CR-123948

TR-796-808
SEPTEMBER 1970

RANDOM EXPERIMENT PROGRAM RESOURCE IMPACT (REPRI) PROGRAM USERS MANUAL

Prepared for:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GEORGE C. MARSHALL SPACE FLIGHT CENTER
Aero-Astrodynamics Laboratory

UNDER CONTRACT NO. NAS8-20082

(NASA-CR-123948) RANDOM EXPERIMENT PROGRAM RESOURCE IMPACT (REPRI) PROGRAM: USER'S MANUAL W.T. Pease, et al (Northrop Corp.) Sep. 1972 108 p CSCL 09B

N73-12174

Unclas
00/08 49049

NORTHROP CORPORATION
ELECTRO-MECHANICAL DIVISION
P. O. BOX 1484
HUNTSVILLE, ALABAMA 35807
TELEPHONE (205)837-0580

TR-796-808

RANDOM EXPERIMENT PROGRAM RESOURCE IMPACT (REPRI) PROGRAM

USERS MANUAL

September 1970

by

W. T. Pease
R. A. Alford

PREPARED FOR:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GEORGE C. MARSHALL SPACE FLIGHT CENTER
AERO-ASTRODYNAMICS LABORATORY

Under Contract NAS8-20082

REVIEWED AND APPROVED BY:

Douglas Raney
D. Raney, Manager
Astrodynamics

W. A. Klabunde
W. A. Klabunde, Manager
Aero-Astrodynamics Program

NORTHROP CORPORATION
HUNTSVILLE, ALABAMA

FOREWORD

The program documented in this report was prepared under Contract NAS8-20082, Appendix C, Schedule Order C-82. This work was done at the direction of the Technical Coordinator, James Mabry, of the Operational Analysis Branch (AERO-MX) of the Aero-Astrodynamics Laboratory, George C. Marshall Space Flight Center. This task was performed by R. A. Alford and W. T. Pease of the Orbital Analysis and Mission Planning organization of Northrop-Huntsville. The assistance and guidance of Mr. J. Moore of the Operations Analysis Branch of AAL contributed to the successful and timely completion of this task.

ABSTRACT

This document is a complete user and programmer guide for the REPRI program. This program was developed to perform mission concept, subsystem capability, and experiment support compatibility studies for the Space Station.

The program utilizes Monte Carlo techniques to randomly schedule events in discrete intervals. Resources, logistics, cost, and space station volume are considered in the program.

The program may be used to:

- Constrain available resources and/or space station volume and determine how much of a given experiment program can be accomplished.
- Determine the quantity of resources and/or space station volume is required to support a given experiment program.
- Determine the sensitivity of the experiment program scheduling to availability of resources, space station volume, or to individual experiments.

The speed of the REPRI program allows large sample sets of feasible schedules to be generated and summarized. This allows statistically valid conclusions to be drawn from the data.

TABLE OF CONTENTS

Section I
INTRODUCTION

The Monte Carlo simulation computer program REPRI (Random Experiment Program with Resource Impact) was developed to analyze space station experiment requirements. This program allows statistical methods to be applied to determine the interrelationship of scheduling with resource and logistics requirements.

Some of the guidelines and capabilities incorporated in REPRI are as follows:

- Space station missions can be analyzed using 20 to 120 equal time intervals.
- No ephemeris-related information is considered.
- The ability to summarize and constrain resource requirements in every interval is provided.
- The ability to constrain each experiment to start in or after a given interval is provided.
- All performances of a single experiment are scheduled as closely together as possible.
- The ability to define an experiment as being mutually exclusive with up to three others is provided. A typical application of this provision is the consideration due to conflicting use of the same docking port.
- The development cost of the experiment program including the module development and fabrication cost is summarized for each year.
- The space station storage volume required in each interval is determined for each schedule.
- The cumulative percent of the experiment performances completed is determined.
- After all schedules are generated, the maximum and average values of each parameter are determined for each interval.
- After all schedules are generated, the total number of times that each experiment was scheduled in each interval is determined.

Each schedule that REPRI generates specifies the intervals in which each experiment will be performed and the total amount of resource and logistics requirements needed in each interval. The resource requirements are user defined parameters which can be constrained to a specified maximum in each interval, e.g., manhours, average power, etc. The logistics parameters consist of the supply and experiment weights and volumes to and from orbit in each interval.

REPRI generates and summarizes random schedules. The summary consists of the maximum and mean values of the resource, logistics, and cost requirements. The random schedules are based upon the principle of scheduling the experiments in random order in randomly chosen intervals.

REPRI can be applied to mission planning for the following purposes:

- To determine the sensitivity of scheduling to selected parameters
- To determine mission and station characteristics required to support a particular experiment package
- To generate a large number of random schedules for statistical evaluation.

Each subroutine of REPRI is described in Section II. The input and output are described in Section III and Section IV, respectively. An explanation of the probability theory to be applied to the data generated using REPRI is outlined in Appendix A and a Fortran listing of REPRI is presented in Appendix B. A FLOWTRAN description of the REPRI program is included in Appendix C.

The REPRI program has been used to determine:

- The sensitivity of space station experiment scheduling to changes in available power and men.
- Impact of performing the artificial gravity experiment on the performance of the space station experiment package.
- Average and maximum percentages of space station experiment performances which can probably be scheduled. Also, average and maximum experiment support cost curves for the period 1970-1986.

These are a few examples of how the program can be used. Some of the study results obtained from the REPRI program have been documented in NASA TMX-64557 entitled "Explanation of Random Experiment Scheduling and its Application to Space Station Analysis", by Mr. J. Moore of AERO-MX.

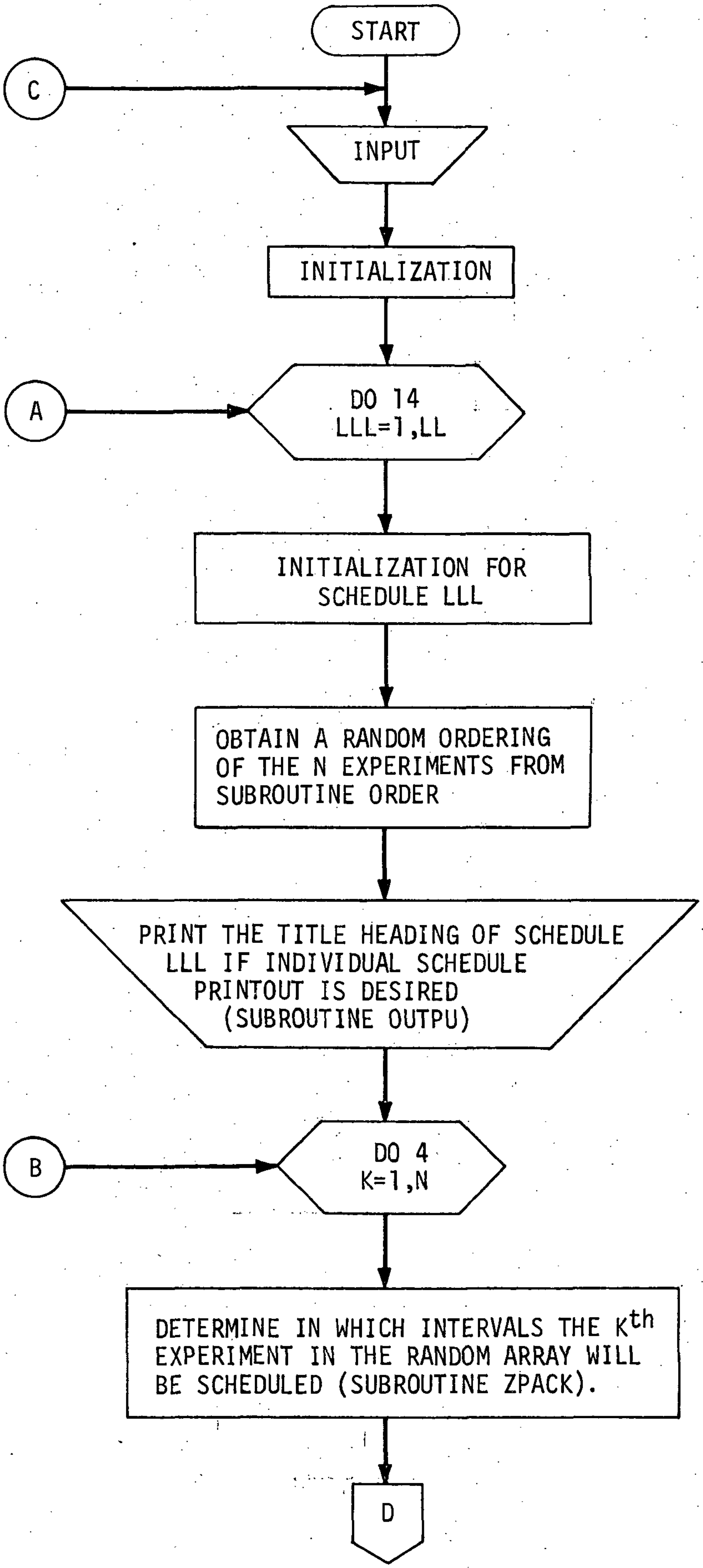
The main point that should be recognized is that the REPRI program allows very large samples of feasible schedules to be generated. Large samples allow statistically valid trends to be established.

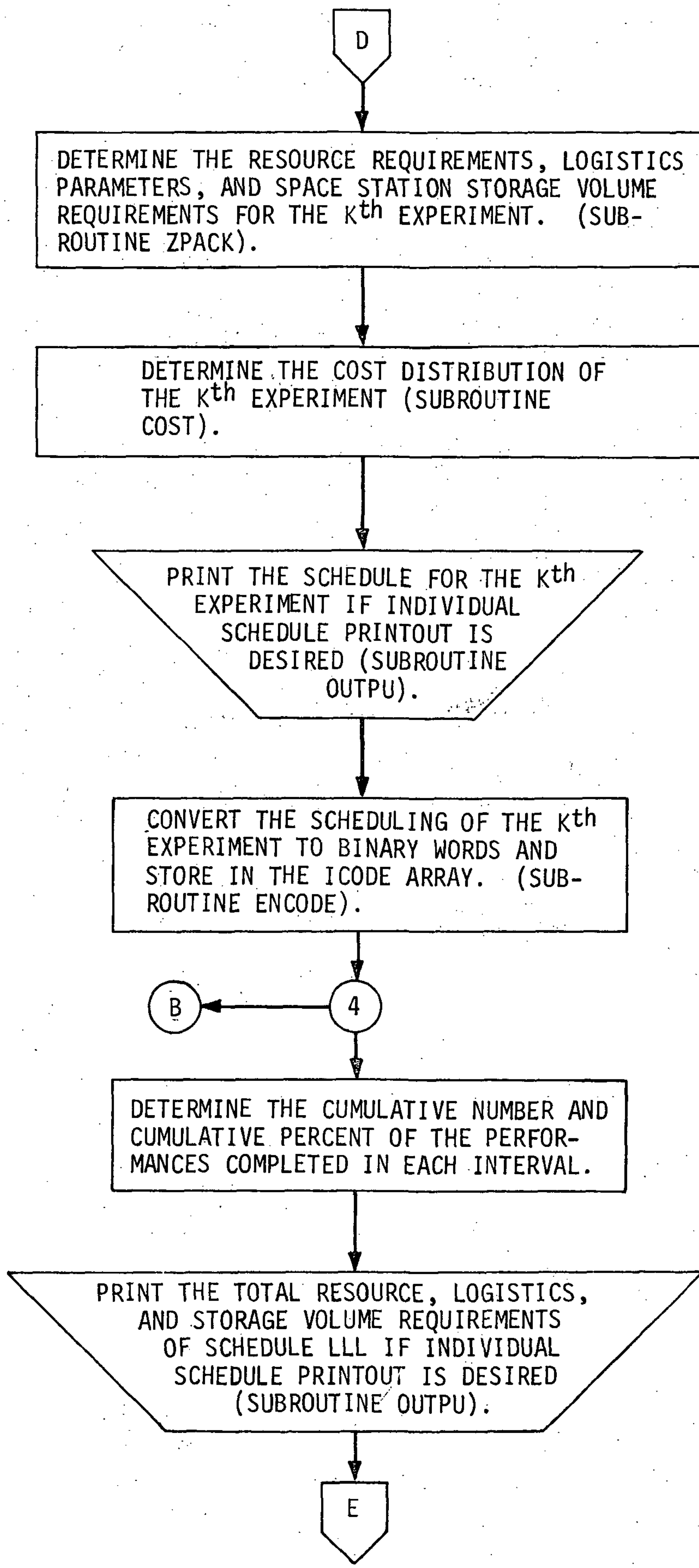
Section II
PROGRAM DESCRIPTION

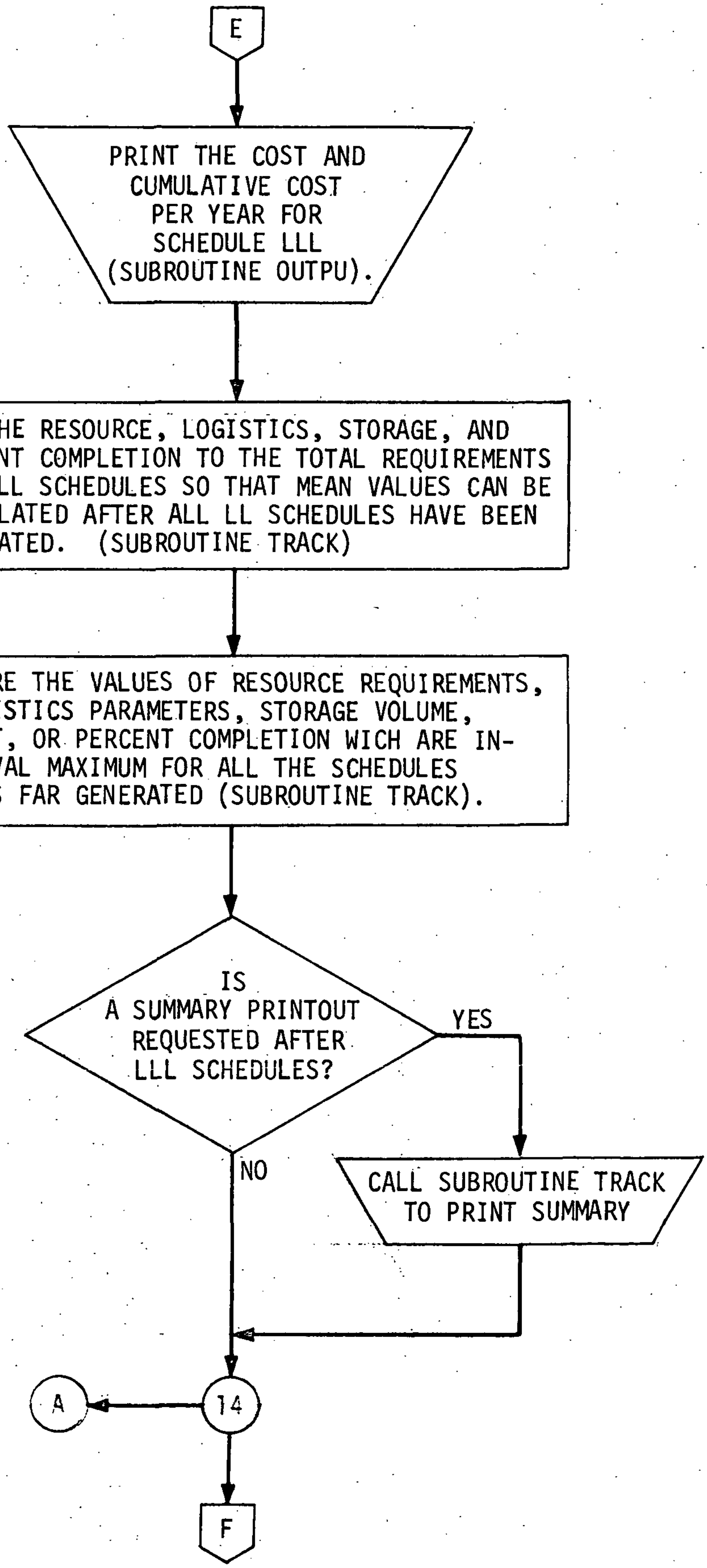
2.1 GENERAL

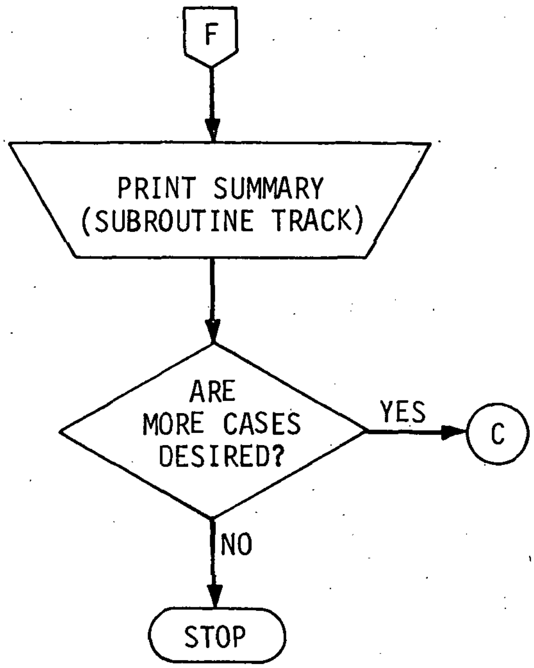
The organization of the REPRI program has been modularized for flexibility. The following block logic shows how the main program links the subprograms together. The order of calling and the functions are described in the block logic. The details of the subroutines are included in subsections 2.2 through 2.10. Only major subroutines are shown in block logic form. Complete FLOWTRAN logic diagrams may be found in Appendix C. A listing of the program may be found in Appendix B.

MAIN PROGRAM




D
DETERMINE THE RESOURCE REQUIREMENTS, LOGISTICS PARAMETERS, AND SPACE STATION STORAGE VOLUME REQUIREMENTS FOR THE Kth EXPERIMENT. (SUBROUTINE ZPACK).
DETERMINE THE COST DISTRIBUTION OF THE Kth EXPERIMENT (SUBROUTINE COST).
PRINT THE SCHEDULE FOR THE Kth EXPERIMENT IF INDIVIDUAL SCHEDULE PRINTOUT IS DESIRED (SUBROUTINE OUTPU).
CONVERT THE SCHEDULING OF THE Kth EXPERIMENT TO BINARY WORDS AND STORE IN THE ICODE ARRAY. (SUBROUTINE ENCODE).
B
4
DETERMINE THE CUMULATIVE NUMBER AND CUMULATIVE PERCENT OF THE PERFORMANCES COMPLETED IN EACH INTERVAL.
PRINT THE TOTAL RESOURCE, LOGISTICS, AND STORAGE VOLUME REQUIREMENTS OF SCHEDULE LLL IF INDIVIDUAL SCHEDULE PRINTOUT IS DESIRED (SUBROUTINE OUTPU).
E


E
PRINT THE COST AND CUMULATIVE COST PER YEAR FOR SCHEDULE LLL (SUBROUTINE OUTPU).
ADD THE RESOURCE, LOGISTICS, STORAGE, AND PERCENT COMPLETION TO THE TOTAL REQUIREMENTS FOR ALL SCHEDULES SO THAT MEAN VALUES CAN BE CALCULATED AFTER ALL LL SCHEDULES HAVE BEEN GENERATED. (SUBROUTINE TRACK)
STORE THE VALUES OF RESOURCE REQUIREMENTS, LOGISTICS PARAMETERS, STORAGE VOLUME, COST, OR PERCENT COMPLETION WICH ARE INTERVAL MAXIMUM FOR ALL THE SCHEDULES THUS FAR GENERATED (SUBROUTINE TRACK).
IS A SUMMARY PRINTOUT REQUESTED AFTER LLL SCHEDULES?
YES
NO
CALL SUBROUTINE TRACK TO PRINT SUMMARY
A
14
F


F
PRINT SUMMARY
(SUBROUTINE TRACK)
ARE
MORE CASES
DESIRED?
YES
C
NO
STOP

2.2 SUBROUTINE INPUT

Subroutine INPUT reads in the data for each case and prints out this information in tabular form. A detailed discussion of the input is presented in Section III.

2.3 SUBROUTINE ZPACK

Each time subroutine ZPACK is called, one experiment is scheduled. If a forced start interval has not been specified, ZPACK will select a random interval in which it will attempt to schedule the experiment designated in the calling argument. The experiment will be scheduled in this interval if the following conditions are met:

- There is a sufficient amount of the four constrained resources available in the interval to perform the experiment
- There is sufficient volume aboard the space station to store the experiment volume plus its supply volume in the interval
- No experiment which is mutually exclusive with the one being scheduled has already been scheduled in the interval
- The interval number is equal to or greater than the minimum start interval assigned to that experiment.

If these are not met, additional intervals will be randomly selected either until the conditions are met or until all intervals have been checked. If additional performances are required, ZPACK will attempt to schedule them subject to the above conditions in the intervals immediately following the random interval. If all the required performances cannot be scheduled in the following intervals, ZPACK will attempt to schedule those remaining in the intervals preceding the random interval.

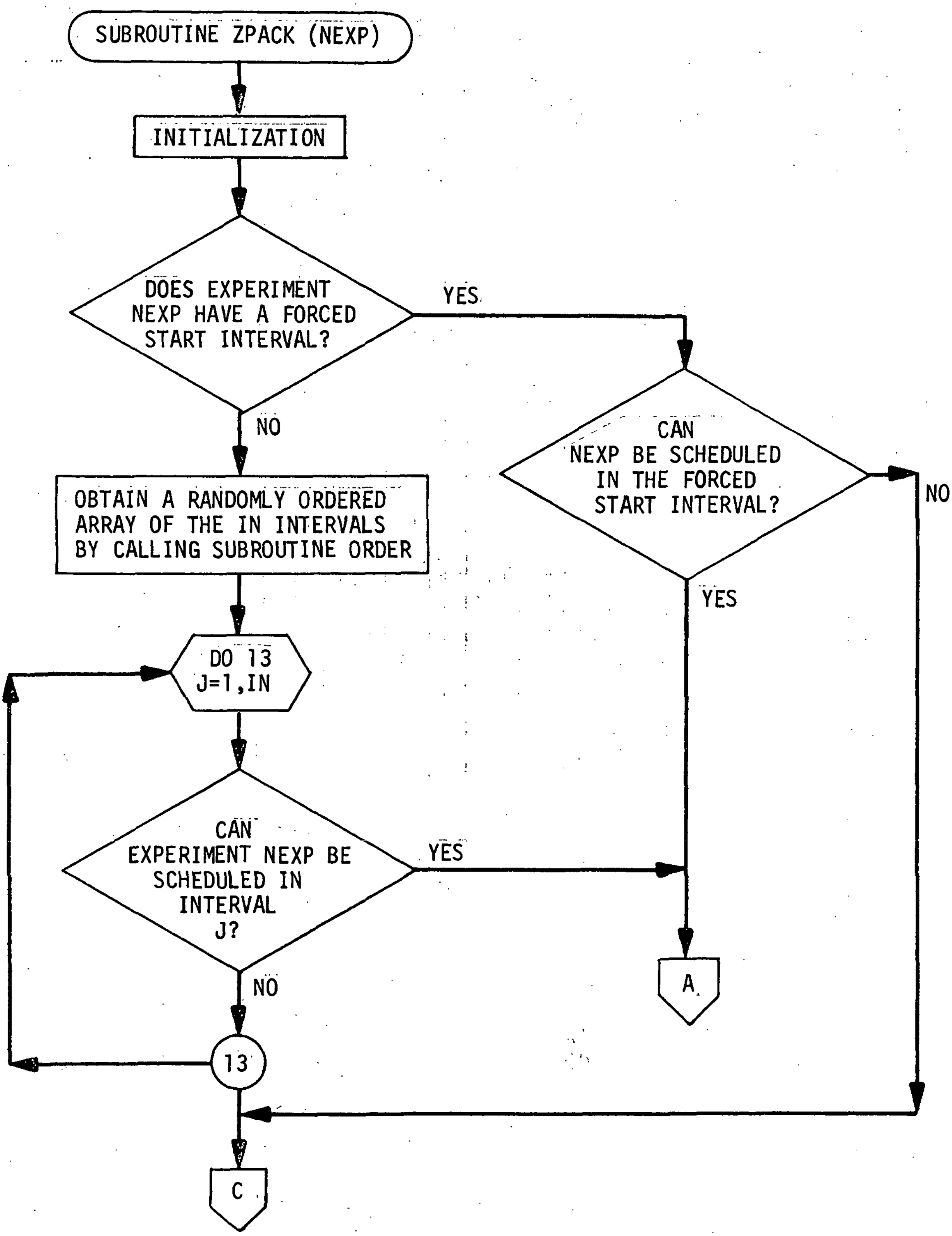
When scheduling has been completed for the designated experiment, its resource requirements (e.g., manhours, bit rate, power, kilowatt hours, etc.), supply weight and volume, return weight and volume, and space station storage volume requirements are summed for the intervals in which it was scheduled. The ITOTL array is zeroed at the start of each schedule and contains the total requirements of the above parameters after all the experiments have been scheduled. One of the four resource requirements has 15 subcategories (e.g., skill types are subcategories of manhours) which are also summed in the ITOTL array.

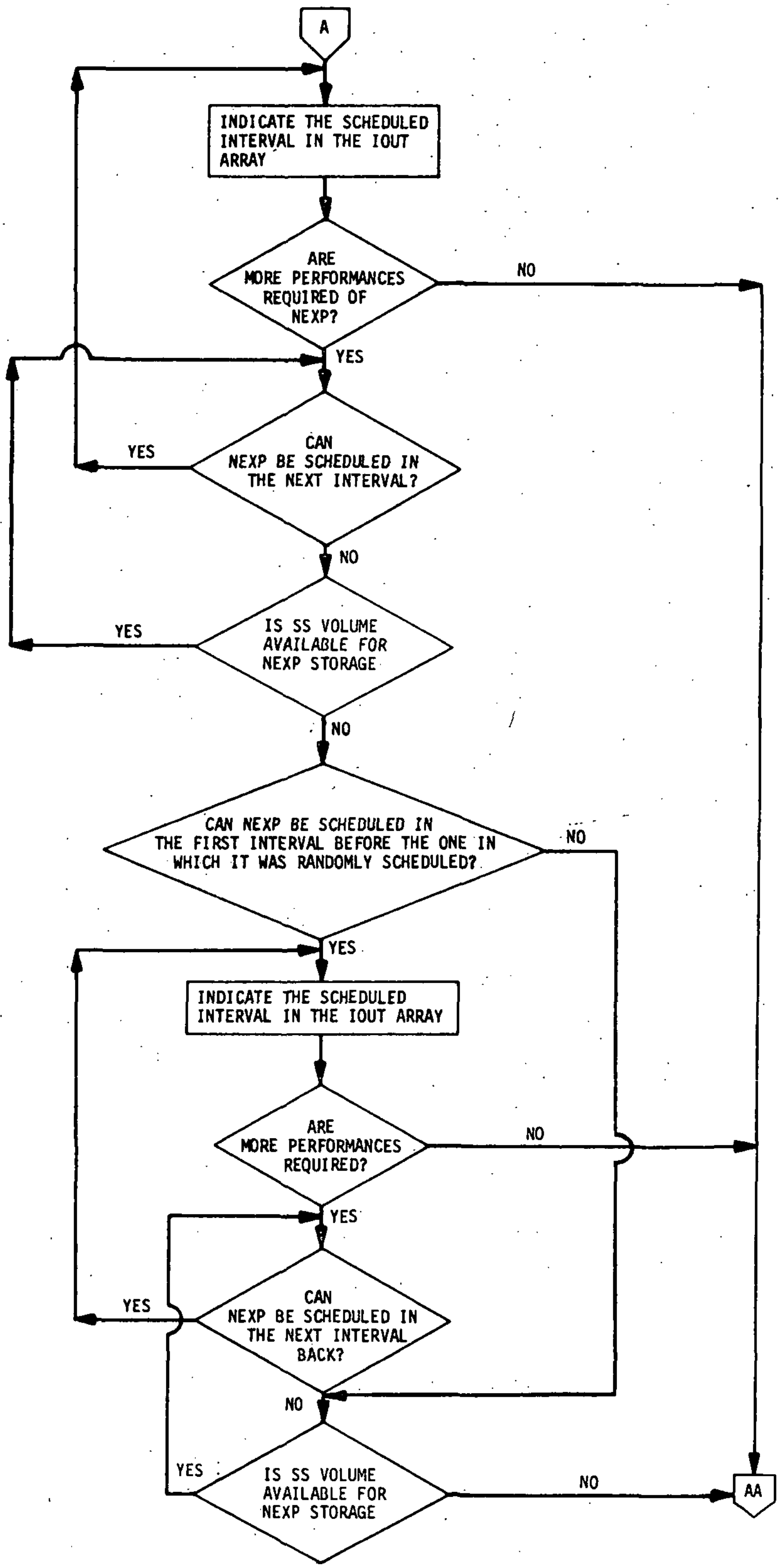
The experiment weight and volume to orbit in the i^{th} interval is determined by summing the weights and volumes of all the experiments which are scheduled for the first time in the i^{th} interval.

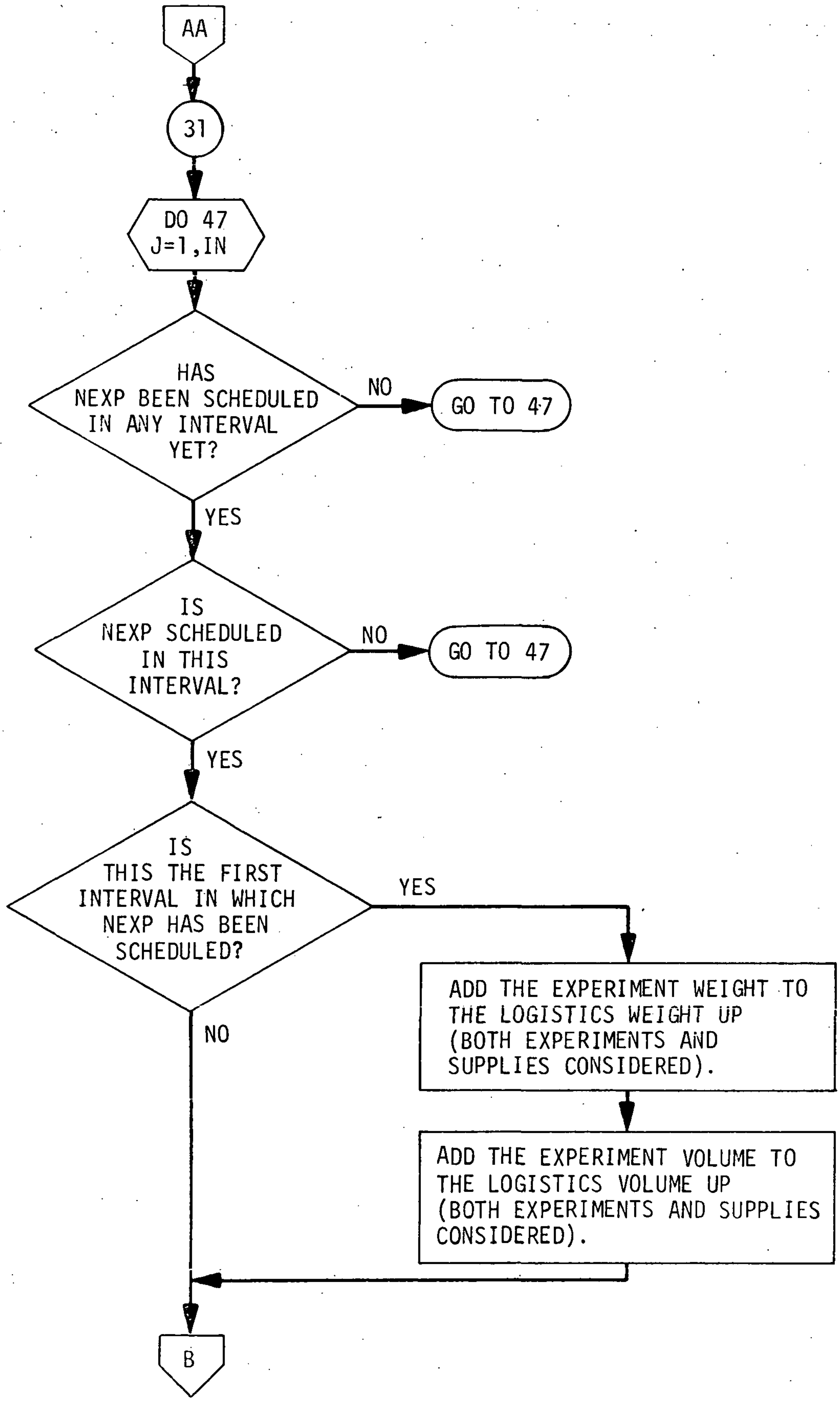
The experiment weight and volume returned from orbit in the i^{th} interval is determined by summing the weights and volumes of all the experiments which are scheduled for the last time in the i^{th} interval.

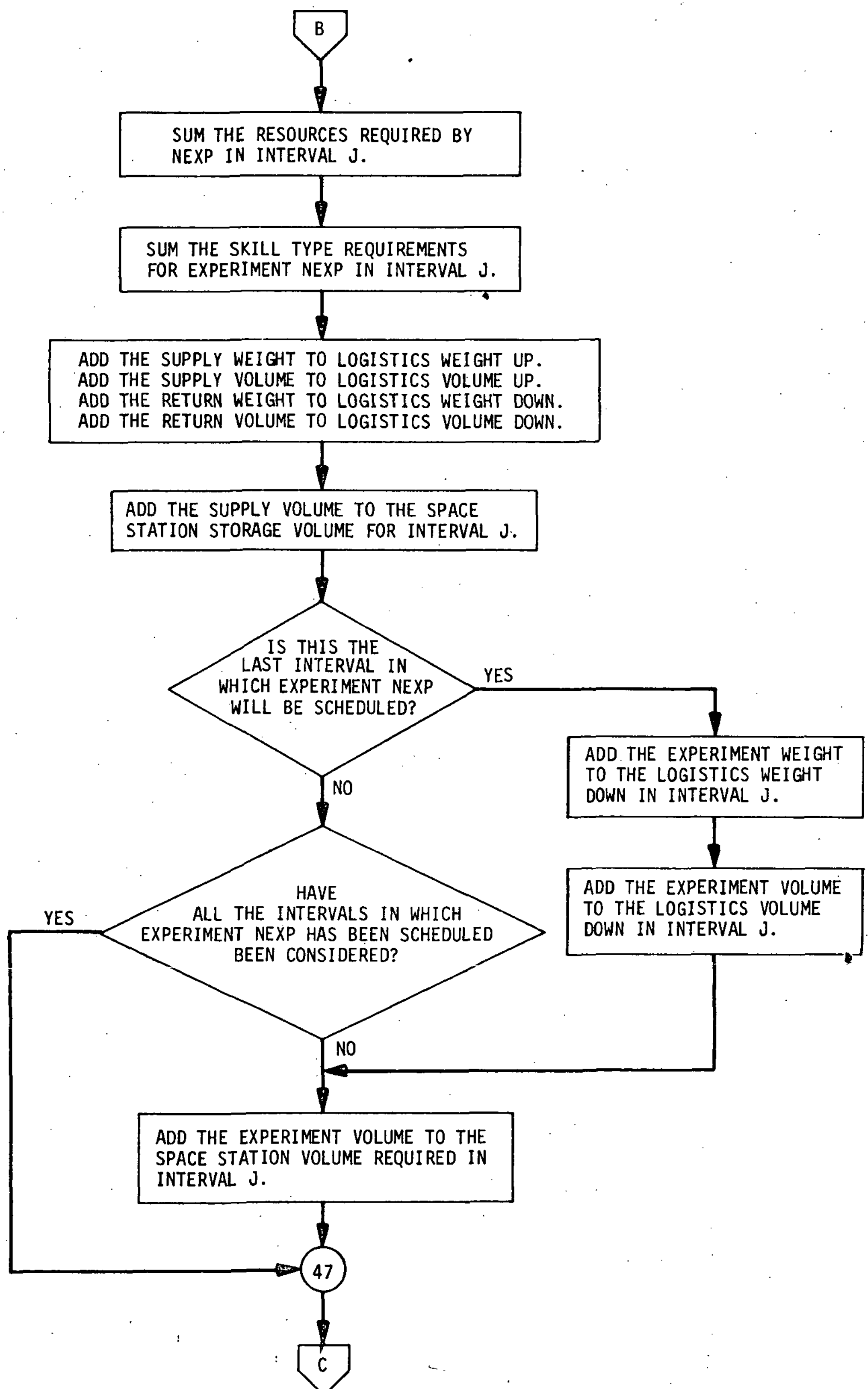
The space station storage volume requirements in the i^{th} interval are determined by summing the supply volumes of all the experiments scheduled in that interval plus the experiment volume of those experiments for which the i^{th} interval falls between the first and last intervals in which they were scheduled.

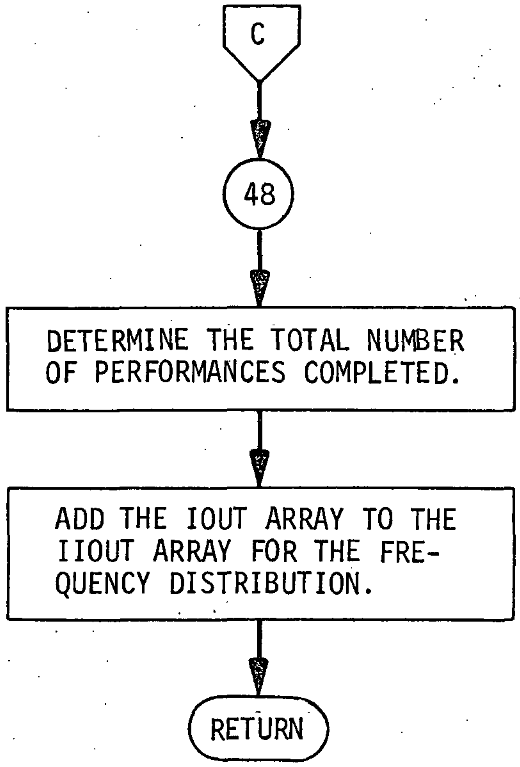
BLOCK LOGIC FOR SUBROUTINE ZPACK




A
INDICATE THE SCHEDULED INTERVAL IN THE IOUT ARRAY
ARE MORE PERFORMANCES REQUIRED OF NEXP?
NO
YES
CAN NEXP BE SCHEDULED IN THE NEXT INTERVAL?
YES
NO
IS SS VOLUME AVAILABLE FOR NEXP STORAGE
YES
NO
CAN NEXP BE SCHEDULED IN THE FIRST INTERVAL BEFORE THE ONE IN WHICH IT WAS RANDOMLY SCHEDULED?
NO
YES
INDICATE THE SCHEDULED INTERVAL IN THE IOUT ARRAY
ARE MORE PERFORMANCES REQUIRED?
NO
YES
CAN NEXP BE SCHEDULED IN THE NEXT INTERVAL BACK?
YES
NO
IS SS VOLUME AVAILABLE FOR NEXP STORAGE
YES
NO
AA


AA
31
DO 47
J=1,IN
HAS
NEXP BEEN SCHEDULED
IN ANY INTERVAL
YET?
NO
GO TO 47
YES
IS
NEXP SCHEDULED
IN THIS
INTERVAL?
NO
GO TO 47
YES
IS
THIS THE FIRST
INTERVAL IN WHICH
NEXP HAS BEEN
SCHEDULED?
YES
ADD THE EXPERIMENT WEIGHT TO
THE LOGISTICS WEIGHT UP
(BOTH EXPERIMENTS AND
SUPPLIES CONSIDERED).
ADD THE EXPERIMENT VOLUME TO
THE LOGISTICS VOLUME UP
(BOTH EXPERIMENTS AND SUPPLIES
CONSIDERED).
NO
B


B
SUM THE RESOURCES REQUIRED BY NEXP IN INTERVAL J.
SUM THE SKILL TYPE REQUIREMENTS FOR EXPERIMENT NEXP IN INTERVAL J.
ADD THE SUPPLY WEIGHT TO LOGISTICS WEIGHT UP.
ADD THE SUPPLY VOLUME TO LOGISTICS VOLUME UP.
ADD THE RETURN WEIGHT TO LOGISTICS WEIGHT DOWN.
ADD THE RETURN VOLUME TO LOGISTICS VOLUME DOWN.
ADD THE SUPPLY VOLUME TO THE SPACE STATION STORAGE VOLUME FOR INTERVAL J.
IS THIS THE LAST INTERVAL IN WHICH EXPERIMENT NEXP WILL BE SCHEDULED?
YES
ADD THE EXPERIMENT WEIGHT TO THE LOGISTICS WEIGHT DOWN IN INTERVAL J.
ADD THE EXPERIMENT VOLUME TO THE LOGISTICS VOLUME DOWN IN INTERVAL J.
NO
HAVE ALL THE INTERVALS IN WHICH EXPERIMENT NEXP HAS BEEN SCHEDULED BEEN CONSIDERED?
YES
NO
ADD THE EXPERIMENT VOLUME TO THE SPACE STATION VOLUME REQUIRED IN INTERVAL J.
47
C

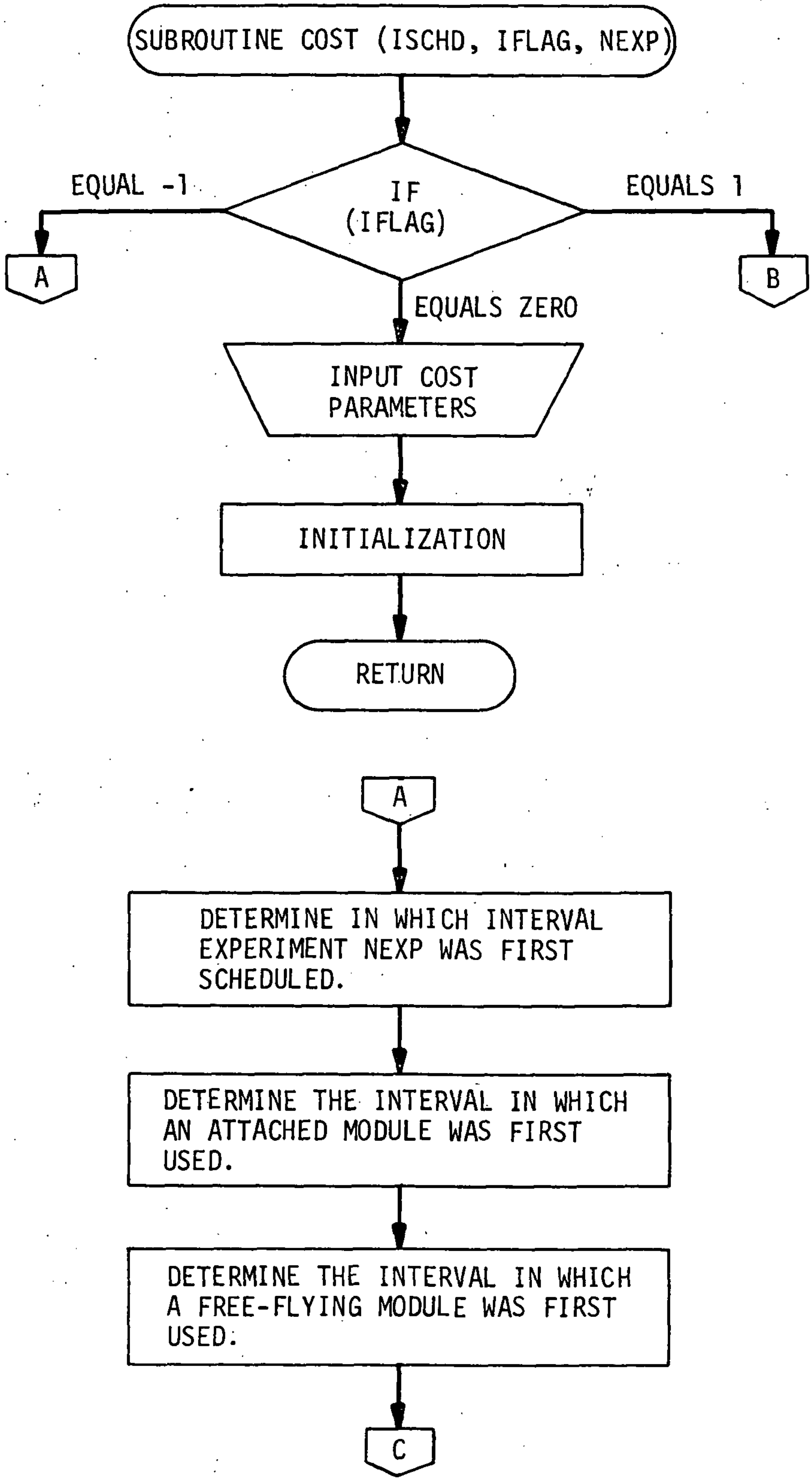

C
48
DETERMINE THE TOTAL NUMBER OF PERFORMANCES COMPLETED.
ADD THE IOUT ARRAY TO THE IIOUT ARRAY FOR THE FRE-QUENCY DISTRIBUTION.
RETURN

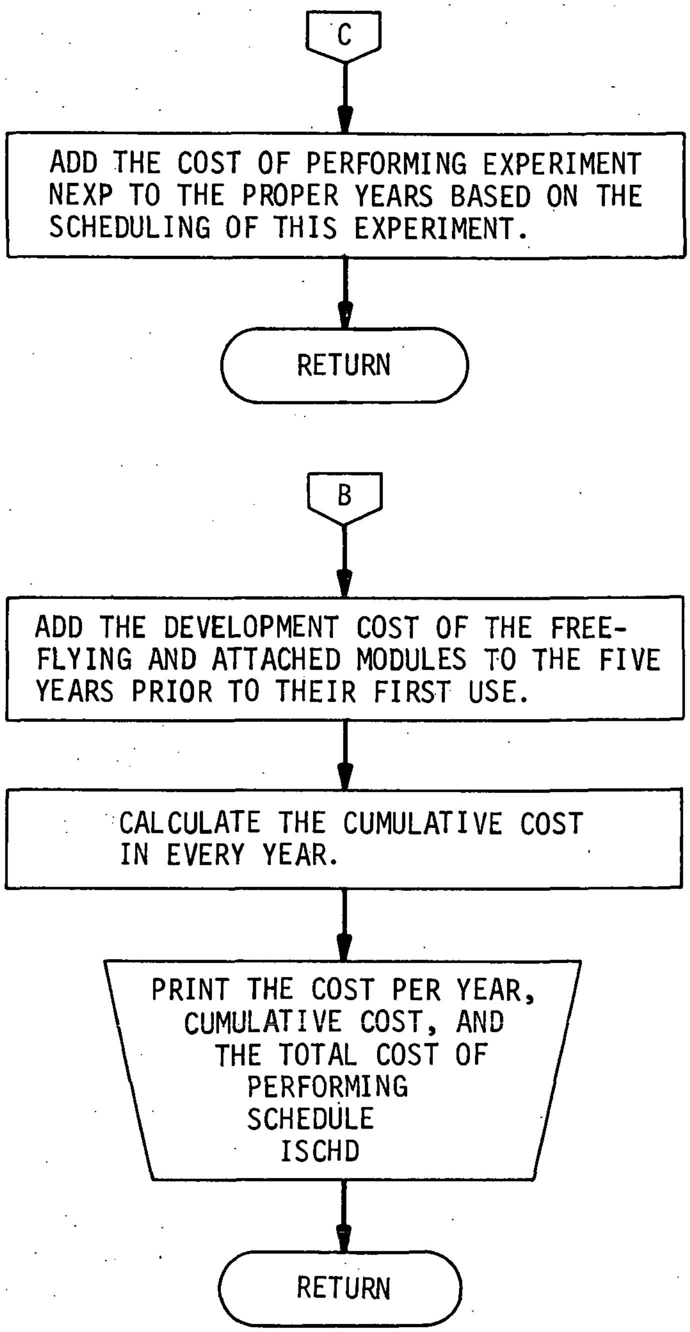
2.4 SUBROUTINE COST

Subroutine COST calculates the yearly cost of performing each schedule based on a 10-year experiment program. Funding is assumed to begin seven years prior to the start of the program. The development cost of each experiment is input for eight years prior to its first scheduled performance. The cost of developing the free-flying and attached modules is input for the five years prior to their respective first uses.

Subroutine COST has three entry points determined from the value of the calling argument parameter IFLAG. If IFLAG equals zero, the development costs of each experiment and the modules are read from card input. If IFLAG is negative, the cost of developing the experiment designated by the calling argument parameter NEXP is added to the XOCST array for the eight years prior to its first scheduled performance. At the completion of each schedule, subroutine COST is called with IFLAG positive. The development cost of both module types are summed in the XOCST array for the five years prior to their respective first uses. The cumulative cost is then calculated and the yearly cost and the cumulative cost are printed.

BLOCK LOGIC FOR SUBROUTINE COST




C
ADD THE COST OF PERFORMING EXPERIMENT NEXP TO THE PROPER YEARS BASED ON THE SCHEDULING OF THIS EXPERIMENT.
RETURN
B
ADD THE DEVELOPMENT COST OF THE FREE-FLYING AND ATTACHED MODULES TO THE FIVE YEARS PRIOR TO THEIR FIRST USE.
CALCULATE THE CUMULATIVE COST IN EVERY YEAR.
PRINT THE COST PER YEAR, CUMULATIVE COST, AND THE TOTAL COST OF PERFORMING SCHEDULE ISCHD
RETURN

2.5 SUBROUTINE RANDXX

Subroutine RANDXX generates random numbers that are uniformly distributed between zero and one. An input integer is forced to overflow the computers integer capacity by multiplication; this eliminates the most significant figures of the product. A real number between zero and one is then obtained by dividing this integer by the largest integer the computer can represent.

2.6 SUBROUTINE ORDER

Subroutine ORDER will randomly order an array of n integers by calling subroutine RANDXX n times.

2.7 SUBROUTINE ENCODE

Subroutine ENCODE will store the IOUT array in the ICODE array under the proper experiment number. Ten integers are packed into one integer word (e.g., 1,0,0,1,1,0,0,1,1, and 0 becomes 1001100110).

2.8 FUNCTION ICON

The function subprogram ICON is called with the number of the experiment being scheduled (NEXP) and the interval number (IN) in which scheduling is being considered.

The purpose of the function is to determine if a conflict exists with experiments which have previously been scheduled in the interval. If a conflict with another experiment exists, the value of ICON is set equal to 1. If no conflict exists, then ICON becomes 0. If experiment number 37 conflicts with the experiment during the interval, then ICON is set equal to 2. The ICOD(K,37) array is input as the scheduling proposed for the artificial-gravity timeline.

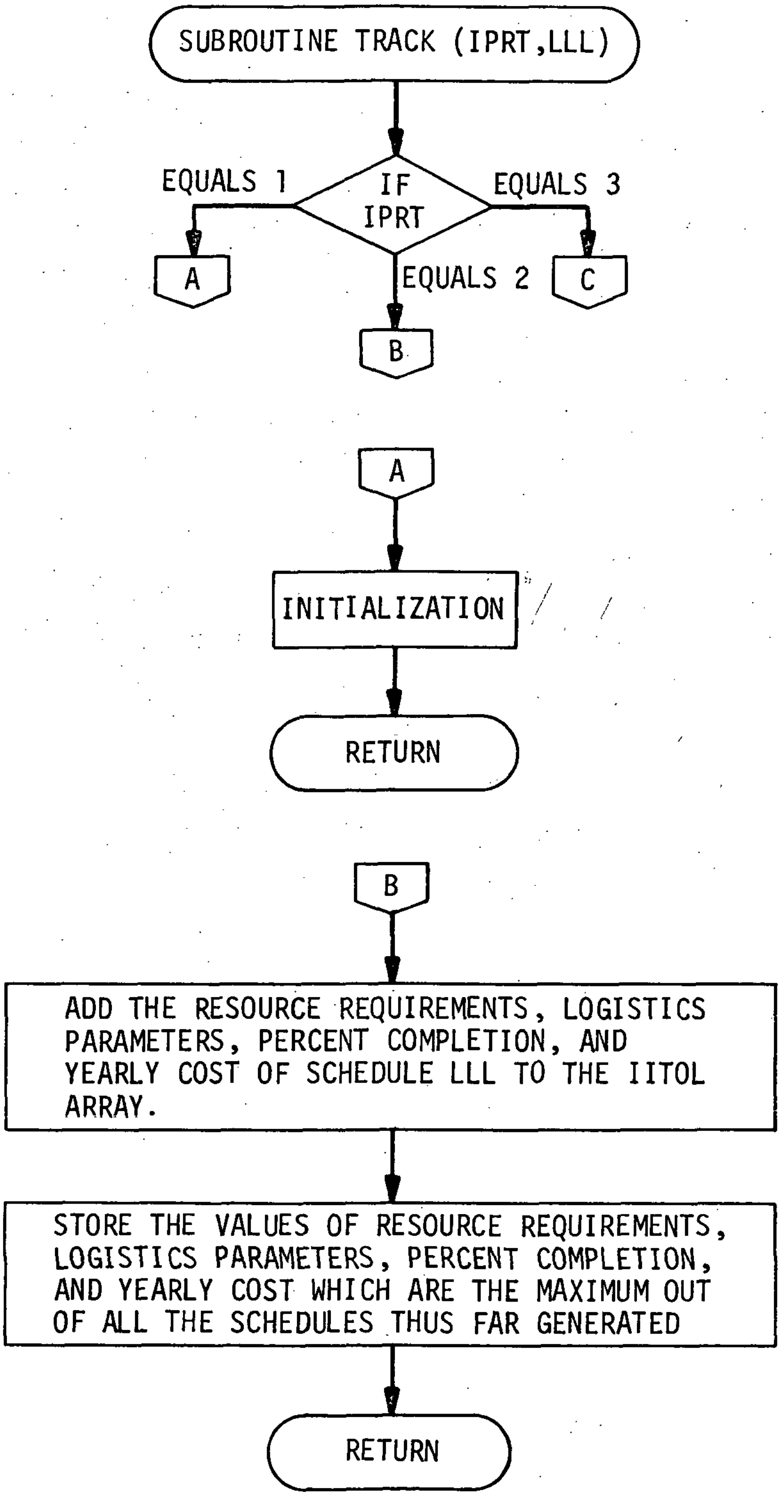
The array IFLCK contains the experiment numbers which conflict with the experiment being scheduled (NEXP). Previous scheduling is packed into ICODE under the appropriate experiment numbers (see subroutine ENCODE).

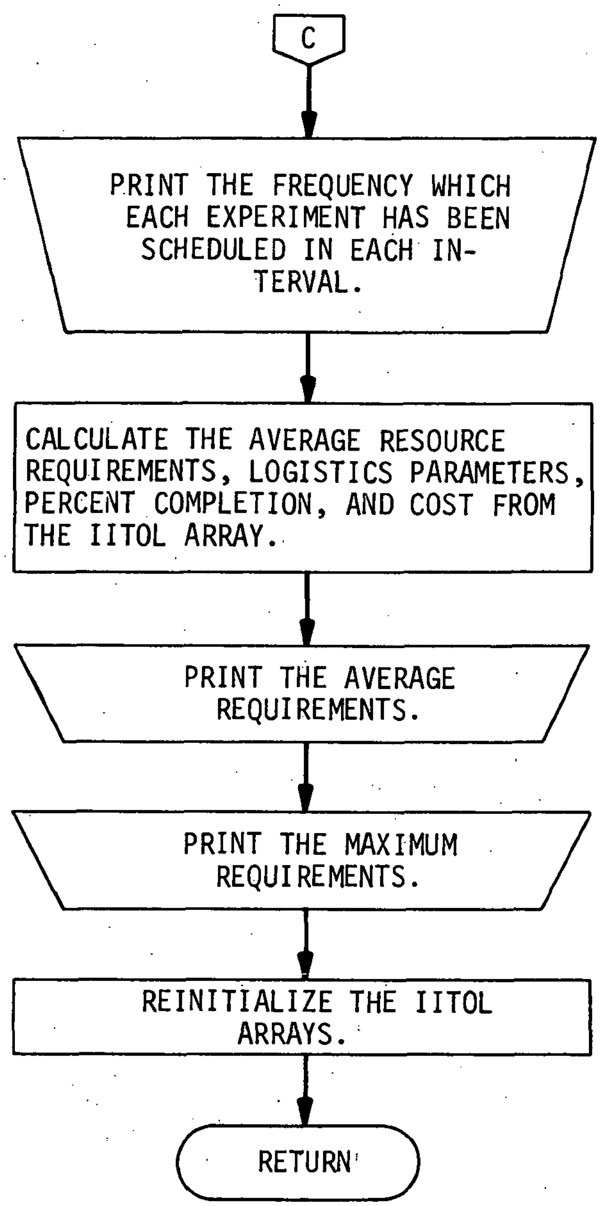
The check for conflicts is made by testing for the presence of a (1) in the digit of the ICODE array which represents the N^{th} interval for each experiment which might conflict.

2.9 SUBROUTINE TRACK

Subroutine TRACK tests the values of each parameter in each interval after the completion of each schedule in order to retain their maximum values. A running total of each parameter is also kept for each interval in order to calculate their mean values after all schedules have been generated. The number of times that each experiment has been scheduled in each interval is summed in the IIOUT array. Subroutine TRACK also prints the IIOUT array and the maximum and mean values of each parameter after a specified number of schedules have been generated.

BLOCK LOGIC FOR SUBROUTINE TRACK




C
PRINT THE FREQUENCY WHICH EACH EXPERIMENT HAS BEEN SCHEDULED IN EACH INTERVAL.
CALCULATE THE AVERAGE RESOURCE REQUIREMENTS, LOGISTICS PARAMETERS, PERCENT COMPLETION, AND COST FROM THE IITOL ARRAY.
PRINT THE AVERAGE REQUIREMENTS.
PRINT THE MAXIMUM REQUIREMENTS.
REINITIALIZE THE IITOL ARRAYS.
RETURN

2.10 SUBROUTINE OUTPU

This subroutine prints the first ISEN2 individual schedules. ISEN2 is input by the user. The printed output consists of the following:

- The intervals in which each experiment has been scheduled
- The amount of each of the four constrainted resources which was used in each interval
- The amount of the subcatagories of the fourth constrained resource which was used in each interval
- The experiment weight and volume to and from orbit required in each interval
- The logistics weight and volume to and from orbit required in each interval
- The amount of space station storage volume required in each interval
- The percentage of the experiment program thus far completed in each interval.

A detailed description and sample of output appears in subsection 4.3.

Section III
INPUT DESCRIPTION

3.1 GENERAL

The data input of REPRI is divided into five major categories of cards. These are:

- Run description cards
- Case description card
- Experiment description cards
- Resource description cards
- Cost description cards.

An input stream for a run with two cases is shown below. These categories will be discussed in detail in subsections 3.2 through 3.6.

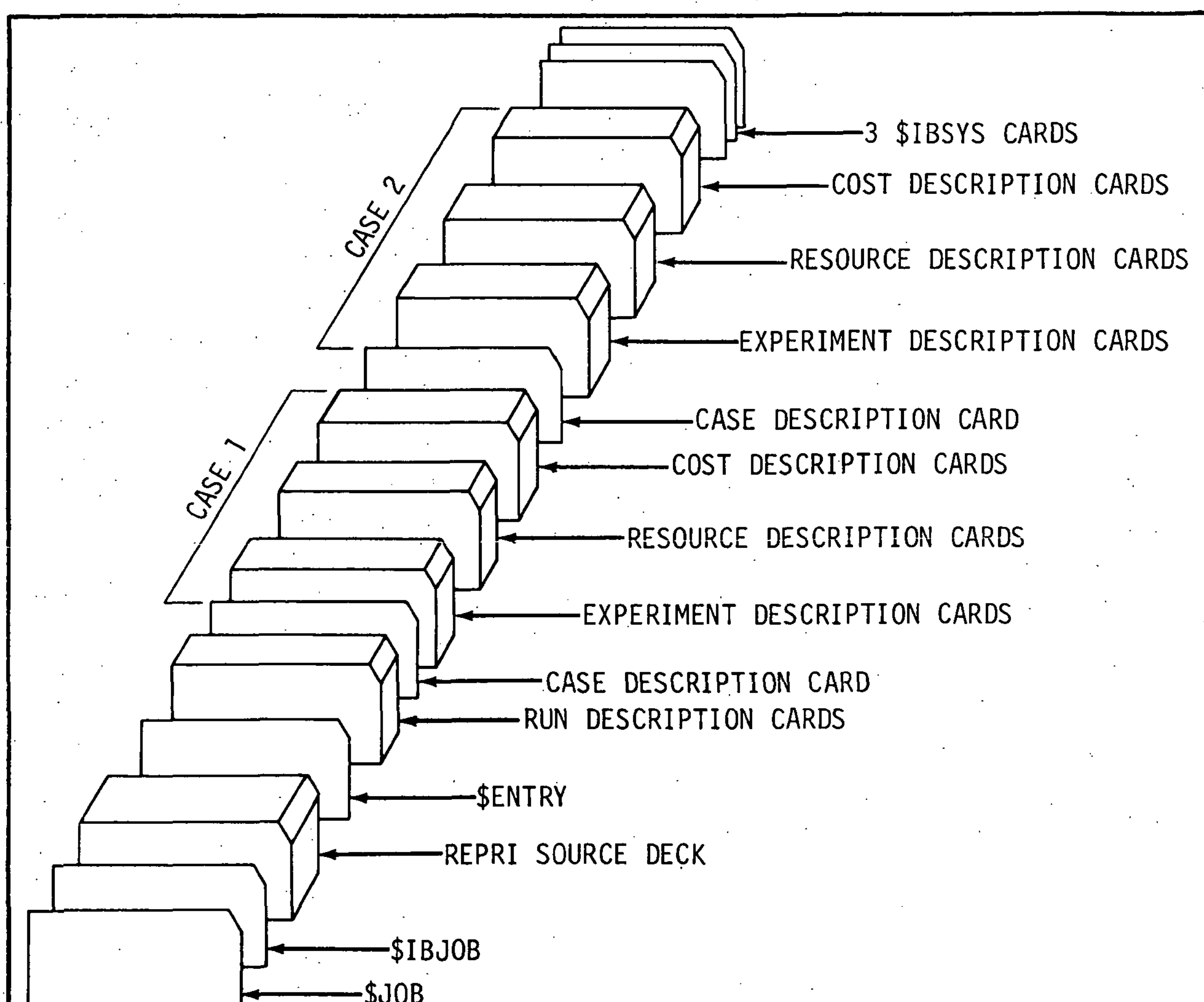


Figure 3-1. TYPICAL DECK SETUP

3.2 RUN DESCRIPTION CARDS

The following cards are input once per run:

CARD	COLUMN	VARIABLE	FORMAT	DESCRIPTION
1	1-3	ISEN2	I3	The number of individual schedules to be printed. Schedule one through schedule ISEN2 will be printed for each case.
	4-6	ICASE	I3	The number of cases to be run.
2	1-50	(LAVG(I),I=1,10)	10I5	The number of schedules after which an intermediate summary is desired. Up to 10 can be requested. The numbers must be input in ascending order. A final summary will always be printed regardless of the values in LAVG.

3.3 CASE DESCRIPTION CARD

This card is the first card required for each case.

CARD	COLUMN	VARIABLE	FORMAT	DESCRIPTION
1	1-5	N	I5	Number of experiments ($N\leq 36$)
	6-10	IN	I5	Number of intervals ($IN\leq 120$)
	11-15	LL	I5	Number of schedules to be generated
	16-20	NSKL	I5	Number of subcatagories of the fourth resource considered ($NSKL\leq 15$)
	21-30	IRAND	I10	A six-digit odd integer which is used to initialize the random number generator.

3.4 EXPERIMENT DESCRIPTION CARDS

The following two cards are input once for each experiment. Hence, 2*N cards are required.

CARD	COLUMN	VARIABLE	FORMAT	DESCRIPTION
1	1-5	MEXP(I)	I5	The four digit integer WXXX where W is the mode of accomodation; W=0 for integral modules, W=1 for attached modules, or W=2 for free-flying modules. XXX is the experiment identification number of the I^{th} experiment.
	6-10	IDATE(I)	I5	The interval in which the I^{th} experiment must start. If IDATE(I)=0, the experiment can start in any interval equal to or greater than JDATE(I).
	11-15	JDATE(I)	I5	The earliest interval in which the I^{th} experiment may start. If JDATE(I)=0, the experiment can start in any interval.
	16-20	IREPT(I)	I5	The number of required performances of the I^{th} experiment.
	21-25	IDATA(I,1)	I5	The amount of the first resource required to perform the I^{th} experiment during one interval, e.g., average watts.
	26-30	IDATA(I,2)	I5	The amount of the second resource required to perform the I^{th} experiment during one interval, e.g., kilowatts.
	31-35	IDATA(I,3)	I5	The amount of the third resource required to perform the I^{th} experiment during one interval, e.g., bit rate.
	36-40	IDATA(I,20)	I5	The supply weight required in one interval to perform the I^{th} experiment.

CARD	COLUMN	VARIABLE	FORMAT	DESCRIPTION
	41-45	IDATA(I,21)	I5	The supply volume required in one interval to perform the I^{th} experiment.
	46-50	IDATA(I,22)	I5	The return weight required in one interval to perform the I^{th} experiment.
	51-55	IDATA(I,23)	I5	The return volume required in one interval to perform the I^{th} experiment.
	56-60	IDATA(I,24)	I5	The weight of the I^{th} experiment equipment.
	61-65	IDATA(I,25)	I5	The volume of the I^{th} experiment equipment.
2	1-5	NREQD(I,1)	I5	The number of different subcategories of the fourth resource that the I^{th} experiment requires. NREQD(I,1)$\leq$5.
	6-10 16-20 26-30 36-40 46-50	NREQD(I,n)	I5	Subcategory number required by the I^{th} experiment (n=2,4,6,8,10). NREQD(I,n)$\leq$15 Only five subcategories may be used for each experiment.
	11-15 21-25 31-35 41-45 51-55	NREQD(I,n+1)	I5	The amount of resource subcategory NREQD(I,n) required to perform the I^{th} experiment.
	56-60 61-65 66-70	JCON(I,1) JCON(I,2) JCON(I,3)	I5	The identification number (MEXP(n)) of the n^{th} experiment that is mutually exclusive with the I^{th} experiment. Up to three conflicts can be specified for each experiment. If conflict with artificial-gravity experiment is desired, a 37 is input in JCON(I,3).

3.5 RESOURCE DESCRIPTION CARDS

The following cards are input once per case:

CARD	COLUMN	VARIABLE	FORMAT	DESCRIPTION
1	1-8	ANAME(J),BNAME(J)	2A4	Eight alphameric figures describing the J^{th} resource considered.
2	1-80	ICTRT(I,J)	10I8	The amount of the J^{th} resource which is available in the I^{th} interval. The values are input in 10 intervals per card, I=1, IN.
3-8	Repeat the preceding two cards for J=1,4:			
9-10	1-60	ICODE(I,37),I=1,12	6I10	An array specifying the intervals that the experiments for which JCON(n,3)=37 cannot be scheduled. One digit represents one interval. There are ten digits in a word. An integer array of up to 12 words are input with six words per card. A one prevents scheduling, and a zero permits scheduling. This option can be used to model the artificial gravity experiment. Ones are input for the intervals in which the artificial gravity experiment will be performed, and JCON(n,3)=37 for the experiments which cannot be performed during the artificial gravity experimentation. Two cards must be input even if less than 60 intervals are used.
NOTE:	The 60 digits on each card correspond to 60 intervals.			
11-12	1-8,9-16, etc.	ANAME(J), BNAME(J), J=1,NSKL	20A4	NSKL sets of eight figures of alphameric description of the subcategories of the fourth resource. NSKL $\leq$ 15. If NSKL $\leq$10, card 12 is not needed.
13	1-8	ANAME(26), BNAME(26)	2A4	Eight figures of alphameric description of the space station storage volume.

CARD	COLUMN	VARIABLE	FORMAT	DESCRIPTION
14-	1-8,9-16, etc.	ICTRT(I,5)	10I8	The amount of space station storage volume available in the I^{th} interval. The values are input in ten intervals per card (I=1,IN). The number of cards required is ((IN+9)/10) truncated.

3.6 COST DISTRIBUTION CARDS

CARD	COLUMN	VARIABLE	FORMAT	DESCRIPTION
1	2-10	IYR	I9	An integer specifying the number of intervals in one year.
2-(n+1)	1-8,9-16, etc.	XCPST(I,J)	10F8.1	The cost of developing the J^{th} experiment in the $(9-I)^{th}$ year prior to its first scheduled performance, I=1,8. This card is input once for each experiment. Cost cards must be in the same order as experiment cards.
(n+2)	1-8,9-16, etc.	FFD(I)	10F8.1	The cost of developing the free-flying modules in the $(6-I)^{th}$ year prior to their first use, I=1,5.
n+3	1-8,9-16, etc.	AMD(I)	10F8.1	The cost of developing the attached modules in the $(6-I)^{th}$ year prior to their first use, I=1,5.

Section IV
OUTPUT DESCRIPTION

4.1 GENERAL

The output falls in three categories. These are input, individual schedules, and a summary of all the scheduling. Each of these will be described in the following. A typical example of each piece of output has been included in this section.

4.2 OUTPUT DESCRIBING INPUT

The first two lines of output that appear (see next page) are the card images of the two run description cards. The information printed is described in Section 3.1. The cards are printed card 2 and then card 1.

The logistics requirements table (see next page) is printed next. This includes all the logistics information input for each experiment. Weight and volume are non-dimensional in the program. The weights and volumes shown in the typical output are pounds and cubic feet. The mode of accommodation (free flyer (FF), attached module (AM), and integral module (I)) is also indicated in this table for reference purposes.

The correspondence between output and input variables is shown below for the I^{th} experiment:

OUTPUT COLUMN	INPUT VARIABLE
EXP WT	IDATA (I,24)
EXP VOL	IDATA (I,25)
SUPPLY WT	IDATA (I,20)
SUPPLY VOL	IDATA (I,21)
RETURN WT	IDATA (I,22)
RETURN VOL	IDATA (I,23)

-0 -0 -0 -0 -0 -0 -0 -0 -0 -0
34 20 200 9 525811

LOGISTICS REQUIREMENTS

	EXP WT	EXP VOL	SUPPLY WT	SUPPLY VOL	RETURN WT	RETURN VOL
FPE 10 FF	23244	8125	72	6	24	0
FPE 20 FF	24388	10598	222	6	222	6
FPE 30 FF	22695	10598	510	12	492	12
FPE 40 AM	18048	6888	30	0	30	0
FPE 50 FF	31828	10598	726	12	138	6
FPE 60 I	157	8	0	0	48	0
FPE 71 I	312	11	108	6	6	0
FPE 72 I	214	13	114	6	12	0
FPE 73 I	321	5	108	0	1	0
FPE 74 I	838	21	114	0	12	0
FPE 75 I	161	5	108	0	12	0
FPE 80 FF	40424	8125	1158	24	588	18
FPE 91 AM	16805	505	2760	120	60	6
FPE 92 AM	15115	10598	2760	114	54	0
FPE 100 I	912	104	78	6	18	0
FPE 110 AM	51001	5980	1806	480	15560	422
FPE 120 FF	7275	6888	5772	90	30	0
FPE 131 I	1450	319	72	6	66	6
FPE 132 AM	13800	9500	78	0	66	0
FPE 140 I	587	31	138	6	138	6
FPE 150 I	2453	103	72	12	36	0
FPE 160 I	1566	103	390	24	288	12
FPE 170 I	1895	49	480	6	480	6
FPE 180 I	670	42	282	6	162	6
FPE 200 AM	21916	6888	84	6	528	18
FPE 210 AM	19607	6888	156	6	6	0
FPE 220 I	7420	120	18	0	18	0
FPE 244 I	2050	316	48	6	156	6
FPE 245 I	570	60	48	0	156	0
FPE 246 I	3170	131	48	0	156	6
FPE 247 I	330	19	48	0	156	0
FPE 250 I	361	37	66	6	60	0
FPE 260 I	564	42	108	6	108	6
FPE 270 I	1287	53	828	30	192	24

A table of scheduling requirements is output next. This includes all scheduling related parameters which were input for each experiment. The table shown on the next page is the typical output from a 20 interval case. The correspondence between the output and the input variables is shown below:

OUTPUT COLUMN	INPUT VARIABLE
FORCED START INTERVAL	IDATE(I)
MINIMUM START INTERVAL	JDATE(I)
NUMBER OF PERFORMANCES	IREPT(I)
CONFLICTS 1	JCON(I,1)
2	JCON(I,2)
3	JCON(I,3)

The output below the table shows the alphameric titles and constrained values input for each resource. These are explained in subsection 3.10 under cards 1-8. The constrained values for each interval are printed 10 intervals to the row with interval 1-10 from left to right on the first row and subsequent intervals following.

The artificial-gravity experiment timeline which is input is shown under the title "ARTIFICIAL G" in the output. Time is the card image of cards 9 and 10 of the resource description cards described in subsection 3.5. In the attached example there was no artificial gravity experiment scheduled. Blanks indicate the same meaning as a zero. If artificial gravity had been scheduled in the fifth interval, the first word would have been "bbbb100000" instead of "bbbbbbbbb0".

SCHEDULING REQUIREMENTS

	FORCED START INTERVAL	MINIMUM START INTERVAL	NUMBER OF PERFORMANCES	CONFLICTS 1	2	3
FPE 10 FF		4	20			
FPE 20 FF		4	20			
FPE 30 FF	4	4	20			
FPE 40 AM		4	4	1210		
FPE 50 FF		4	4			
FPE 60 I	3	3	1			
FPE 71 I		1	3			
FPE 72 I		1	3			
FPE 73 I		1	8			
FPE 74 I		1	8			
FPE 75 I		1	8			
FPE 80 FF		4	20	1200		
FPE 91 AM		4	20			
FPE 92 AM		4	20			
FPE 100 I		1	20			
FPE 110 AM		4	20			
FPE 120 FF		4	4			
FPE 131 I		1	20			
FPE 132 AM		4	20			
FPE 140 I	1	1	7			
FPE 150 I		1	20			
FPE 160 I		1	4			
FPE 170 I		1	4			
FPE 180 I		1	4			
FPE 200 AM		4	10	2080		
FPE 210 AM		4	20	1040		
FPE 220 I	1	4	1			
FPE 244 I		1	4			
FPE 245 I	1	1	2			
FPE 246 I	1	1	2			
FPE 247 I	1	1	4			
FPE 250 I		1	20			
FPE 260 I		1	20			
FPE 270 I		1	4			

A WATTS
99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999
99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999
KW-HRS
99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999
99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999
BIT RT
99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999
99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999
MAN HRS
99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999
99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999 99999999
ARTIFICIAL G
0 0 -0 -0 -0 -0
-0 -0 -0 -0 -0 -0

The resource requirements input for each experiment are tabularized next (shown on next page). The correspondence between the output and the input variables is shown below:

OUTPUT COLUMN	INPUT VARIABLE
A WATTS	IDATA(I,1)
KW-HRS	IDATA(I,2)
BIT-RT	IDATA(I,3)
MAN HRS	IDATA(I,4)

The program is non-dimensional in regard to resources. The dimension of the variables shown in the typical output are watts, kilowatt hours, bits/day x 10^{+7}, and manhours/day. Manhours/day is calculated internally as the sum of the skill hours/day.

RESOURCE REQUIREMENTS

	A WATTS	KW-HRS	BIT RT	MAN HRS
FPE 2010	25	0	99	876
FPE 2020	37	0	3660	246
FPE 2030	50	0	25600	480
FPE 1040	1120	4326	25	252
FPE 2050	50	0	45	480
~~FPE 60~~	~~31~~	~~96~~	~~17~~	~~102~~
FPE 71	0	24	3	1002
FPE 72	1	6	2	150
FPE 73	81	1530	2	132
FPE 74	150	84	2	144
FPE 75	150	0	2	174
~~FPE 2080~~	~~56~~	~~0~~	~~864~~	~~336~~
FPE 1091	2559	10692	5	1248
FPE 1092	1000	0	5	18
FPE 100	433	504	14	576
FPE 1110	1780	4638	49000	1380
FPE 2120	17	0	53	132
~~FPE 131~~	~~290~~	~~228~~	~~66~~	~~4266~~
FPE 1132	1155	4320	78	5010
FPE 140	157	150	0	1224
FPE 150	2000	14256	0	2106
FPE 160	149	1542	7	458
FPE 170	80	215	1	744
~~FPE 180~~	~~22~~	~~84~~	~~0~~	~~276~~
FPE 1200	20	4704	1	432
FPE 1210	1000	4320	161	204
FPE 220	568	6396	3	1326
FPE 244	5	24	0	378
FPE 245	0	18	0	114
~~FPE 246~~	~~1~~	~~0~~	~~0~~	~~120~~
FPE 247	75	0	0	144
FPE 250	43	258	0	204
FPE 260	81	408	0	246
FPE 270	45	132	4	390

The tables on the following pages show the skill types and manhours of each required for each experiment. It should be noted that the skill names may be changed from "SKILL 1", "SKILL 2", etc., simply by changing the cards starting at card 11 in subsection 3.5. The correspondence between the output and the input variables is explained in card 2 of subsection 3.4.

SKILL TYPE REQUIREMENTS

		SKILL 1	SKILL 2	SKILL 3	SKILL 4	SKILL 5
FPE	10	438				
FPE	20	144				
FPE	30	288				
FPE	40	192				
FPE	50	275				
FPE	60					
FPE	71					
FPE	72	60				
FPE	73	90				
FPE	74	102				
FPE	75	30				
FPE	80	234				
FPE	91		522	725		
FPE	92			18		
FPE	100		396			
FPE	110					
FPE	120					
FPE	131				1295	1134
FPE	132				450	672
FPE	140					306
FPE	150					
FPE	160	318				
FPE	170					
FPE	180					
FPE	200					
FPE	210	174				
FPE	220	480				
FPE	244					
FPE	245					
FPE	246	30				
FPE	247				42	60
FPE	250		174			
FPE	260			204		
FPE	270	348				

SKILL TYPE REQUIREMENTS

		SKILL 6	SKILL 7	SKILL 8	SKILL 9
FPE	10		438		
FPE	20			102	
FPE	30		30	162	
FPE	40			60	
FPE	50		102	102	
FPE	60			102	
FPE	71		102	900	
FPE	72		30	60	
FPE	73		42		
FPE	74		42		
FPE	75		144		
FPE	80		42	60	
FPE	91				
FPE	92				
FPE	100		180		
FPE	110		114	1266	
FPE	120		60	72	
FPE	131				1836
FPE	132	1602	90		2196
FPE	140		366	12	540
FPE	150		702	702	702
FPE	160		60	90	
FPE	170		612	132	
FPE	180		144	132	
FPE	200		318	114	
FPE	210			30	
FPE	220		846		
FPE	244	42	144	192	
FPE	245			114	
FPE	246		30	60	
FPE	247		42		
FPE	250			30	
FPE	260			42	
FPE	270			42	

The next information printed (shown on next page) is headed by a single number which corresponds to IYR described in subsection 3.6. The subsequent lines of print give the FPE number and the associated cost for each year of development for up to eight years. Note that if an experiment required only five years of development, then the first three years are zero. The two lines of print below the experiment costs are the development cost of the free flying and attached modules respectively for a five year development. This is discussed in subsection 3.6.

2

2010	-0.0	-0.0	-0.0	4500.0	18000.0	27000.0	27500.0	18500.0
2020	-0.0	7500.0	45000.0	45000.0	50000.0	52500.0	57500.0	27500.0
2030	-0.0	6000.0	35000.0	5070.0	5620.0	5940.0	15940.0	22420.0
1040	-0.0	-0.0	-0.0	-0.0	4500.0	15000.0	20500.0	14500.0
2050	-0.0	-0.0	-0.0	-0.0	17110.0	21270.0	21170.0	13410.0
60	-0.0	-0.0	-0.0	-0.0	-0.0	1500.0	3000.0	3000.0
71	-0.0	-0.0	-0.0	-0.0	-0.0	230.0	750.0	750.0
72	-0.0	-0.0	-0.0	-0.0	-0.0	230.0	1130.0	1130.0
73	-0.0	-0.0	-0.0	-0.0	380.0	1500.0	1500.0	1500.0
74	-0.0	-0.0	-0.0	-0.0	190.0	1130.0	1130.0	1130.0
75	-0.0	-0.0	-0.0	-0.0	190.0	1130.0	1130.0	1130.0
2080	-0.0	-0.0	4950.0	12900.0	19370.0	19370.0	14880.0	21830.0
1091	-0.0	-0.0	-0.0	2540.0	3140.0	3750.0	6220.0	3820.0
1092	-0.0	-0.0	-0.0	290.0	440.0	2250.0	9250.0	9250.0
100	-0.0	-0.0	-0.0	900.0	1350.0	1550.0	1880.0	1170.0
1110	-0.0	-0.0	3000.0	4500.0	4500.0	15000.0	28000.0	28000.0
2120	-0.0	-0.0	-0.0	-0.0	210.0	2080.0	5110.0	4110.0
131	-0.0	-0.0	-0.0	650.0	5550.0	19080.0	29600.0	19080.0
1132	-0.0	-0.0	-0.0	290.0	2890.0	8400.0	18520.0	13900.0
140	-0.0	-0.0	9000.0	19500.0	22500.0	27000.0	24000.0	6000.0
150	-0.0	-0.0	9000.0	22500.0	30000.0	35000.0	36000.0	9000.0
160	-0.0	900.0	3750.0	4500.0	4500.0	9000.0	9000.0	6000.0
170	-0.0	-0.0	450.0	900.0	1500.0	3000.0	2700.0	750.0
180	-0.0	-0.0	900.0	1500.0	3000.0	4500.0	3000.0	1500.0
1200	-0.0	500.0	1350.0	1950.0	4500.0	4500.0	11500.0	7000.0
1210	-0.0	-0.0	-0.0	-0.0	1500.0	3250.0	20500.0	10750.0
220	-0.0	-0.0	500.0	2250.0	3000.0	7500.0	6000.0	3000.0
244	-0.0	-0.0	-0.0	5570.0	8640.0	5570.0	1910.0	190.0
245	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	290.0
246	-0.0	-0.0	-0.0	-0.0	-0.0	1570.0	1250.0	140.0
247	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0
250	-0.0	-0.0	150.0	1100.0	1770.0	2530.0	3300.0	3150.0
260	-0.0	-0.0	-0.0	2540.0	3140.0	3750.0	6220.0	3820.0
270	-0.0	-0.0	-0.0	-0.0	1500.0	1500.0	9000.0	9000.0
51900.0	97900.0	58500.0	32400.0	9200.0				
18310.0	27020.0	20350.0	9550.0	2750.0				

4.3 INDIVIDUAL SCHEDULE OUTPUT

A typical schedule is shown on the next three pages. The experiments (FPE's) are shown from top to bottom in the order in which they were scheduled. The "1" or "0" in the schedule matrix indicates scheduled or not schedule respectively. The performances desired and actually scheduled for each experiment are shown in the columns on the right side along with the mode of accommodation.

The items shown on the summary of resources are

A WATTS	average watts summed over all FPE's active (watts)
KW HRS	kilowatt hours required (kilowatt hours)
BIT RT	bits per day generated by FPE's (BPD x 10^{+7})
MAN HRS	manhours required during interval (manhours)
SKILL 1, etc	skill hours required during each interval
S WT UP	supply weight required to orbit during interval (pounds)
S VOL UP	supply volume required to orbit during interval (cubic feet)
RET WT	weight to be returned from orbit (pounds)
RET VOL	volume to be returned from orbit (cubic feet)
E WT UP	experiment weight to be launched to orbit (pounds)
E VOL UP	experiment volume to be launched to orbit (cubic feet)
SS VOL	space station volume required (cubic feet)
PC COMPL	cumulative percent of performance completed (%)
COST PER YEAR	year, cost per year, and cumulative cost from top to bottom. (Cost in \$ x 10^6.) The units of cost on output are the units of cost used to input divided by 10^3.

SCHEDULE INTERVAL	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	PERFORMANCES REQ	PERFORMANCES SCHD	MODE OF ACCOMMODATION
FPE 100	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	20	I
FPE 244	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	4	4	I
FPE 210	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	17	AM
FPE 200	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	10	10	AM
FPE 220	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	I
FPE 245	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	I
FPE 80	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	20	7	FF
FPE 40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	-0	AM
FPE 10	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	17	EF
FPE 247	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	I
FPE 110	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	17	AM
FPE 72	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	3	I
FPE 73	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	8	8	I
FPE 132	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	17	AM
FPE 180	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	4	4	I
FPE 131	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	20	I

FPE 150	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1
FPE 250	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4	4	1
FPE 30	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	20	1
FPE 246	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	17	FF
FPE 150	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	1
FPE 92	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	20	1
FPE 120	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	20	17	AM
FPE 270	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	FF
FPE 75	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	4	4	1
FPE 170	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	8	8	1
FPE 20	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4	4	1
FPE 71	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	20	17	FF
FPE 50	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	3	3	1
FPE 140	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	FF
FPE 74	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	7	7	1
FPE 91	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	8	8	1
FPE 250	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	17	AM
																					20	20	1

SUMMARY OF RESOURCES

INTERVAL	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
K WATTS	3556	5090	3121	10797	10852	10551	10551	10783	10855	10958	10744	10812	10093	10595	10871	10854	10700	10559	10559	10559
KW-HRS	22216	15522	15906	39912	40125	40122	40122	41382	41250	41250	45038	44495	45026	46025	45942	45966	45906	45906	45882	45882
SIT RT	83	50	70	79560	79607	79505	79505	79508	79553	79565	78702	78748	78750	78759	78750	78706	78698	78696	78693	78693
MAN HRS	10325	9000	9018	17104	20184	20054	20034	17883	17893	17982	18304	18016	18150	18150	17874	18744	18978	18804	17802	17802
SKILL 1	310	30	30	1565	1702	1902	1902	1872	1578	1726	1474	1175	1265	1255	1256	1266	1164	1134	1134	1134
SKILL 2	570	570	570	1092	1072	1092	1072	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092
SKILL 3	204	204	204	948	948	948	948	948	948	948	948	948	948	948	948	948	948	948	948	948
SKILL 4	1338	1338	1335	1758	1745	1745	1745	1745	1745	1745	1745	1745	1745	1746	1745	1746	1746	1746	1745	1745
SKILL 5	1500	1500	1500	2172	2112	2112	2112	1805	1805	1805	1806	1805	1806	1805	1805	1806	1806	1806	1305	1805
SKILL 6	0	0	0	1632	1632	1602	1632	1602	1602	1602	1602	1602	1602	1602	1602	1602	1644	1644	1544	1644
SKILL 7	2165	1320	1320	2054	2705	2575	2675	2370	1578	1342	2252	2252	2304	2304	2150	2202	2304	2160	2058	2058
SKILL 8	950	950	948	2530	2742	2582	2582	2715	2464	2464	2570	2552	2652	2652	2520	3348	3540	3540	2540	2540
SKILL 9	3078	3078	3078	5274	5274	5274	5274	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734
S WT UP	595	576	546	11045	12254	12090	12095	11514	10422	10530	9738	15120	15228	15226	14946	9282	9215	9108	9000	9000
S VOL UP	42	42	45	845	854	958	958	945	829	826	915	882	882	582	575	792	798	798	792	792
RET WT	8332	4734	799	19185	19534	19338	21182	32769	13850	30096	20340	16515	16517	19187	25631	19169	18635	18775	18457	235140
RET VOL	144	215	25	525	531	518	532	11147	458	5513	547	432	482	524	7364	497	487	493	482	71088
L WT UP	17917	0	37122	3557	33723	0	0	1555	538	151	22565	7276	321	0	0	312	2050	0	0	0
E VOL UP	755	0	21	73979	10647	0	0	133	21	5	5930	5888	5	0	0	11	316	0	0	0
SS VOL	1358	856	724	1557	1615	1596	1576	1533	1557	1552	1572	1555	1560	1565	1512	1437	1740	1735	1718	1718
PC COMPL	3	5	3	12	15	22	27	32	35	41	45	51	56	51	66	71	76	80	35	89

COST PER YEAR (MILLION DOLLARS)

1970	1971	1972	1973	1974	1975	1976	1977	1978	1979
-0.00	25.50	100.65	191.39	302.54	472.22	322.95	252.10	121.54	26.28
-0.00	25.50	125.15	316.55	520.57	992.82	1315.71	1567.32	1688.85	1709.14

1980	1981	1982	1983	1984	1985	1986	1987	TOTAL
30.42	23.93	9.62	2.66	0.56	0.00	0.00	0.00	1775.34
1739.57	1763.50	1773.11	1775.77	1775.34	1775.34	1775.34	1775.34	

4.4 OUTPUT SUMMARIZING SCHEDULE FOR ENTIRE CASE

A sample of the case summary printed for all scheduling is shown on the next four pages. The number of times each experiment was scheduled in each interval is shown in the frequency distribution of scheduling. The average number of performances scheduled is shown on the right.

The average and maximum values utilized in each interval for the case run are summarized in a format similar to that for the individual schedule.

FREQUENCY DISTRIBUTION OF SCHEDULING FOR 200 SCHEDULES

INTERVAL		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	PERFORMANCES REQ	SCHD
FPE	10	0	0	0	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	20	17
FPE	20	0	0	0	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	20	17
FPE	30	0	0	0	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	20	17
FPE	40	0	0	0	5	10	18	25	24	26	25	22	24	27	22	24	21	33	31	24	21	4	2
FPE	50	0	0	0	14	23	34	48	45	53	50	47	47	42	47	50	49	82	59	55	45	4	4
FPE	60	0	0	200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
FPE	71	10	17	31	39	43	34	24	19	22	25	33	33	31	28	34	34	35	46	36	24	3	3
FPE	72	10	17	27	25	35	39	36	29	20	19	23	33	39	35	32	30	34	49	40	27	3	3
FPE	73	5	14	25	38	41	54	55	71	77	78	76	50	153	145	134	122	113	108	98	82	8	8
FPE	74	4	10	20	29	44	52	50	69	82	87	83	86	156	148	140	131	114	103	97	85	8	8
FPE	75	9	15	24	32	39	49	52	72	74	75	74	82	151	151	138	128	117	108	102	86	8	8
FPE	80	0	0	0	173	170	170	157	151	157	155	96	95	95	104	108	112	119	123	123	123	20	11
FPE	91	0	0	0	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	20	17
FPE	92	0	0	0	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	20	17
FPE	100	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	20	20
FPE	110	0	0	0	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	20	17
																						4	4

FPE 120	0	0	0	17	24	43	57	50	52	49	43	45	50	41	40	35	74	67	58	49	20	20
FPE 131	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	20	17
FPE 132	0	0	0	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	7	7
FPE 140	200	200	200	200	200	200	200	0	0	0	0	0	0	0	0	0	0	0	0	0	20	20
FPE 150	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	4	4
FPE 160	10	18	29	40	39	45	48	47	46	42	39	37	38	35	35	34	57	59	49	42	4	4
FPE 170	12	24	35	43	47	45	42	38	38	35	40	45	47	43	35	29	56	51	47	40	4	4
FPE 180	10	16	28	35	34	39	40	42	41	35	34	35	35	42	36	31	60	67	52	56	10	5
FPE 200	0	0	0	8	15	19	29	39	43	45	104	104	104	95	59	55	75	65	61	59	20	13
FPE 210	0	0	0	155	154	153	151	155	155	158	164	153	152	153	153	152	145	145	145	148	1	1
FPE 220	200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4
FPE 244	3	17	25	37	41	46	47	45	42	39	31	39	41	39	41	37	68	59	53	42	2	2
FPE 245	200	200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2
FPE 246	200	200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4
FPE 247	200	200	200	200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	20
FPE 250	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	20	20
FPE 260	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	4	4
FPE 270	7	24	29	42	47	47	51	51	46	36	33	27	30	33	42	45	70	58	45	33		

AVERAGE RESOURCES UTILIZED IN 200 SCHEDULES

INTERVAL	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
A WATTS	3685	3146	3210	10687	10645	10714	10775	10834	10553	10661	10655	10671	10827	10785	10775	10734	10755	10709	10549	10590
KW-HRS	22459	15125	16409	39857	40040	40472	40965	41145	41301	41350	42773	42815	43503	43102	42882	42644	42556	42117	41741	41427
BIT RT	84	82	100	79413	79405	79413	79405	79383	79357	79359	79107	79108	79109	79142	79150	79174	79212	79224	79218	79211
MAN HRS	10508	9361	9445	19305	19255	19322	19354	18111	18139	18113	18145	18192	18365	18311	18297	18232	18662	18552	18342	18107
SKILL 1	547	119	130	1414	1445	1491	1533	1527	1533	1511	1425	1423	1507	1507	1521	1515	1648	1593	1519	1450
SKILL 2	570	570	570	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092
SKILL 3	234	234	234	948	948	948	948	948	948	948	948	948	948	948	948	948	948	948	948	948
SKILL 4	1338	1338	1338	1788	1745	1745	1745	1745	1746	1745	1746	1745	1746	1745	1745	1746	1746	1746	1745	1746
SKILL 5	1500	1500	1500	2172	2112	2112	2112	1806	1806	1805	1805	1805	1806	1805	1805	1806	1806	1806	1805	1806
SKILL 6	2	4	5	1510	1611	1512	1512	1511	1511	1610	1507	1510	1611	1610	1611	1610	1616	1614	1613	1611
SKILL 7	2434	1450	1490	2290	2281	2313	2338	1970	1989	1961	2054	2096	2190	2155	2108	2065	2217	2156	2102	2030
SKILL 8	1035	1097	1124	2712	2745	2735	2739	2556	2580	2565	2721	2736	2732	2712	2732	2716	2855	2863	2782	2689
SKILL 9	3078	3078	3078	5274	5274	5274	5274	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734
S WT UP	510	737	952	10975	11183	11307	12330	11892	11940	11785	11270	11422	11600	11387	11363	11258	12791	12460	12035	11608
S VOL UP	46	50	54	631	655	840	854	843	843	839	830	832	833	830	831	830	660	853	844	835
REF WT	8398	4870	10107	20380	19250	20015	24543	22284	23209	34592	21837	23024	22666	22853	22636	25382	22851	22721	21539250233	
REF VOL	145	221	35	727	556	717	2359	1545	2252	4488	1547	2059	2079	2021	2004	2912	1868	1946	1532	74051
E WT UP	13523	452		642222077	3324	4199	5112	4502	4552	3848	9995	3511	4382	4763	4799	3551	14315	875	196	0
E VOL UP	433	24	41	70145	1131	1532	1948	1573	1652	1558	3358	1380	1471	1454	1649	1203	5372	166	69	0
SS VOL	1039	952	805	1604	1599	1522	1544	1567	1553	1570	1545	1561	1578	1570	1572	1560	1678	1646	1614	1575
PC COMPL	3	5	8	13	17	22	27	32	36	41	45	50	55	60	64	69	74	79	84	88

AVERAGE COST PER YEAR

1970	1971	1972	1973	1974	1975	1976	1977	1978	1979
0.05	25.97	101.86	193.22	303.22	355.67	300.32	228.07	111.60	31.51
0.05	26.03	127.89	321.11	624.33	951.00	1231.32	1509.39	1620.98	1652.49

1980	1981	1982	1983	1984	1985	1986	1987	TOTAL
34.53	34.54	30.30	22.98	12.14	0.45	0.00	0.00	1787.43
1687.02	1721.56	1751.86	1774.84	1786.93	1787.43	1787.43	1787.43	

MAXIMUM VALUE OF RESOURCES FOR 200 SCHEDULES

INTERVAL	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
K WATTS	3957	3459	3519	11230	11205	11205	11211	11044	11239	11089	11084	11200	11157	11155	11173	11128	11272	11166	11148	11148
KW-HRS	23054	16816	17156	44594	46188	47534	47755	47515	47616	47516	47490	47700	47860	47940	47632	47700	47790	47622	47622	47615
BIT RT	92	90	112	79521	77518	79555	79556	79555	79555	79555	79557	79582	79664	79664	79653	79666	79671	79673	79661	79661
MAN HRS	11054	10396	11850	21545	21012	21090	21125	20195	20256	20054	20345	20345	20550	20304	20388	20214	20586	20418	20346	19944
SKILL 1	915	458	798	2075	2094	2075	2310	2310	2340	2322	2322	2124	2166	2442	2208	2472	2442	2400	2154	2094
SKILL 2	570	570	570	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092	1092
SKILL 3	204	204	234	948	948	948	948	948	948	948	948	948	948	948	948	948	948	948	948	948
SKILL 4	1338	1338	1338	1785	1745	1745	1746	1745	1745	1745	1745	1745	1745	1745	1746	1746	1746	1746	1746	1746
SKILL 5	1500	1500	1500	2172	2112	2112	2112	1805	1805	1805	1805	1805	1505	1505	1805	1806	1806	1806	1806	1805
SKILL 6	42	42	42	1644	1644	1544	1644	1644	1644	1544	1544	1544	1644	1644	1644	1544	1644	1644	1544	1644
SKILL 7	2922	2252	2232	3458	3240	3198	3240	2934	2814	2644	3060	3060	2945	3018	2958	2856	3078	3078	2934	2844
SKILL 8	1992	2142	2244	3834	3720	3785	3702	3840	3840	3762	3774	3774	3804	3752	3852	3780	3876	3876	3815	3708
SKILL 9	3978	3078	3078	5274	5274	5274	5274	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734	4734
S WT UP	1038	2258	2436	17532	17375	17412	18534	18270	18330	17466	17465	17220	17328	17316	17634	17634	17958	18072	17832	17832
S VOL UP	78	54	114	972	955	954	978	969	955	942	935	935	935	942	942	948	966	972	960	954
RET WT	8974	5558	1509	51614	39191	50455	59428	92415	90753	91751	52549	58393	72399	69923	57655	70652	74793	52875	57225	294662
RET VOL	158	251	74	5945	7599	8505	18035	17268	19205	17259	11402	17968	17974	17968	17968	18017	18105	11408	17956	89886
E WT UP	20958	5553	59502	25973	33723	53744	49875	34623	49875	32555	55744	39104	34716	72252	47700	47700	61192	40736	19607	0
E VOL UP	1529	430	487	61557	10651	17455	17486	13914	17491	13737	17802	17486	13781	16723	15013	17499	17857	8136	6888	0
SS VOL	1383	1354	1349	2152	2115	2045	1990	2127	2015	2017	2017	2022	2027	2064	2050	2050	2075	2011	2011	1963
PC COMPL	4	7	10	15	21	27	32	37	41	46	50	55	59	64	68	73	78	81	85	89

1970	1971	1972	1973	1974	1975	1976	1977	1978	1979
0.90	29.55	108.40	206.38	318.75	585.30	330.52	270.62	154.73	74.59
0.90	30.45	135.75	341.57	553.08	1030.77	1356.53	1593.54	1747.19	1791.23

1980	1981	1982	1983	1984	1985	1986	1987	TOTAL
23.73	67.31	61.55	56.58	43.55	11.31	0.00	0.00	1830.84
1815.22	1827.54	1830.84	1830.84	1830.84	1830.84	1830.84	1830.84	

Appendix A
STATISTICAL BACKGROUND FOR REPRI

REPRI was designed for Phase B compatibility analysis of Space Station mission concepts, sub-system capabilities, logistics support, and experiment packages. Since no one schedule could suffice to establish trends or sensitivities for the entire ten year program, REPRI was designed to generate a large sampling of feasible schedules. In order to avoid biasing of the schedules, random scheduling was utilized. Only truly random feasible schedules are statistically valid for identifying program trends, since any deterministic scheduling algorithm must schedule based on some reasoning which may induce bias.

Random scheduling as applied is simply a Monte Carlo simulation of the scheduling with all experiments having equal probabilities of being scheduled in any interval. The flexibility with which the Monte Carlo technique allows new parameters to be added makes it highly desirable. Many analysis techniques are modeled completely on the parameters to be analyzed. A possible disadvantage of the Monte Carlo technique might have been the amount of time required to generate the large samples necessary to gain acceptable confidence levels. This problem has been avoided by careful programming.

Briefly, the Monte Carlo technique as applied in the REPRI consists of:

- Randomly generating the order in which the events are to be scheduled.
- Randomly selecting the starting time of an event from among the set of all possible starting times; i.e., from among the candidates.

When a complete schedule has been generated, the values of the selected parameters are computed and stored. The program automatically repeats the scheduling process until the desired number of schedules are generated.

Once all the desired schedules and associated values of each parameter are generated, the results can be analyzed to determine the degree to which given parameters are dependent upon the schedule. Information concerning the maximum and average ranges of all parameters being considered is also obtained. The probability and confidence level that each parameter will lie within the range determined for it is a function only of the number of schedules used to determine the range.

The rest of this section will serve to demonstrate the probability theory used to determine the probabilities and confidence levels to be applied to the data generated using the REPRI program.

Let S_1, S_2 ..., denote the schedules in the set of all possible randomly generated schedules, and let f_i denote the i^{th} payoff function (i.e., parameter) defined on all possible schedules. The sequence of values of f_i, given by $f_i(S_1)$, $f_i(S_2)$ ..., can be considered as a sequence of independent random variables with an unknown distribution, F. Then $F_i(x)$ is the probability that a randomly generated schedule will have an f_i value smaller than x, i.e., (see Figure A-1).

$$F_i(x) = P_r\,[f_i\,(S) < x] = \int_0^x y_i\,(x)\,dx$$

where S is an arbitrary random schedule.

Let ξ_p be the p-percentile of the unknown distribution F_i, i.e., p is the number such that

$$F_i\,(\xi_p) = P_r\,[f_i\,(S) < \xi_p] = p = \int_0^{\xi_p} y_i\,(x)\,dx,\ 0 \leq p \leq 1 \qquad \text{(A-1)}$$

Suppose that the outcome of the f_i value greater than or equal to ξ_p from a given schedule be designated as a success and the outcome of an f_i value less than ξ_p be designated as a failure. Hence, by equation (A-1), the probability of a failure for any random schedule is given by p and conversely the probability of a success is given by (1-p).

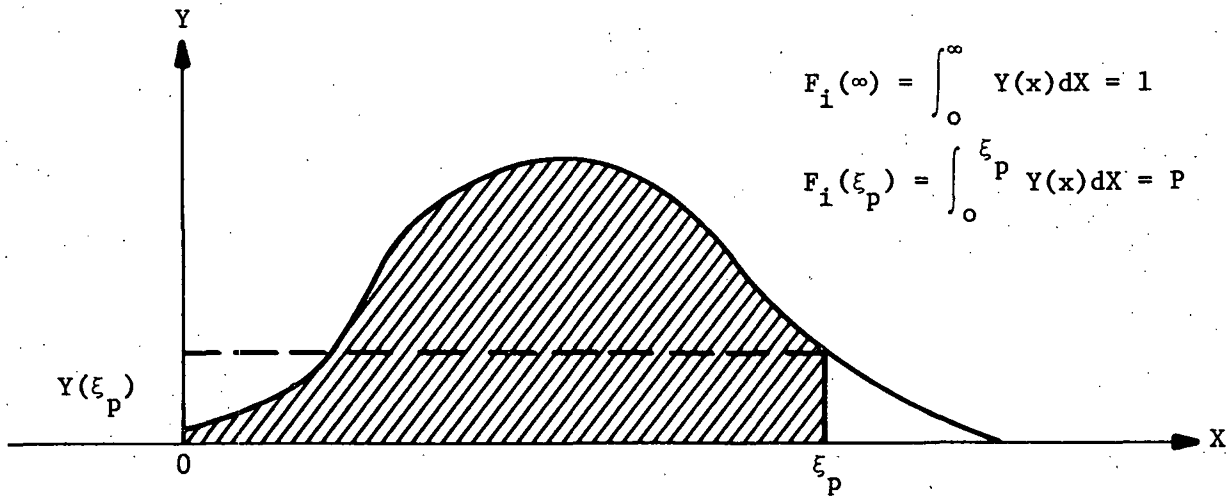
If a finite number of schedules, N, are generated, the probability that at least one of these N schedules will have an f_i value greater than or equal to ξ_p may be expressed as the probability of at least one success in N independent trials, and this probability may be denoted by

$$P_r\,[\text{at least one } f_i(S_k) \geq \xi_p],\ k = 1, 2, \ldots, N$$

By Bernoulli's Theorem[1], the probability of exactly j failures and (n-j) successes in n independent trials is given by the expression

$$P_r = \frac{n!}{j!\,(n-j)!}\,p^j\,(1-p)^{n-j}$$

1. *Coolidge, J. L. An Introduction to Mathematical Probability, Dover Publications, Inc., New York, N. Y., 1962, p. 32.*



$$F_i(\infty) = \int_o^{\infty} Y(x)dX = 1$$

$$F_i(\xi_p) = \int_o^{\xi_p} Y(x)dX = P$$

Note: The probability, $F_i\ (\infty) = P_r\ [f_i(S)<\infty\ ] = 1$, that a random f_i value will lie in the interval $[o, \infty]$ is represented by the total area under the curve from 0 to ∞.

The probability, $F_i\ (\xi_p) = P_r\ [f_i(S)\ <\xi_p] = p$, that a random f_i value will lie in the interval $[0, \xi_p]$ is represented by the shaded area under the curve from 0 to ξ_p.

Figure A-1. DISTRIBUTION CURVE FOR THE SCHEDULE PARAMETER f_i

Since an outcome of at least one success in n trials includes all possible outcomes except that of exactly n failures, the probability of at least one success is given by

$$1 - [P_r(\text{n failures in n trials})]$$

Thus, the probability of at least one f_i value greater than or equal to ξ_p in the N random schedules generated may be experssed as

$$P_r\ [\text{at least one } f_i\ (S_k) \geq \xi_p] = 1 - [P_r\ (\text{all N } f_i\ (S_k) < \xi)],$$
$$k = 1, 2, \ldots, N \qquad \text{(A-3)}$$

Replacing n by N, and j by N in expression (A-2), we have

$$P_r\ [\text{all N } f_i\ (S_k) < \xi_p] = \frac{N!}{N!\ (N-N)!} p^N\ (1-p)^{N-N} = p^N$$
$$k = 1, 2, \ldots, N$$

and substituting p^N in equation (A-3) we have

$$P_r\ [\text{at least one } f_i\ (S_k) \leq\ _p] = 1-p^N,\ k = 1, 2, \ldots, N \qquad \text{(A-4)}$$

The value of p is sometimes called the confidence level.

As an example of the application of equation (A-4), suppose it is desired to know the number, N, of schedules required to have the probability be 0.95 that the maximum (minimum) value for the f_i parameter in the sample S_1, ..., S_N of N schedules is larger (smaller) than 99 percent of all possible values. Then the preceding equation becomes

$$0.95 = 1 - (0.99)^N$$

In general, the exact solution of this equation will not yield an integer value for N, hence, the equation is solved for the samllest value for N for which the probability is at least 0.95, i.e., the smallest integer N is sought for which the inequality given by

$$0.95 \leq 1 - (0.99)^N$$

holds. The value of N is found to be N = 296.

Figure A-2 is a graph of P_r as a function of N for value of p of 0.90, 0.95, and 0.99. Figure A-3 is a plot of p as a function of N for values of P_r of 0.90, 0.95, and 0.99. Using these graphs, the number of schedules which must be generated to achieve the desired p and P_r values can be readily obtained.

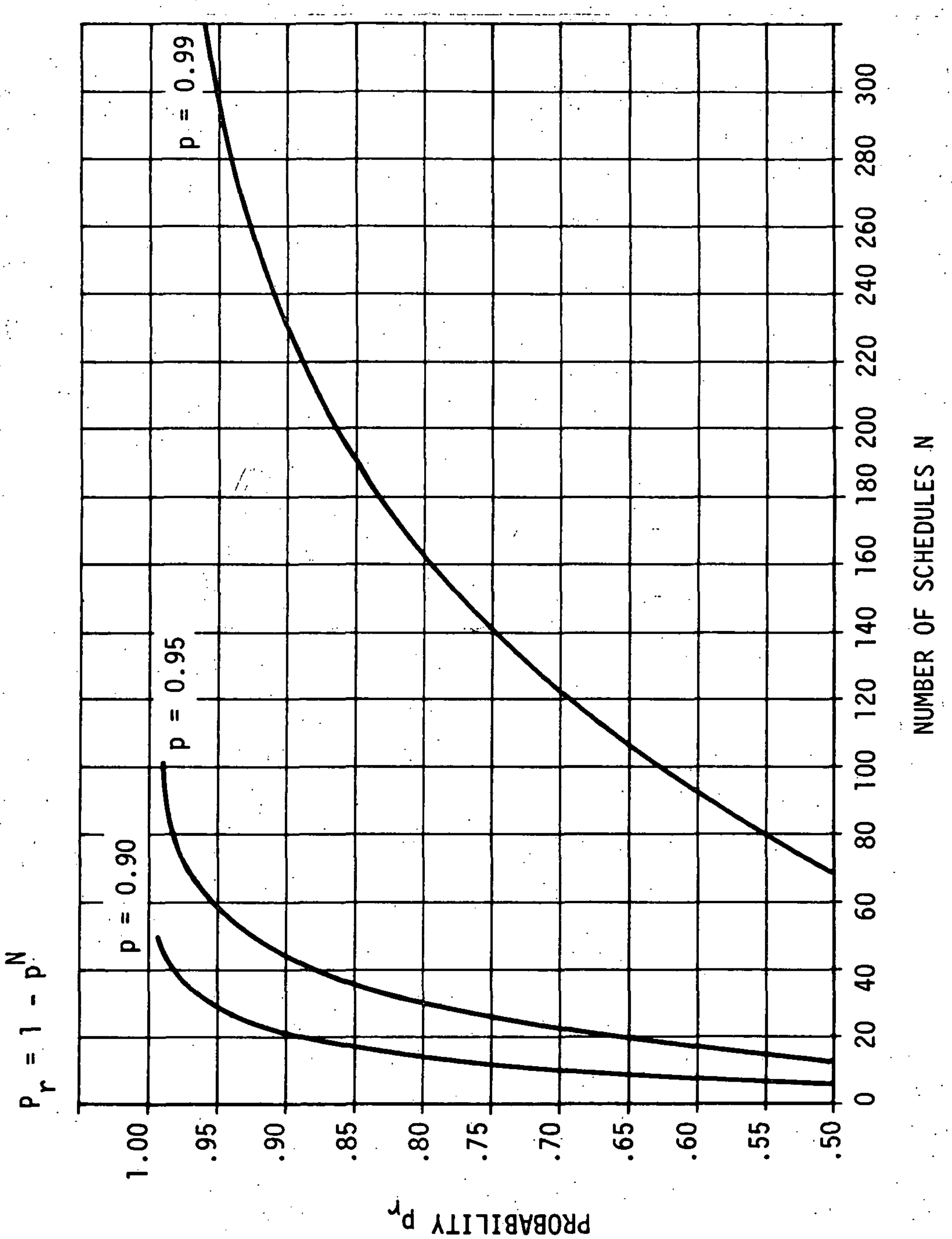


Figure A-2. PROBABILITY AS A FUNCTION OF THE NUMBER OF SCHEDULES

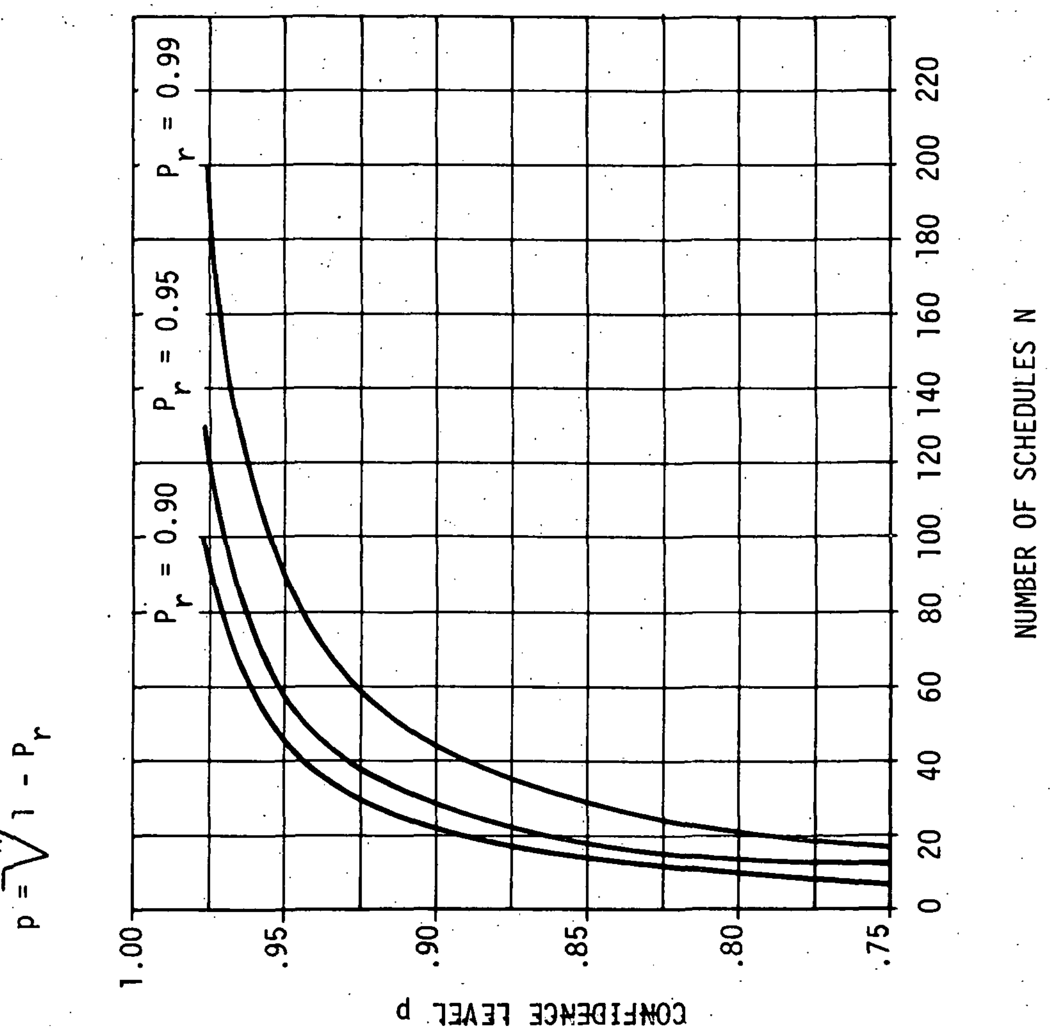


FIGURE A-3. CONFIDENCE LEVEL AS A FUNCTION OF NUMBER OF SCHEDULES

Appendix B

FORTRAN SOURCE LISTING OF REPRI PROGRAM

```
$JOB            NASA/KIRK              410160,00,12
$IBJOB          MAP,FILES,GO,NODECK
$IBFTC MAIN
      DIMENSION IDATA(36,25),ITOTL(120,27),ICTRT(120,5),IOUT(120),
     *          JDATE(36),IREPT(36),IDATE(36),ANAME(27),BNAME(27),
     *          MEXP(36),IMOD(3),XOCST(18),XCOST(9,36),IPERF(36),
     *          JCON(36,3),XXCST(18),IIOUT(120,36),NREQD(36,11),
     *          IEXP(36),LAVG(10)
      COMMON IRAND,N,M,IN,IDATA,ITOTL,ICTRT,IOUT,IREPT,IDATE,MM,
     *       JDATE,MEXP,XCOST,IPERF,IRAN,XOCST,JCON,ISEN2,ZCOST,
     *       APER,NSKL ,LL,XXCST,IIN,IIOUT,NREQD,IAM,IFM,ICODE(12,37)
      COMMON/ALPHA/ANAME,BNAME,IMOD
C
      READ(5,    1) ISEN2,ICASE
      WRITE(6,1)  ISEN2,ICASE
    1 FORMAT(2I3)
      READ(5,    2) (LAVG(I), I=1,10)
      WRITE(6,2) (LAVG(I),I=1,10)
    2 FORMAT(10I5)
C
C     ISEN2 - NUMBER OF SCHEDULES TO BE PRINTED
C             (SCHEDULE ONE THROUGH SCHEDULE ISEN2)
C     ICASE - NO. OF CASES(COMPLETE DATA PACKS)
C     LAVG - NUMBER OF SCHEDULES AFTER WHICH A SUMMARY IS DESIRED.
C     INPUT UP TO TEN VALUES IN ASCENDING ORDER.  A FINAL
C     SUMMARY IS PRINTED REGARDLESS OF THE VALUES IN LAVG.
C
C
C     MM= NUMBER OF RESOURCES FOR WHICH UTILIZATION IS COMPUTED
    3 MM=27
      IFLAG=0
      LXXX=1
      CALL INPUT
      WRITE(6,    4)
      CALL COST(LLL,IFLAG,IDUM)
      CALL TRACK(1,LLL)
      WRITE(6,    4)
    4 FORMAT(1H1)
C
C
    5 DO    14 LLL=1,LL
      DO     6 J=1,18
    6 XOCST(J)=0.
      DO 16 I=1,12
      DO 16 J=1,8
   16 ICODE(I,J)=0
      ZCOST=0.0
      IAM=1000
      IFM=1000
      DO     8 I=1,IN
      DO     7 J=1,MM
    7 ITOTL(I,J)=0
    8 IOUT(I)=0
      CALL ORDER(N,IEXP)
```

```
      CALL OUTPU (LLL,0,IDUM)
C
C     EXPERIMENT SCHEDULING LOOP
      DO    10 K=1,N
      CALL ZPACK(IEXP(K))
      CALL COST(LLL,-1,IEXP(K))
      CALL OUTPU (LLL,+1,IEXP(K))
   10 CALL ENCODE(IEXP(K),ICODE,IOUT,IN)
C
      DO    11 I=2,IN
   11 ITOTL(I,27)=ITOTL(I,27)+ITOTL(I-1,27)
      DO    12 I=1,IN
   12 ITOTL(I,27)=IFIX(FLOAT(ITOTL(I,27))*100./APER+0.5)
      CALL OUTPU (LLL,+2,IDUM)
      CALL COST(LLL,1,IDUM)
      CALL TRACK(2,LLL)
      IF(LLL-LAVG(LXXX))14,13,14
   13 CALL TRACK(3,LLL)
      LXXX=LXXX + 1
   14 CONTINUE
C
C
      CALL TRACK(3,LL)
      ICASE=ICASE-1
      WRITE(6,    4)
      IF(ICASE)15,15,3
   15 STOP
      END

$IBFTC BLCK    DECK
      BLOCK DATA
      DIMENSION IMOD(3),ANAME(27),BNAME(27)
      COMMON/ALPHA/ANAME,BNAME,IMOD
      DATA IMOD/2H I,2HAM,2HFF/
      DATA ANAME(20),BNAME(20)/4HS WT,4H UP /
      DATA ANAME(21),BNAME(21)/4HS VO,4HL UP/
      DATA ANAME(22),BNAME(22)/4HRET ,4HWT  /
      DATA ANAME(23),BNAME(23)/4HRET ,4HVOL /
      DATA ANAME(24),BNAME(24)/4HE WT,4H UP /
      DATA ANAME(25),BNAME(25)/4HE VO,4HL UP/
      DATA ANAME(27),BNAME(27)/4HPC C,4HOMPL/
C     ANAME(20) = SUPPLY   WEIGHT TO BE CARRIED UP
C     ANAME(21) = SUPPLY   VOLUME TO BE CARRIED UP
C     ANAME(22) = LOGISTIC WEIGHT TO BE CARRIED DOWN  (INCLUDES BOTH
C                 EXPERIMENTS AND SUPPLIES)
C     ANAME(23) = LOGISTIC VOLUME TO BE CARRIED DOWN  (INCLUDES BOTH
C                 EXPERIMENTS AND SUPPLIES)
C     ANAME(24) = EXPERIMENT WEIGHT TO BE CARRIED UP
C     ANAME(25) = EXPERIMENT VOLUME TO BE CARRIED UP
C     ANAME(27) = PERCENT OF EXPERIMENTS THAT WERE SCHEDULED
C     IMOD   = FPE ACCOMMODATION
C     IMOD(1)  = I
C     IMOD(2)  = AM
C     IMOD(3)  = FF
      END
```

```
$IBFTC OTPT    DECK
      SUBROUTINE OUTPU (ISCHD,JUMP,NEXP)                                OTPT  1
      DIMENSION IDATA(36,25),ITOTL(120,27),ICTRT(120,5),IOUT(120),      OTPT  2
     *          JDATE(36),IREPT(36),IDATE(36),ANAME(27),BNAME(27),      OTPT  3
     *          MEXP(36),IMOD(3),XOCST(18),XCOST(8,36),IPERF(36),       OTPT  4
     *          JCON(36,3),XXCST(18),IIOUT(120,36),NREQD(36,11)         OTPT  5
      COMMON IRAND,N,M,IN,IDATA,ITOTL,ICTRT,IOUT,IREPT,IDATE,MM,        OTPT  6
     *       JDATE,MEXP,XCOST,IPERF,IRAN,XOCST,JCON,ISEN2,ZCOST,        OTPT  7
     *       APER,NSKL ,LL,XXCST,IIN,IIOUT,NREQD,IAM,IFM,ICODE(12,37)   OTPT  8
      COMMON/ALPHA/ANAME,BNAME,IMOD                                     OTPT  9
      IF(ISEN2-ISCHD)18,1,1                                             OTPT 10
C                                                                       OTPT 11
C     PRINT SCHEDULE                                                    OTPT 12
C                                                                       OTPT 13
C                                                                       OTPT 14
    1 IF (JUMP-1)2,6,10                                                 OTPT 15
C                                                                       OTPT 16
C     PRINT SCHEDULE HEADING                                            OTPT 17
    2 WRITE(6,   3) ISCHD                                               OTPT 18
    3 FORMAT(  9H SCHEDULE,1X,I4,75X,12HPERFORMANCES,9H  MODE OF)       OTPT 19
      WRITE(6,   4) (I, I=1,IN)                                         OTPT 20
    4 FORMAT (1X,8HINTERVAL ,2X,I1,19I4,13H    REQ  SCHD,15H  ACCOMMODATOTPT 21
     *ION/, (8X,20I4))                                                  OTPT 22
      WRITE(6,   5)                                                     OTPT 23
    5 FORMAT (1X,109(1H-))                                              OTPT 24
      RETURN                                                            OTPT 25
C                                                                       OTPT 26
C     PRINT SCHEDULE OF EXPERIMENT NEXP                                 OTPT 27
    6 MODX=MEXP(NEXP)/1000+1                                            OTPT 28
      MEXP2=MEXP(NEXP)-(MEXP(NEXP)/1000)*1000                           OTPT 29
      WRITE(6,   9) MEXP2, (IOUT(I), I=1,IIN)                           OTPT 30
      WRITE(6,  17) IREPT(NEXP),IPERF(NEXP),IMOD(MODX)                  OTPT 34
   17 FORMAT(1H+,88X,2I6,7X,A2)                                         OTPT 56
      IXX=IIN + 1                                                       OTPT 31
      IF(IN-20)18,18,7                                                  O3PT 32
    7 WRITE(6,  15) (IOUT(I), I=IXX,IN)                                 OTPT 33
    9 FORMAT(1X,3HFPE,1X,I3,20I4)                                       OTPT 35
      RETURN                                                            OTPT 36
C                                                                       OTPT 37
C     PRINT RESOURCE SUMMARY                                            OTPT 38
   10 WRITE(6,   5)                                                     OTPT 39
      WRITE(6,  11)                                                     OTPT 40
   11 FORMAT(1X,20HSUMMARY OF RESOURCES)                                OTPT 41
      WRITE(6,  12) (I, I=1,IN)                                         OTPT 42
   12 FORMAT(10H INTERVAL  ,20I6/, (10X,20I6))                          OTPT 43
      DO   13 I=1,M                                                     OTPT 44
   13 WRITE(6,  16) ANAME(I),BNAME(I),(ITOTL(J,I), J=1,IN)              OTPT 45
      DO   14 I=20,27                                                   OTPT 46
   14 WRITE(6,  16) ANAME(I),BNAME(I),(ITOTL(J,I), J=1,IN)              OTPT 47
   15 FORMAT(8X,20I4)                                                   OTPT 54
   16 FORMAT(1X,2A4,1X,20I6/, ( 10X,20I6))                              OTPT 55
   18 RETURN                                                            OTPT 57
      END                                                               OTPT 58
```

```
$IBFTC COST    DECK
      SUBROUTINE COST(ISCHD,IFLAG,NEXP)
      DIMENSION IDATA(36,25),ITOTL(120,27),ICTRT(120,5),IOUT(120),
     *          JDATE(36),IREPT(36),IDATE(36),FFD(5),AMD(5),
     *          MEXP(36),XOCST(18),XCOST(8,36),IPERF(36),
     *          JCON(36,3),XXCST(18),IIOUT(120,36),NREQD(36,11)
      COMMON IRAND,N,M,IN,IDATA,ITOTL,ICTRT,IOUT,IREPT,IDATE,MM,
     *       JDATE,MEXP,XCOST,IPERF,IRAN,XOCST,JCON,ISEN2,ZCOST,
     *       APER,NSKL  ,LL,XXCST,IIN,IIOUT,NREQD,IAM,IFM,ICODE(12,37)
      IF(IFLAG)6,1,17
C
C
C     INITIALIZATION AND INPUT OF COST VARIABLES
    1 READ(5,    4) IYR
      WRITE(6,    4) IYR
C     IYR = NO. OF INTERVALS/YEAR
      DO    3 J=1,N
      READ(5,    2)(XCOST(I,J),I=1,8)
    2 FORMAT(10F8.1)
    3 WRITE(6,    4) MEXP(J),(XCOST(I,J),I=1,8)
    4 FORMAT(1X,I9,9F9.1)
      READ(5,    2) (FFD(I), I=1,5)
      WRITE(6,    5)(FFD(I), I=1,5)
    5 FORMAT(1X,10F9.1)
      READ(5,    2)  (AMD(I),I=1,5)
      WRITE(6,    5)(AMD(I),I=1,5)
C
C     XCOST(I,J) - THE COST OF EXPERIMENT J REQUIRED THE (9-I)TH
C                  YEAR PRIOR TO THE FIRST SCHEDULED PERFORMANCE
C                  OF EXPERIMENT J.
C     FFD(I)     - THE COST OF DEVELOPING A FREE-FLYING MODULE THAT
C                  IS REQUIRED IN THE (6-I)TH YEAR PRIOR TO THE
C                  FIRST USE OF A FF MODULE.
C     AMD(I)     - THE COST OF DEVELOPING AN ATTACHED MODULE THAT
C                  IS REQUIRED IN THE (6-I)TH YEAR PRIOR TO THE
C                  FIRST USE OF AN ATTACHED MODULE.
C
C     THE NUMBER OF YEARS THE COST OF AN EXPERIMENT PACKAGE IS SPREAD
C     OVER IS ASSUMED TO BE 8.
C     THE NUMBER OF YEARS THE DEVELOPEMENT COST OF A MODULE IS SPREAD
C     OVER IS ASSUMED TO BE 5.
C
      YR=IYR
      IYTP=8*IYR
      IYTPM2=IYTP-2
      IYTP5=5*IYR
      NSYR=7*IYR
      RETURN
C
C
    6 DO    7 IFIRST=1,IN
      IF(IOUT(IFIRST))8,7,8
    7 CONTINUE
      RETURN
```

```
C       IFIRST - FIRST INTERVAL NEXP WAS SCHEDULED IN
      8 KK=MEXP(NEXP)/1000
        IF(KK-1)13,9,11
      9 IF(IFIRST-IAM)10,13,13
     10 IAM=IFIRST
C       IAM - FIRST INTERVAL IN WHICH AN ATTACHED MODULE IS USED
        GO TO    13
     11 IF(IFIRST-IFM)12,13,13
     12 IFM=IFIRST
C       IFM - FIRST INTERVAL IN WHICH A FREE FLYING MODULE IS USED
C
     13 DO    16 K=1,IYTP
        KK=(IYTP-K)/IYR+1
        II=(IFIRST+IYTPM2)/IYR
        IF(II)14,14,15
     14 II=(IYR-II)/IYR
     15 XOCST(II)=XCOST(KK,NEXP)/YR+XOCST(II)
     16 IFIRST =IFIRST-1
        RETURN
C
C
C       LOGIC TO CONSIDER MODULE DEVELOPMENT
     17 DO    18 I=1,IYTP5
        II=((IAM-I)+NSYR+IYR-1)/IYR
        KK=5-(I-1)/IYR
        XOCST(II)=XOCST(II)+AMD(KK)/YR
        II=((IFM-I)+NSYR+IYR-1)/IYR
     18 XOCST(II)=XOCST(II)+FFD(KK)/YR
        DO    19 I=1,18
        XOCST(I)=XOCST(I)*.001
     19 ZCOST=ZCOST+XOCST(I)
C       ZCOST - TOTAL COST
C
C       CALCULATION OF CUMULATIVE COST (XXCST)
        XXCST(1)=XOCST(1)
        DO    20 I=2,18
     20 XXCST(I)=XXCST(I-1)+XOCST(I)
C
C
        IF(ISEN2-ISCHD)27,21,21
     21 WRITE(6,   22)
     22 FORMAT(34H COST PER YEAR  (MILLION DOLLARS) )
        WRITE(6,   23) (I, I=1970,1979)
     23 FORMAT(1X ,10I10)
        WRITE(6,   24) (XOCST(I), I=1,10)
     24 FORMAT(1X,10F10.2)
        WRITE(6,   24) (XXCST(I), I=1,10)
        WRITE(6,   25) (I,I=1980,1987)
     25 FORMAT(    1X, 8I10,4X,5HTOTAL )
        WRITE(6,   24)(XOCST(I),I=11,18),ZCOST
        WRITE(6,   24) (XXCST(I), I=11,18)
        WRITE(6,   26)
     26 FORMAT(1H1)
     27 RETURN
        END
```

```
$IBFTC ZPCK     DECK
      SUBROUTINE ZPACK(NEXP)                                            ZPCK  1
      DIMENSION IDATA(36,25),ITOTL(120,27),ICTRT(120,5),IOUT(120),      ZPCK  2
     *          JDATE(36),IREPT(36),IDATE(36),ANAME(27),BNAME(27),      ZPCK  3
     *          MEXP(36),IMOD(3),XOCST(18),XCOST(8,36),IPERF(36),       ZPCK  4
     *          JCON(36,3),XXCST(18),IIOUT(120,36),NREQD(36,11),        ZPCK  5
     *          ISTOR(120),KFT(3)                                       ZPCK  6
      COMMON IRAND,N,M,IN,IDATA,ITOTL,ICTRT,IOUT,IREPT,IDATE,MM,        ZPCK  7
     *       JDATE,MEXP,XCOST,IPERF,IRAN,XOCST,JCON,ISEN2,ZCOST,        ZPCK  8
     *       APER,NSKL  ,LL,XXCST,IIN,IIOUT,NREQD,IAM,IFM,ICODE(12,37)  ZPCK  9
      COMMON/ALPHA/ANAME,BNAME,IMOD                                     ZPCK 10
      M1= 4                                                             ZPCK 11
C     MODTY = FLAG WHICH INDICATES MODE OF ACCOMODATION
      MODTY=MEXP(NEXP)/1000                                             ZPCK 12
      NRPT=IREPT(NEXP)                                                  ZPCK 13
C     NRPT - NUMBER OF REQUIRED PERFORMANCES FOR EXPERIMENT NEXP        ZPCK 14
      DO    1 I=1,3                                                     ZPCK 15
    1 KFT(I)=JCON(NEXP,I)                                               ZPCK 16
C     KFT = DEFINES FPE'S WHICH CONFLICT WITH FPE BEING SCHEDULED.
      IF(MODTY)2,2,3                                                    ZPCK 17
    2 ISAVI=IDATA(NEXP,21)    +IDATA(NEXP,25)
      GO TO    4                                                        ZPCK 19
    3 ISAVI=IDATA(NEXP,21)
C     ISAVI = SPACE STATION STORAGE VOLUME REQUIRED TO STORE FPE BEING
C             SCHEDULED.
    4 IF(NRPT)48,48,5                                                   ZPCK 21
C                                                                       ZPCK 22
C     TEST TO CHECK IF EXPERIMENT NEXP HAS A FORCED START DATE          ZPCK 23
    5 IF(IDATE(NEXP))9,9,6                                              ZPCK 24
    6 INTVL=IDATE(NEXP)                                                 ZPCK 25
      DO    7 K=1,M1                                                    ZPCK 26
C     CHECK FIRST 4 CONSTRAINTS (MANPOWER, ELECTRICAL POWER, ETC.).
      IF(ICTRT(INTVL,K)-ITOTL(INTVL,K)-IDATA(NEXP,K))48,7,7             ZPCK 27
    7 CONTINUE                                                          ZPCK 28
C     CHECK STATION STORAGE VOLUME.
      IF(ICTRT(INTVL, 5)-ITOTL(INTVL,26)-ISAVI)48,8,8                   ZPCK 29
C     CHECK FOR OTHER FPE'S WHICH CONFLICT.
    8 IF(ICON(ICODE,-1,INTVL,KFT)) 14,14,48
C                                                                       ZPCK 31
C     THIS SECTION SELECTS THE START INTERVAL.
C                                                                       ZPCK 32
    9 CALL ORDER(IN,ISTOR)                                              ZPCK 33
C     ISTOR IS A RANDOM ORDERING OF THE IN INTERVALS                    ZPCK 34
      DO   13 J=1,IN                                                    ZPCK 35
      INTVL=ISTOR(J)                                                    ZPCK 36
C     EXPERIMENT NEXP CANNOT BE SCHEDULED BEFORE JDATE(NEXP)            ZPCK 37
      IF(INTVL-JDATE(NEXP))13,10,10                                     ZPCK 38
   10 DO   11 K=1,M1                                                    ZPCK 39
      IF(ICTRT(INTVL,K)-ITOTL(INTVL,K)-IDATA(NEXP,K))13,11,11           ZPCK 40
   11 CONTINUE                                                          ZPCK 41
      IF(ICTRT(INTVL, 5)-ITOTL(INTVL,26)-ISAVI)13,12,12                 ZPCK 42
   12 IF(ICON(ICODE,-1,INTVL,KFT)) 14,14,13
   13 CONTINUE                                                          ZPCK 44
      GO TO    48                                                       ZPCK 45
```

```
C                                                                       ZPCK  46
   14 IOUT(INTVL)=1                                                     ZPCK  47
C     NEXP HAS BEEN SCHEDULED IN INTVL                                  ZPCK  48
      ISAVE=INTVL                                                       ZPCK  49
C     ISAVE = FIRST INTERVAL NEXP HAS BEEN SCHEDULED IN                 ZPCK  50
      NRPT=NRPT-1                                                       ZPCK  51
   15 IF(NRPT)31,31,16                                                  ZPCK  52
   16 INTVL=INTVL+1                                                     ZPCK  53
      IF(INTVL-IN)17,17,23                                              ZPCK  54
C                                                                       ZPCK  55
C     LOOP FOR SCHEDULING AFTER ISAVE                                   ZPCK  56
   17 DO    19 K=1,M1                                                   ZPCK  57
      IF(ICTRT(INTVL,K)-ITOTL(INTVL,K)-IDATA(NEXP,K))18,19,19           ZPCK  58
   18 IF(ICTRT(INTVL, 5)-ITOTL(INTVL,26)-ISAVI)23,16,16                 ZPCK  59
   19 CONTINUE                                                          ZPCK  60
      IF(ICTRT(INTVL, 5)-ITOTL(INTVL,26)-ISAVI)23,20,20                 ZPCK  61
   20 ITEST=ICON(ICODE,MODTY,INTVL,KFT)+1
      GO TO (21,23,16), ITEST
   21 IOUT(INTVL)=1                                                     ZPCK  63
      NRPT=NRPT-1                                                       ZPCK  64
      IF(INTVL-ISAVE)22,15,15                                           ZPCK  65
   22 IF(NRPT)31,31,24                                                  ZPCK  66
C                                                                       ZPCK  67
C     LOOP FOR SCHEDULING BEFORE ISAVE                                  ZPCK  68
   23 INTVL=ISAVE                                                       ZPCK  69
   24 INTVL=INTVL-1                                                     ZPCK  70
      IF(INTVL)31,31,25                                                 ZPCK  71
   25 IF(INTVL-IDATE(NEXP))31,26,26                                     ZPCK  72
   26 IF(INTVL-JDATE(NEXP))31,27,27                                     ZPCK  73
   27 DO    29 K=1,M1                                                   ZPCK  74
      IF(ICTRT(INTVL,K)-ITOTL(INTVL,K)-IDATA(NEXP,K))28,29,29           ZPCK  75
   28 IF(ICTRT(INTVL, 5)-ITOTL(INTVL,26)-ISAVI)31,24,24                 ZPCK  76
   29 CONTINUE                                                          ZPCK  77
      IF(ICTRT(INTVL, 5)-ITOTL(INTVL,26)-ISAVI)31,30,30                 ZPCK  78
   30 ITEST=ICON(ICODE,MODTY,INTVL,KFT)+1
      GO TO (21,31,24), ITEST
C     ALL SCHEDULING HAS BEEN COMPLETED FOR NEXP                        ZPCK  80
C                                                                       ZPCK  81
C     SUMMATION OF RESOURCE REQUIREMENTS                                ZPCK  82
C                                                                       ZPCK  83
   31 ISAVE =0                                                          ZPCK  84
      DO    47 J=1,IN                                                   ZPCK  85
      ISAVE=ISAVE+IOUT(J)                                               ZPCK  86
      IF(ISAVE)47,47,32                                                 ZPCK  87
   32 IF(IOUT(J))47,43,35                                               ZPCK  90
   35 IF(ISAVE-1)47,36,37                                               ZPCK  91
   36 ITOTL(J,24)=ITOTL(J,24)+IDATA(NEXP,24)
      ITOTL(J,25)=ITOTL(J,25)+IDATA(NEXP,25)
   37 DO    38 K=1,4                                                    ZPCK  94
   38 ITOTL(J,K)=ITOTL(J,K)+IDATA(NEXP,K)                               ZPCK  95
C                                                                       ZPCK  96
C     SKILL TYPE ALGORITHM                                              ZPCK  97
      IF(NREQD(NEXP,1))41,41,39                                         ZPCK  98
   39 INDEX=NREQD(NEXP,1)*2                                             ZPCK  99
      DO    40 I=2,INDEX,2                                              ZPCK 100
```

```
      I2=NREQD(NEXP,I)+M1
   40 ITOTL(J,I2)=ITOTL(J,I2)+NREQD(NEXP,I+1)
C
   41 DO   42 K=20,23
   42 ITOTL(J,K)=ITOTL(J,K)+IDATA(NEXP,K)
      ITOTL(J,26)=ITOTL(J,26)+IDATA(NEXP,21)*3
      ITOTL(J,27)=ITOTL(J,27)+IOUT(J)
   43 IF(ISAVE-IREPT(NEXP)+NRPT)45,44,47
   44 ITOTL(J,22)=ITOTL(J,22)+IDATA(NEXP,24)
      ITOTL(J,23)=ITOTL(J,23)+IDATA(NEXP,25)
      ISAVE =ISAVE+1
   45 IF(MODTY)46,46,47
   46 ITOTL(J,26)=ITOTL(J,26)+IDATA(NEXP,25)
   47 CONTINUE
C
C     CALCULATION OF THE NO OF PERFORMANCES COMPLETED
   48 IPERF(NEXP)=IREPT(NEXP)-NRPT
      DO   49 I=1,IN
   49 IIOUT(I,NEXP)=IOUT(I)+IIOUT(I,NEXP)
      RETURN
      END
```

```
$IBFTC ICON     DECK
      FUNCTION ICON(ICODE,MODTY,IN,IFLCK)
      DIMENSION ICODE(12,37),IFLCK(3)
      ICON = 0
C     FUNCTION ICON CHECKS TO SEE IF EXPERIMENTS IFLCK(1-3) HAVE
C     BEEN SCHEDULED IN INTERVAL IN
C
      IWORD=(IN-1)/10+1
      IBIT=10**(IWORD*10-IN)
      DO    2 I=1,3
      M=IFLCK(I)
      IF(M)2,2,1
    1 K=ICODE(IWORD,M)/IBIT
      ICON= K-(K/10)*10
      IF(ICON)3,2,3
    2 CONTINUE
      RETURN
    3 IF(M.EQ.37.AND.MODTY.EQ.0) ICON=ICON+1
      RETURN
C
C     ICON = 2 - ARTIFICIAL GRAVITY CONFLICT
C     ICON=1   - CONFLICT
C     ICON=0   - NO CONFLICT
      END
```

```
$IBFTC ORDR    DECK
      SUBROUTINE ORDER(N,IORD)                                          ORDR   1
      DIMENSION IORD(1)                                                 ORDR   2
      IN=N                                                              ORDR   4
      DO     1  I=1,N                                                   ORDR   5
    1 IORD(I)=I                                                         ORDR   6
      DO     2  I=1,N                                                   ORDR   7
      CALL RANDXX(RNBR,0,NN)                                            ORDR   8
      XX=IN                                                             ORDR   9
      IX=RNBR*XX                                                        ORDR  10
      IX=IX+1                                                           ORDR  11
      ISAV=IORD(IX)                                                     ORDR  12
      IORD(IX)=IORD(IN)                                                 ORDR  13
      IORD(IN)=ISAV                                                     ORDR  14
      IN=IN-1                                                           ORDR  15
    2 CONTINUE                                                          ORDR  16
      RETURN                                                            ORDR  17
      END                                                               ORDR  18
```

```
$IBFTC RAND    DECK
      SUBROUTINE RANDXX(R,IFLAG,NN)                                     RAND   1
      COMMON N                                                          RAND   2
      IF(IFLAG)2,1,2                                                    RAND   3
    1 N=N*78125                                                         RAND   4
      R = FLOAT(N)/ .34359738E+11                                       RAND   5
      RETURN                                                            RAND   6
    2 IF((2*(NN/2)-NN).EQ.0) NN=NN+1                                    RAND   7
      N=IABS(NN)*1953125                                                RAND   8
      RETURN                                                            RAND   9
      END                                                               RAND  10
```

```
$IBFTC ECDE    DECK
      SUBROUTINE ENCODE (K,ICODE,IOUT,IN)                               ECDE   1
      DIMENSION ICODE(12,37),IOUT(1)                                    ECDE   2
C                                                                       ECDE   3
C     PACKS IOUT(IN) FOR EXPERIMENT K INTO ICODE(12,K)                  ECDE   4
C                                                                       ECDE   5
      IX=(IN + 9)/10                                                    ECDE   6
      DO     1  I=1,IX                                                  ECDE   7
      ICODE(I,K)= 0                                                     ECDE   8
      DO     1  KK=1,10                                                 ECDE   9
      MM=(I-1)*10+KK                                                    ECDE  10
      IF(MM-IN)1,1,2                                                    ECDE  11
    1 ICODE(I,K)=IOUT(MM)*(10**(10-KK))+ICODE(I,K)                      ECDE  12
    2 DO     3  KK=1,IN                                                 ECDE  13
    3 IOUT(KK)=0                                                        ECDE  14
      RETURN                                                            ECDE  15
      END                                                               ECDE  16
```

```
$IBFTC INPT     DECK
      SUBROUTINE INPUT
      DIMENSION IDATA(36,25),ITOTL(120,27),ICTRT(120,5),IOUT(120),
     *          JDATE(36),IREPT(36),IDATE(36),ANAME(27),BNAME(27),
     *          MEXP(36),IMOD(3),XOCST(18),XCOST(8,36),IPERF(36),
     *          JCON(36,3),XXCST(18),IIOUT(120,36),NREQD(36,11)
      DIMENSION A(11),JCONX(3)
      DIMENSION B(9),KLOUT(5)
      COMMON IRAND,N,M,IN,IDATA,ITOTL,ICTRT,IOUT,IREPT,IDATE,MM,
     *       JDATE,MEXP,XCOST,IPERF,IRAN,XOCST,JCON,ISEN2,ZCOST,
     *       APER,NSKL ,LL,XXCST,IIN,IIOUT,NREQD,IAM,IFM,ICODE(12,37)
      COMMON/ALPHA/ANAME,BNAME,IMOD
      DATA IBLNK/6H      /
      DATA A4,XI4/6H,7X,A4,6H,7X,I4/
      DATA A42,XI42/6H1X,A4,,6H1X,I4,/
      DATA A/61H(1X,3HFPE,I4,1X,A2,7X,I4,11X,I4,11X,I4,5X,1X,I4,1X,I4,1X
     *,I4,)/
      DATA B/49H(4H FPE,I5,1X,     4X,A6,4X,A6,4X,A6,4X,A6,4X,A6,)/
      DATA BI/6H2X,I8,/
      DATA BA/6H4X,A6,/
C     N  = NUMBER OF EXPERIMENTS
C     IN = NUMBER OF UNIT TIME INTERVALS
C     LL = NUMBER OF SCHEDULES
C
      READ(5,112) N,IN,LL,NSKL,IRAND
      WRITE(6,112)N,IN,LL,NSKL,IRAND
  112 FORMAT(4I5,I10)
      IIN=20
      IF(IN-20)500,501,501
  500 IIN=IN
  501 CONTINUE
C     M = NUMBER OF RESOURCES TO BE CONSTRAINED
      M=4+NSKL
      CALL RANDXX(APER,1,IRAND)
      APER=0.
      WRITE(6,8699)
 8699 FORMAT(1H1,29X,22HLOGISTICS REQUIREMENTS//)
      WRITE(6,8700)
 8700 FORMAT(13X,6HEXP WT,4X,7HEXP VOL,3X,9HSUPPLY WT,2X,10HSUPPLY VOL,
     *          1X,9HRETURN WT,2X,10HRETURN VOL/)
C     MEXP(I) = EXPERIMENT NUMBER OF ITH EXPERIMENT
C     IDATE(I) = THE INTERVAL IN WHICH EXPERIMENT I IS FORCED TO START
C     JDATE(I) = FIRST INTERVAL IN WHICH EXPERIMENT I MAY START
C     IREPT(I) = NUMBER OF INTERVALS IN WHICH ITH EXPERIMENT IS TO BE
C                SCHEDULED.
C     IDATA(I,J) = THE AMOUNT OF RESOURCE J REQUIRED BY EXPERIMENT I
C     DURING ONE UNIT TIME INTERVAL.
C     IDATA(J,1 ) = BIT RATE
C     IDATA(J,2 ) = AVE WATTS    REQUIRED TO PERFORM  EXPERIMENT J
C     IDATA(J,3 ) = KW-HOURS     REQUIRED TO PERFORM  EXPERIMENT J DURING
C                   UNIT TIME INTERVAL
C     IDATA(J,4 ) = MAN HOURS    REQUIRED TO PERFORM  EXPERIMENT J DURING
C                   UNIT TIME INTERVAL
C     IDATA(J,20) = SUPPLY WT    REQUIRED TO PERFORM  EXPERIMENT J DURING
C                   UNIT TIME INTERVAL
```

```
C       IDATA(J,21) = SUPPLY VOL  REQUIRED TO PERFORM  EXPERIMENT J DURING
C                     UNIT TIME INTERVAL
C       IDATA(J,22) = RETURN WT PER    UNIT TIME INTERVAL
C       IDATA(J,23) = RETURN VOL PER   UNIT TIME INTERVAL
C       IDATA(J,24) = EXP WT
C       IDATA(J,25) = EXP VOL
        DO 111 I=1,N
        READ(5,110) MEXP(I),IDATE(I),JDATE(I),IREPT(I),(IDATA(I,J),J=1,3)
       *                           ,(IDATA(I,J),J=20,25)
C       JCON(I) = THE EXPERIMENT WHICH CONFLICTS WITH ITH EXPERIMENT
        READ(5,110) (NREQD(I,J), J=1,11),(JCON(I,J), J=1,3)
  110   FORMAT(15I5)
        IDATA(I,4)=0
        IF(NREQD(I,1)) 81,81,82
   82   K=NREQD(I,1)*2+1
        DO 80 J=3,K,2
   80   IDATA(I,4)=NREQD(I,J)+IDATA(I,4)
   81   APER=APER+FLOAT(IREPT(I))
        MEXP2=MEXP(I)-(MEXP(I)/1000)*1000
        MODX=(MEXP(I)-MEXP2)/1000+1
  111   WRITE(6,8701) MEXP2,IMOD(MODX),IDATA(I,24),IDATA(I,25),
       *                (IDATA(I,J), J=20,23)
 8701   FORMAT(4H FPE,I4,1X,A2,2X,6(I5,6X))
        WRITE(6,5550)
 5550   FORMAT(1H1,23X,23HSCHEDULING REQUIREMENTS//)
        WRITE(6,5560)
 5560   FORMAT(13X,12HFORCED START,3X,13HMINIMUM START,4X,9HNUMBER OF ,7X,
       *        9HCONFLICTS,/,15X,8HINTERVAL,7X,8HINTERVAL,6X,
       *        12HPERFORMANCES,4X,1H1,4X,1H2,4X,1H3/)
        DO 450 J=1,N
        IF(IDATE(J)) 819,818,819
  818   A(4)=A4
        IDATEX=IBLNK
        GO TO 820
  819   A(4)=XI4
        IDATEX=IDATE(J)
  820   DO 821 K=1,3
        IF(JCON(J,K)) 830,831,830
  831   A(K+7)=A42
        JCONX(K)=IBLNK
        GO TO 821
  830   A(K+7)=XI42
        JCONX(K)=JCON(J,K)
  821   CONTINUE
        IF(JDATE(J).EQ.0) JDATE(J)=1
        MEXP2=MEXP(J)-(MEXP(J)/1000)*1000
        MODX=(MEXP(J)-MEXP2)/1000+1
  450   WRITE(6,A)MEXP2,IMOD(MODX),IDATEX,JDATE(J),IREPT(J),
       *             (JCONX(K),K=1,3)
        DO 205 K=1,3
        DO 205 I=1,N
        DO 204 J=1,N
        IF(JCON(I,K)-MEXP(J))204,202,204
  204   CONTINUE
        IF(JCON(I,K)-37)133,205,133
```

```
  133 JCON(I,K)=0
      GO TO 205
  202 JCON(I,K)=J
  205 CONTINUE
      WRITE(6,465)
  465 FORMAT(1H1)
C     ANAME(J) = THE EIGHT   LETTERS USED TO DESCRIBE RESOURCE J ON
C     PRINT OUT.
C     ANAME(1 ) = AVERAGE WATTS
C     ANAME(2 ) = KILOWATT-HOURS
C     ANAME(3 ) = BIT RATE
C     ANAME(4 ) = MAN HOURS
C     ANAME(5-19) = SKILL TYPES 1-15
C     ANAME(26) = SPACE STATION STORAGE VOLUME REQUIRED ASSUMING THAT
C                 EACH EXPERIMENT IS RETURNED TO EARTH AFTER COMPLETION
      DO 9 J=1,4
      READ(5,11) ANAME(J),BNAME(J)
   11 FORMAT(2A4)
C     ICTRT(I,J) = THE AMOUNT OF RESOURCE J AVAILABLE DURING INTERVAL I.
      READ(5,118) (ICTRT(I,J), I=1,IN)
  118 FORMAT(10I8)
      WRITE(6,88) ANAME(J),BNAME(J)
   88 FORMAT(1X,2A4)
    9 WRITE(6,119) (ICTRT(I,J), I=1,IN)
  119 FORMAT(1X,10I9)
      READ(5,14)(ANAME(J),BNAME(J), J=5,M)
   14 FORMAT(20A4)
      READ(5,11) ANAME(26),BNAME(26)
      READ(5,118) (ICTRT(I,5), I=1,IN)
      READ (5,90) (ICODE(I,37),I=1,12)
   90 FORMAT(6I10)
      WRITE (6,91) (ICODE(I,37),I=1,12)
   91 FORMAT(13H ARTIFICIAL G/(1X,6I10))
C     ARTIFICIAL G CONFLICT MUST BE INPUT IN JCON(NEXP,3)
      WRITE(6,931)
  931 FORMAT(1H1,23X,21HRESOURCE REQUIREMENTS//)
      WRITE(6,9000)(ANAME(K),BNAME(K), K=1,4)
 9000 FORMAT(13X,4(2X,2A4))
      DO 1111 I=1,N
 1111 WRITE(6,9001)MEXP(I),(IDATA(I,K), K=1,4)
 9001 FORMAT(4H FPE,I5,1X,4(2X,I8))
      NO=1
      IX=(NSKL/5)+1
      DO 8888 I=1,IX
      WRITE(6,9002)
 9002 FORMAT(1H1,18X,23HSKILL TYPE REQUIREMENTS//)
      NO1=I*5
      IF(I.EQ.IX) NO1=NSKL
      IJ=NO1+4
      KK=NO+4
      WRITE(6,9003)(ANAME(KJ),BNAME(KJ), KJ=KK,IJ)
 9003 FORMAT(10X,5(2X,2A4))
      DO 8887 IJ=1,N
      DO 804 KJ=1,5
      B(KJ+3)=BA
```

```
  804 KLOUT(KJ)=IBLNK
      IZ=NREQD(IJ,1)*2
      DO 803 K=NO,NO1
      NO2=K-(I-1)*5
      DO 8886 KK=2,IZ,2
      IF(NREQD(IJ,KK).NE.K) GO TO 8886
      B(NO2+3)=BI
      KLOUT(NO2)=NREQD(IJ,KK+1)
 8886 CONTINUE
  803 CONTINUE
      MEXP2=MEXP(IJ)-(MEXP(IJ)/1000)*1000
 8887 WRITE(6,B) MEXP2,(KLOUT(K), K=1,5)
 8888 NO=NO1+1
      RETURN
      END


$IBFTC TRCK
      SUBROUTINE TRACK(IPRT,LLL)
      DIMENSION IDATA(36,25),ITOTL(120,27),ICTRT(120,5),IOUT(120),
     *          JDATE(36),IREPT(36),IDATE(36),ANAME(27),BNAME(27),
     *          MEXP(36),IMOD(3),XOCST(18),XCOST(8,36),IPERF(36),
     *          JCON(36,3),XXCST(18),IIOUT(120,36),NREQD(36,11),
     *          MAX(120,27),XMAX(18),XMAX1(18),
     *          IPERA(36),IITOT(120,27),XCCST(18),XTCST(18)
      COMMON IRAND,N,M,IN,IDATA,ITOTL,ICTRT,IOUT,IREPT,IDATE,MM,
     *       JDATE,MEXP,XCOST,IPERF,IRAN,XOCST,JCON,ISEN2,ZCOST,
     *       APER,NSKL ,LL,XXCST,IIN,IIOUT,NREQD,IAM,IFM,ICODE(12,37)
      COMMON/ALPHA/ANAME,BNAME,IMOD
C
      GO TO (1,4,8),IPRT
C
C     INITIALIZATION
C
    1 DO 60 I=1,IN
      DO 60 J=1,MM
   60 MAX(I,J)=-0.34359738E+09
      DO 61 I=1,18
      XTCST(I)=0.
      XMAX(I)=-1.0E+12
   61 XMAX1(I)=-1.0E+12
      ZMAX=-1.0E+12
      ZTCST=0.
      DO 330 I=1,N
  330 IPERA(I)=0
      DO 210 I=1,IN
      DO 211 J=1,N
  211 IIOUT(I,J)=0
      DO 210 J=1,MM
  210 IITOT(I,J)=0
      RETURN
C
C
C     TOTAL AND MAXIMUM CALCULATIONS
```

```
C
    4 DO 500 I=1,IN
      DO 601 K=1,MM
  601 IITOT(I,K)=ITOTL(I,K)+IITOT(I,K)
      DO 433 J=1,MM
      IF(MAX(I,J)-ITOTL(I,J)) 430,433,433
  430 MAX(I,J)=ITOTL(I,J)
  433 CONTINUE
  500 CONTINUE
      DO 320 I=1,N
  320 IPERA(I)=IPERA(I)+IPERF(I)
      DO 7113 I=1,18
      XTCST(I)=XTCST(I)+XOCST(I)
      IF(XMAX(I)-XOCST(I)) 633,634,634
  633 XMAX(I)=XOCST(I)
  634 IF(XXCST(I)-XMAX1(I)) 7113,7113,7111
 7111 XMAX1(I)=XXCST(I)
 7113 CONTINUE
      ZTCST=ZTCST+ZCOST
      IF(ZCOST-ZMAX) 8113,8113,8110
 8110 ZMAX=ZCOST
 8113 CONTINUE
      RETURN
C
C     OUTPUT
C
    8 PRINT 511,LLL
  511 FORMAT(1H1,41H FREQUENCY DISTRIBUTION OF SCHEDULING FOR ,I5,11H  S
     1CHEDULES  )
      RL=LLL
      PRINT 203, (I, I=1,IN)
  203 FORMAT(1X,8HINTERVAL ,I3,19I4,5(/8X,20I4))
      PRINT 204
      PRINT 205
  204 FORMAT(1H+,87X,14H  PERFORMANCES)
  205 FORMAT(1X,87(1H-),4X,9HREQ  SCHD)
      DO 509 J=1,N
      IPERR   =IFIX(FLOAT(IPERA(J))/RL+0.5)
      MEXP2=MEXP(J)-(MEXP(J)/1000)*1000
      PRINT 1301,IREPT(J),IPERR
 1301 FORMAT(89X,2I6)
  509 PRINT 1300,MEXP2,(IIOUT(I,J),I=1,IN)
 1300 FORMAT(1H+,3HFPE,1X,I3,20I4/,(8X,20I4))
      PRINT 206
  206 FORMAT(1X,100(1H-))
      PRINT 502, LLL
  502 FORMAT(/1X,29HAVERAGE RESOURCES UTILIZED IN,2X,I3,2X,9HSCHEDULES)
      PRINT 12, (I, I=1,IN)
      DO 314 I=1,IN
      DO 314 J=1,MM
  314 IITOT(I,J)=IFIX(FLOAT(IITOT(I,J))/RL+0.5)
      DO 301 K=1,M
  301 PRINT 14, ANAME(K),BNAME(K), (IITOT(J,K), J=1,IN)
      DO 302 K=20,27
  302 PRINT  14, ANAME(K),BNAME(K), (IITOT(J,K), J=1,IN)
      PRINT 208
```

```
208   FORMAT(1X,21HAVERAGE COST PER YEAR)
      DO 45 I=1,18
   45 XTCST(I)=XTCST(I)/RL
      ZACST=ZTCST/RL
      XCCST(1)=XTCST(1)
      DO 480 I=2,18
  480 XCCST(I)=XCCST(I-1)+XTCST(I)
      PRINT 16, (I, I=1970,1979)
      PRINT 17, (XTCST(I), I=1,10)
      PRINT 17, (XCCST(I), I=1,10)
      PRINT 18, (I, I=1980,1987)
      PRINT 17, (XTCST(I), I=11,18), ZACST
      PRINT 17, (XCCST(I), I=11,18)
   12 FORMAT(10H INTERVAL  ,20I6/5(10X,20I6/))
   14 FORMAT(1X,2A4,1X,20I6/5(10X,20I6/))
   16 FORMAT(1X,10I10)
   17 FORMAT(1X,10F10.2)
   18 FORMAT(1X,8I10,4X,5HTOTAL)
C
C
C
      PRINT 680, LLL
  680 FORMAT(1H1,30HMAXIMUM VALUE OF RESOURCES FOR,I4,10H SCHEDULES/)
      PRINT 12, (I, I=1,IN)
      DO 13 K=1,M
   13 PRINT 14, ANAME(K),BNAME(K),(MAX(J,K), J=1,IN)
      DO 7220 K=20,27
 7220 PRINT 14, ANAME(K),BNAME(K),(MAX(J,K), J=1,IN)
      PRINT 16, (I,I=1970,1979)
      PRINT 17, (XMAX(I), I=1,10)
      PRINT 17, (XMAX1(I), I=1,10)
      PRINT 18, (I,I=1980,1987)
      PRINT 17, (XMAX(I), I=11,18), ZMAX
      PRINT 17, (XMAX1(I), I=11,18)
      DO 451 I=1,IN
      DO 451 K=1,MM
  451 IITOT(I,K)=IITOT(I,K)*LLL
      DO 452 I=1,18
  452 XTCST(I)=XTCST(I)*RL
      RETURN
      END


$ENTRY
  2  6

   34    20   200     9    525811
 2010     0     4    20    25     0    99    72     6    24    023244 8125
    2     1   438     7   438
 2020     0     4    20    37     0  3660   222     6   222    62438810598
    2     1   144     8   102
 2030     4     4    20    50    025600   510    12   492   122269610598
    3     1   288     7    30     8   162
 1040     0     4     4  1120  4326    26    30     0    30    018048 6888
    2     1   192     8    60                                    1210
```

2050	0	4	4	50	0	46	726	12	138	63182810598		
3	1	276	7	102	8	102						
0060	3	3	1	31	96	17	0	0	48	0	157	8
1	8	102										
0071	0	1	3	0	24	3	108	6	6	0	312	11
2	7	102	8	90								
0072	0	1	3	1	6	2	114	6	12	0	214	13
3	1	60	7	30	8	60						
0073	0	1	8	81	1530	2	108	0	1	0	321	5
2	1	90	7	42								
0074	0	1	8	150	84	2	114	0	12	0	838	21
2	1	102	7	42								
0075	0	1	8	150	0	2	108	0	12	0	161	5
2	1	30	7	144								
2080	0	4	20	56	0	864	1158	24	588	1840424		8125
3	1	234	7	42	8	60					1200	
1091	0	4	20	255910692		6	2760	120	60	616805		505
2	2	522	3	726								
1092	0	4	20	1000	0	6	2760	114	54	01511510598		
1	3	18										
0100	0	1	20	433	504	14	78	6	18	0	912	104
2	2	396	7	18								
1110	0	4	20	1780	463849000		1806	48	1656	422510010598		
2	7	114	8	1266								
2120	0	4	4	17	0	53	5772	90	30	0	7276	6888
2	7	60	8	72								
0131	0	0	20	290	228	66	72	6	66	6	1450	319
3	4	1296	5	1134	9	1836						
1132	0	4	20	1155	4320	78	78	0	66	013800		9500
5	4	450	5	672	6	1602	7	90	9	2196		
0140	1	0	7	167	150	0	138	6	138	6	587	31
4	5	306	7	366	8	12	9	540				
0150	0	1	20	200014256		0	72	12	36	0	2453	103
3	7	702	8	702	9	702						
0160	0	1	4	149	1542	7	390	24	288	12	1566	103
3	1	318	7	60	8	90						
0170	0	0	4	80	216	1	480	6	480	6	1895	49
2	7	612	8	132								
0180	0	0	4	22	84	0	282	6	162	6	670	42
2	7	144	8	132								
1200	0	4	10	20	4704	1	84	6	528	1821916		6888
2	7	318	8	114							2080	
1210	0	4	20	1000	4320	161	156	6	6	019607		6888
2	1	174	8	30							1040	
0220	1	4	1	568	6396	3	18	0	18	0	7420	120
2	1	480	7	846								
0244	0	1	4	5	24	0	48	6	156	6	2050	316
3	6	42	7	144	8	192						
0245	1	0	2	0	18	0	48	0	156	0	670	60
1	8	114										
0246	1	0	2	1	0	0	48	0	156	6	3170	131
3	1	30	7	30	8	60						
0247	1	0	4	75	0	0	48	0	156	0	330	19
3	4	42	5	60	7	42						
0250	0	1	20	43	258	0	66	6	60	0	361	37
2	2	174	8	30								

```
0260     0    1   20   81  408     0  108     6  108     6  564    42
   2     3  204    8   42
0270     0    1    4   45  132     4  828    30  192    24 1287    53
   2     1  348    8   42
A WATTS
99999999999999999999999999999999999999999999999999999999999999999999999999999999
99999999999999999999999999999999999999999999999999999999999999999999999999999999
KW-HRS
99999999999999999999999999999999999999999999999999999999999999999999999999999999
99999999999999999999999999999999999999999999999999999999999999999999999999999999
BIT RT
99999999999999999999999999999999999999999999999999999999999999999999999999999999
99999999999999999999999999999999999999999999999999999999999999999999999999999999
MAN HRS
99999999999999999999999999999999999999999999999999999999999999999999999999999999
99999999999999999999999999999999999999999999999999999999999999999999999999999999
SKILL 1 SKILL 2 SKILL 3 SKILL 4 SKILL 5 SKILL 6 SKILL 7 SKILL 8 SKILL 9
SS VOL
99999999999999999999999999999999999999999999999999999999999999999999999999999999
99999999999999999999999999999999999999999999999999999999999999999999999999999999
00000000000000000000
```

2

		4500.	18000.	27000.	27500.	18500.
7500.	45000.	45000.	60000.	52500.	57500.	27500.
6000.	36000.	6070.	5620.	6940.	16940.	22420.
			4500.	15000.	20500.	14500.
			17110.	21270.	21170.	13410.
				1500.	3000.	3000.
				230.	750.	750.
				230.	1130.	1130.
			380.	1500.	1500.	1500.
			190.	1130.	1130.	1130.
			190.	1130.	1130.	1130.
	4950.	12990.	19370.	19370.	14880.	21880.
		2540.	3140.	3750.	6220.	3820.
		290.	440.	2250.	9250.	9250.
		900.	1350.	1650.	1880.	1170.
	3000.	4500.	4500.	15000.	28000.	28000.
			210.	2080.	5110.	4110.
		650.	6560.	19080.	29600.	19080.
		290.	2890.	8400.	18520.	13900.
	9000.	19500.	22500.	27000.	24000.	6000.
	9000.	22500.	30000.	36000.	36000.	9000.
900.	3750.	4500.	4500.	9000.	9000.	6000.
	450.	900.	1500.	3000.	2700.	750.
	900.	1500.	3000.	4500.	3000.	1500.
600.	1350.	1950.	4500.	4500.	11500.	7000.
			1500.	8250.	20500.	10750.
	600.	2250.	3000.	7500.	6000.	3000.
		5570.	8640.	5570.	1910.	190.
						290.
				1570.	1260.	140.
	150.	1100.	1770.	2630.	3300.	3150.
		2540.	3140.	3750.	6220.	3820.
			1500.	1500.	9000.	9000.

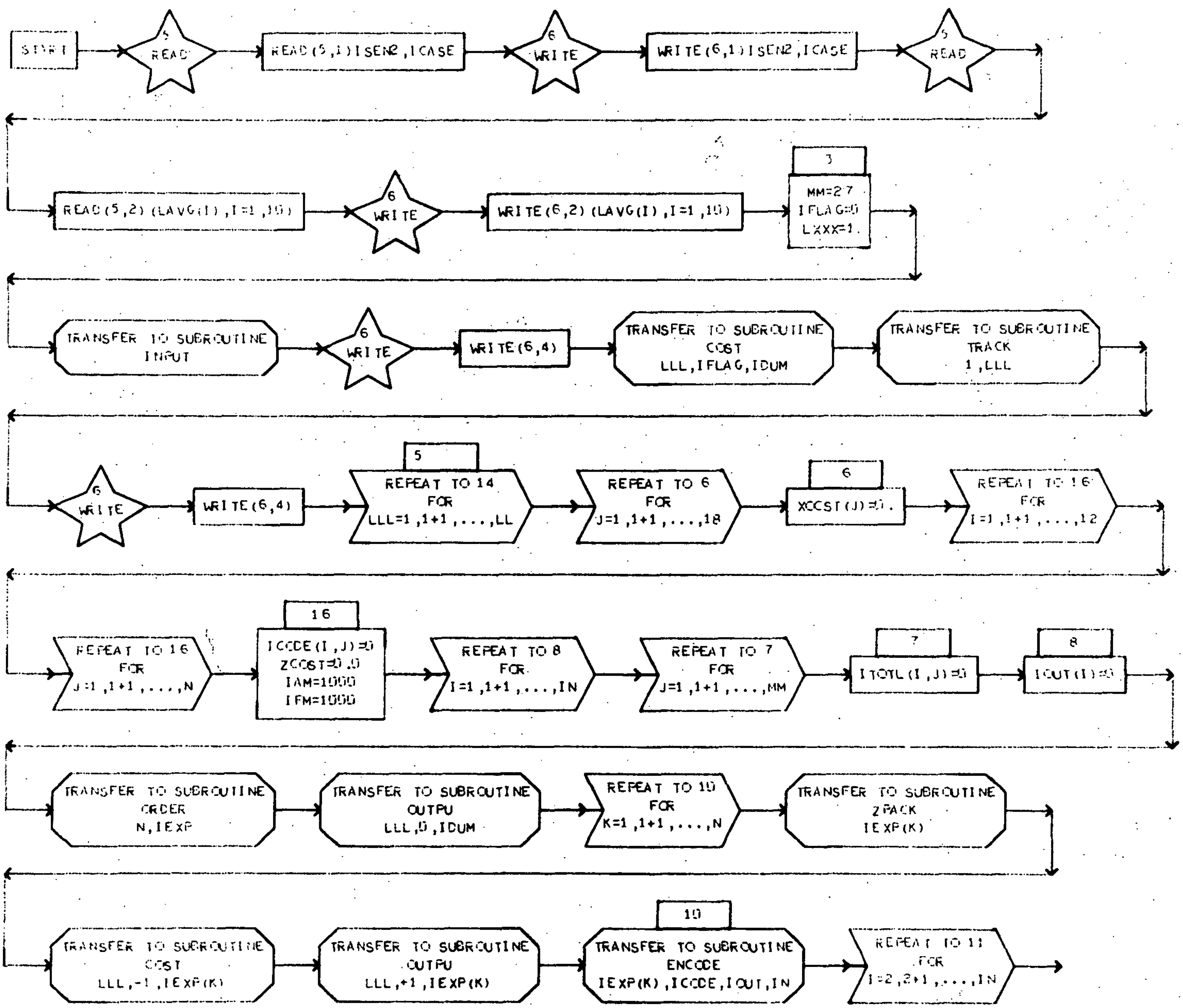
61900. 97900. 68600. 32400. 9200.
18310. 29020. 20360. 9560. 2750.
$IBSYS
$IBSYS
$IBSYS

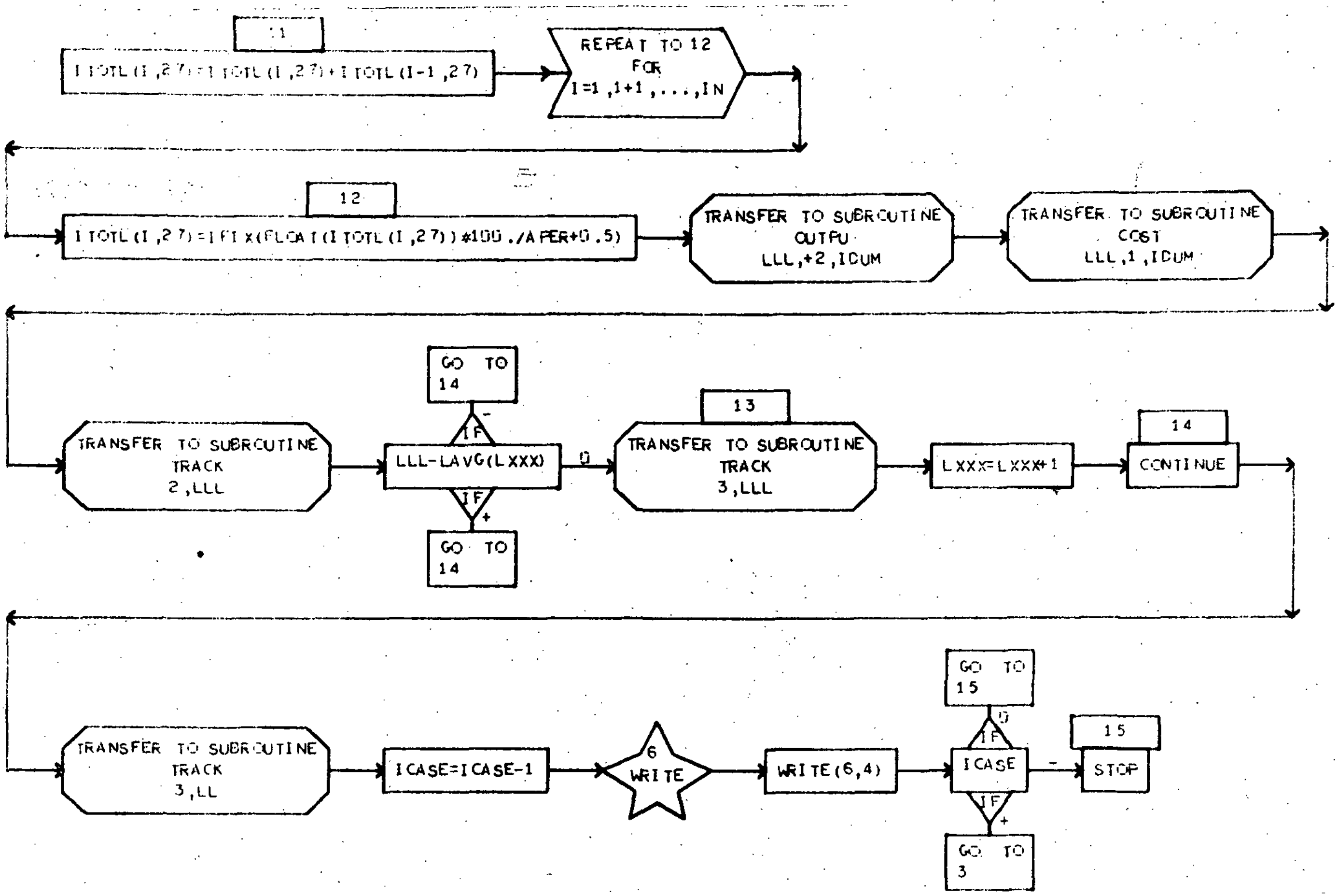
Appendix C

FLOWTRAN LOGIC PLOTS OF REPRI

DIMENSIONED VARIABLES

SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES
IDATA	36,25	ITOTL	120,27	ICTRT	120,5	ICUT	120	JDATE	36
IREPT	36	IDATE	36	NAME	27	BNAME	27	MEXP	36
IMOD	3	XCCST	18	XCOST	8,36	IPERF	36	JCON	36,3
XXCST	18	ICUT	120,36	NREQD	36,1	IEXP	36	LAVG	10

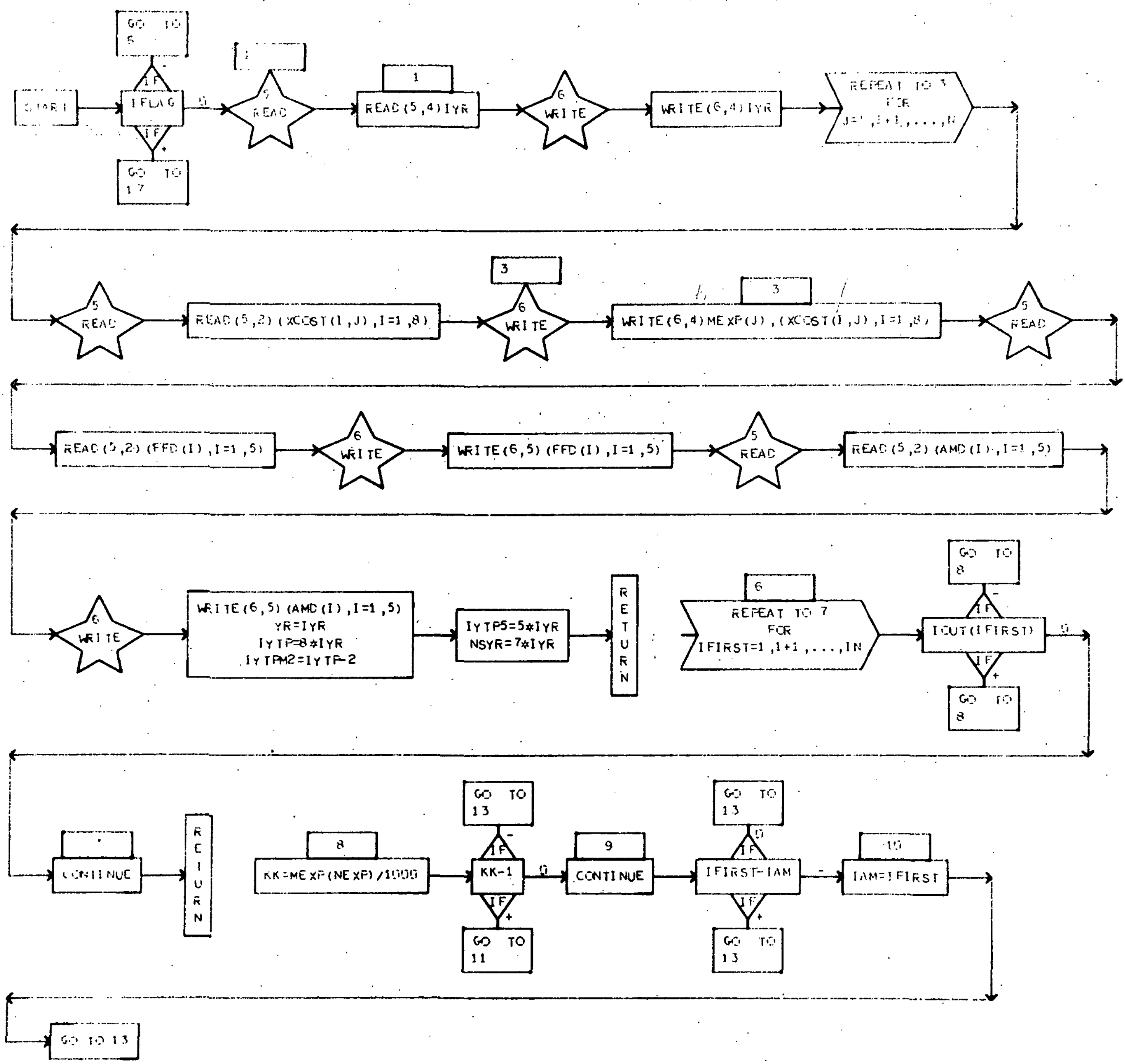



11
ITOTL(I,27)=ITOTL(I,27)+ITOTL(I-1,27)
REPEAT TO 12
FOR
I=1,1+1,...,IN
12
ITOTL(I,27)=IFIX(FLOAT(ITOTL(I,27))*100./APER+0.5)
TRANSFER TO SUBROUTINE
OUTPU
LLL,+2,ICUM
TRANSFER TO SUBROUTINE
COST
LLL,1,ICUM
TRANSFER TO SUBROUTINE
TRACK
2,LLL
GO TO
14
IF
LLL-LAVG(LXXX)
GO TO
14
13
TRANSFER TO SUBROUTINE
TRACK
3,LLL
LXXX=LXXX+1
14
CONTINUE
TRANSFER TO SUBROUTINE
TRACK
3,LL
ICASE=ICASE-1
6
WRITE
WRITE(6,4)
GO TO
15
ICASE
GO TO
3
15
STOP

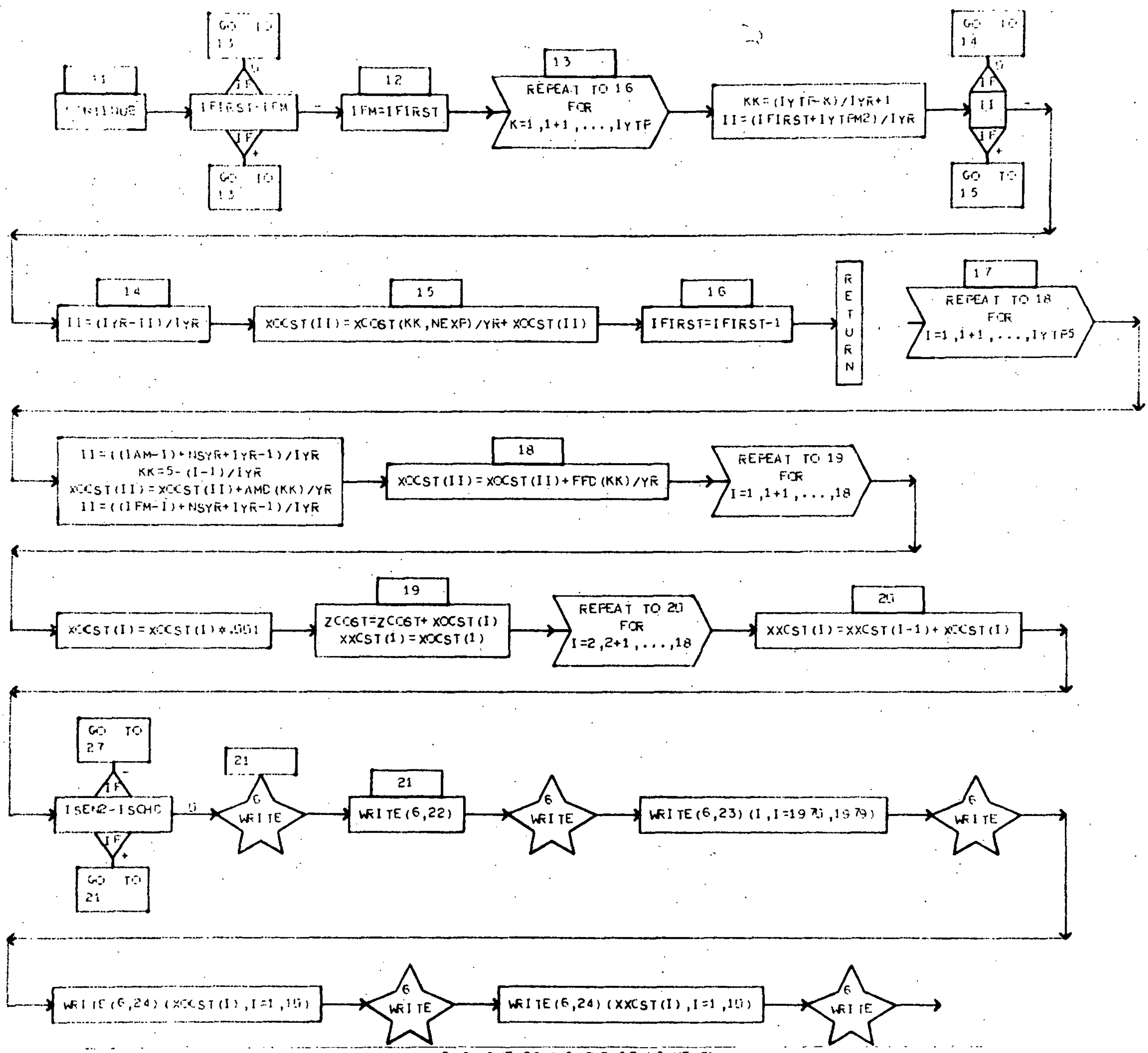
DIMENSIONED VARIABLES

SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES
IDATA	36,25	ITOTL	120,27	ICTRT	120,5	ICUT	120	JDATE	36
IREPT	36	IDATE	36	FD	5	AMD	5	MEXP	36
XCOST	18	XCOST	8,36	IPERF	36	JCON	36,3	XXCST	18
IICUT	120,36	NREQD	36,11						

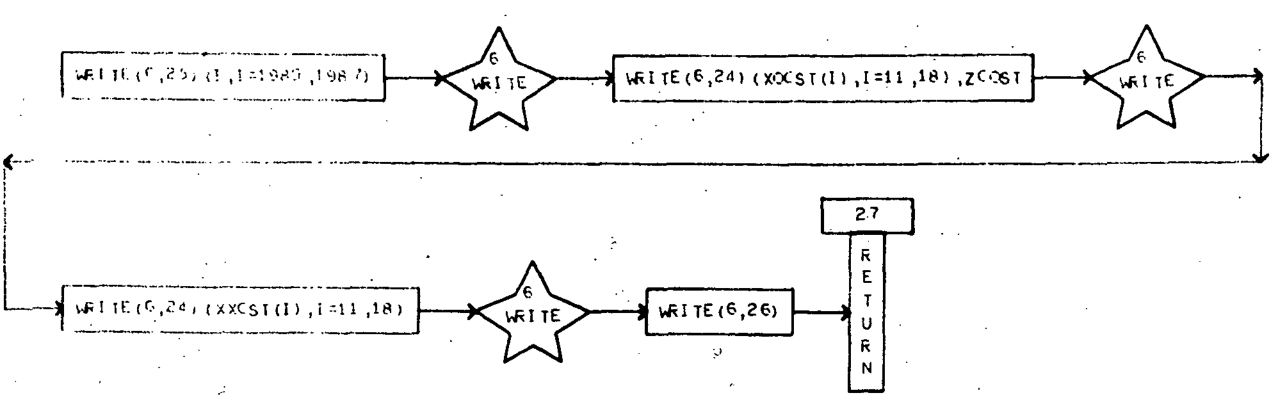
SUBROUTINE COST(ISCHD,IFLAG,NEXP)



SUBROUTINE COST(ISCHD,IFLAG,NEXP)

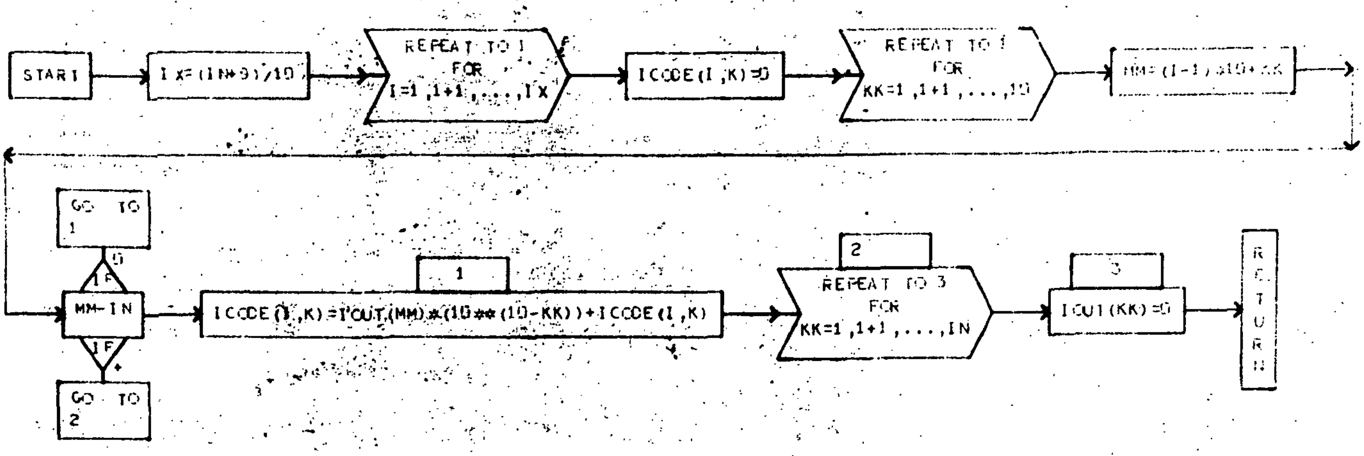


SUBROUTINE COST(ISCHD,IFLAG,NEXP)



SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES
ICODE	12,37	IOUT	1						

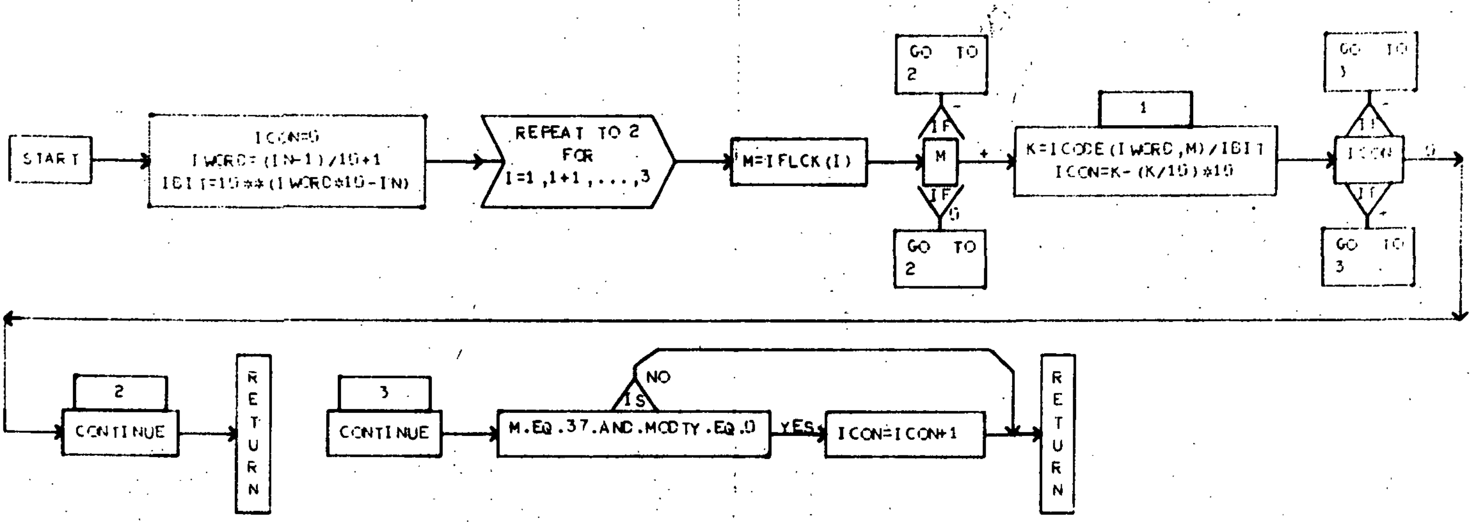
SUBROUTINE ENCODE (K,ICODE,IOUT,IN)



D I M E N S I O N E D   V A R I A B L E S

SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES
ICODE	12,37	IFLCK	3						

FUNCTION ICON(ICODE,MODTY,IN,IFLCK) PAGE 1

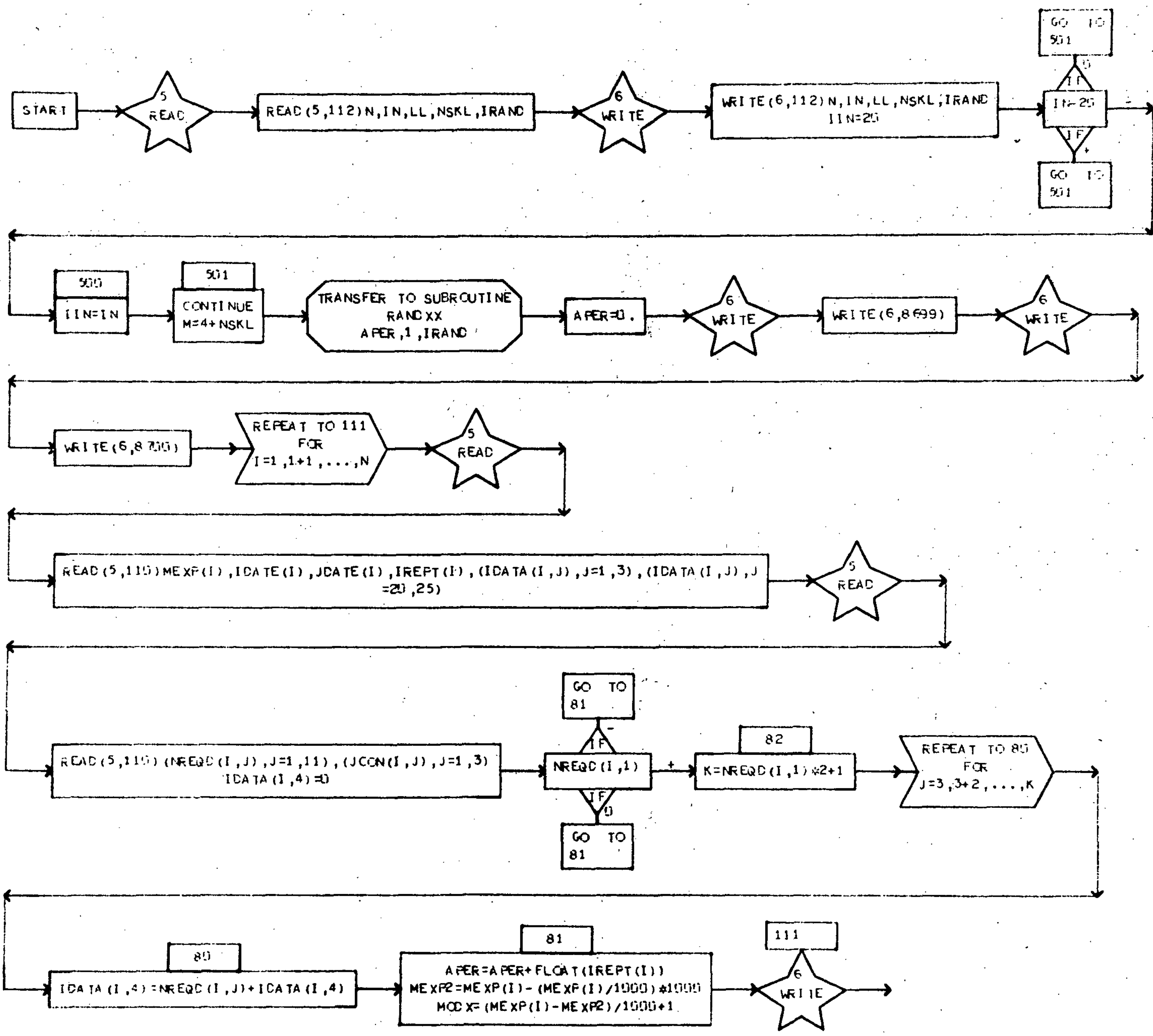


DIMENSIONED VARIABLES

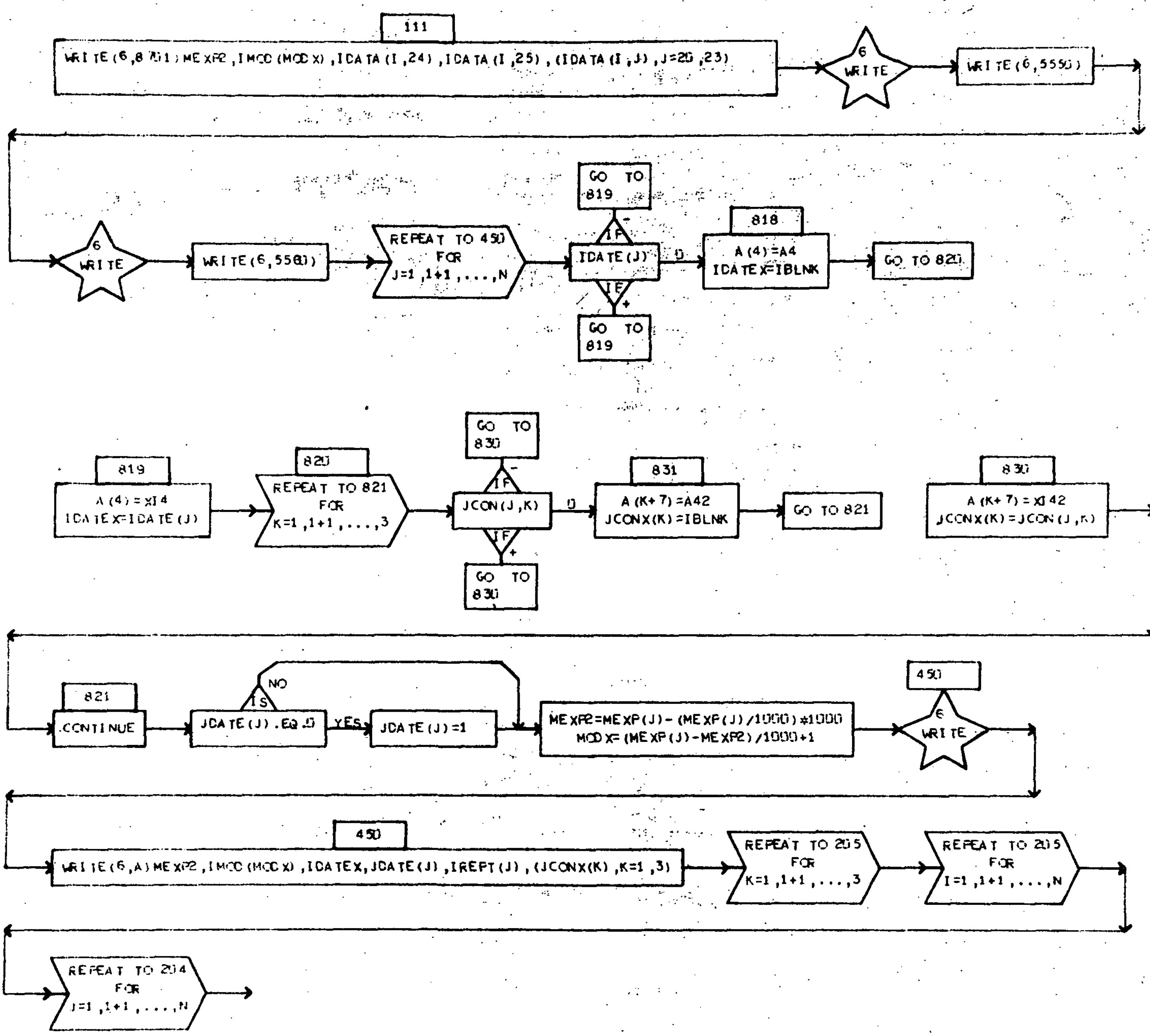
SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES
IDATA	36,25	ITOTL	120,27	ICTRT	120,5	ICUT	120	JDATE	36
IREPT	36	IDATE	36	NAME	27	BNAME	27	MEXP	36
IMOD	3	XOCST	18	XCOST	8,36	IPERF	36	JCON	36,3
XXCST	18	ICUT	120,36	NREQD	36,1	A	11	JCONX	3
D	9	KLOUT	5						

SUBROUTINE INPUT

PAGE 1

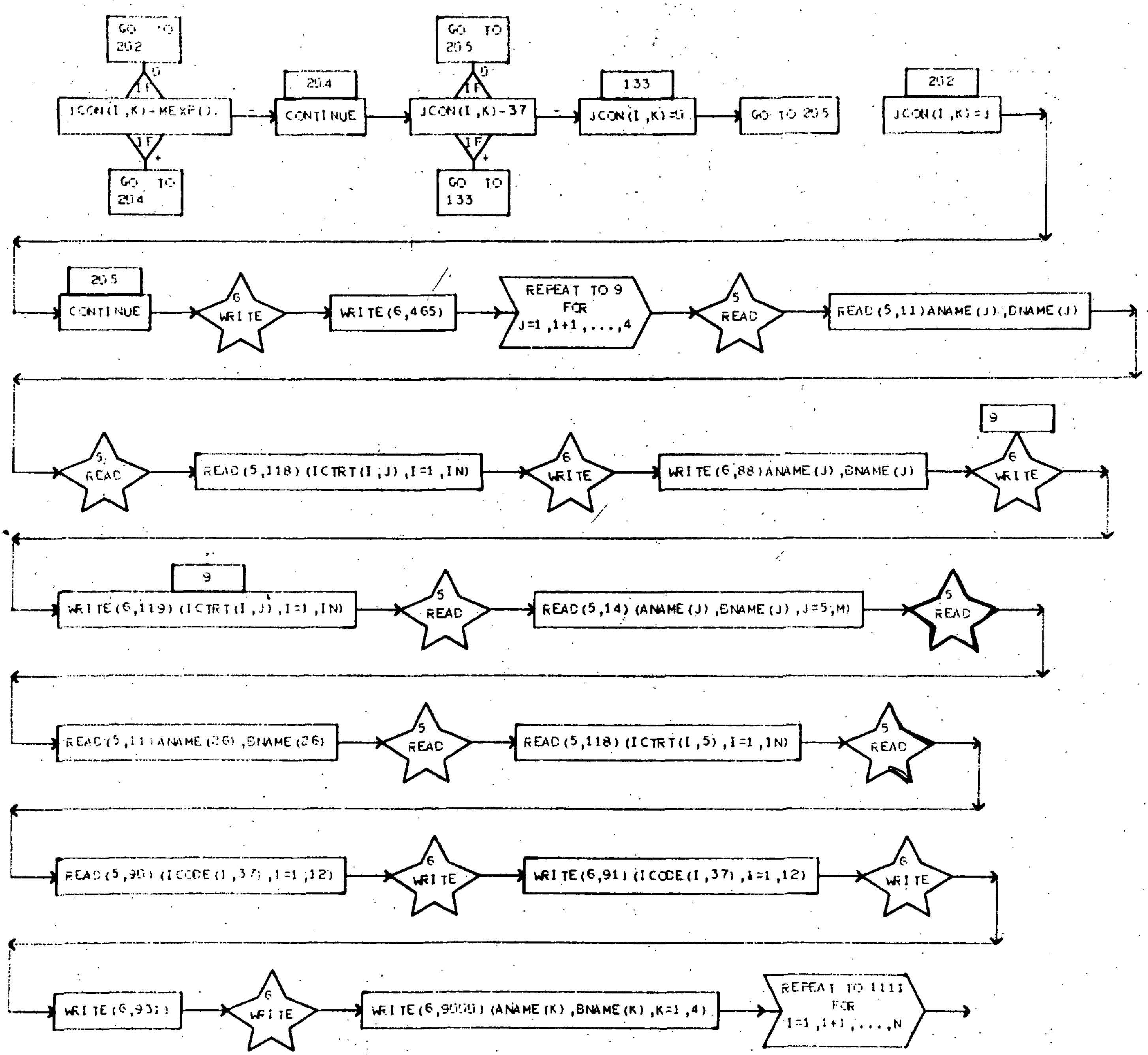


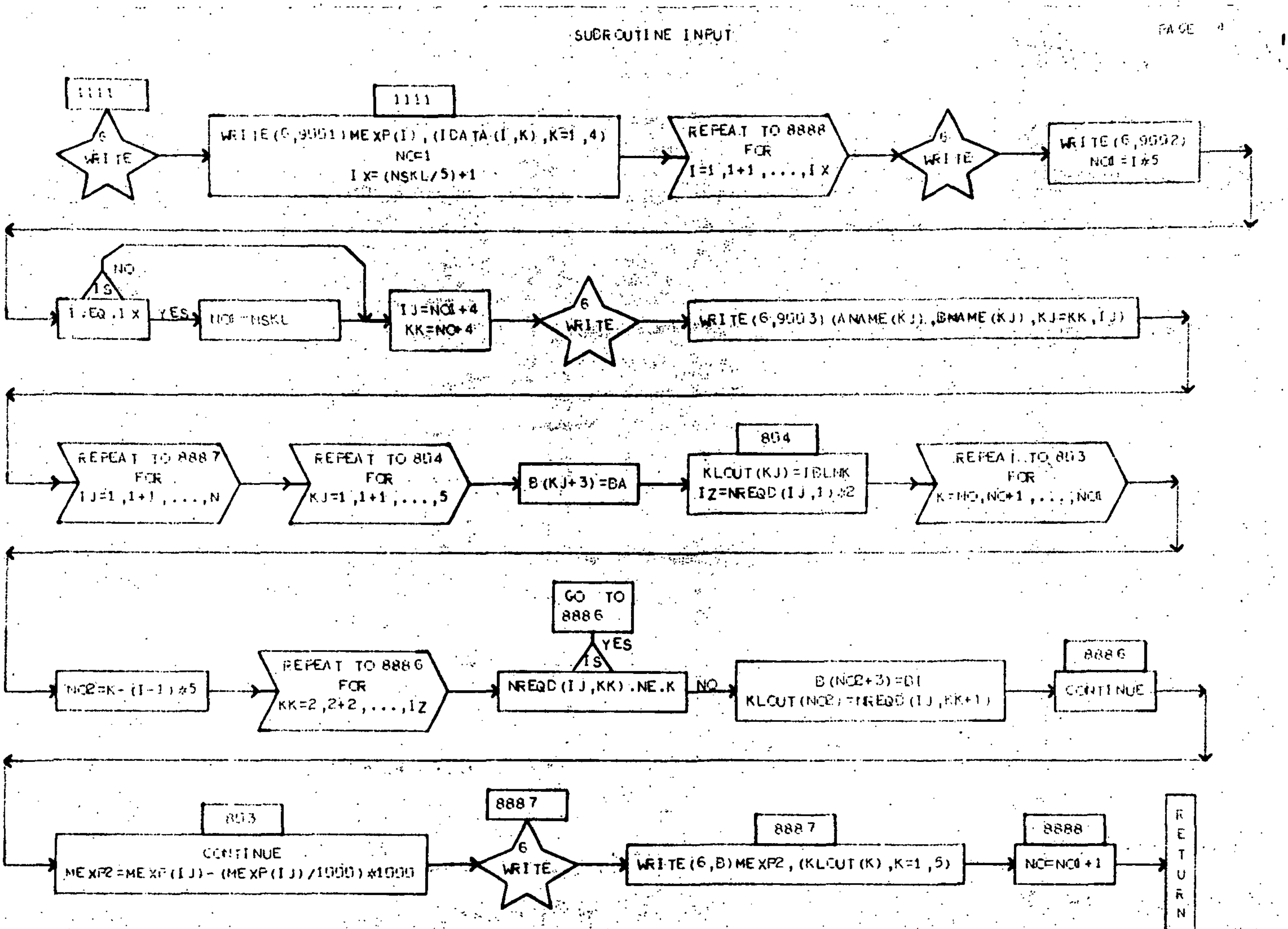
SUBROUTINE INPUT PAGE 2



SUBROUTINE INPUT

PAGE 3

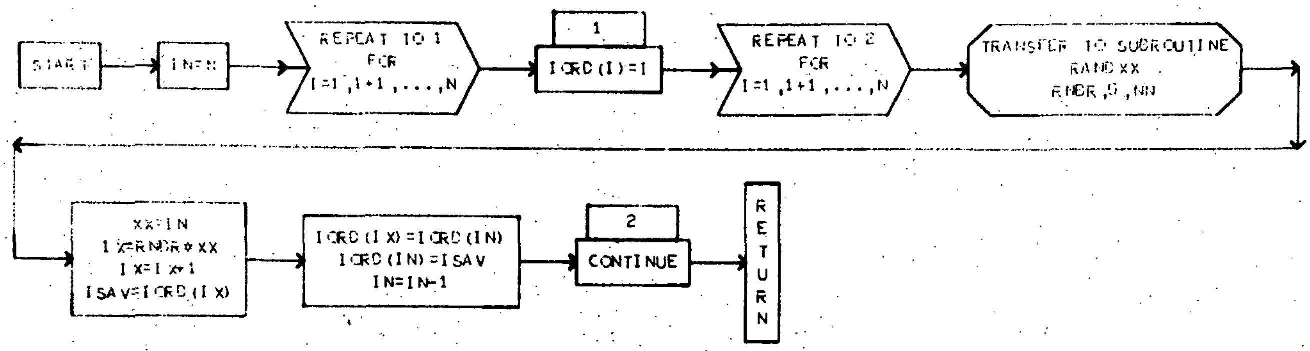



SUBROUTINE INPUT
PAGE
1111
6 WRITE
1111
WRITE(6,9001)MEXP(I),(IDATA(I,K),K=1,4)
NC=1
IX=(NSKL/5)+1
REPEAT TO 8888 FOR I=1,1+1,...,IX
6 WRITE
WRITE(6,9002)
NC1=I*5
IS I.EQ.IX
NO
YES
NC1=NSKL
IJ=NC1+4
KK=NC+4
6 WRITE
WRITE(6,9003)(ANAME(KJ),BNAME(KJ),KJ=KK,IJ)
REPEAT TO 8887 FOR IJ=1,1+1,...,N
REPEAT TO 804 FOR KJ=1,1+1,...,5
B(KJ+3)=BA
804
KLCUT(KJ)=IBLNK
IZ=NREQD(IJ,1)*2
REPEAT TO 803 FOR K=NC,NC+1,...,NC1
NC2=K-(I-1)*5
REPEAT TO 8886 FOR KK=2,2+2,...,IZ
GO TO 8886
YES
IS NREQD(IJ,KK).NE.K
NO
B(NC2+3)=BI
KLCUT(NC2)=NREQD(IJ,KK+1)
8886
CONTINUE
803
CONTINUE
MEXP2=MEXP(IJ)-(MEXP(IJ)/1000)*1000
8887
6 WRITE
8887
WRITE(6,B)MEXP2,(KLCUT(K),K=1,5)
8888
NC=NC1+1
RETURN

DIMENSIONED VARIABLES

SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES
IORD	1								

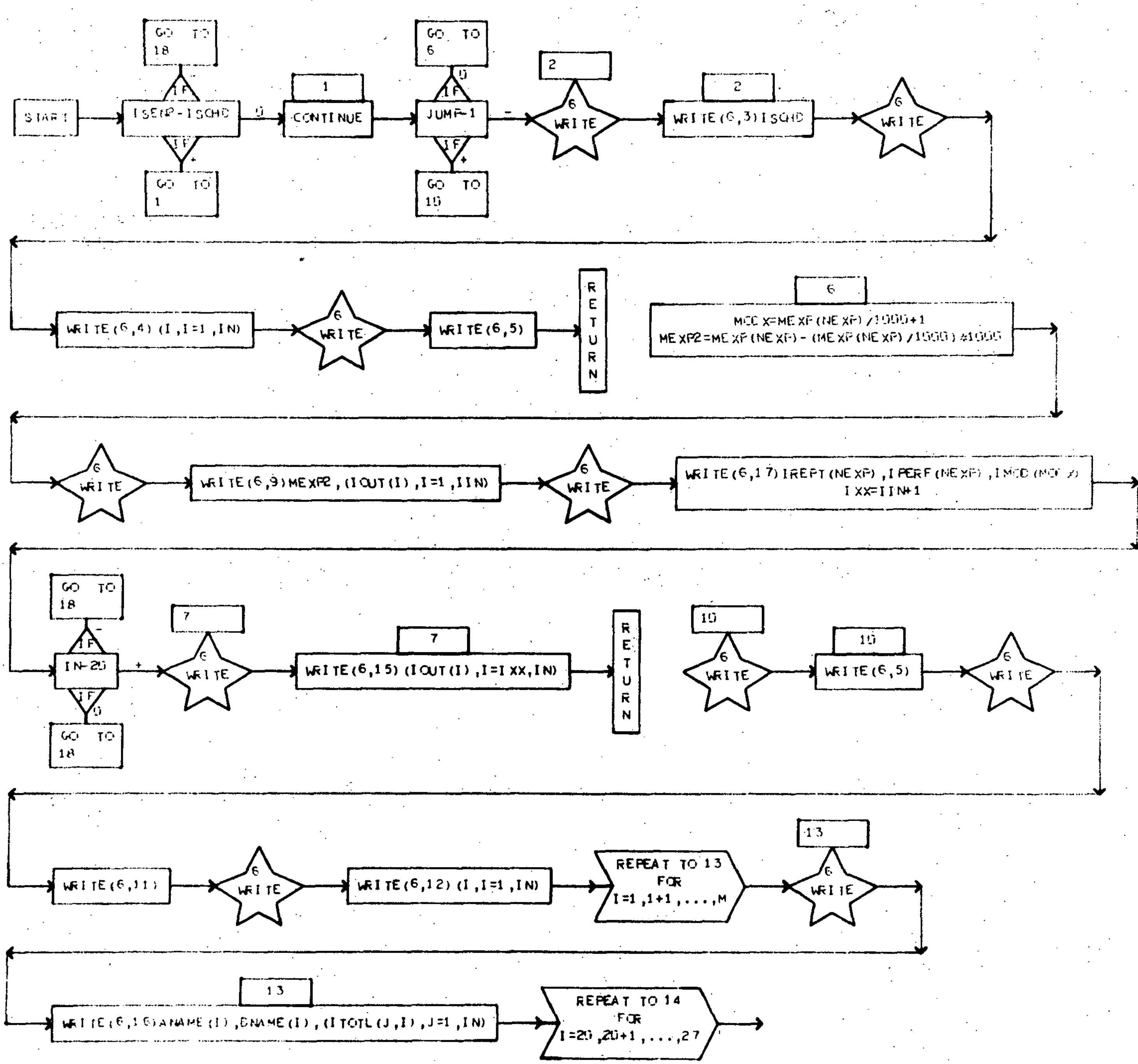
SUBROUTINE ORDER(N,IORD)



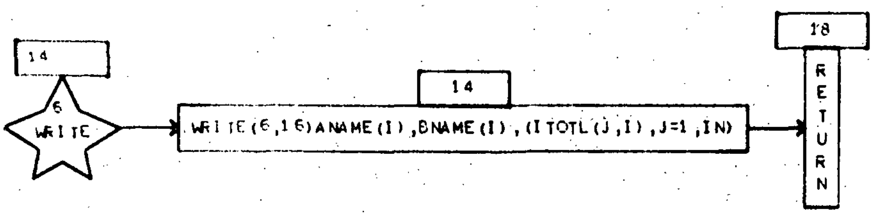
DIMENSIONED VARIABLES

SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES
IDATA	36,25	ITOTL	120,27	ICTRT	120,5	IOUT	120	JDATE	36
IREPT	36	IDATE	36	NAME	27	BNAME	27	MEXP	36
IMOD	3	XOCST	18	XCOST	8,36	IPERF	36	JCON	36,3
XXCST	18	IOUT	120,36	NREQD	36,1				

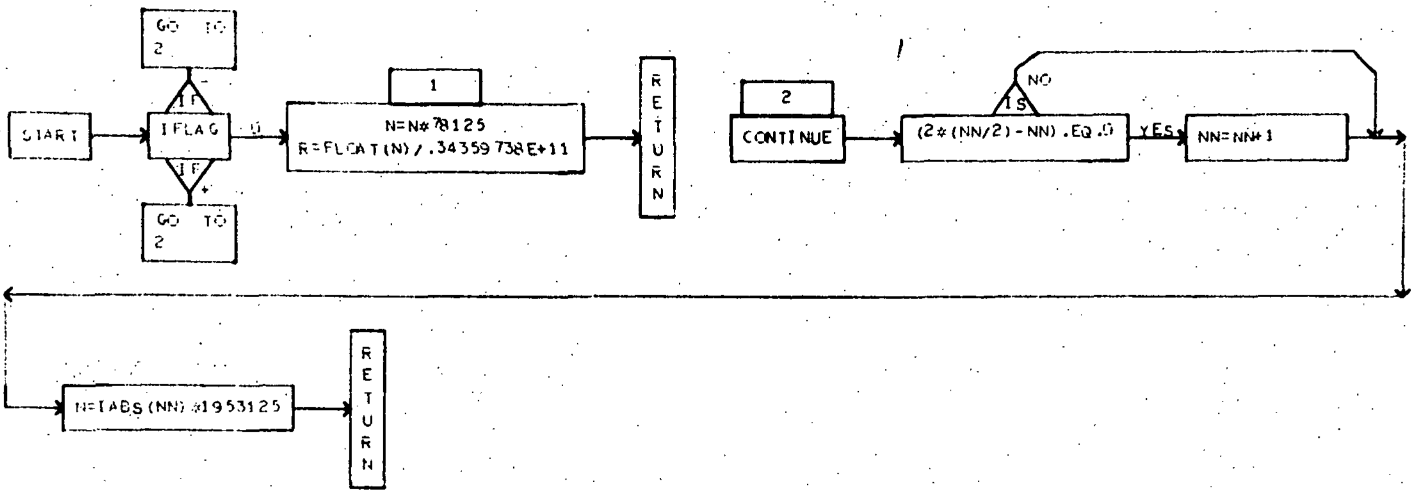
SUBROUTINE OUTPU (ISCHD,JUMP,NEXP)



SUBROUTINE OUTPU (ISCHD,JUMP,NEXP) PAGE 2



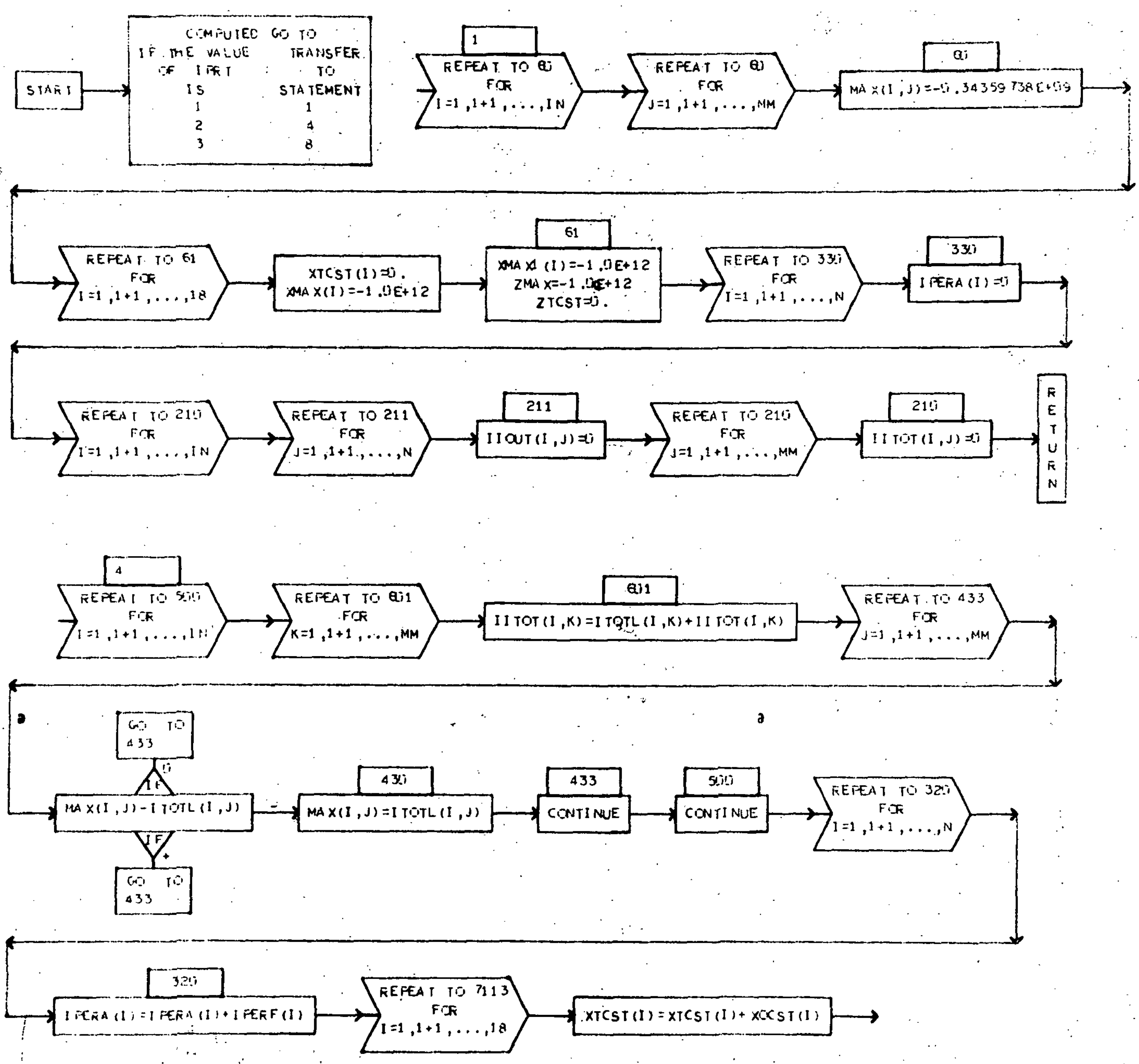
SUBROUTINE RANDXX(R,IFLAG,NN)



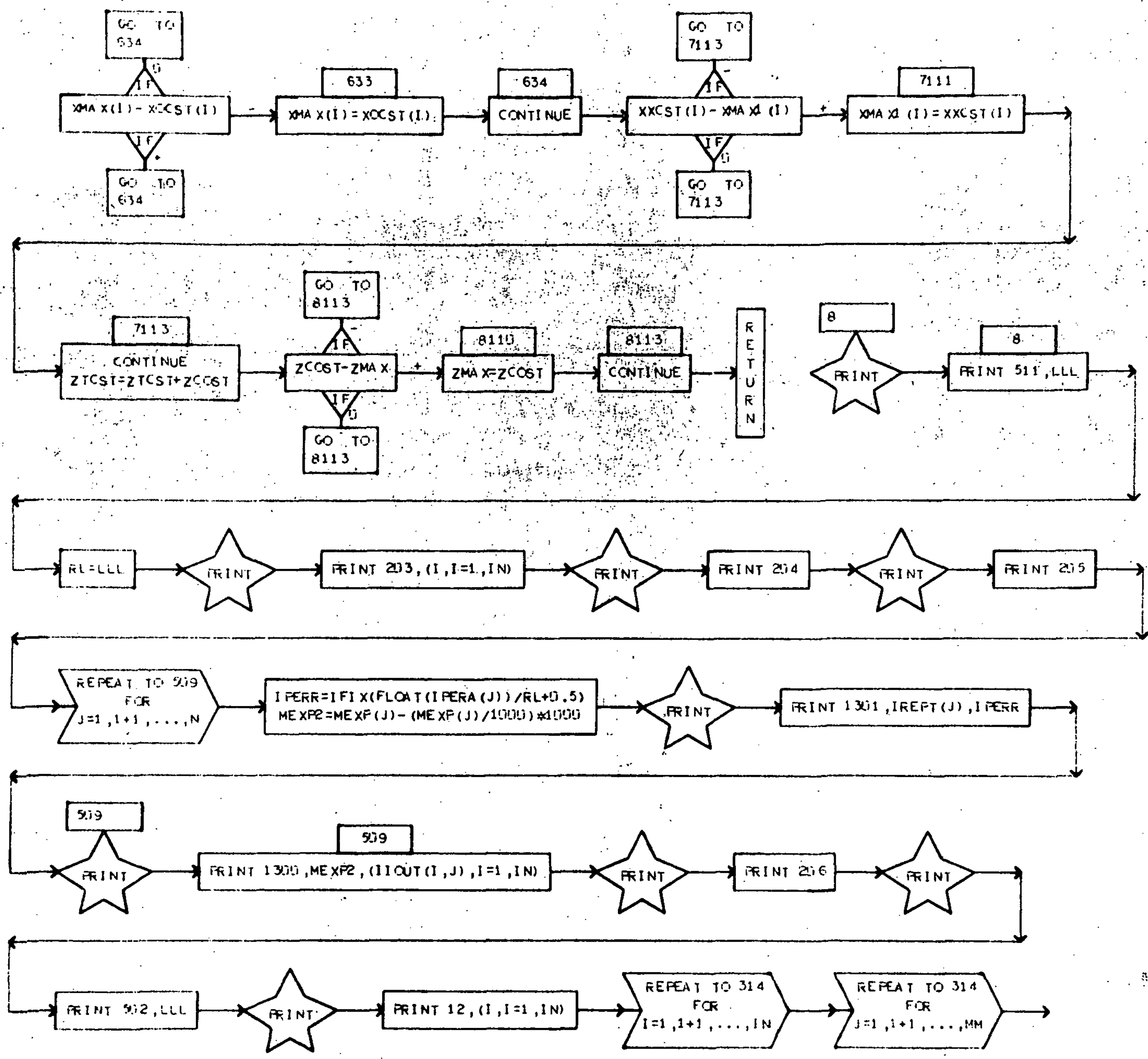
DIMENSIONED VARIABLES

SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES
IDATA	36,25	ITOTL	120,27	ICTRT	120,5	ICUT	120	JDATE	36
IREPT	36	IDATE	36	NAME	27	BNAME	27	MEXP	36
IMOD	3	XCCST	18	XCOST	8,36	IPERF	36	JCON	36,3
XXCST	18	ICUT	120,36	NREQD	36,1	MAX	120,27	XMA	18
XMAX	18	IPERA	36	IITOT	120,7	XCCST	18	XTCST	18

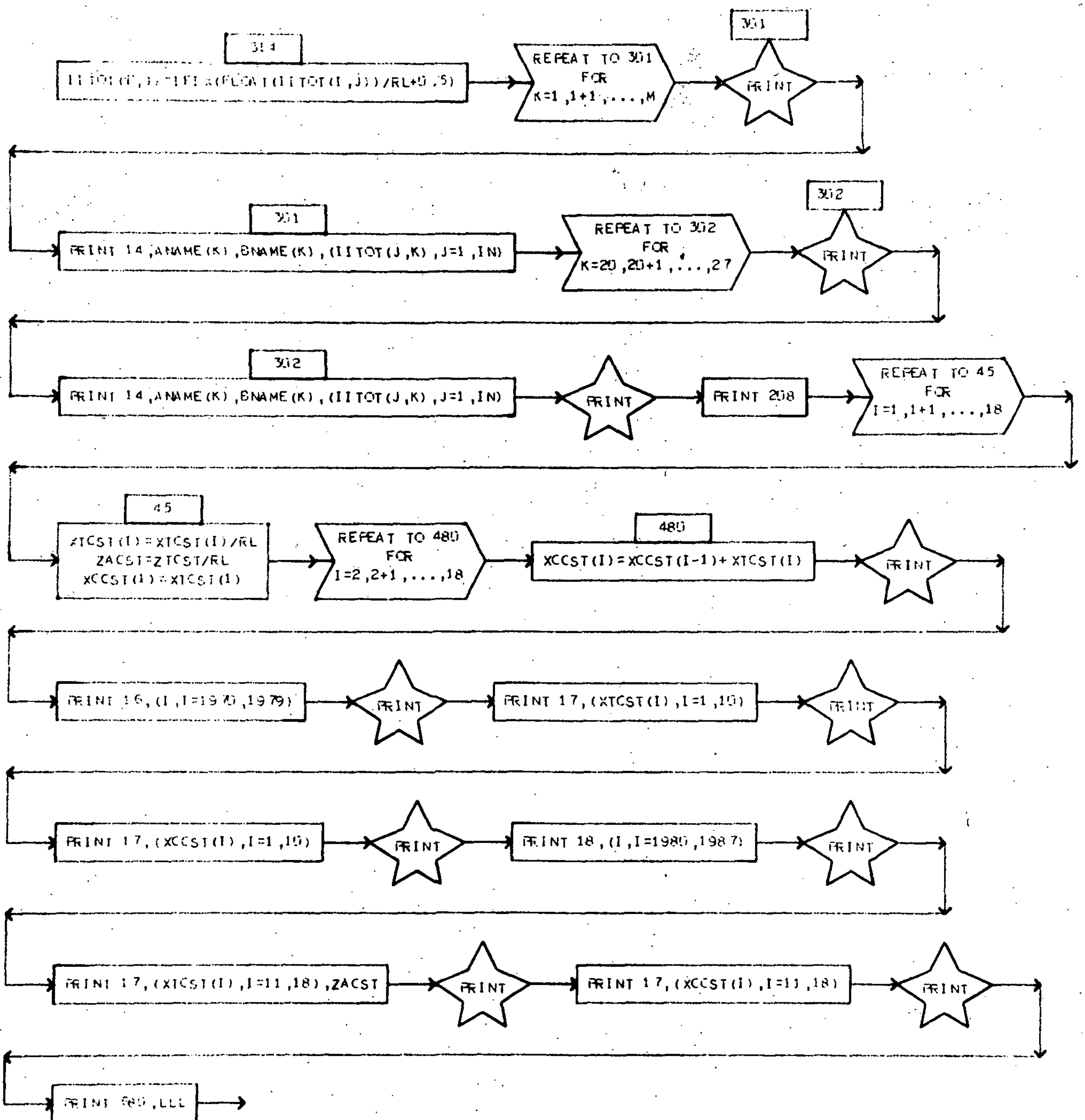
SUBROUTINE TRACK(IPRT,LLL)



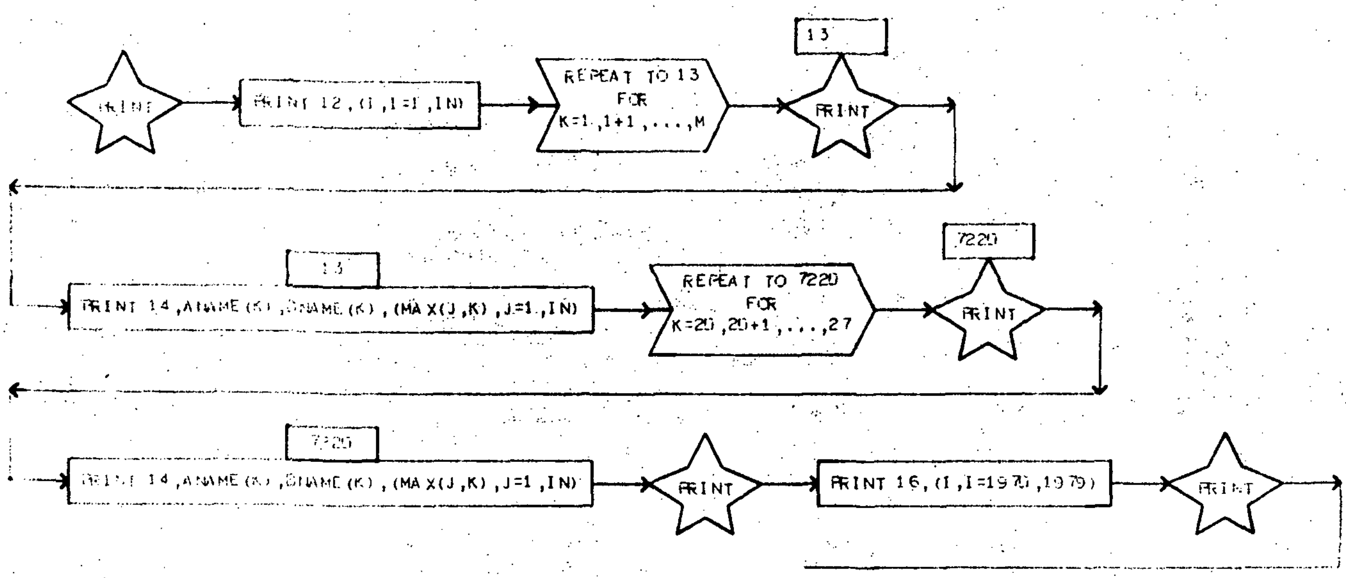
SUBROUTINE TRACK(IPRT,LLL)



SUBROUTINE TRACK(IPRT,LLL)



SUBROUTINE TRACK(IPRT,LLL)

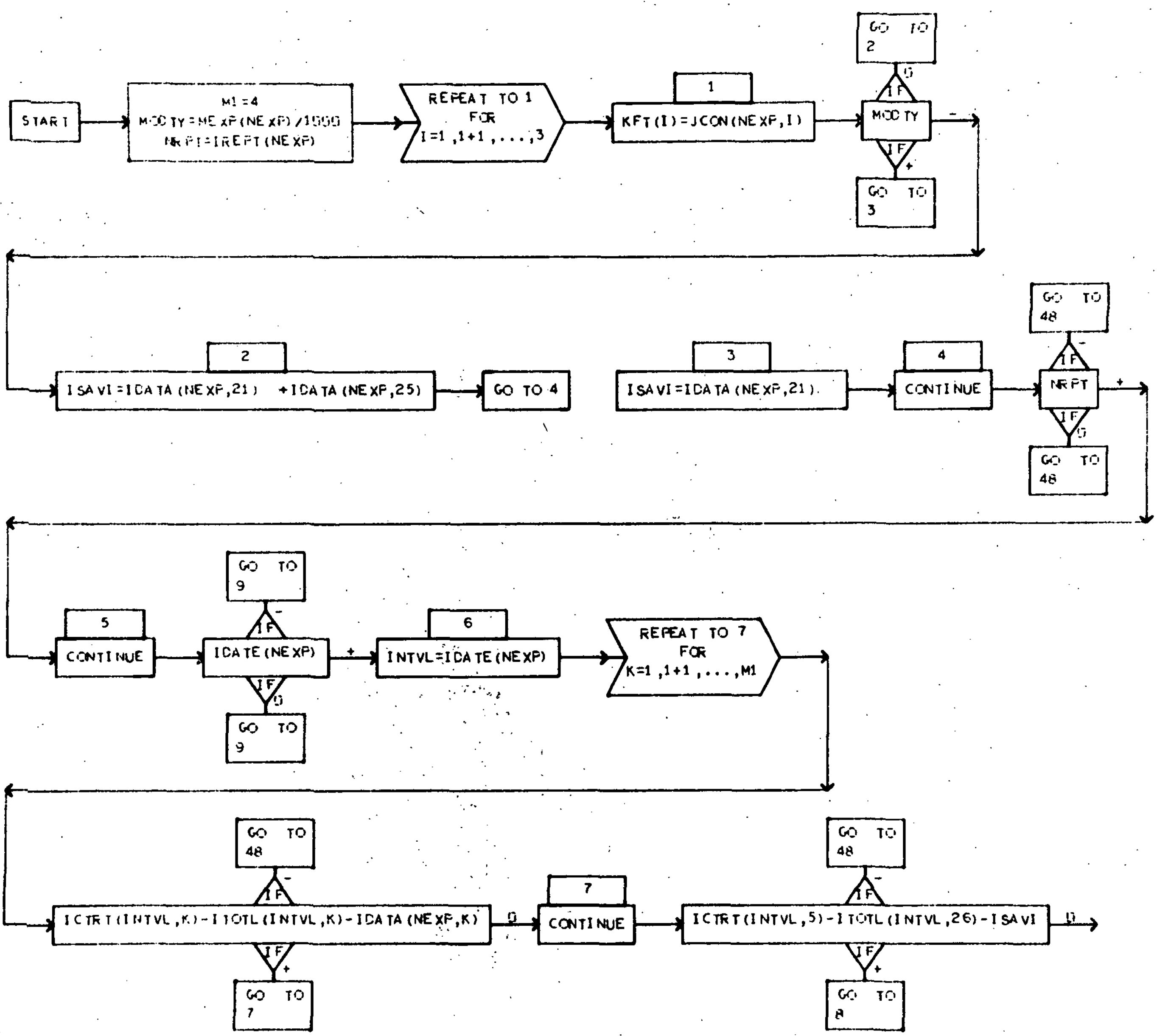


43RAN2SEND TO ...RUDY HAYES BIN 200 ROOM M-128

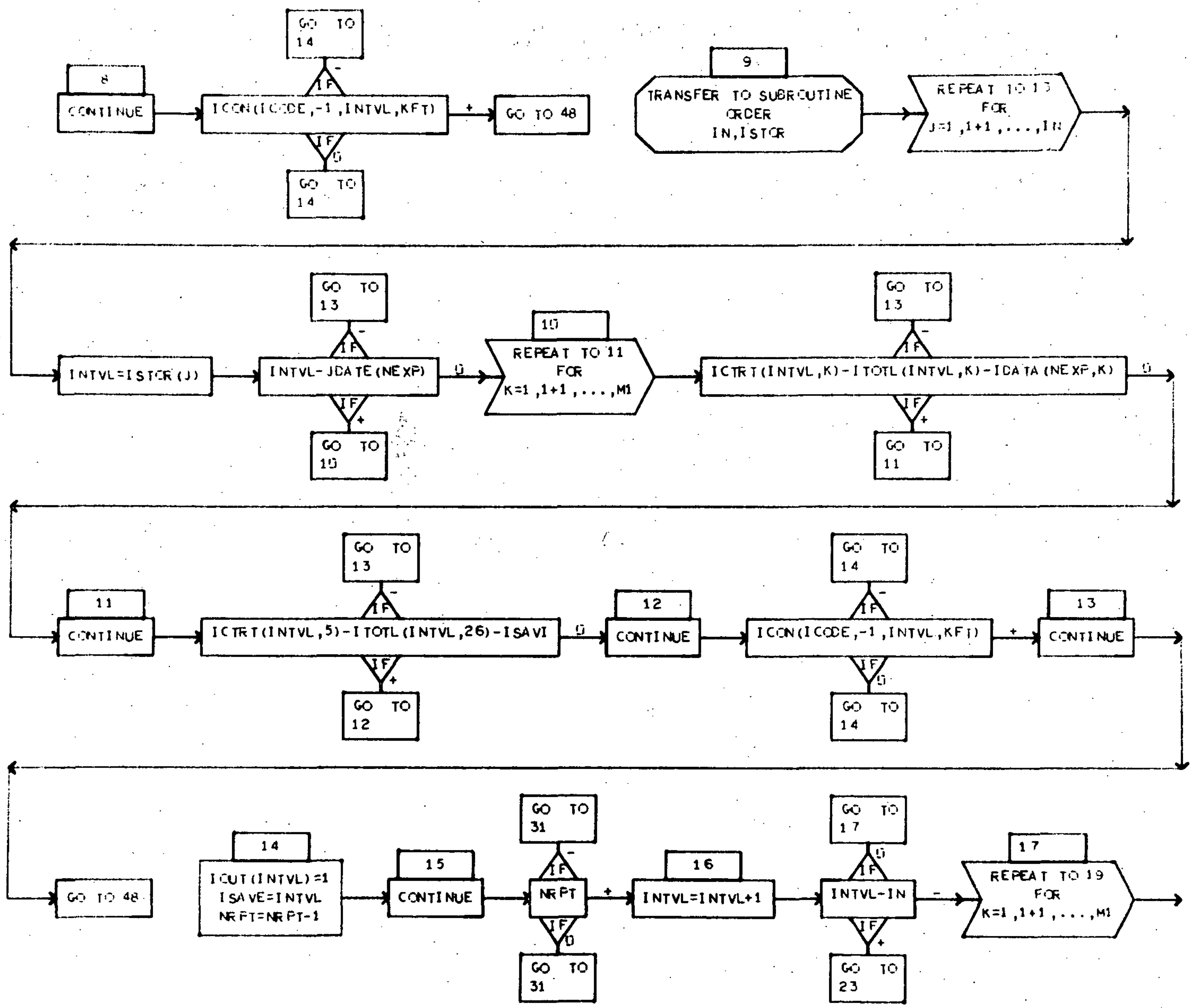
DIMENSIONED VARIABLES

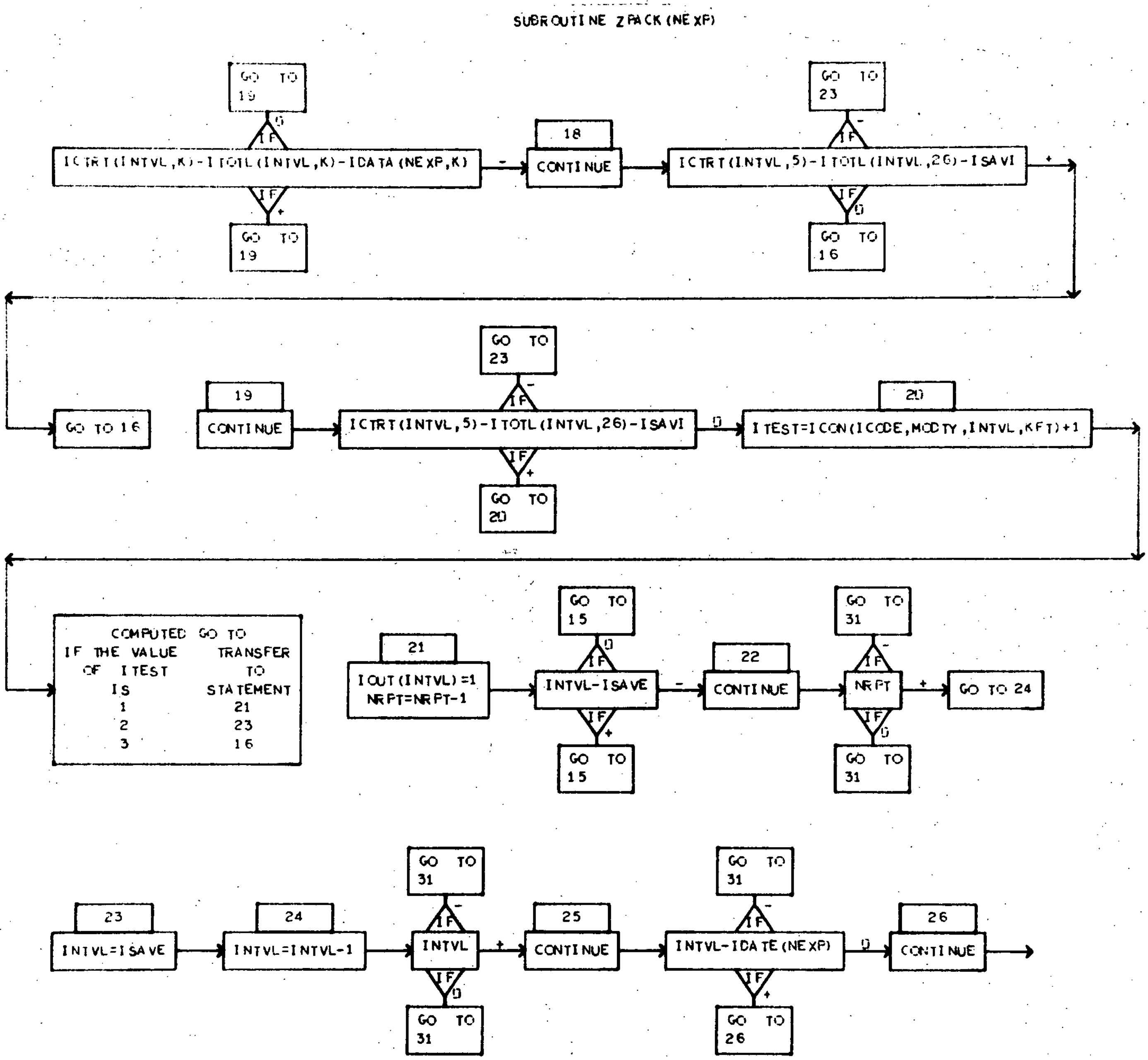
SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES	SYMBOL	STORAGES
IDATA	36,25	ITOTL	120,27	ICTRT	120,5	ICUT	120	JDATE	36
IREPT	36	IDATE	36	NAME	27	BNAME	27	MEXP	36
IMOD	3	XCCST	18	XCOST	8,36	IPERF	36	JCON	36,3
XXCST	18	ICUT	120,36	NREQD	36,1	ISTOR	120	KFT	3

SUBROUTINE ZPACK (NEXP)

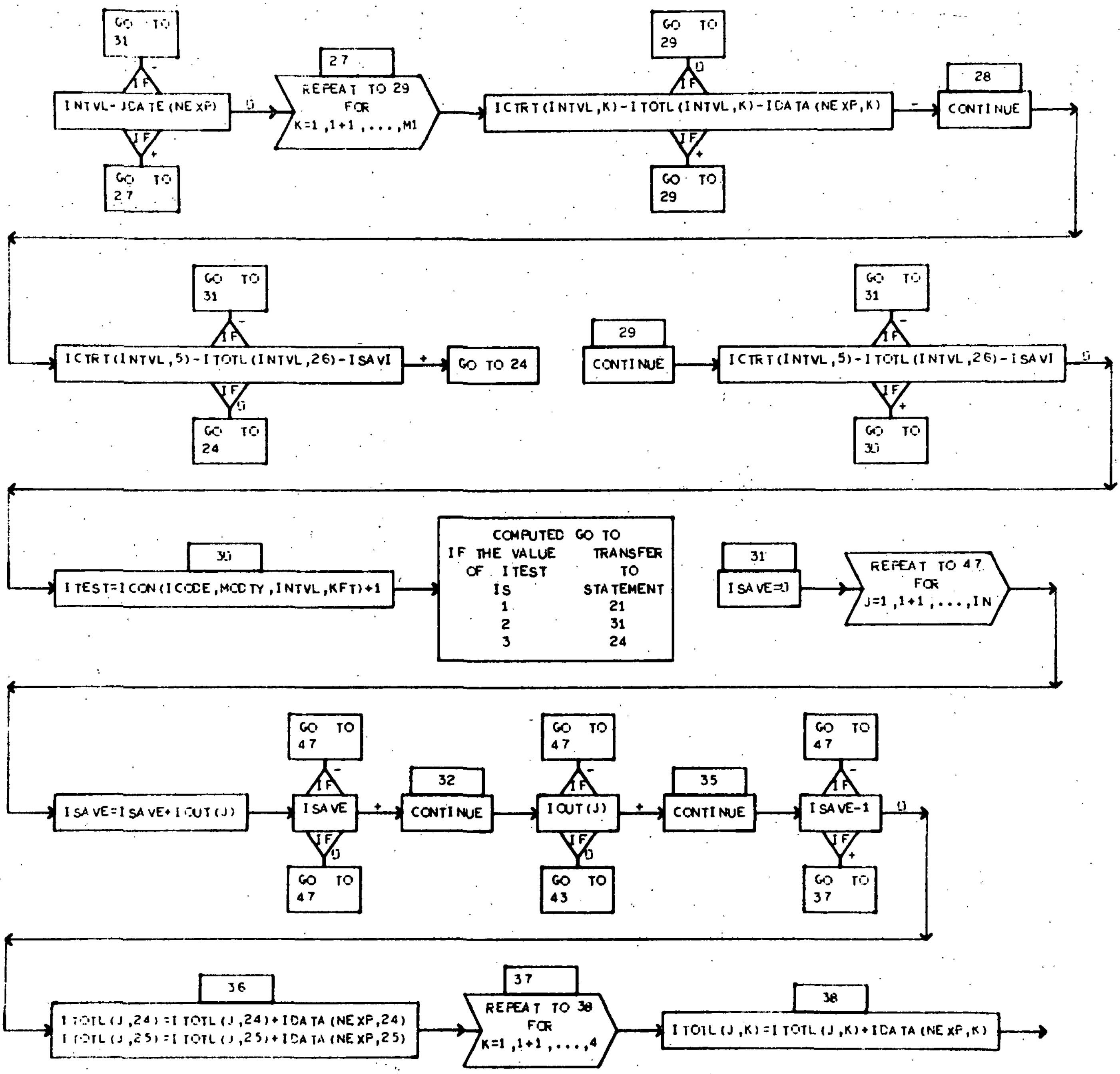


SUBROUTINE ZPACK(NEXP)




SUBROUTINE ZPACK(NEXP)
ICTRT(INTVL,K)-ITOTL(INTVL,K)-IDATA(NEXP,K)
GO TO 19
GO TO 19
18
CONTINUE
ICTRT(INTVL,5)-ITOTL(INTVL,26)-ISAVI
GO TO 23
GO TO 16
GO TO 16
19
CONTINUE
ICTRT(INTVL,5)-ITOTL(INTVL,26)-ISAVI
GO TO 23
GO TO 20
20
ITEST=ICON(ICODE,MODTY,INTVL,KFT)+1
COMPUTED GO TO
IF THE VALUE OF ITEST IS 1 2 3
TRANSFER TO STATEMENT 21 23 16
21
IOUT(INTVL)=1
NRPT=NRPT-1
INTVL-ISAVE
GO TO 15
GO TO 15
22
CONTINUE
NRPT
GO TO 31
GO TO 31
GO TO 24
23
INTVL=ISAVE
24
INTVL=INTVL-1
INTVL
GO TO 31
GO TO 31
25
CONTINUE
INTVL-IDATE(NEXP)
GO TO 31
GO TO 26
26
CONTINUE

SUBROUTINE ZPACK (NEXP)



SUBROUTINE ZPACK(NEXP)