :      2.10000     2.10000     2.10000     2.10000
  YEAR     92 :      2.25000     2.25000     2.25000     2.25000
  YEAR     93 :      2.40000     2.40000     2.40000     2.40000
  YEAR     94 :      2.58000     2.58000     2.58000     2.58000
  YEAR     95 :      2.76000     2.76000     2.76000     2.76000

           LOAD FACTOR FOR EACH SEASON EACH YEAR
  YEAR     81 :      0.60000     0.60000     0.60000     0.60000
  YEAR     82 :      0.60000     0.60000     0.60000     0.60000
  YEAR     83 :      0.60000     0.60000     0.60000     0.60000
  YEAR     84 :      0.60000     0.60000     0.60000     0.60000
  YEAR     85 :      0.60000     0.60000     0.60000     0.60000
  YEAR     86 :      0.60000     0.60000     0.60000     0.60000
  YEAR     87 :      0.60000     0.60000     0.60000     0.60000
  YEAR     88 :      0.60000     0.60000     0.60000     0.60000
  YEAR     89 :      0.60000     0.60000     0.60000     0.60000
  YEAR     90 :      0.60000     0.60000     0.60000     0.60000
  YEAR     91 :      0.60000     0.60000     0.60000     0.60000
  YEAR     92 :      0.60000     0.60000     0.60000     0.60000
  YEAR     93 :      0.60000     0.60000     0.60000     0.60000
  YEAR     94 :      0.60000     0.60000     0.60000     0.60000
  YEAR     95 :      0.60000     0.60000     0.60000     0.60000

    IS YOUR INPUT COMPLETE ?   ENTER YES OR NO.
y
```

DOES A FILE OF REFERENCE LOAD DATA EXIST? ENTER YES OR NO.

y

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
SP	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	1.51652D+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	5.99416E-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	2.18293D+07	5.99808E-01
2	1.34529E+04	4.96065E+03	8.04969D+03	1.75805D+07	5.98360E-01
3	1.63131E+04	4.85852E+03	9.77086D+03	2.13396D+07	5.98957E-01
4	1.41151E+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.hydro.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	839.714	9050.9
1982	1707.95	1878.75	840.223	9843.0
1983	988.95	2188.95	849.959	10362.1
1984	859.95	845.95	580.760	11081.0
1985	238.68	1438.68	605.452	11842.3
1986	0.0	2072.79	586.030	12688.2
1987	1166.89	2366.89	381.675	13534.1
1988	1451.84	2651.84	310.671	14549.2
1989	1521.37	2721.37	267.419	15479.6
1990	1837.14	3037.14	219.374	16663.9
1991	877.49	865.24	230.411	17763.5
1992	0.0	2218.67	242.719	19032.3
1993	2349.86	3549.86	150.151	20301.1
1994	2527.25	3727.25	137.108	21823.7
1995	2704.67	3904.67	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 SPECITY:
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 THEN 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 LOLP 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    5    2    1
        8    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
  60 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	562.918	165.656	2073.325	2	0	2
0.0	548.908	225.581	2712.466	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	4456.297	7	2	4
0.0	527.690	167.605	5028.672	7	3	5
0.0	554.537	307.419	5654.934	7	4	7
0.0	562.415	201.187	6257.578	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	8090.066	12	8	11
0.0	583.575	239.518	8689.605	15	9	12
0.0	604.551	107.986	9297.750	15	10	13

NEW UPPER BOUND FOR FATHOMING IS 9297.75

***** 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	2	0	0
	5	3	4
YEAR 1986	5	0	0
	8	3	4
YEAR 1987	5	0	1
	8	3	5
YEAR 1988	5	1	2
	8	4	6
YEAR 1989	5	2	6
	8	5	10
YEAR 1990	6	3	8
	9	6	10
YEAR 1991	10	3	7
	13	6	11

YEAR 1992	10	3	7
	13	6	11
YEAR 1993	11	6	10
	14	9	14
YEAR 1994	12	7	11
	15	10	15
YEAR 1995	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: 0.915411E+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	1
0.0	588.608	232.839	1438.623	0	0	2
0.0	562.919	165.656	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	512.615	234.522	4455.074	7	1	4
0.0	524.325	159.739	5021.992	7	1	5
0.0	553.416	307.419	5647.133	7	2	7
0.0	543.535	288.707	6248.398	7	3	9
0.0	554.811	150.351	6828.262	10	3	9
0.0	549.956	248.511	7411.344	10	3	11
0.0	561.562	178.132	7990.715	11	6	12
0.0	563.878	266.085	8572.324	12	7	14
0.0	575.593	185.955	9154.109	14	8	15

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	5	0	0
	8	3	4
YEAR 1987	5	0	1
	8	3	5
YEAR 1988	5	0	2
	8	3	6

```
YEAR 1989      5    0    6
               8    3   10
YEAR 1990      6    1    6
               9    4   10
YEAR 1991      8    1    7
              11    4   11
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     165.656     2073.325       2  0  2
   0.0       549.976     215.468     2709.469       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  6
   0.0       533.393     300.032     5654.402       6  1  8
   0.0       525.391     281.769     6236.141       6  1 10
   0.0       537.051     150.351     6788.246       9  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     8498.168      11  5 15
   0.0       572.232     122.021     9074.461      14  6 15
 NEW UPPER BOUND FOR FATHOMING IS        9074.46

*****  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	8
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 9
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	165.656	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	319.478	5051.008	6	1	6
0.0	533.393	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	532.403	239.130	7921.656	9	2	14
0.0	546.828	188.008	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	617.392	247.930	741.357	0	0	1
0.0	588.608	232.839	1438.623	0	0	2
0.0	562.918	165.656	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	319.478	5051.008	6	1	6
0.0	533.393	300.032	5654.402	8	1	8
0.0	525.391	281.769	6236.141	6	1	10
0.0	537.051	150.351	6798.246	8	1	10
0.0	533.969	248.511	7365.344	9	1	12
0.0	530.616	233.384	7919.293	9	1	14
0.0	538.554	234.122	8473.449	12	1	15
0.0	561.233	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	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	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:  0.903711E+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     165.656     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     319.478     5051.008       6  1  6
   0.0       533.393     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       530.616     233.384     7919.293       9  1 14
   0.0       538.554     234.122     8473.449      12  1 15
   0.0       559.766     116.953     9037.109      15  1 15
NEW UPPER BOUND FOR FATHOMING IS          9037.11

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 7	9	3	LOLP+	UNSERVED ENERGY (MWH)	TOTAL OPERATING COST*	CAPITAL COST**	SALVAGE VALUE*	OBJECTIVE FUNCTION*
1981	0	0	1	0.0	0.15360E+05	617.399	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	0.26608E+05	562.924	71.784	93.872	2073.325
1984	2	1	3	0.0	0.30080E+05	548.983	86.187	129.281	2709.489
1985	4	1	3	0.0	0.34720E+05	534.839	53.571	92.532	3297.892
1986	6	1	3	0.0	0.33184E+05	528.463	45.737	91.473	3872.104
1987	6	1	4	0.0	0.37056E+05	528.167	51.028	119.065	4451.289
1988	6	1	6	0.0	0.26768E+05	514.539	85.184	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	6798.246
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	7919.293
1994	12	1	15	0.0	0.0	538.564	15.608	218.514	8473.449
1995	15	1	15	0.0	0.0	559.775	3.888	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

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	139.266	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	40.900	37.198	33.362	30.664	25.055
12	48.126	45.969	43.485	40.940	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	9.892

COST OF OPERATION BY PLANT TYPE

MILLIONS OF 1980 DOLLARS

PLANT TYPE	1981	1982	1983	1984	1985	1986	1987	1988
16	30.997	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	34.464	30.190	27.591	22.413
19	24.535	21.574	17.423	14.937	12.028	9.849	9.140	6.400
20	15.405	13.488	10.696	9.187	7.290	5.985	5.620	3.915
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	6.049	3.754
23	18.554	15.920	12.615	10.924	8.660	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	4.934	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

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	0.009	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	0.006
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	9.310	8.095
42	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
43	0.748	0.651	0.568	0.492	0.428	0.372	0.324	0.281
TOTAL	617.399	588.615	562.924	549.983	534.839	528.483	528.167	514.539

COST OF OPERATION BY PLANT TYPE

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	284.491	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	65.198	62.363	59.652	76.078	90.961	1145.389
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	9.966	9.533	9.118	8.722	8.343	7.980	7.633	92.529
10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	27.199	23.359	22.648	19.832	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	11.937	388.595
14	16.083	12.216	13.244	10.526	8.616	7.785	9.135	280.796
15	14.254	11.250	11.922	9.880	7.924	7.457	8.495	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	9.203	9.550	7.676	6.539	5.968	7.358	198.446
17	12.183	9.633	9.872	8.233	6.759	6.047	7.177	212.077
18	24.914	21.330	20.569	17.561	15.495	14.384	14.614	416.375
19	9.712	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	3.953	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	5.991	4.443	4.928	3.769	2.977	2.879	3.721	102.765
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	1.685	1.689	2.383	50.684
29	2.672	2.049	2.408	1.964	1.468	1.449	2.085	47.499
30	1.534	1.087	1.300	1.130	0.892	0.902	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	0.497	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	0.089	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	533.978	530.624	538.564	559.775	8183.738

CONSTRUCTION COSTS* BY PLANT TYPE

MILLIONS OF 1980 DOLLARS

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

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	641.127
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 DYNO 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 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)

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 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 (MILLIOON DOLLARS)
27 RATIO OF CURRENT YR WK CAPTL/PREV YR OP 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
--
05

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.
?
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.
?
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 (MILLIOON DOLLARS)
27 RATIO OF CURRENT YR WK CAPTL/PREV YR OP 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.
?
n


DO YOU WANT TO CHANGE ANY YEARLY FINANCIAL DATA?
ENTER YES OR NO.
n


ANY CHANGES IN FINANCIAL DATA?
ENTER YES OR NO.
?
n


IS YOUR INPUT FILE FOR FINANCIAL DATA COMPLETE?
ENTER YES OR NO.
?
y


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)	1952.28	2216.28	2480.28	2797.08	3113.88	3430.68	3747.48	4117.07
- ACCUM DEPRECIATION	111.80	182.40	261.81	351.61	451.81	562.41	683.41	816.41
NET PLANT VALUE	1840.48	2033.88	2218.47	2445.47	2662.07	2868.27	3064.06	3300.66
+CWIP	950.31	1058.50	1189.09	1266.48	1365.47	1510.06	1702.05	1859.04
NET UTILITY PLANT	2790.79	3092.38	3407.56	3711.95	4027.54	4378.32	4766.11	5159.70
NET WORKING CAPITAL	371.69	173.66	134.54	140.22	146.19	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 LIABILTIES	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
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
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
- DEFFERED 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	1640.79	1748.18	1891.89	2039.34	2144.18	2141.96	2101.01
- OPER AND MAINT COSTS	7.77	6.83	6.02	5.31	4.70	4.17	3.71
- FUEL COSTS	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
- DEFFERED 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	1451.09
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	324.77	334.61	354.68	373.53	393.25	412.82	432.64	454.47
DEPRECIATION	61.80	70.60	79.40	89.80	100.20	110.60	121.00	133.00
DEFERRED INCOME TAXES	33.62	36.90	39.57	42.56	44.90	46.63	47.79	49.31
TOTAL INTERNAL	420.20	442.11	473.65	505.89	538.35	570.05	601.43	636.78
EXTERNAL								
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)	359.59	372.19	394.59	394.19	415.79	461.39	508.79	526.59
PREFERRED DIVIDENDS	1.32	1.32	2.15	3.12	4.11	5.26	6.56	7.87
COMMON DIVIDENDS	258.76	266.63	282.02	296.33	311.31	326.05	340.86	357.28
DEBT RETIREMENT	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
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 CERAS.
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.