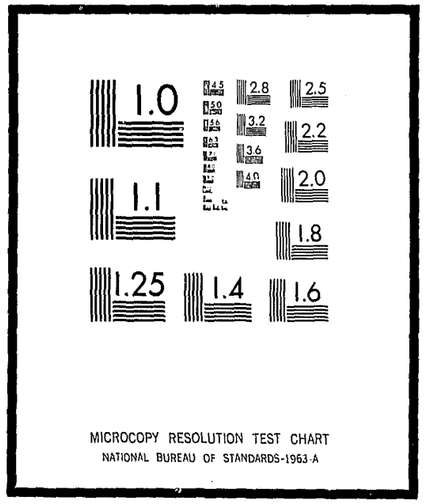
NCJRS



$00996.00.000650
ACCESSION NUMBER: 00996.00.000650
TITLE: SIMBAD - SIMULATION AS A BASIS FOR SOCIAL AGENT'S DECISIONS, FINAL REPORT
PUBLICATION DATE: 690630
AUTHOR(S): ANON
NUMBER OF PAGES: 150
ISSUING AGENCY: SOUTHERN CALIFORNIA UNIV
SPONSORING AGENCY: LEAA
GRANT/CONTRACT: 359; 168
SUBJECT/CONTENT: PROBATION AND PAROLE (ADULT)
CORRECTIONS
RECIDIVISM
CRIME STATISTICS
MODELING TECHNIQUES
PROBATION STATISTICS
JUVENILES
PROBATION AND PAROLE DECISION MAKING
MATHEMATICAL MODEL
PROBATION
PREDICTION
SOCIAL WORKER

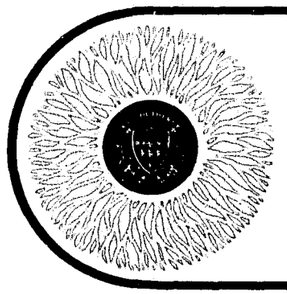
ANNOTATION:
A SYSTEM FOR COMPUTER-AIDED PROBATION DECISION MAKING.

ABSTRACT:
THE ACRONYM SIMBAD IS MADE UP OF LETTERS FROM THE PHRASE, SIMULATION AS A BASIS FOR SOCIAL AGENTS' DECISIONS. THE SYSTEM WAS DEVELOPED TO EXPERIMENT IN THE USE OF A COMPUTER AS A GUIDE TO PROBATION DECISION MAKING AND FOR CONVENIENT AND FAST STORAGE AND RETRIEVAL OF DATA ABOUT DELINQUENT JUVENILES. IT IS DESIGNED AS A TOOL IN THE COMPARISON OF A SINGLE CASE WITH MANY OTHERS AND THE PREDICTION OF WHAT THE OUTCOME MAY BE ON THE BASIS OF WHAT HAS HAPPENED IN THE PAST. MATHEMATICAL MODELS OF THE PROBATION PROCESS HAVE BEEN DEVELOPED. THE OBJECTIVE IS TO CREATE CHANGE EFFECTED BY AUTOMATIC UPDATING, INCREMENTING, AND EVALUATING FEATURES OF THE SYSTEM. (AUTHOR ABSTRACT MODIFIED)

public systems
research institute

SIMBAD

A SYSTEM FOR COMPUTER-AIDED PROBATION DECISION MAKING



University of Southern California

SIMBAD

Simulation as a Basis for Social Agents' Decisions

FINAL REPORT

The work described in this report has been supported in part by Public Health Service Grant MH-06597 from the National Institute of Mental Health; Grants 168 and 359 from the Office of Law Enforcement Assistance; and grants from the Ford Foundation.

Public Systems Research Institute
University of Southern California
June 30, 1969

TABLE OF CONTENTS

SIMULATION AS A BASIS OF SOCIAL AGENTS' DECISIONS

HISTORY OF THE PROJECT

SIMBAD is an outgrowth of two previous studies. The first of these, reported in 1961,* was based on intensive taped interviews with 178 youngsters on formal probation and assigned to the Santa Monica area office of the Los Angeles Probation Department. The primary purpose of the interviews was to ask open-ended questions which would give probationers an opportunity to describe in their own words what they thought was helpful or harmful in their experience while on probation. The study revealed an immense variability in the amount and nature of the supervision provided.

Some youngsters were never seen by their probation officers after the initial court experience. Many were seen on an average of once a month, while others were seen for extensive interviews as often as once a week. The effects of the probation process on the youngsters themselves were also perceived as highly variable. For some it was a significant experience, affecting them personally and in their relations with parents, friends, and school. Others could not recall the name of their probation officer. A severe limitation on the interpretations that could be made from information obtained in this earlier study was that judgments were made solely on the basis of perceptions. The open-ended nature of the questions asked also presented monumental difficulties in the attempt to extract data which were comparable or communicable in any form other than that in which they were received.

A second study, which came to be called "The Probation Project," tried to correct some of the major inadequacies of the first study and to

*McEachern, A.W. (ed.) Views of Authority: Probationers & Probation Offices. Los Angeles: Youth Studies Center, Univ. of So. Calif., 1961.

explore further some of the issues raised by that study. The main objective of this project was to assess the relative effectiveness of different dispositions and supervision practices in the probation system. A rather complicated measure of recidivism (described elsewhere) was devised as the criterion of success or failure, and this criterion was applied to two major categories of probation activity: the position assigned and the treatment given.

When the two major categorizations were combined, four classes of youngsters were described:

A. No position and no treatment (N=820):
Those dismissed either at intake (79.3%) or court (8.0%) and never seen, or those placed on informal probation but never seen (11.7%), or those referred to some other agency and not seen by probation (1.7%).

B. No position but received some treatment (N=469):
Primarily those placed on informal probation who were seen (77.8%) but also including those who were dismissed and subsequently contacted (22.0%), and those referred to another agency and also contacted by a probation officer (0.2%).

C. Position, but no treatment (N=246):
Wards of the court who were never seen after the initial court experience.

D. Position and treatment (N=755):
Wards of the court who received treatment.

Secondary aims of the study included description of the background characteristics of all youngsters referred to probation in these Southern California counties over a two-month period, and a comparison of these youngsters with the general population of youngsters in these areas. Another aim was to examine the relationships among the seriousness of the offense for which youngsters were referred, dispositions and treatment they received, and any of a large number of background characteristics.

In order to achieve the aims of the study, data were collected on all juveniles referred to eight participating Southern California probation departments in October and November, 1963. With the exception of Los Angeles County, these counties comprise all of Southern California, one of the fastest-growing regions in the country in terms of population. The area has been subjected to all the common problems associated with accelerated residential development and the influx of highly mobile persons with diverse racial and ethnic backgrounds. Nevertheless, the counties vary from a sparsely populated rural county (Imperial) to a county which contains the eighteenth-largest city in the nation (San Diego).

Another reason why these counties were selected for the study was to take into account wide differences known to exist in administrative policies of the various probation departments. This allowed for comparisons of youngsters with similar social backgrounds and offense histories who received different dispositions and kinds of treatment, in order to assess the relative effectiveness of different procedures. Los Angeles was excluded from the study primarily because it is unique in the nation in its size and decentralization.

The basic data consisted of complete background information, collected by project staff on uniform forms, on all juveniles referred to these probation departments for delinquent acts over a two-month period. These included: personal characteristics, delinquent history, school experience, socio-economic status, and family history and structure. Reason for referral, detention history, the court process, and initial disposition and placement were also described; and all cases were followed, whatever the initial disposition, for a period of one year. During that one-year period, records were kept of any subsequent referral to probation and of any treatment contact with the juvenile and others related to his case. Moreover, data describing the probation officers' characteristics, official positions, and caseload were obtained. These

basic data describe the probation histories of 2,290 juveniles who were referred to the seven departments studied during the months of October and November, 1963. The final contact cards (records of treatment contacts) and re-referral records were obtained in May, 1965.

The first study had given us considerable "insight" into the effects of probation on its clients. The difficulty was in bringing these insights out where they could be of use to others. The second study provided even more information and provided it in a form suitable to statistical analysis and summarization. It was now possible to present the data, but was it practical? Art, as we know, has always been longer than life, but with the coming of the computers, this ancient discrepancy has become utterly bewildering in its implications for both the researcher and his audience. Before the advent of the computer, the researcher was compelled to select a human-sized volume of analysis which could be reduced to a human-sized volume of reading matter for his audience. But now the possible analyses are endless and life is still brief. We have reached a point where only a computer can read and interpret the mass of data which a computer can store and manipulate.

A little earlier in this report we reported a few of the findings from the Probation Project in respect to the four classes of youngsters described. Many of these "findings" are, of course, confounded by the nature of the criterion we have used, so that without multivariate analyses we are unlikely to be able to say exactly which process variables are related to the criterion independently of the background characteristics and delinquent histories of the youngsters. Furthermore, they are not simple one-to-one relationships or perfect linear correlations. So far, we have performed analyses in which the zero-order relationships between 283 variables and the criterion were examined. But such a vast quantity of detail is very difficult to report and literally impossible to assimilate. If we were to go beyond this stage of analysis

to the multivariate analyses clearly essential if we are to isolate the effects of single variables, we would be so inextricably "embroiled" in the facts that no probation officer would recognize the relationship between our "findings" and his "problems," even were we able to communicate with him.

As Leslie Wilkins (1967, p.3)* suggests in commenting on "crime" statistics: "The figures and other information are interpreted to mean that we must do something, but give us no indication of what to do. Information collection has for too long lost its direct connection with decisions. When the facts bear no relationship that we can see to acts, we may doubt the scientific nature of the items we call 'facts.'"

Suppose, nevertheless, that we are able to predict with a fair degree of accuracy the outcome with some mathematically sophisticated combination of these 283 variables. Theoretically, an officer could take the formulae we arrived at and apply them to each of his daily decisions for each of the youngsters in his caseload. Consider the probation officer, desk calculator by his left hand, pages and pages of formulae at his right, telling him how to calculate the multiple curvilinear regression weights he needs in order to be sure he is taking account of the best information available, all the while the youngster stares at him, confused, frightened and uncertain about his future. This is obviously very primitive science fiction - no probation officer, even had he the training and inclination to perform such a task while his client waited, would have the time or energy to do anything else but push the buttons on the calculator. When we consider, as well, the complexity of a traditionally conceived multivariate model, in which optimal use is made of the many bases of classification, normally and non-normally distributed variables, linear and curvilinear relationships of a very high order, we are led to conclude again with Wilkins (1967, p.9):

*A list of references appears on page 88.

> ...we have got our models wrong. We must move away from simple cause-effect, deterministic roles towards concepts of complex systems, strategies and rational decision processes in conditions of incomplete knowledge and even incomplete criteria and objectives which are likely to be subject to change.

One approach to this very challenging task is outlined in the following sections.

SIMBAD

The acronym that serves as the title for this section of the report is made up of appropriately selected letters from the phrase "Simulation as a Basis of Social Agents' Decisions." This project, an outgrowth of the Probation Project, is intended to partially bridge the gap between the results of research and the decisions of practitioners. The basic objective of this project is to introduce new knowledge and new technology into the practice of probation. Participating probation departments will have remote, real-time access to a computer facility which will provide probability estimates of success for disposition and treatment decisions at any point in the probation process. This has been accomplished through the development of mathematical models of the probation process, based on a large body of data from the research we have just described. Research findings will, in effect, be immediately available at the moment they are needed - when decisions are made. A major objective is to create, not merely initial change, but a true process of change continually effected by the automatic updating, incrementing, and evaluating features of the system.

That such an approach is appropriate will be evident if the nature of new knowledge about human behavior is considered. New knowledge in the behavioral sciences can be thought of as falling into one of two categories. It may be in the nature of one or more specific relationships which have been empirically demonstrated to occur with a fairly high degree of reliability. Since levels of reliability that are considered high in the behavioral sciences may range from accounting for say 10% of the variance to perhaps 65%, the ways in which practioners would apply such "knowledge" is no doubt as obscure to the scientist as it is to the practitioner. But the subject matter of the behavioral sciences and the sophistication of the scientist often lead to relatively complete empirical models based, for example, on multiple regression or discriminant function procedures, in

which case the practitioner would need to measure several variables in the same way the scientist did, apply the appropriate weights, and arrive at a reasonably close approximation to the prediction he would like to make. This is a procedure in which practitioners could be trained, but its rigors would probably result in an insensitivity to the exceptional case, the overpowering variable that the empirical model failed to take into account.

It will be evident that a procedure which allowed a practicing social agent to base his decisions on the best available empirical knowledge, while it left him free to exploit his own experience in regard to the idiosyncracies of an individual case, would enhance the informational base of his decisions; and to the extent that both bases for his decisions were valid, increase the likelihood of a successful outcome.

The specific objectives of the program are to establish such a basis for decisions in the field of probation, by providing a simulation of the probation process which can be used as a prognostic tool; to evaluate the effects of introducing such a system into at least three operating departments; and to build into the system automatic updating and evaluation procedures. It should also be sufficiently flexible to allow individual probation departments to evaluate experimental treatment programs as well as their overall operation. Although these objectives are easily within the bounds of current technology, their realization is not simply a matter of writing another program or two. They must be approached in stages, at each of which systematic evaluations can be carried out and modifications introduced to conform to the operational needs of participating probation departments.

The first stage is the acquisition of a body of empirical data on the basis of which the initial simulation programs can be developed and evaluated. These data have been acquired in the study of the

probation process described above, in a study of authorities surrounding and influencing juveniles referred to probation (McEachern and Taylor, 1966), and of experimental treatment programs in the three probation departments (Hill, Stoller, and Straub, 1967) involved in the present research.

The following stages describe in sequential steps the specific objectives of the project:

A. Develop an operating model in the form of a computer program of the probation process from the point of initial intake decisions, investigations, court proceedings, court decisions, disposition, treatment and placement alternatives, to subsequent re-referral. This overall program is called the SIMBAD system.

B. Keep records of each department's caseload, so that within one year, the original data on which the initial probabilities were based can be replaced entirely with current data which is being continually updated. The nature of the data to be collected for this sample will be determined, initially, on the basis of analyses of the data already collected. As conditions change, it may be expected that some factors will become less relevant and others more relevant. It is also expected that analyses of the original data will indicate areas in which more or different types of data must be collected in order to arrive at more precise probabilities. The nature of the information gathered will, accordingly, be reviewed periodically to determine what questions can be dropped and what need to be added. This updating and incrementation feature should prove particularly valuable in allowing departments to test their own assumptions as to what factors most significantly affect particular decisions.

C. Provide the overall SIMBAD system and the necessary equipment to probation departments, and train their staff in its use and significance.

D. Develop and incorporate auxiliary programs at the request of, and to meet special needs of, probation departments. In order to insure that probation departments be willing and capable of maintaining the system, we will be obligated to assist in making it as versatile and flexible as possible. The operative consideration here is whether the probation administrator can convince his county board of supervisors that the system justifies the expense. For this reason, we are planning a certain flexibility in the development of at least the following auxiliary models:

1. A model to assist in answering questions relevant to problems of caseload management. Information is already available from the previous study of probation, and more will be gathered in the course of following the sample in this project, which will assist the probation officer in establishing priorities and allocating time in the management of his caseload. Caseload figures are universally high, and all departments identify this as a crucial problem.

2. Special models to assist in the evaluation of experimental programs, either presently in operation or contemplated during the period of the project, within the participating departments. The sample incrementation feature of the SIMBAD system would, with some modifications, be of immense help in this respect by providing current and relevant material for matching purposes.

3. Planning models to assist in such matters as budget and caseload projections, the need for and deployment of personnel and resources in the coming year, and so forth. The assumption is that a carefully controlled sample will permit more accurate projections than can be obtained from the record-keeping systems currently employed.

Hidden in these limited descriptions of the specific objectives of the project are a number of design characteristics whose significance will be much clearer when the procedures are detailed. Of the greatest importance, however, is the fact that the initial model will be based primarily on data which will be several years out of date by the time the simulation models are first put into operation in probation departments. Although it is probably true that certain basic relationships within these data (and in the initial models) will have some degree of permanence, it is equally true that a number of conditions which influence the success of probation will have changed. Personnel turnover is one very obvious example. More subtle, though perhaps of as much importance, would be styles of delinquent behavior, police referral practices, judges' evaluations, population changes, and even economic changes within a community, all of which can have a significant effect on the outcome of probation practice. The only solution to the problems inherent in the instability of the environment, clients, and organization with which probation must contend, is to insure updating as rapidly as possible.

Finally, although it can be better described as a guiding principle than as a specific objective, the operation and updating of the system in probation departments will be designed to provide the greatest amount of usable information with the least amount of additional record-keeping for purposes of research. Consequently, the system has been designed as itself a "model" which can be adapted and carried over into the operation of other agencies with a minimum of excess research baggage.

THE CURRENT FORM OF THE SIMBAD OPERATING SYSTEM

Because of the degree of complexity involved, it is an almost impossible task to communicate in words the technical routines of the functioning of a computerized system such as SIMBAD. We are therefore here confining our presentation to the following material:

A. A simple listing outlining the program.

B. A series of simplified flow diagrams describing the basic logic of the system.

C. A listing of the entire program and subroutines (coding).

D. A summary of the basic operative capabilities of the system and of user interaction as demonstrated in detail in the appended Reference Manual. The prefatory remarks to the Manual also contain a simple description of the main functions and features of the system.

A. OUTLINE OF SIMBAD COMPUTER PROGRAM

1. Initialize communications equipment.
2. Read internal data files from disk.
3. Set status word to 0 for each line.
4. Examine communication status for next line.

   If negative, go to 4.
   If positive go to appropriate communications error routine and return to 4.
   If zero go to specific routine depending on status word for this line.

5. Status 0 — User initiating contact with computer.

   Transmit "PLEASE TYPE IN YOUR SIMBAD ACCOUNT NUMBER."
   Set status word to 1.

6. Status 1 — Welcome message has been transmitted.

   Initiate communications read for user's account number.
   Set status word to 2.

7. Status 2 — User's account number has been received.

   If account number is invalid, go to 5.
   If account number is valid, edit "SIMBAD READY AT (time) (date)."
   Set status word to 3.
   Go to 4.

8. Status 3 — Finished transmitting SIMBAD READY message.

   Initiate transmission of " - " (hyphen).
   Set status word to 4.
   Go to 4.

9. Status 4 — Finished transmitting hyphen.

   Initiate communications read for user's command statement.
   Set status word to 5.
   Go to 4.

10. Status 5 — Finished receiving user's command statement.

    Scan statement for lexicon words and integers.
    If statement is valid go to specified command routine.
    If statement is invalid initiate transmission of appropriate diagnostic.
    Set status word to 3.
    Go to 4.

11. Status 6 — Finished transmitting terminal message.

    Close user's communication line.
    Set status to 7.
    Go to 4.

12. Status 7 — Finished closing user's line.

    Initialize communication line for next caller.
    Set status word to 0.
    Go to 4.

COMMAND ROUTINES RESPONDING TO USER'S STATEMENT

UPDATE — Read in specified case file if necessary and modify according to user's statement.

Initiate transmission of "U (file I.D.) (item number) (value) (old value)."

Set status word to 3.
Go to 4.

DELETE — Clear specified file from disk storage.

Initiate transmission of "D (file I.D.)."

Set status word to 3.
Go to 4.

NEW — Create empty file on disk storage.

Initiate transmission of "NEW I.D. NUMBER IS (I.D. number)."

Set status word to 3.
Go to 4.

CLEAR — Set Summary Definition Matrix to zeros.

Initiate transmission of "SUMMARY MATRIX CLEARED."

Set status word to 3.
Go to 4.

DISPLAY — Initiate transmission of specified items within specified file.

Set status word to 3.
Go to 4.

DEFINE — Modify Summary Definition Matrix according to user's statement.

Set status word to 3.
Go to 4.

MESSAGE — Type user's statement on console typewriter in computer room.

Set status word to 3.
Go to 4.

TEST — Set switches within program to prevent permanent modification of case data on disk storage.

Set status word to 3.
Go to 4.

END TEST — Restore internal switches to allow full operation.

Set status word to 3.
Go to 4.

DONE — Store item in memory disk (if any).

Initiate transmission of "OFF AT (time)."

Set status word to 6.
Go to 4.

SUMMARY — Scan disk file and accumulate counters according to Summary Definition Matrix.

Send appropriate header information and values to delayed storage for subsequent listing or initiate transmission of same.

Set status word to 3.
Go to 4.

QUESTION — Use Bayes' theorem to provide probabilities for specified question about specified kid or entire data base.

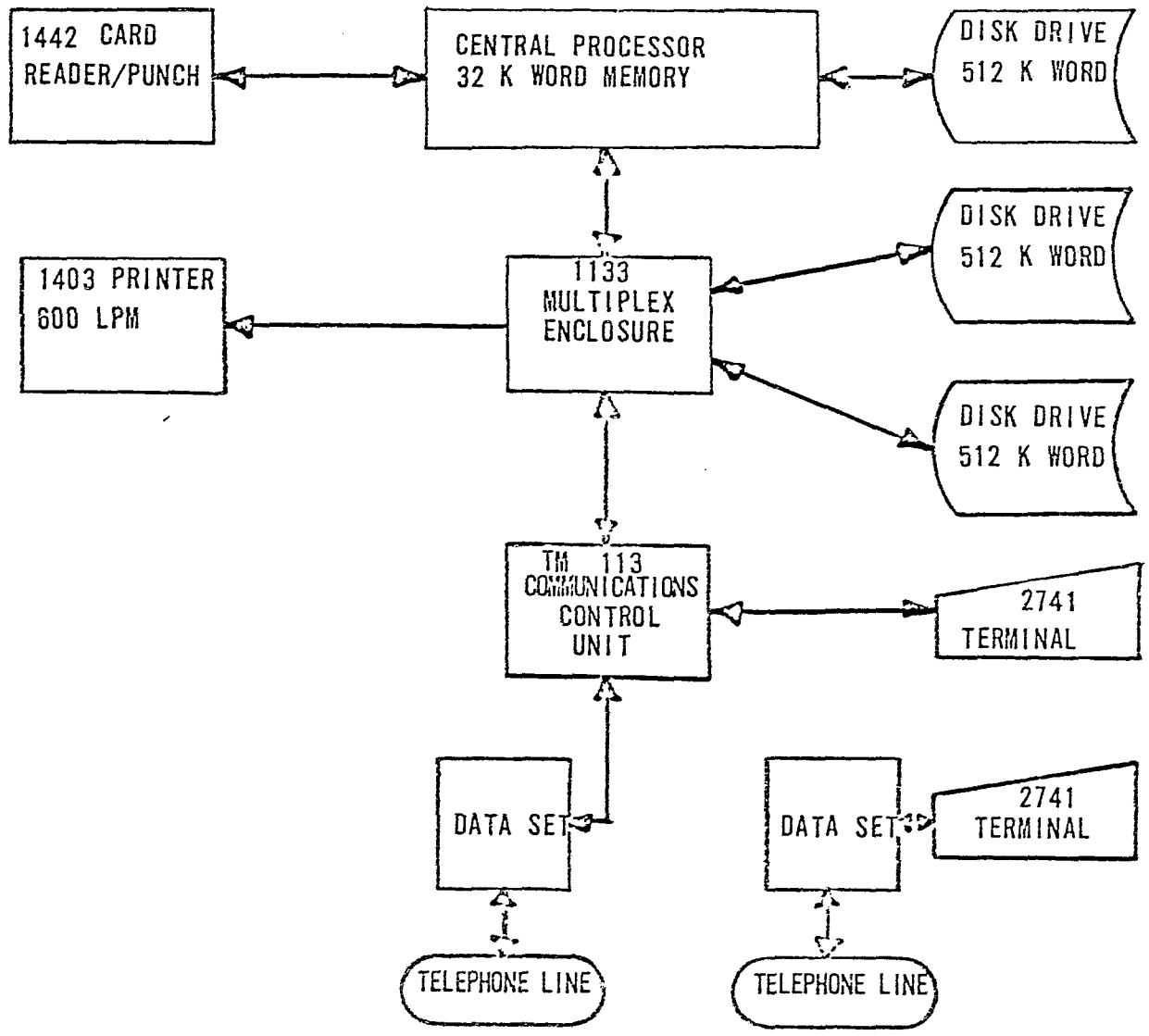
Send output to delayed storage for subsequent listing or initiate transmission.

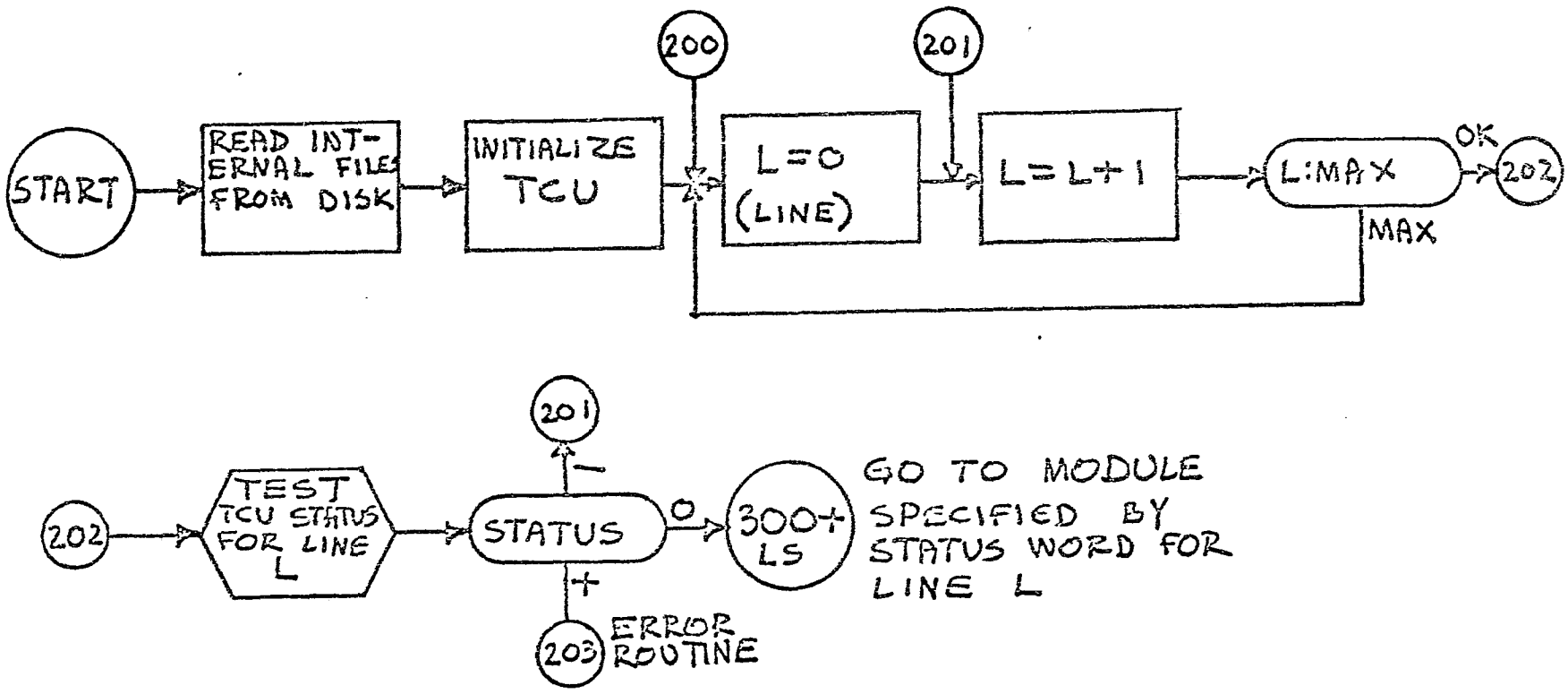
Set status word to 3.
Go to 4.

FILE — Get values of all non-zero items in specified file.
Send data to delayed storage for subsequent listing or initiate transmission.

Set status word to 3.
Go to 4.

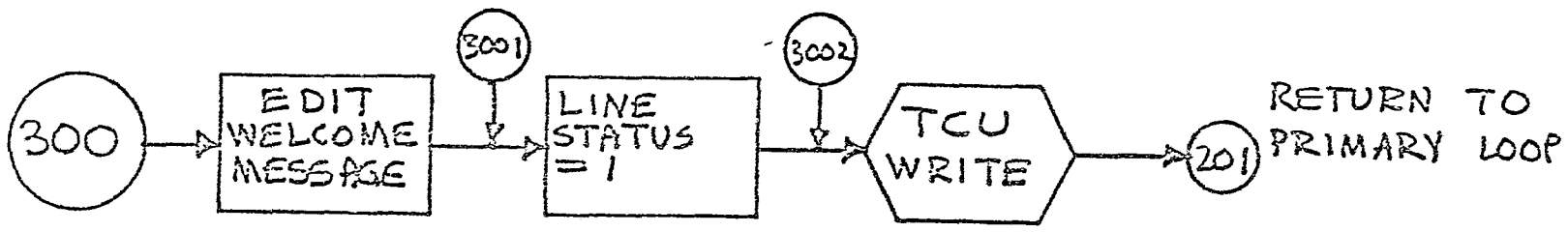
SIMBAD COMPUTER CONFIGURATION (I.B.M. 1130)



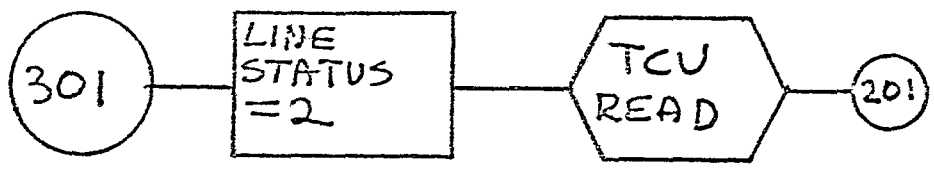


PRIMARY COMMUNICATIONS LOOP. THE STATUS OF EACH COMMUNICATIONS LINE IS EXAMINED AND THE APPROPRIATE SERVICE ROUTINE IS CALLED

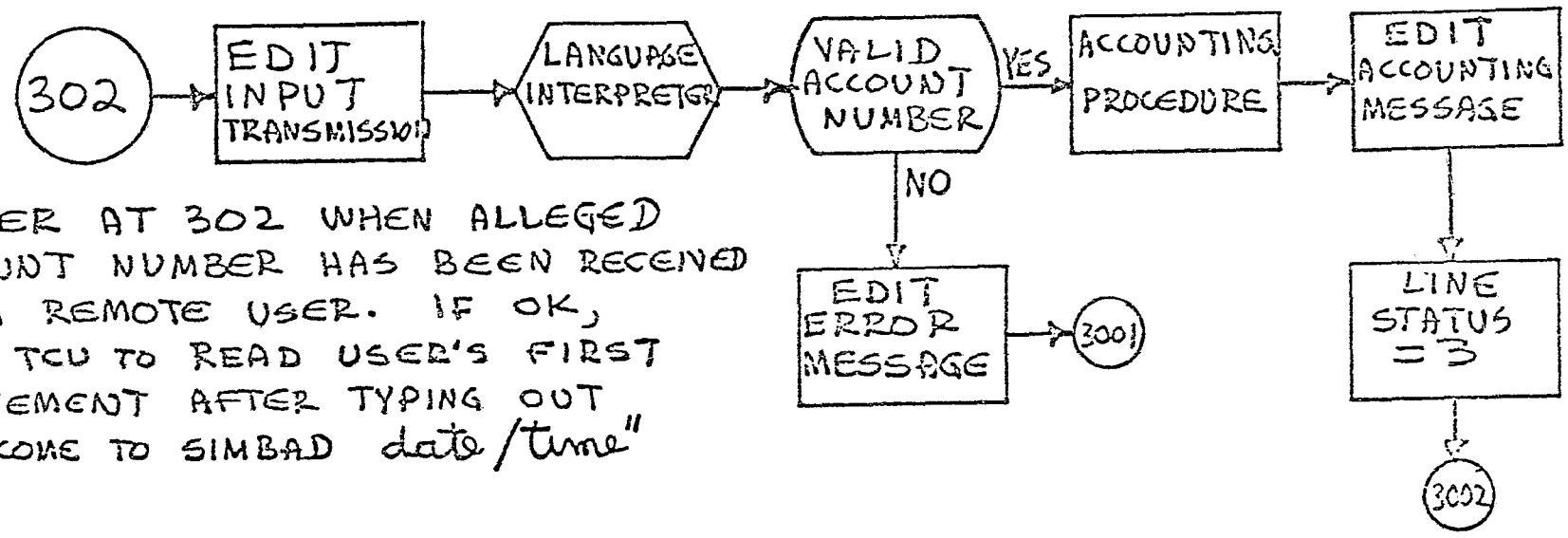
SIMPLIFIED FLOW CHART OF THE SIMBAD COMMUNICATIONS MONITOR



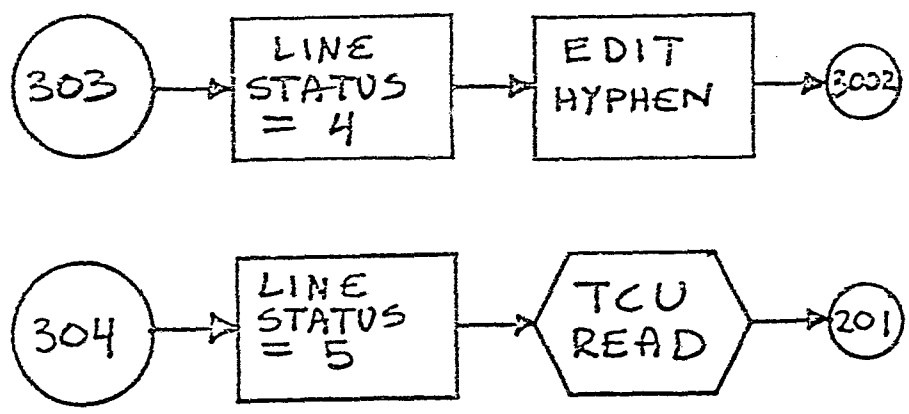
ENTER AT 300 WHEN REMOTE TERMINAL IS FIRST CALLING IN

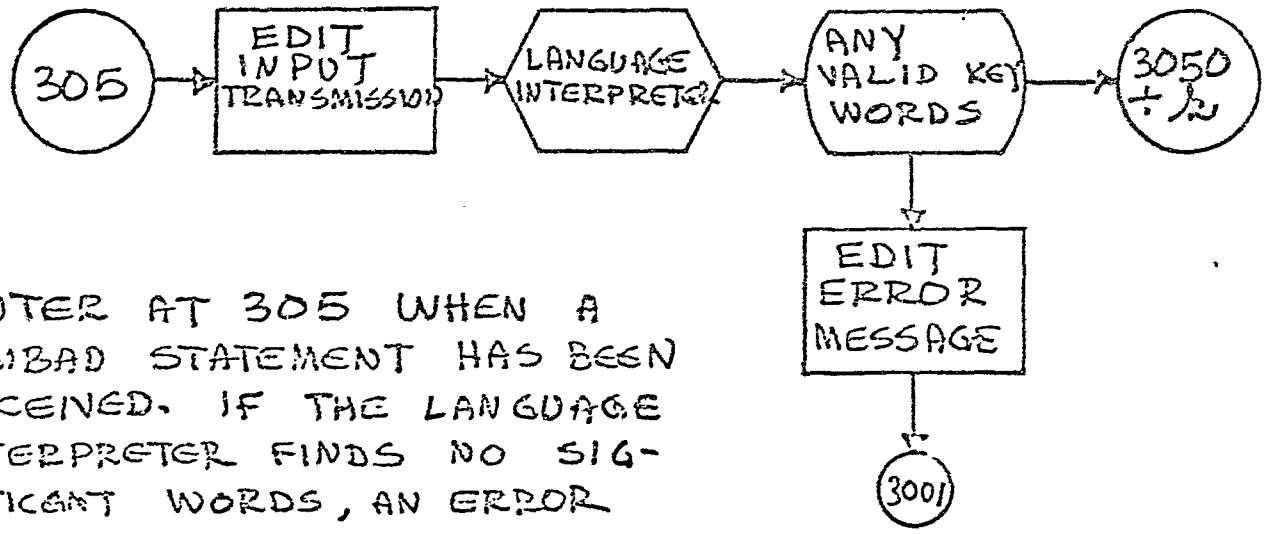


ENTER AT 301 WHEN TRANSMISSION OF "PLEASE ENTER ACCOUNT NUMBER" HAS BEEN FINISHED



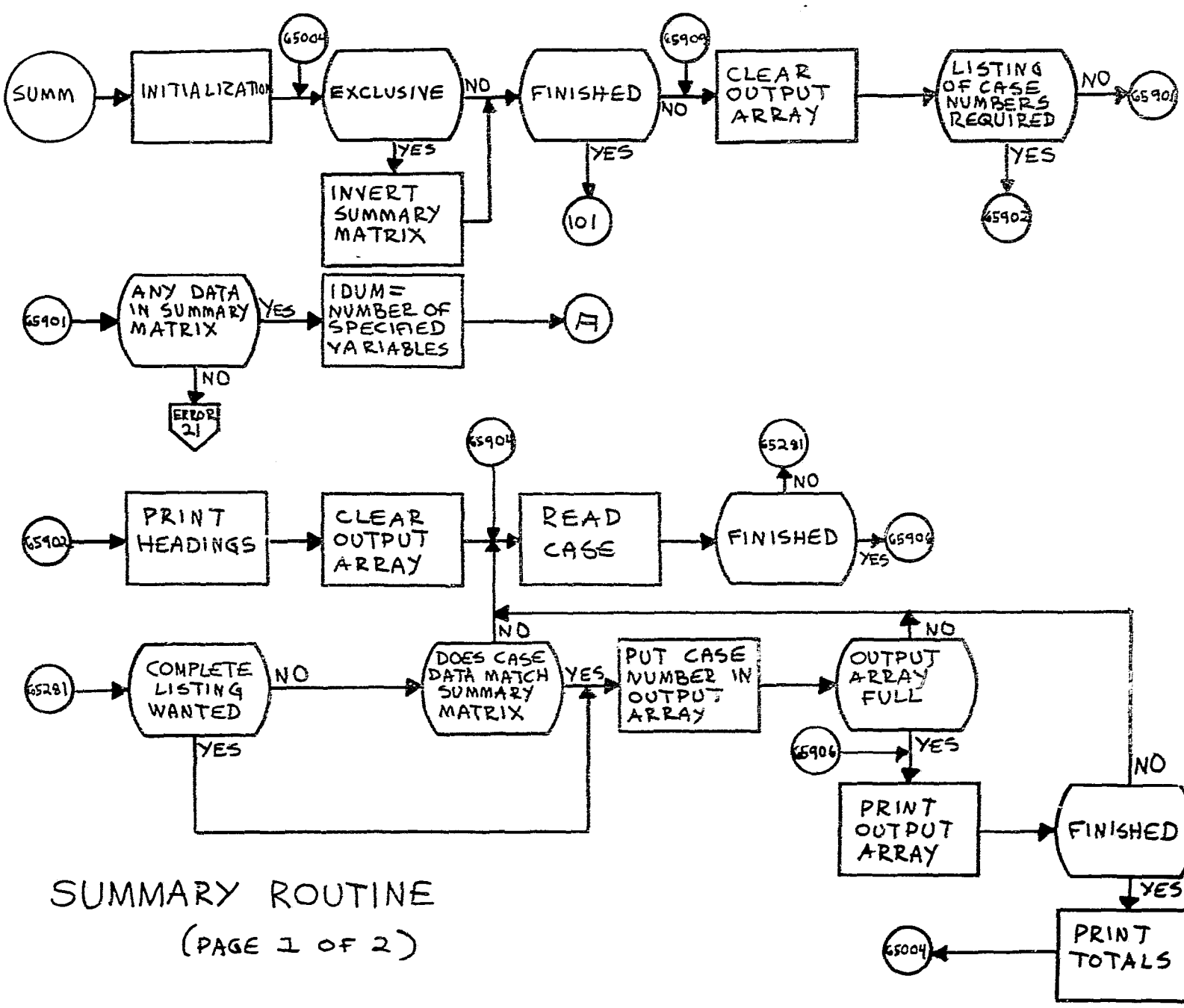
ENTER AT 302 WHEN ALLEGED ACCOUNT NUMBER HAS BEEN RECEIVED FROM REMOTE USER. IF OK, SET TCU TO READ USER'S FIRST STATEMENT AFTER TYPING OUT "WELCOME TO SIMBAD date/time"



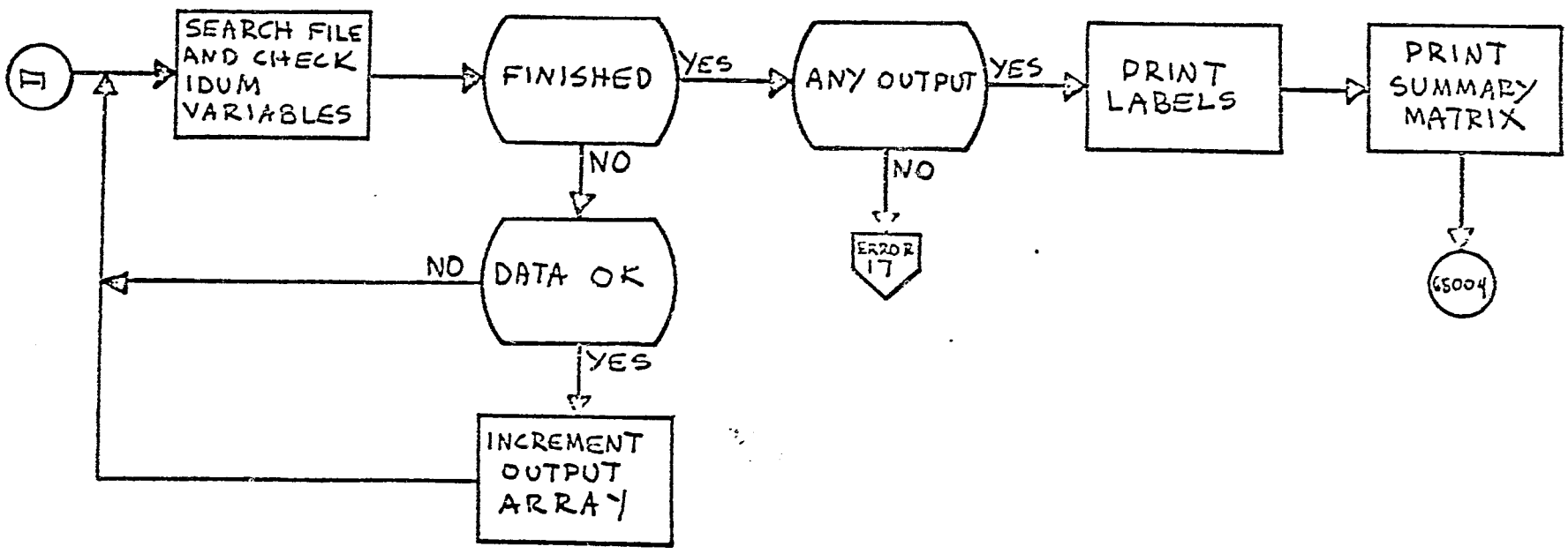


k IS THE CODED VALUE OF THE MOST SIGNIFICANT KEY WORD.

ENTER AT 305 WHEN A SIMBAD STATEMENT HAS BEEN RECEIVED. IF THE LANGUAGE INTERPRETER FINDS NO SIGNIFICANT WORDS, AN ERROR MESSAGE IS SENT TO THE REMOTE USER. OTHERWISE CONTROL IS SENT OUTSIDE THE MONITOR TO THE PARTICULAR MODULE IMPLIED BY THE KEY WORDS IN THE USER'S STATEMENT.



SUMMARY ROUTINE
(PAGE 1 OF 2)

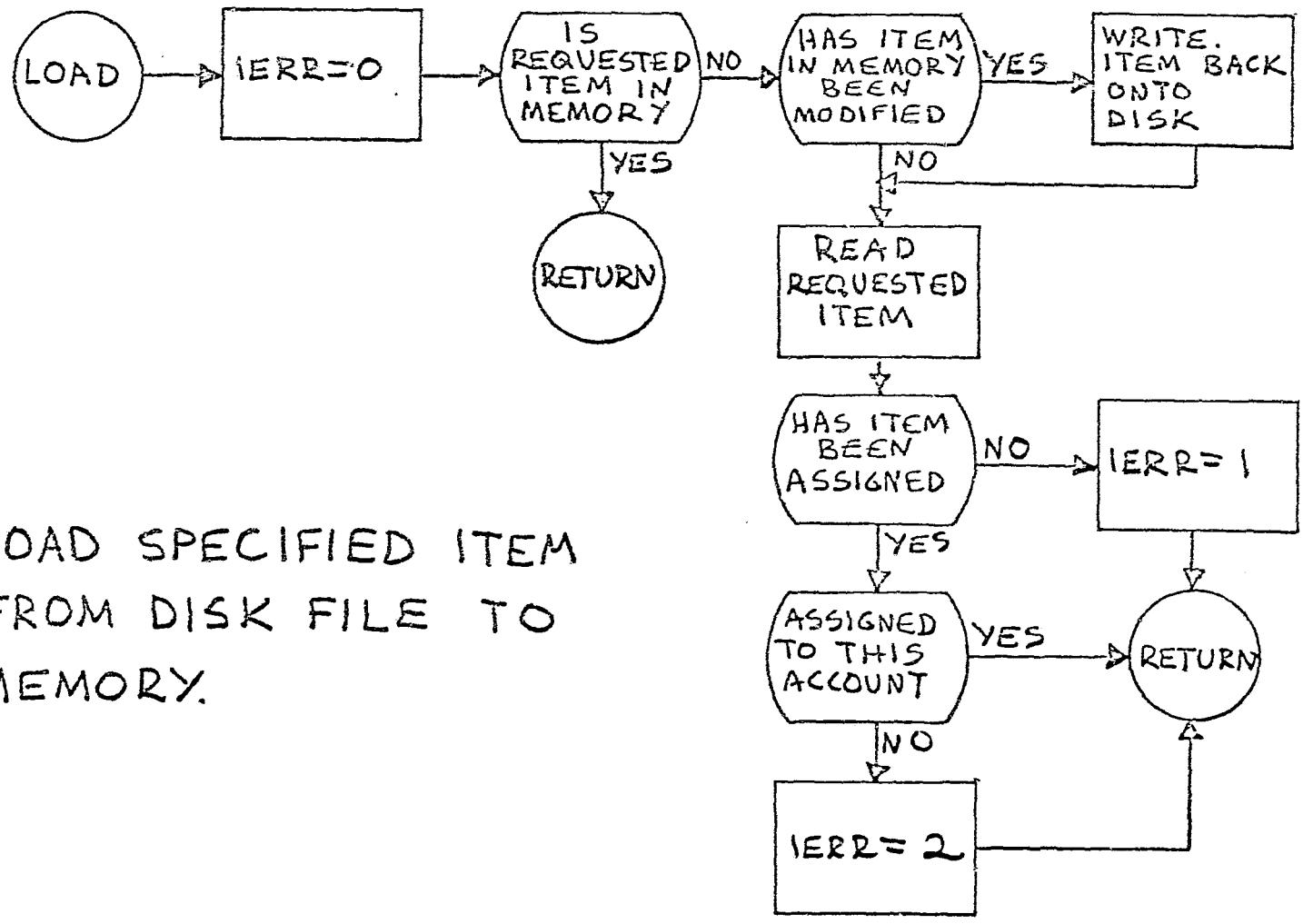


SUMMARY ROUTINE
(PAGE 2 OF 2)

DELAY SUBROUTINE

STORE LINE OF TEXT ON
DISK FOR LATER PRINTING

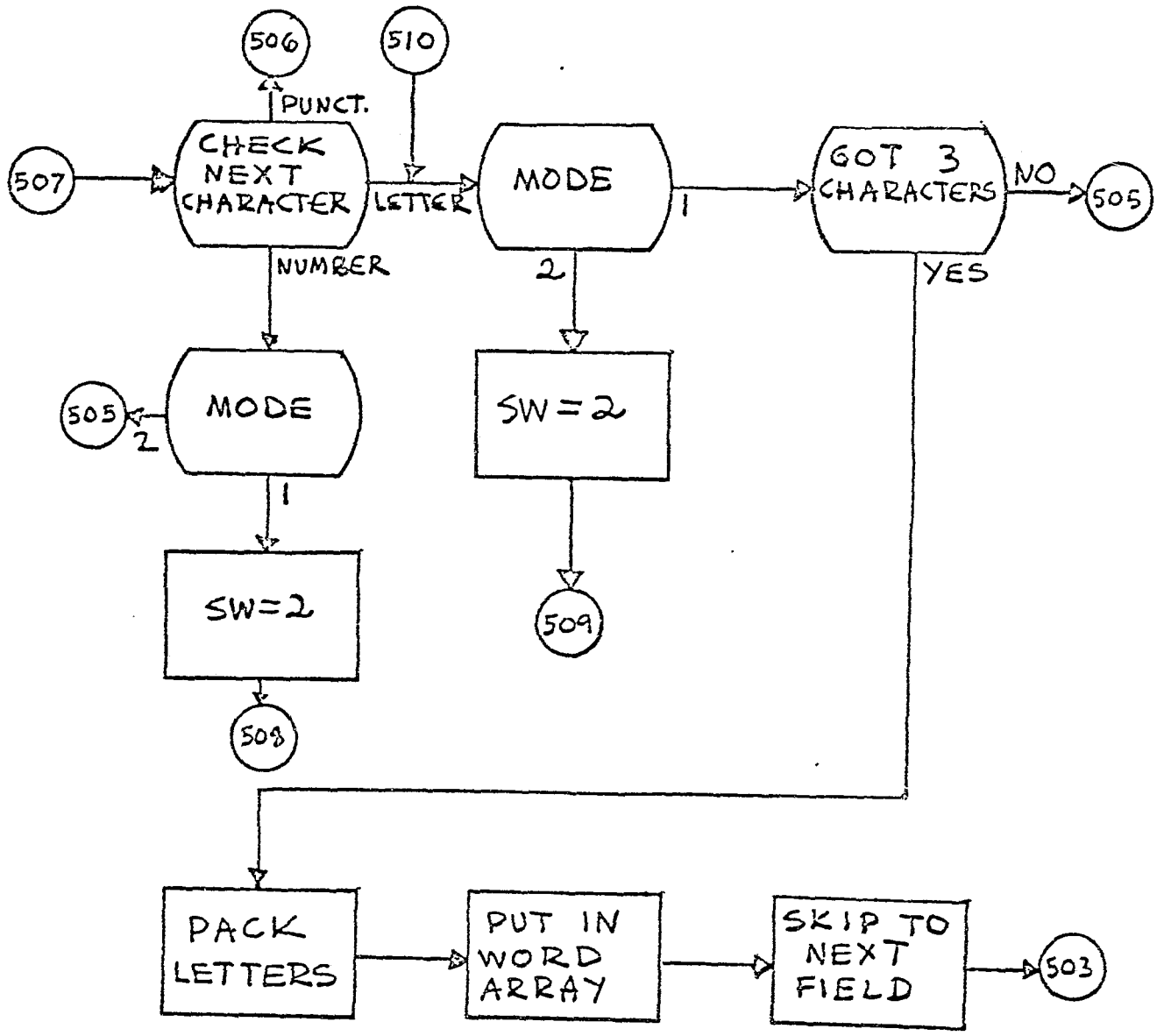
LOAD SUBROUTINE



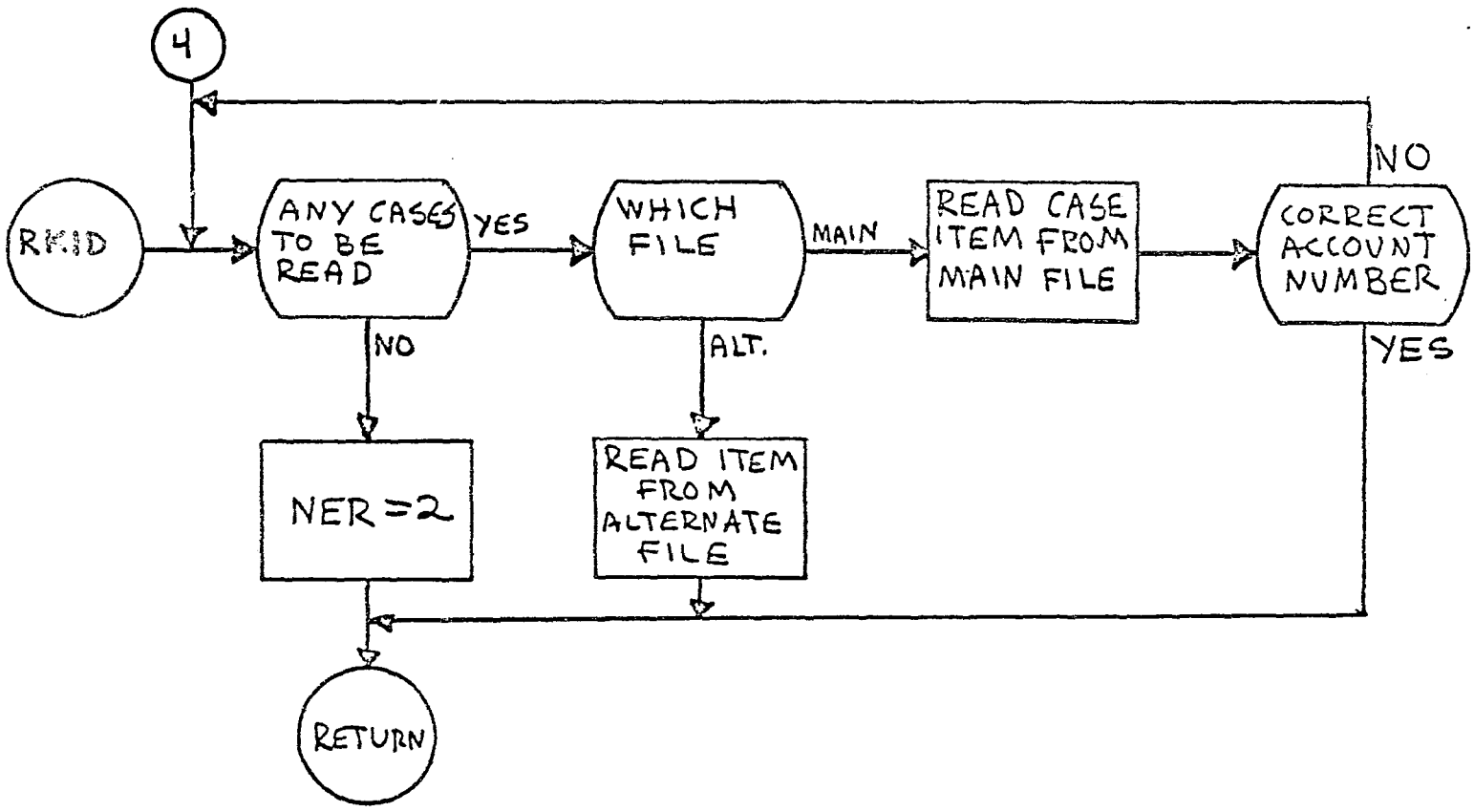
LOAD SPECIFIED ITEM FROM DISK FILE TO MEMORY.

SCAN SUBROUTINE

SCAN



RKID SUBROUTINE



```
// JOB
// FOR
*     IOCS(1403 PRINTER,CARD,DISK,TYPEWRITER,KEYBOARD)
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*TRANSFER TRACE
*ARITHMETIC TRACE
**    S I M B A D  M A I N  P R O G R A M



      DIMENSION ITEM(80,2), LCA(50), LA(50), IOA(20), LCOA(10)
      DIMENSION LECA(50), LEPA(10), LSB(20,2), LIST(181),M(125,2)
      DIMENSION ICHAR(40), LOA(30)
      DIMENSION LSUM(81, 10, 2), NOUT(10, 10, 10)
      DIMENSION IAB(10), NDAR(8)
      DIMENSION LABEL (79,11,3),IOUT(115),IVMAX(79)

C     DIMENSIONS FOR QUESTION ANSWERING

      DIMENSION MATR(3,10,62),NATR(2,10,62),NPRI3(3,2)
      DIMENSION NPRI2(2,2),NDUM(80),NVAL(80),NPDH(9,80),NPOS(9)

      DIMENSION NST(810,2),EPOS(10,3)
      COMMON IOTCU(200)
      EQUIVALENCE (LSUM(81,10,2), NST(810,2))

C     ------------------------------------------------------------

C     D E F I N E  F I L E  S E C T I O N

C     SIMBAD EXECUTION DECK

C/ JOB    0003 0008 0007
C/ XEQ SIMBA    3
CFILES(100,LEX,0008),(102,CASE,0008),(103,SIMAC,0008),(104,SIMER,0008)
CFILES(105,SIMCT,0008),(106,SIMAT,0008),(201,DOP1,0008),(202,DOP2,0008)
CFILES(101,LABEL,0008),(107,SIMDL,0008),(300,SCRAP,0008)

      DEFINE FILE 100(2, 50, U, ITEMP)
      DEFINE FILE 101(36, 79, U, N)
      DEFINE FILE 102(1500 , 80, U, ITEMP)
      DEFINE FILE 103(100 , 16, U, NACT)
      DEFINE FILE 104(32, 80, U, ITEMP)
      DEFINE FILE 105 (1,2,U,ITEMP)
      DEFINE FILE 106(62, 50, U, ITEMP)
      DEFINE FILE 107(4, 80, U, JUNK)

      DEFINE FILE 201(150, 64, U, ITEMP)
      DEFINE FILE 202(150, 64, U, ITEMP)
      DEFINE FILE 300(2290, 65, U, JUNK)
C     ------------------------------------------------------------
C     'CHARACTER' ARRAY REQUIRED BY A1/A3 CONVERSION SUBROUTINES
```



```
      DATA ICHAR /
     --14016,-11968,-11712,-11456,-10944,-11200,-14784,-15040,-7360,
     --6848,-16064,-10688,-7616,-7104,-10432,-9920,-6592,-14272,
     --15296,-15552,-15808,-14528,-10176,-6336,-6080,-5824,16448,-4032,
     --3776,-3520,-3264,-3008,-2752,-2496,-2240,-1984,-1728,19264,
     -24640,20032/

      DATA IVMAX / 6,2,3,5,4,6,2,5,7,2,7,6,6,5,5,5,5,6,3,6,7,4,6,5,2,2,
     *2,2,9,5,5,5,2,6,7,2,2,6,3,3,4,2,2,2,2,2,2,7,2,4,5,6,2,4,3,4,2,3,4,
     *4,5,4,17*1 /

C     ------------------------------------------------------------
      DATA NDAR/8*24896/

C     INTERNAL SIMBAD PARAMETERS


      NOL = 2
      DO 1 J=1,3
      DO 1 I=1,NOL
      NPRI3(J,I) = 333
      NPRI2(1,I) = 500
1     NPRI2(2,I) = 500

      NAFQ = 3
      NLI = 63
      NLX = NLI + 1
      NFPO = 400
      NLPO = 499
      NFC = 500
C---TEMP NUMBER OF LAST CASE
      NLC = 1000
      NQ200 = 200
      NQ300 = 300
      NFQ = 210
      NLQ = 379
      NFQ2 = NFQ + 100
      NQ1 = NFQ+1
      NQ2 = NFQ + NLI + 10
      NQ3 = NQ1 + 100
      NQ4 = NQ2 +100
      NQ5 = 300
      NFVD = 10
      NFVD1 = NFVD +1
      NLVD = 90
      NFA = 1000
      NLA = 1999
      NRES = 9
      NSW1 = 1
      NFIV = 0
      NLIV = 9
C     NYR IS TENS VALUE OF CURRENT YEAR
      NYR = 6
      NFP = 2000
```

```
      NLP = 2009
      LINE = 1
      NFKW = 1
      NLKW = 99
      NFSW = 100
      NLSW = 199
      NIV = 119
      NEC = 50
      KPO = 101
      NEP = 10
      NACT = 1
      NAMAX = 1000
      NCAN = 24896
      NMES = 1
      NCASE = 1500
      NLSB = 20
      NEG = -1
      NL = 21*256
      NEXC = 1
      KEXC = 103
      MAIL = 100
      MSIZE = 121
      M0 = MSIZE +4
      M1 = MSIZE +3
      M2 = MSIZE +2
      M3 = MSIZE +1
      ITIME = 0
      NDATE = 0
      IRED = 13632
      IBLK = 5184
C     NPR IS IN IDUM LOOP IN SUMMARY ROUTINE
      NPR = 3

C     ---------------------------------------------------------------

C     INDEX VALUES FOR LSB ARRAY

      NCACC = 1
      NCKW = 2
      NCID = 3
      NCIV = 4
      NCVD = 5
      NCD =  6
      IMOD = 7
      ITEST = 8
      NRC = 9
      IMAIL = 10
      NCQ = 11
      NTIME = 12
      NALT = 13
C     LSB SLOTS 14 THRU 20 ARE NOT YET ASSIGNED

C     ---------------------------------------------------------------
```



```
C     READ IN THE LEXICON AND LEXICON CODE ARRAYS

      READ (100'1) LA
      READ (100'2) LCA


C     READ IN THE MATR AND NATR FILES
      DO 2 K = 1, 62
2     READ (106'K) ((NATR(I,J,K),J=1,10),I=1,2),((MATR(I,J,K),J=1,10),I=
     *1,3)
      N = 1
      DO 4 J = 1, 11
      DO 4 K = 1, 3
4     READ(101'N) (LABEL(I,J,K), I = 1,79)
      DO 6 I = 1, NLI
      DO 6 J = 1, 10
      DO 6 K = 1, NOL
6     LSUM(I, J, K) = 0

C     ---------------------------------------------------------------

C     FILL THE EMPTY ITEM ARRAYS (LECA AND LEPA) WITH THE ID'S
C     OF AVAILABLE EMPTY ITEMS ON DISK

      CALL FILLX(LEPA, NEP, NFPO, NLPO, ITEM, LINE, NFPO, IERR)
      CALL FILLX(LECA, NEC, NFC, NLC, ITEM, LINE, NFPO, IERR)
      IF (IERR-1) 50, 54, 50
50    WRITE (1, 9050)
9050  FORMAT (///'NO DISK SPACE FOR NEW CASE ITEMS'//)
      PAUSE 1

C     ---------------------------------------------------------------

C     ZERO OUT THE LSB'S

54    DO 56 I = 1, NLSB
      DO 56 J = 1, NOL
56    LSB(I, J) = 0

C     ---------------------------------------------------------------

C     REQUEST TODAY'S DATE FROM KEYBOARD

      WRITE (1, 957)
957   FORMAT ('TYPE IN TODAY''S DATE'/'FORMAT- MMDDY '/'MM=MONTH, DD=DAY
     *, Y= LAST DIGIT OF YEAR   ')
      READ(6,958) NDATE
958   FORMAT (I5)

C     EDIT TODAY'S DATE

      CALL MSIA (LIST, 1, 5, NDATE, -1)
      CALL MOVE (LIST, 1, 2, NDAR, 1)
      CALL MOVE (LIST, 3, 4, NDAR, 4)
```

```
      CALL MSIA (NDAR, 7, 7, NYR, 0)
      NDAR(8) = LIST(5)


C     ------------------------------------------------------------------

C     GET READY TO GO ON-LINE

      WRITE (1, 9062)
9062  FORMAT('SET DATA SWITCH (10+LINE) FOR DATA SET ELIMINATORS'/)
      CALL TCUIR

C     ------------------------------------------------------------------

C     REQUEST THE PRESENT TIME (24 HOUR SYSTEM) FROM KEYBOARD

      WRITE (1, 959)
959   FORMAT ('TYPE IN THE PRESENT TIME (I4)')
      READ (6,9959) ITIME
9959  FORMAT (I4)
      WRITE (1, 960) ITIME
960   FORMAT(/'LINES OPENED AT',I5/'START TCU TO GO ON-LINE'/)

C     ------------------------------------------------------------------

C     OPEN ALL COMMUNICATION LINES

      DO 60 LINE = 1, NOL
      ITEM(1, LINE) = 0
      M(M0, LINE) = 0
      M(M1, LINE) = 0
      M(M2, LINE) = MSIZE
      M(M3, LINE) = MSIZE
      LSB(IMOD, LINE) = 1
      LSB(NRC, LINE) = 1
      LSB(ITEST, LINE) = 1
      LSB(NALT, LINE) = 1
      CALL TCU (5, LINE, M(M1, LINE))
      WRITE (1,9970) LINE
9970  FORMAT ('LINE'I2' OPENED')
      WRITE (107'LINE) NEG
60    CONTINUE

C     ------------------------------------------------------------------

C     PRIMARY COMMUNICATIONS LOOP

100   LINE = 0
C     IF DATA SWITCH 14 IS UP, LIST ALL THE ASSIGNED ID NUMBERS

      CALL DATSW(14, I)
      GO TO (7000, 101), I
101   LINE = LINE +1
      IF (LINE - NOL) 102, 102, 100
```



```
C     TEST TCU STATUS FOR CURRENT LINE

102   CALL TCU (0, LINE, M(M1, LINE))
      IF (M(M1, LINE)) 101, 105, 110

105   ITEMP = M(M0, LINE) +1
      GO TO (1000, 1100, 1200, 1300, 1400, 1500, 1600, 1700, 1800, 1900
     *,ITEMP

C     ------------------------------------------------------------------

C     TCU ERROR 'GO TO' MATRIX

110   IF (M(M0, LINE)) 111, 101, 111
111   NTER = M(M1, LINE)
      NSTAT = M(M0, LINE) +1
      WRITE (1,991) LINE, M(M0, LINE), NTER
991   FORMAT ('L'I2'S'I2'E'I2)

      GO TO (201, 202, 203, 204, 204), NTER

201   GO TO (101,1000,1000,1000,2011,101,101,101,101),NSTAT

2011  NERR = 1
      GO TO 9999

202   GO TO(2021,2021,2021,2021,2022,2022,2022,2022,2022),NSTAT

2021  DO 20211 I = 1, NLSB
20211 LSB(I,LINE) = 0
970   ITEM(1, LINE) = 0
      M(M0, LINE) = 0
      M(M1, LINE) = 0
      M(M2, LINE) = MSIZE
      M(M3, LINE) = MSIZE
      LSB(IMOD, LINE) = 1
      LSB(NRC, LINE) = 1
      LSB(ITEST, LINE) = 1
      CALL TCU (5, LINE, M(M1, LINE))
      WRITE (1,9970) LINE
      GO TO 101

2022  CALL TIME (ITIME)
      WRITE (1,92022) LSB(NCACC, LINE), LINE, ITIME
92022 FORMAT (/I4' DISCONNECTED LINE'I2' AT'I5/)
      LSB(IMAIL, LINE) = ITIME
      CALL ACCT(LSB,LINE,NLSB,NACT,NAMAX,NDATE)
      IF(LSB(IMOD, LINE) - 2) 2021, 2024, 2021

2024  NREC = LSB(NCID, LINE) - NFPO +1
      WRITE (102'NREC) (ITEM(I,LINE), I = 1, NLI)
      GO TO 2021
```

```
203   GO TO (1000,1000,1000,2050,2050,2050,1604,2050,2050),NSTAT
2050  WRITE (NEG,92050) NL
      GO TO 1299
92050 FORMAT ('**INTERRUPTED**'A1'-')


204   GO TO (1000,1000,1000,101,2041,101,101,101,101,101),NSTAT
2041  WRITE(NEG,92041) NL
92041 FORMAT ('COMMUNICATIONS ERROR. RE-ENTER STATEMENT'A1'-')
      GO TO 1299


C     ------------------------------------------------------------------

C     STATUS = 0 (FIRST CALLING IN)

1000  WRITE (NEG, 91000) NL, NL, NL
91000 FORMAT (3A1,'PLEASE TYPE IN YOUR SIMBAD ACCOUNT NUMBER  ')
      M(MO, LINE) = 1
      DO 1003 I = 1, NLSB
      LSB (I, LINE) = 0
1003  ITEM(I, LINE) = 0
      LSB(IMOD, LINE) = 1
      LSB(ITEST, LINE) = 1
1001  CALL TCU (3, LINE, M(M1, LINE))
      GO TO 101

C     ------------------------------------------------------------------

C     STATUS = 1 (FINISHED SENDING INITIAL MESSAGE)

1100  M(MO, LINE) = 2
1101  CALL TCU (2, LINE, M(M1, LINE))
      GO TO 101

C     ------------------------------------------------------------------

C     STATUS = 2 (FINISHED RECEIVING ACCOUNT NUMBER)

1200  READ (NEG, 91200) LIST
91200 FORMAT (120A1)
      ITEMP = M(M2, LINE)
      CALL SCAN (LIST, ITEMP, LA, LCA, LCOA, IOA, LOA)
      CALL FIND (IOA, NFA, NLP, INUM, IVAL)
      IF (INUM - 1) 1000, 1204, 1000
1204  LSB (NCACC, LINE) = IVAL
      CALL TIME (ITIME)
      LSB(NTIME, LINE) = ITIME
      LSB (IMOD, LINE) = 1
      LSB(ITEST, LINE) = 1
      WRITE (1, 91204) IVAL, LINE, ITIME
91204 FORMAT (/I5,' ON LINE',I2,' AT',I5/)
      CALL ACCT (LSB, LINE, NLSB, NACT, NAMAX, NDATE)
      WRITE (NEG,91201) NL, (NDAR(I),I=1,8),ITIME,NL,NL
```



```
91201 FORMAT (A1'SIMBAD READY '8A1,I5,2A1'-')
1299  M(MO, LINE) = 3
      GO TO 1001

C     ------------------------------------------------------------------

C     STATUS = 3 (FINISHED SENDING WELCOME MESSAGE)

1300  M(MO, LINE) = 4
      GO TO 1101

C     ------------------------------------------------------------------

C     STATUS = 4 (FINISHED RECEIVING USER STATEMENT)

1400  READ (NEG, 91200) LIST
      ITEMP = M(M2,LINE)+1
      LIST(ITEMP) = 16448

C     CONVERT ANY IBM/2741 LOWER CASE CHARACTERS TO UPPER CASE

      DO 1403 I = 1, ITEMP
      IF (LIST(I)) 1401, 1403, 1403
1401  IF (LIST(I) + 16320) 1402, 1403, 1403
1402  LIST(I) = LIST(I) + 16384
1403  CONTINUE
      CALL DATSW (LINE, I)
      GO TO (1404, 1405), I
1404  CALL TIME (ITIME)
      WRITE (5, 91400) LINE, ITIME, (LIST(KK), KK = 1, ITEMP)
91400 FORMAT (' LINE',I2,' AT',I5,'-',100A1)
1405  CALL SCAN (LIST, ITEMP, LA, LCA, LCOA, IOA, LOA)

C     CHECK FOR KEY WORDS IN USER'S STATEMENT


C     IF THE LAST CHARACTER OF STATEMENT IS A '/', HAVE HIM RETYPE IT

      I = M(M2, LINE)
      IF (LIST(I) - NCAN) 14052, 14051, 14052
14051 WRITE(NEG, 91405) NL, NL
91405 FORMAT (A1,'RE-TYPE',A1,'-')
      GO TO 1299

C     CHECK FOR THE WORD 'MESSAGE' ANYWHERE IN STATEMENT

14052 CALL FIND (LCOA, NMES, NMES, INUM, IVAL)
      IF (IVAL) 14053, 14053, 5100

14053 CALL FIND (LCOA, NFKW, NLKW, INUM, IVAL)
      IF (INUM) 1406, 1406, 2000
2000  GO TO (5100,5200,5300,5400,5500,5600,5700,5800,5900,6000,6100,6200
     *,6300, 6400,6600), IVAL
```

```
C     CHECK FOR VALID ITEM ID NUMBER

1406  CALL FIND (IOA, NFPO, NLC, INUM, IVAL)
      IF (INUM) 1409, 1409, 1407

C     THERE IS AN ID NUMBER. CHECK FOR 'ALT' IN USER STATEMENT

1407  CALL FIND (LCOA, 102, 102, INUM, JUNK)
      IF (INUM) 14072, 14072, 14076

C     NO 'ALT' LOAD SPECIFIED REGULAR ITEM

14072 CALL LOAD(IVAL,LSB,LINE,NCACC,NCID,IMOD,ITEM,NLI,NFPO,NERR)
      IF (NERR) 1409, 1409, 14073
14073 NERR = NERR +3
      GO TO 9999

C     'ALT' DATA SPECIFIED. MAKE SURE USER IS IN 'TEST' MODE

14076 IF (LSB(ITEST, LINE) -1) 14079, 14[illegible], 14079

C     NOT IN 'TEST' MODE - ERROR

14078 NERR = 22
      GO TO 9999

C     LOAD SPECIFIED ALTERNATE ITEM
14079 I = IVAL -NFPO +1
      IF (I - 2290) 14082, 14082, 1408
1408  NERR = 23
      GO TO 9999

14082 READ (300'I) K, (ITEM(J,LINE), J=2,63)
      LSB(NCID, LINE) = IVAL



C     CHECK FOR A QUESTION NUMBER IN USER'S STATEMENT

1409  CALL FIND (IOA, NQ1, NQ2, INUM, IVAL)
      IF (INUM) 14091, 14091, 1410
14091 CALL FIND (IOA, NQ3, NQ4, INUM, IVAL)
      IF (INUM) 1412, 1412, 1410
1410  LSB(NCQ, LINE) = IVAL
      GO TO 6200

C     CHECK FOR QUESTION 200 OR 300
1412  CALL FIND (IOA,NQ200,NQ200,INUM,IVAL)
      IF (INUM) 14121,14121,1410
14121 CALL FIND (IOA,NQ300,NQ300,INUM,IVAL)
      IF (INUM) 1413,1413,1410
C     CHECK FOR VALID INTEGER VALUES IN USER'S STATEMENT
```



```
1413  CALL FIND (IOA,NFIV,NLIV,INUM,IVAL)
      IF (INUM-1) 1424, 1416, 1414
1414  NERR = 2
      GO TO 9999
1416  LSB(NCIV, LINE) = IVAL

C     CHECK FOR VARIABLE DESIGNATORS IN USER'S STATEMENT

      CALL FIND (IOA,NFVD1,NLVD,INUM,IVAL)
      IF (INUM - 1) 1418,1420,1418
1418  NERR = 3
      GO TO 9999
1420  LSB(NCVD, LINE) = IVAL
      LSB(NCKW, LINE) = 10
      GO TO 6000

C     CHECK FOR VARIABLE DESIGNATORS IN USER'S STATEMENT

1424  CALL FIND (IOA, NFVD1,NLVD, INUM, IVAL)
      IF (INUM) 1426, 1426, 1430
1426  NERR = 1
      GO TO 9999
1430  LSB (NCKW, LINE) = 11
      GO TO 6100

C     ------------------------------------------------------------

C     STATUS = 5 (ENTER HERE AFTER CREATING DELAYED OUTPUT FILE

1500  LSB(NCIV, LINE) = 1
      CALL FIND(LCOA,MAIL,MAIL,LSB(IMAIL,LINE), IVAL)
      IF (LSB(IMAIL, LINE)) 1501, 1501, 15001
15001 CALL TIME (ITIME)
      WRITE (5,91500) LSB(NCACC,LINE),(NOAR(I),I=1,8), ITIME
91500 FORMAT ('1    OUTPUT FOR ACCOUNT'I6,8X'DATE '8A1,5X'TIME'I5)
1501  NFILE = LINE+200
      NCT = LSB(NCIV, LINE)
      READ (NFILE'NCT) L, (LIST(I), I = 121, 181)
      LSB(NCIV, LINE) = LSB(NCIV, LINE) +1
      CALL UNPAC (LIST, 121, 181, LIST, 1)
      IF (LSB(IMAIL, LINE)) 1504, 1504, 1510
1504  WRITE (NEG, 91501) (LIST(I), I = 1, L)
      M(MO, LINE) = 6
      GO TO 1001
91501 FORMAT (120A1)
1510  DO 15102 I = 1, L
      IF (LIST(I) - 5440) 15102, 15101, 15102
15101 LIST(I) = 16448
15102 CONTINUE
      WRITE (5,91510) (LIST(I), I = 1, L)
91510 FORMAT (1H0,120A1)

C     ------------------------------------------------------------
```

```
C     STATUS = 6 (RETURN HERE TO PRINT SUBSEQUENT LINES OF DELAYED FILE

1600  IF (LSB(NCIV, LINE) - LSB(NRC, LINE)) 1501, 1604, 1604
1604  IF (LSB(IMAIL,LINE))1603, 1603, 1608
1603  WRITE (NEG, 91604) NL
      LSB(NRC, LINE) = 1
      GO TO 1299
91604 FORMAT (A1,'-')

1608  WRITE (5,91608)
      WRITE (NEG,91610) NL
      LSB(IMAIL,LINE) = 0
      LSB(NRC, LINE) = 1
      GO TO 1299
91608 FORMAT (1H1)
91610 FORMAT ('OUTPUT HAS BEEN PRINTED AND WILL BE SENT'A1'-')

C     ------------------------------------------------------------

C     STATUS = 8 (RETURN AFTER SENDING DIAGNOSTIC MESSAGE)

1800  WRITE (NEG, 91800) IBLK, NL
91800 FORMAT (2A1,'-')
      GO TO 1299

C     ------------------------------------------------------------

C     MESSAGE

5100  CALL TIME (ITIME)
      WRITE(1,95100)LSB(NCACC,LINE),LINE,ITIME,IRED,(LIST(I),I=1,ITEMP)
95100 FORMAT(/'FROM USER',I5,',LINE',I2,' AT',I5,'  ',A1,90A1)
      WRITE (1, 95102) IBLK
95102 FORMAT (A1/)
      WRITE (NEG, 95101) NL
95101 FORMAT ('MESSAGE SENT',A1,'-')
      GO TO 1299

C     ------------------------------------------------------------

C     GENERATE NEW PO OR CASE ITEM ON DISK

5200  IF (LSB(ITEST, LINE) -1) 5204, 5208, 5204
5204  NERR = 8
      GO TO 9999

C     CHECK FOR THE CODE FOR 'PO'

5208  CALL FIND (LCOA, KPO, KPO, INUM, IVAL)
      IF (INUM) 5210, 5210, 5240

C     GET NEXT AVAILABLE 'CASE' ID
```



```
5210  IF (LECA(1) - 1) 5212, 5212, 5218
5212  IF (LSB(IMOD,LINE)-2) 52110, 5211, 52110
5211  NREC = LSB(NCID,LINE) - NFPO +1
      WRITE (102'NREC) (ITEM(I,LINE), I = 1, NLI)
      LSB(IMOD, LINE) = 1
52110 CALL FILLX(LECA, NEC, NFC, NLC, ITEM, LINE, NFPO, IERR)
      IF (IERR-1) 5218, 5214, 5218
5214  NERR = 9
      GO TO 9999
5218  I = LECA(1)
      IVAL = LECA(I)
      LECA(1) = LECA(1) -1

C     INCREMENT CASE COUNT ON DISK

      READ (105'1) I, J
      I = I +1
      WRITE (105'1) I, J

5220  CALL LOAD(IVAL,LSB,LINE,NCACC,NCID,IMOD,ITEM,    NLI,NFPO,IERR)
      LSB(NCID, LINE) = IVAL

      LSB (IMOD, LINE) = 2
C     ZERO OUT THE NEW ITEM

      GO TO (5225, 5234), NSW1
5225  DO 5230 I = 1, NLI
5230  ITEM(I, LINE) = 0
      ITEM(1, LINE) = LSB(NCACC, LINE)
      WRITE (NEG, 95230) LSB(NCID, LINE), NL
5232  CALL ACCT (LSB, LINE, NLSB, NACT, NAMAX, NDATE)
95230 FORMAT ('NEW I.D. NUMBER IS', I6, A1,'-')
      GO TO 1299
5234  DO 5236 I = 1, NLI
5236  ITEM(I, LINE) = LIST(I)
      NSW1 = 1
      WRITE (107'LINE) NEG
      ITEM(1, LINE) = LSB(NCACC, LINE)
      WRITE (NEG, 95236) LIST(1), LSB(NCID, LINE), NL
      GO TO 5232
95236 FORMAT ('CASE'I6' RESTORED.  NEW I.D. IS'I6,A1'-')

C     GET NEXT AVAILABLE 'PO' ITEM ID

5240  IF (LEPA(1) - 1) 5242, 5242, 5248
5242  IF (LSB(IMOD,LINE) - 2) 5244, 5243, 5244
5243  NREC = LSB(NCID, LINE) - NFPO +1
      WRITE (102'NREC) (ITEM(I, LINE), I = 1, NLI)
      LSB(IMOD, LINE) = 1
5244  CALL FILLX(LEPA, NEP, NFPO, NLPO, ITEM, LINE, NFPO, IERR)
      IF (NERR - 1) 5248, 5214, 5248
5248  I = LEPA(1)
      IVAL = LEPA(I)
      LEPA(1) = LEPA(1) -1
```

```
C     INCREMENT PO COUNT ON DISK

      READ (105'1) I, J
      J = J +1
      WRITE (105'1) I, J

      GO TO 5220

C     ------------------------------------------------------------

C     TEST


C     CHECK STATEMENT FOR 'FINISHED'

5300  CALL FIND (LCOA, 5, 5, INUM, IVAL)
      IF (INUM) 5302, 5302, 5504
5302  IF (LSB(IMOD, LINE) -2) 5306, 5304, 5306
5304  ITEMP = LSB(NCID, LINE) - NFPO+1
      WRITE (102'ITEMP) (ITEM(I, LINE), I = 1, NLI)
      LSB(IMOD, LINE) = 1
5306  LSB(ITEST, LINE) = 2
      WRITE(NEG, 95300) NL
95300 FORMAT ('YOU ARE NOW IN TEST MODE',A1,'-')
      GO TO 1299

C     ------------------------------------------------------------

C     DELETE PO OR CASE ITEM

5400  IF (LSB(ITEST, LINE) -1) 5404, 5408, 5404
5404  NERR = 6
      GO TO 9999

5408  CALL FIND (IOA, NFPO, NLC, INUM, IVAL)
      IF (INUM - 1) 5410, 5412, 5410
5410  NERR = 7
      GO TO 9999
5412  CALL LOAD(IVAL,LSB,LINE,NCACC,NCID,IMOD,ITEM,     NLI,NFPO,NERR)
      IF (NERR) 5416, 5413, 5416
5413  ITEM(1, LINE) = LSB(NCID, LINE)
      WRITE (107'LINE) (ITEM(I,LINE),I=1,NLI)
      ITEM(1, LINE) = 0
      NREC = IVAL - NFPO +1
      WRITE (102'NREC) (ITEM(I,LINE), I = 1, NLI)
      IF (LSB(NCID, LINE) - NLPO) 5424, 5424, 5434
5416  NERR = NERR +4
      GO TO 9999
5424  IF (LEPA(1) - NEP) 5426, 5427, 5427
5426  ITEMP = LEPA(1) +1
      LEPA(ITEMP) = LSB(NCID, LINE)
      LEPA(1) = ITEMP
```



```
C     DECREMENT PO COUNT ON DISK
5427  READ (105'1) I, J
      J = J -1
      WRITE (105'1) I, J

5430  CALL ACCT (LSB, LINE, NLSB, NACT, NAMAX, NDATE)
      WRITE (NEG, 95430) LSB(NCID,LINE), NL
95430 FORMAT ('D',I6,A1,'-')
      LSB(IMOD, LINE) = 1
      IDDEL = LSB(NCID, LINE)
      LSB(NCID, LINE) = 0
      GO TO 1299
5434  IF (LECA(1) - NEC) 5438, 5440, 5440
5438  ITEMP = LECA(1) +1
      LECA(ITEMP) = LSB(NCID, LINE)
      LECA(1) = ITEMP

C     DECREMENT CASE COUNT ON DISK
5440  READ (105'1) I, J
      I = I -1
      WRITE (105'1) I, J

      GO TO 5430

C     ------------------------------------------------------------

C     QUIT

C     SEE IF THE WORD 'TEST' APPEARS IN USER'S STATEMENT

5500  CALL FIND (LCOA, 3, 3, INUM, IVAL)
      IF (INUM) 5508, 5508, 5504

C     THE STATEMENT MEANT 'TERMINATE TEST MODE'

5504  LSB(ITEST, LINE) = 1
      LSB(NCID, LINE) = 0
      ITEM(1, LINE) = 0
      WRITE (NEG, 95504) NL
95504 FORMAT ('YOU ARE NOW IN NORMAL MODE',A1,'-')
      GO TO 1299

C     THE STATEMENT MEANT 'QUIT ALTOGETHER'

5508  CALL TIME (ITIME)
      WRITE (1, 91701) LSB(NCACC, LINE), LINE, ITIME
91701 FORMAT (/I4,' OFF LINE',I2,' AT',I5/)
      LSB(IMAIL, LINE) = ITIME
      CALL ACCT (LSB, LINE, NLSB, NACT, NAMAX, NDATE)
      IF (LSB(IMOD, LINE) -2) 5512, 5510, 5512
5510  NREC = LSB(NCID, LINE) - NFPO +1
      WRITE (102'NREC) (ITEM(I, LINE), I = 1, NLI)
5512  IF (LSB(NCKW, LINE)) 5513, 5514, 5514
```

```
5513  M(MO, LINE) = 0
      GO TO 101
5514  WRITE (NEG, 95508) NL, ITIME, NL, NL, NL
95508 FORMAT (A1,'OFF AT',I5,3A1)
      M(MO, LINE) = 7
      GO TO 1001

C     RETURN HERE AFTER FINAL MESSAGE HAS BEEN SENT.  CLOSE LINE

1700  DO 1704 I = 1, NLSB
1704  LSB(I, LINE) = 0
      M(MO, LINE) = 0

C     DISCONNECT

C     IF THIS LINE IS ON A DATASET ELIMINATOR, DON'T DOSCONNECT

      I = LINE +10
      CALL DATSW (I, J)
      GO TO (1000, 1706), J

1706  CALL TCU (6, LINE, M(M1, LINE))
      CALL TCU (0,LINE,M(M1,LINE))
1708  CONTINUE
      IF (M(M1,LINE)) 1708,970,970

C     ------------------------------------------------------------

C     SUMMARY ROUTINE

5600  IF (LSB(IMOD, LINE) - 2) 5604, 5602, 5604

5602  NREC = LSB(NCID, LINE) - NFPO +1
      WRITE (102'NREC) (ITEM(I,LINE), I = 1, NLI)
      LSB(IMOD, LINE) = 1
5604  DO 5605 I = 1, 10
5605  LSUM(NLX,I,LINE) = 0

C     CHECK FOR 'ALT' IN USER STATEMENT AND SET SWITCH ACCORDINGLY

      LSB(NALT, LINE) = 1
      CALL FIND (LCOA, 102, 102, INUM, IVAL)
      IF (INUM) 56053, 56053, 56052
56052 LSB(NALT, LINE) = 2

C     CHECK FOR WORD 'EXTERNAL' IN USER STATEMENT AND SET 'NEXC'

56053 CALL FIND (LCOA, KEXC, KEXC, INUM, IVAL)
      NEXC = 1
      IF (INUM) 56054, 56055, 56054
56055 NEXC = 2


C     SEE IF THE PROPER NUMBER OF INTEGERS IS SPECIFIED
```



```
56054 CALL FIND (IOA,NFVDI,NLVD,INUM,IVAL)
      IF (INUM) 5606, 56066, 5608
5606  NERR = 16
      GO TO 9999
56066 NIST = 1
      GO TO 5612

5608  IF (INUM - 10) 5610, 5610, 5606

5610  NIST = 2
      J = 2
      DO 5612 I = 1, INUM
      LSUM (NLX,I,LINE) = IOA(J)
5612  J = J +1
      LSB(IMAIL, LINE) = 0
      CALL FIND (LCOA, MAIL, MAIL, INUM, IVAL)
      IF (INUM) 6500, 6500, 5613
5613  LSB(IMAIL, LINE) = 1
      GO TO 6500


C     ------------------------------------------------------------

C     STATUS REPORT (TEMPORARY VERSION)

5700  LSB(NRC,LINE) = 1
      JJ = 0
      IF (LSB(IMOD, LINE) - 2) 5701, 57001, 5701
57001 J = LSB(NCID, LINE) - NFPO +1
      WRITE (102'J) (ITEM(I,LINE),I=1,NLI)
5701  J = NFC + 150
      WRITE (NEG, 95701) LSB(NCACC, LINE), NL
95701 FORMAT ('FILES FOR ACCOUNT'I6,A1)
      CALL DELAY(M, MO, M1, M2, LINE, LIST, LSB, NRC, 24)
      DO 5708 I = NFPO, J
      CALL TCU (0,LINE,M(M1,LINE))
      IF (M(M1,LINE)) 57011,57011,110
57011 CONTINUE
      N = I - NFPO +1
      READ (102'N) K
      IF (K - LSB(NCACC, LINE)) 5708, 5702, 5708
5702  JJ = JJ +1
      NST(JJ,1) = I
      NST(JJ, 2) = K
5708  CONTINUE
      IF (JJ) 5713, 5713, 5710
5710  DO 5712 I = 1, JJ
95702 FORMAT (I6, A1)
      WRITE (NEG, 95702) NST(I,1), NL
5712  CALL DELAY (M, MO, M1, M2, LINE,LIST,LSB,NRC, 7)
5713  READ (105'1) I, J
      WRITE (NEG, 95700) I, NL
      CALL DELAY(M,MO,M1,M2,LINE,LIST,LSB,NRC,13)
```

```
95700 FORMAT (I6,' CASES',A1)
      GO TO 101

C     ------------------------------------------------------------------

C     SET OR DISPLAY THE NPR ARRAYS

5800  CALL FIND (IOA, 1, 1000, INUM, IVAL)
      IF (INUM) 5810, 5804, 5810

5804  WRITE (NEG, 95804) (NPRI3(I,LINE),I=1,3),(NPRI2(I,LINE),I=1,2),NL
95804 FORMAT ('NPRI3'3I4'  NPRI2'2I4,A1'-')
      GO TO 1299

5810  IF (INUM-5) 5814, 5812, 5814
5812  NPRI3(1,LINE) = IOA(2)
      NPRI3(2,LINE) = IOA(3)
      NPRI3(3,LINE) = IOA(4)
      NPRI2(1,LINE) = IOA(5)
      NPRI2(2,LINE) = IOA(6)
      GO TO 5804

5814  WRITE (NEG, 95814) NL
95814 FORMAT('YOU MUST SPECIFY 5 VALUES, 3 FOR NPRI3 AND 2 FOR NPRI2'A1
     *-')
      GO TO 1299

C     ------------------------------------------------------------------

5900  IF (LSB(ITEST, LINE) -1) 5910, 5901, 5910
5901  CALL FIND (IOA, NFPO, NLC, INUM, IVAL)
      IF (INUM - 1) 5902, 5904, 5902
5902  NERR = 18
      GO TO 9999

5904  READ (107'LINE) (LIST(I), I = 1, NLI)
      IF (LIST(1) - IVAL) 5906, 5908, 5906
5906  NERR = 19
      GO TO 9999
5908  NSW1 = 2
      IF (IVAL - NFC) 5240, 5210, 5210
5910  NERR = 20
      GO TO 9999
C     ------------------------------------------------------------------

C     UPDATE ITEM

6000  LSB (NCKW, LINE) = 10
      IF (LSB(NCID, LINE)) 6001, 6001, 60011
6001  NERR = 4
      GO TO 9999
60011 IF (LSB(ITEST, LINE) -2) 6002, 6004, 6002
6002  LSB(IMOD, LINE) = 2
6004  LTVD = LSB(NCVD, LINE) - NFVD+1
```



```
      NSAVE = ITEM(LTVD, LINE)

C     MAKE SURE THAT THE VALUE SPECIFIED FOR THIS VARIABLE IS LEGAL

      IF (LSB(NCIV, LINE) - IVMAX(LTVD-1))6006, 6006, 6005

C     ILLEGAL

6005  WRITE (NEG, 96005) LSB(NCVD, LINE), IVMAX(LTVD-1), NL
96005 FORMAT ('MAXIMUM VALUE FOR VARIABLE'I3' IS'I3,A1,'-')
      GO TO 1299

6006  ITEM(LTVD, LINE) = LSB(NCIV, LINE)
      IF (LSB(ITEST, LINE) - 1) 6008, 6008, 6010
6008  CALL ACCT (LSB, LINE, NLSB, NACT, NAMAX, NDATE)
6010  WRITE (NEG, 96000) LSB(NCID, LINE), LSB(NCVD, LINE),LSB(NCIV,LINE)
     *,NSAVE, NL
96000 FORMAT('U'I6,I4,I2' WAS 'I2,A1'-')
      GO TO 1299

C     ------------------------------------------------------------------

C     TYPE INTEGER VALUES CORRESPONDING TO EACH VARIABLE
C     DESIGNATOR IN THE USER'S STATEMENT

6100  LSB (NCKW, LINE) = 11

      CALL FIND (IOA, NFPO, NLC, INUM, IVAL)
      IF (INUM) 6102, 6102, 6101
6101  CALL LOAD(IVAL,LSB,LINE,NCACC,NCID,IMOD,ITEM,    NLI,NFPO,NERR)
      IF (NERR) 6102, 6102, 14073
C     FIND THE NUMBER OF DESIGNATORS IN STATEMENT

6102  CALL FIND (IOA, NFVD1,NLVD, INUM, IVAL)
      IF (INUM) 61021, 61021, 61022
61021 NERR = 10
      GO TO 9999
61022 IF (LSB(NCID, LINE)) 61023, 61023, 61024
61023 NERR = 4
      GO TO 9999
61024 K = 1
      LSB(NRC, LINE) = 1
      DO 6115 I = 1, INUM
6103  K = K+1
      IF (IOA(K) - NFVD1)6103, 6110, 6108
6108  IF (IOA(K) - NLVD) 6110, 6110, 6103
6110  J = IOA(K) - NFVD +1
      WRITE (NEG,96110) LSB(NCID,LINE), IOA(K), ITEM(J, LINE), NL
96110 FORMAT (I6,I4,I2,A1)
      CALL DELAY (M, M0, M1, M2, LINE, LIST, LSB, NRC, 13)
6115  CONTINUE
      GO TO 101

C     ------------------------------------------------------------------
```

```
C     QUESTION ROUTINE

6200  IF (LSB(NCID, LINE)) 6201, 6201, 6202
6201  NERR = 4
      GO TO 9999
6202  WRITE(NEG,96202) LSB(NCQ,LINE), LSB(NCID,LINE), NL
96202 FORMAT ('QUESTION'I4' CASE'I6,A1)
      LSB(NRC, LINE) = 1
      CALL DELAY(M,M0,M1,M2,LINE,LIST,LSB,NRC,24)
      IF (LSB(NCQ, LINE) - NQ5) 6203, 6204, 6204

C     LOWER QUESTION RANGE (WORSE-SAME-BETTER)

6203  NVAR = LSB(NCQ, LINE) - NQ1 + 1
      NX = 2
      GO TO 6210

C     UPPER QUESTION RANGE (RECID-NON RECID)

6204  NVAR = LSB(NCQ, LINE) - NQ3 + 1
      NX = 3

6210  NV = 0
      JJ = 0

      IF (NVAR) 62040, 62040, 62100
62040 NVAR = 63
62100 DO 6214 I = 2, NLI
      IF (ITEM(I, LINE)) 6214, 6214, 6212
6212  IF (NVAR-(I-1)) 6213,6214,6213
6213  NV = NV +1
      JJ = JJ +1
      NDUM(JJ) = I-1
      NVAL(JJ) = ITEM(I, LINE)
6214  CONTINUE
      IF (NV) 6215, 6255, 6215

6215  DO 6224 I = 1, NV
      K = NDUM(I)
      KK = NVAL(I)
      GO TO (6216, 6220, 6216), NX
6216  DO 6218 J = 1, 3
      NPDH(J, I) = MATR(J, KK, K)
6218  CONTINUE
      GO TO 6224

6220  DO 6222 J = 1, 2
      NPDH(J, I) = NATR(J, KK, K)
6222  CONTINUE
6224  CONTINUE

      IF (NVAR - 63) 62241, 62242,62241
62242 KKK = 1
```



```
      GO TO 6232
62241 DO 6230 J = 1, 10
      IF (MATR(1, J, NVAR)) 6228, 6226, 6228
6226  KKK = J -1
      GO TO 6232
6228  CONTINUE
6230  CONTINUE

6232  GO TO (6234, 6242, 6234), NX
6234  DO 6240 I = 1, KKK
      IF (NVAR - 63) 62349, 62369, 62369
62349 DO 6236 J = 1, 3
      L = NV +1
      NPDH(J, L) = MATR(J, I, NVAR)
6236  CONTINUE
      GO TO 62361
62369 L = NV

62361 CALL BAYES (NPRI3, NPOS, NPDH, L, NX,LINE)

      DO 6238 KK = 1, 3
      EPOS(I,KK) = NPOS(KK)
      EPOS(I,KK) = EPOS(I,KK) / 1000.
6238  CONTINUE
6240  CONTINUE
      GO TO 6250

6242  DO 6248 I = 1, KKK
      IF (NVAR - 63) 62429, 62449, 62449
62429 DO 6244 J = 1, 2
      L = NV +1
      NPDH(J, L) = NATR(J, I, NVAR)
6244  CONTINUE
      GO TO 62441
62449 L = NV

62441 CALL BAYES (NPRI2, NPOS, NPDH, L, NX, LINE)

      DO 6246 KK = 1, 2
      EPOS(I,KK) = NPOS(KK)
      EPOS(I,KK) = EPOS(I,KK) / 1000.
6246  CONTINUE
6248  CONTINUE

C     WRITE OUT THE NEPOS ARRAY

6250  N = 1
      DO 62513 K=1,2
      CALL LABL (NVAR,K,N,IOUT,LABEL,NDUM,ICHAR)
      N = 10
62513 CONTINUE
      GO TO (62505,62505,62510),NX
62505 WRITE (NEG,62550) (IOUT(N),N=1,18)
62550 FORMAT (18A1,'    NON-R    RECID')
```

```
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,36)
      GO TO 62515
62510 WRITE (NEG,62551) (IOUT(N),N=1,18)
62551 FORMAT (18A1,'    WORSE     SAME    BETTER')
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,45)
62515 DO 62525 I=1,KKK
      I2 = I + 2
      CALL LABL (NVAR,I2,1,IOUT,LABEL,NDUM,ICHAR)
      WRITE (NEG,62526) NL,(IOUT (N),N=1,9),( EPOS(I,J),J=1,NX)
62526 FORMAT (1A1,9X,9A1,3F9.3)
      IDL = NX * 9 + 19
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,IDL)
62525 CONTINUE
      GO TO 101

6255  NERR = 15
      GO TO 9999

C     ------------------------------------------------------------------

C     SECTION TO MAINTAIN SUMMARY MATRIX

6300  INF = NFVD +1
      CALL FIND (IOA, INF, NLVD, NV, ND)
      ND = ND-NFVD+1
      IF (NV -1) 6304, 6330, 6308
6304  NERR = 11
      GO TO 9999

6308  CALL FIND (IOA, 0, 9, INUM, IVAL)
      IF (INUM) 6310, 6312, 6310
6310  NERR = 12
      GO TO 9999

6312  I = 1
      LSB(NRC, LINE) = 1
C     FOLLOWING INSTRUCTION IS STRANGE
      NV = NV+1
      DO 6320 ITEMP = 1, NV
6314  I = I +1
      K = IOA(I)
      IF (K-INF) 6314, 6318, 6316
6316  IF (K - NLVD) 6318, 6318, 6314
6318  L = K-NFVD+1
      WRITE (NEG,96318) K, (LSUM(L,J,LINE), J=1,10), NL
96318 FORMAT (I4,') ',10I2,A1)
      CALL DELAY(M,MO,M1,M2,LINE,LIST,LSB,NRC,27)
6320  CONTINUE
      GO TO 101

6330  CALL FIND (IOA, 1, 10, INUM, IVAL)
      IF (INUM) 6345, 6312, 6334
6334  DO 6338 I = 1, 10
6338  LSUM(ND, I, LINE) = 0
```



```
      I = 1
      DO 6345 ITEMP = 1, INUM
6340  I = I +1
      J = IOA(I) - 1
      IF (J) 6340, 6344, 6342
6342  IF (J - 9) 6344, 6344, 6340
6344  J = J+1
      LSUM(ND, J, LINE) = 1
6345  CONTINUE
      WRITE (NEG, 96345) NL
96345 FORMAT (A1,'-')
      GO TO 1299

C     ------------------------------------------------------------------

C     ROUTINE TO CLEAR SUMMARY MATRIX

6400  DO 6402 I = 1, NLI
      DO 6402 J = 1,10
6402  LSUM(I, J, LINE) = 0
      WRITE (NEG, 96405) NL
96405 FORMAT ('SUMMARY MATRIX CLEARED',A1,'-')
      GO TO 1299

C     ------------------------------------------------------------------
C     ------------------------------------------------------------------

C     ROUTINE TO PERFORM SUMMARY
6500  READ (105'1) NKIDS
      IF(LSB(IMAIL,LINE)) 6597,6597,6598
6598  NL = 16448
6597  IF (NKIDS) 6570,6570,6562
6562  NAC = LSB(NCACC,LINE)
      IF (LSB(NALT, LINE) -1) 65405, 65405, 65406
65406 NRRA = 0
65405 LSB(NRC,LINE) = 1
      LCT = 0
      JCNT = 0
      NEXCN = 1
      NFK = NFC
      JMC = 0
      IMC = 0
      NMC = 0
      NCT = 0
      JK = 0
65004 GO TO (65005,65908),NEXC
65005 DO 65003 I = 1,NLI
      DO 65003 J = 1,10
      IF(LSUM(I,J,LINE)) 65001,65001,65002
65001 LSUM (I,J,LINE) = 1
      GO TO 65003
65002 LSUM (I,J,LINE) = 0
65003 CONTINUE
```

```
65908 GO TO (65909,101),NFXCN
65909 DO 6590 II=1,10
      IAB(II) = II
      DO 6590 JJ=1,10
      DO 6590 KK=1,10
      NOUT(II,JJ,KK) = 0
6590  CONTINUE
      GO TO (65902,65901),NIST
65901 INX = LSUM(NLX,1,LINE) - (NFVD-1)
      INXX = INX - 1
      INXJ = IVMAX(INXX)
      DO 6535 II=1,INXJ
      IF(LSUM(INX,II,LINE)) 6535,6535,6534
6534  JK = JK + 1
      IAB(JK) = II
6535  CONTINUE
      IF(JK) 65081,65081,65351
65351 DO 65591 I=1,NLI
      DO 65591 J=1,10
      IF (LSUM(I,J,LINE)) 65591,65591,65592
65591 CONTINUE
      NERR = 21
      NL = 21 * 256
      GO TO 9999
65592 IDUM = 0
      DO 6501 I=1,NPR
      IF(LSUM(NLX,I,LINE))6501,6501,6591
6591  IDUM = IDUM + 1
      IF (IDUM-3) 6501,6560,6560
6501  CONTINUE
6560  GO TO (6503,6509,6504),IDUM
6503  KK = 1
      JJ = 1
      GO TO 6515
6504  CALL RKID (NKIDS,ITEM,LINE,NAC,NFK,NLI,NER,NFPO,LCT,LSB,NRRA,NALT)
      IF(NER-2) 6565,6589,6589
6565  IX = LSUM(NLX,3,LINE) - (NFVD-1)
      DO 6508 III=1,10
      IF(LSUM(IX,III,LINE)) 6508,6508,6506
6506  JMC = JMC + 1
      IF(ITEM(IX,LINE)-III   ) 6508,6507,6508
6507  JJ = III
      GO TO 6511
6508  CONTINUE
      IF (JMC) 65081,65081,65192
65081 NERR = 21
      NL = 21 * 256
      GO TO 9999
6509  JJ = 1
6510  CALL RKID (NKIDS,ITEM,LINE,NAC,NFK,NLI,NER,NFPO,LCT,LSB,NRRA,NALT)
      IF (NER-2) 6511,6589,6589
6511  IY = LSUM(NLX,2,LINE) - (NFVD-1)
      DO 6514 JJJ = 1,10
      IF(LSUM(IY,JJJ,LINE)) 6514,6514,6512
```



```
6512  IMC = IMC + 1
      IF(ITEM(IY,LINE)-JJJ   ) 6514,6513,6514
6513  KK = JJJ
      GO TO 6516
6514  CONTINUE
      IF (IMC) 65081,65081,65192
6515  CALL RKID (NKIDS,ITEM,LINE,NAC,NFK,NLI,NER,NFPO,LCT,LSB,NRRA,NALT)
      IF(NER-2) 6516,6589,6589
6516  MX = LSUM(NLX,1,LINE) - (NFVD-1)
      DO 6519 I=1,JK
      INM = IAB(I)
      NMC = NMC + 1
      IF(ITEM(MX,LINE) - INM) 65192,6518,6519
6518  II = I
      GO TO 65191
6519  CONTINUE
      GO TO 65192
65191 NOUT(KK,II,JJ) = NOUT(KK,II,JJ) + 1
      JCNT = JCNT + 1
65192 CALL TCU (0,LINE,M(M1,LINE))
      IF (M(M1,LINE)) 65195,65195,110
65195 GO TO (6515,6510,6504),IDUM
6589  IF (LCT) 6570,6570,6571
6570  NERR = 17
      NL = 21 * 256
      GO TO 9999
6571  IXX = IX - 1
      IF (JCNT) 6570,6570,65711
65711 IYY = IY - 1
      IXJ = IVMAX(IXX)
      IYJ = IVMAX(IYY)
      NX = LSUM(NLX,2,LINE) - (NFVD-1)
      MX = LSUM(NLX,3,LINE) - (NFVD-1)
      CALL FILL (IOUT,1,115,16448)
      NIZ = 0
6595  NIZ = NIZ + 1
      GO TO (6573,6574,6575),NIZ
6573  IX = INX - 1
      N = 1
      IF (IDUM - 1) 6587,6587,6576
6574  IX = NX - 1
      N = 19
      IF(IDUM-2) 6567,6587,6576
6575  IX = MX - 1
      N = 37
      IF(IDUM-3) 6567,6587,6587
6587  NIZ = 3
6576  CALL LABL (IX,1,N,IOUT,LABEL,NDUM,ICHAR)
      N = N + 9
      CALL LABL (IX,2,N,IOUT,LABEL,NDUM,ICHAR)
6567  IF(NIZ-3) 6595,6596,6596
6596  GO TO (6561,6566,6563),IDUM
6561  WRITE (NEG,6554) (IOUT(I),I=1,18),NL,NL,NL
6554  FORMAT ('SUMMARY OF ',21A1)
```

```
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,32)
      GO TO 6542
6566  WRITE (NEG,6555) (IOUT(I),I=1,36),NL,NL,NL
6555  FORMAT ('SUMMARY OF ',18A1,' BY ',21A1)
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,54)
      GO TO 6542
6563  WRITE (NEG,6556) (IOUT(I),I=1,54),NL,NL,NL
6556  FORMAT ('SUMMARY OF ',18A1,' BY ',18A1,' BY ',21A1)
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,76)
6542  IDL = (JK-1) * 10 + 19
      MXX = MX - 1
      MXJ = IVMAX(MXX)
      CALL FILL (IOUT,1,115,16448)
      DO 6528 L=1,MXJ
      IF(LSUM(NLX,3,LINE)) 6582,6582,6581
6581  IF(LSUM(MX,L,LINE)) 6528,6528,6525
6525  IX = MX - 1
      K = L + 2
      CALL LABL (IX,K,1,IOUT,LABEL,NDUM,ICHAR)
      WRITE (NEG,6564) (IOUT(I),I=1,9),NL,NL
6564  FORMAT (20X,11A1)
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,31)
6582  N = 1
      CALL FILL (IOUT,1,115,16448)
      DO 6577 I=1,JK
      IX=INX - 1
      K = IAB(I) + 2
      CALL LABL (IX,K,N,IOUT,LABEL,NDUM,ICHAR)
      N = N + 10
6577  CONTINUE
      N = N - 1
      WRITE (NEG,6568) (IOUT(I),I=1,N)
      ILL = JK * 10 + 10
6568  FORMAT (10X,101A1)
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,ILL)
      NXX = NX - 1
      NXJ = IVMAX(NXX)
      DO 6527 I=1,NXJ
      CALL FILL (IOUT,1,115,16448)
      IF (LSUM(NLX,2,LINE)) 6584,6584,6586
6586  IF(LSUM(NX,I,LINE)) 6527,6527,6526
6526  IX = NX - 1
      K = I + 2
      CALL LABL (IX,K,1,IOUT,LABEL,NDUM,ICHAR)
      IF (LSB(IMAIL,LINE)) 6584,6584,65845
65845 WRITE (NEG,65946) NL,(IOUT(II),II=1,9),(NOUT(I,J,L),J=1,JK)
65946 FORMAT (10A1,I9,9I10)
      GO TO 65841
6584  WRITE (NEG,6594) NL,(IOUT(II),II=1,9),(NOUT(I,J,L),J=1,JK)
6594  FORMAT (10A1,I10,9I10)
      IDL = (JK - 1) * 10 + 20
65841 CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,IDL)
      DO 65847 IM = 1,JK
      NCT = NCT + NOUT(I,IM,L)
```



```
65847 CONTINUE
      IF (LSUM(NLX,2,LINE)) 6583,6583,6527
6527  CONTINUE
6583  WRITE (NEG,6569) NL,NL,NL
6569  FORMAT (3A1)
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,3)
      IF (LSUM(NLX,3,LINE)) 6585,6585,6528
6528  CONTINUE
6585  WRITE (NEG,65695) NCT
65695 FORMAT (     'TOTAL CASES IN SUMMARY ',I5)
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,28)
      NL = 21 * 256
      NEXCN = 2
      GO TO 65004

65902 NRRA = 0
      WRITE  (NEG,65998) LSB(NCACC,LINE)
65998 FORMAT (' FILES FOR ACCOUNT ',I6,' WHICH MATCH THE SUMMARY MATRIX'
     *)
      CALL DELAY (M,MO,M1,M2,LINE,LIST,LSB,NRC,56)
      LLL = 0
65905 L = 0
      DO 65903 I =1,15
65903 IOUT(I) = 0
65904 CALL RKID(NKIDS,ITEM,LINE,NAC,NFK,NLI,NER,NFPO,LCT,LSB,NRRA,NALT)
      IACT = 0
      IBCT = 0
      IF (LSB(NALT,LINE) -1) 65541, 65541, 65542
65541 NXJ = NFK - 1
      GO TO 65543
65542 NXJ = NRRA + NFPO
65543 IF (NER - 2) 65281,65906,65906
65281 DO 65851 I=1,62
      MMM = 0
      JJJ = 0
      GO TO (65851,65272),NEXC
65272 NNN = IVMAX(I)
      DO 65077 IBB=1,NNN
      IF (LSUM(I+1,IBB,LINE)) 65077,65077,65088
65077 CONTINUE
      GO TO 65851
65088 IACT = IACT + 1
      IF (ITEM(I+1,LINE)) 65851,65851,65282
65282 DO 65284 J=1,NNN
      IF (LSUM(I+1,J,LINE)) 65284,65284,65283
65283 MMM = 1
      IF (ITEM(I+1,LINE) - J) 65284,65285,65284
65285 JJJ = 1
      IBCT = IBCT + 1
65284 CONTINUE
      IF (MMM) 65851,65851,65286
65286 IF(JJJ) 65904,65904,65851
65851 CONTINUE
      IF (IACT - IBCT) 65904,65852,65904
```

```
65852 LLL = LLL + 1
      L = L + 1
      IOUT(L) = NXJ
      IF (L - 15) 65904,65906,65906
65906 WRITE (NEG,65907) NL,NL,(IOUT(I),I=1,L)
65907 FORMAT (2A1,1516)
      IDL = L * 6 + 2
      CALL DELAY (M,M0,M1,M2,LINE,LIST,LSB,NRC,IDL)
      IF (NER - 2) 65905, 65999, 65999
65999 WRITE (NEG,65696) NL, NL, LLL
65696 FORMAT (2A1,'TOTAL CASES IN SUMMARY ',I5)
      CALL DELAY (M,M0,M1,M2,LINE,LIST,LSB,NRC,30)
      NL = 21 * 256
      NEXCN = 2
      GO TO 65004

C     ------------------------------------------------------------------

C     TYPE OUT ALL NON-ZERO VARIABLES FOR SPECIFIED KID OR PO

6600  LSB(NRC, LINE) = 1
      CALL FIND (IOA, NFPO, NLC, INUM, IVAL)
      IF (INUM) 6608, 6608, 6604

6604  CALL LOAD (IVAL,LSB,LINE,NCACC,NCID,IMOD,ITEM,NLI,NFPO,NERR)

      IF (NERR) 6608, 6608, 6606
6606  NERR = 5
      GO TO 9999
6608  IF (LSB(NCID,LINE)) 6201,6201,6609

6609  WRITE (NEG, 96608) NL, LSB(NCID, LINE), NL
96608 FORMAT (A1,'DATA IN FILE OF CASE'I6,A1)
      CALL DELAY(M,M0,M1,M2,LINE,LIST,LSB,NRC,28)

      DO 6620 I = 2, NLI
      IF (ITEM(I, LINE)) 6620, 6620, 6610
6610  J = I + NFVD-1
      WRITE (NEG,96610) J, ITEM(I, LINE), NL
      CALL DELAY(M,M0,M1,M2,LINE,LIST,LSB,NRC,10)
6620  CONTINUE
      GO TO 101
96610 FORMAT (I3, I6, A1)

C     ------------------------------------------------------------------

C     LIST ALL ASSIGNED ID NUMBERS AND ACCOUNTS

7000  WRITE (5, 97001)
      IC = 0
      WRITE (1, 97000)
97000 FORMAT ('PUT DOWN SWITCH 14')
97001 FORMAT ('1DISK SUMMARY'/)
      J = NFC + 150
```



```
      DO 7010 I = NFPO, J
      N = I - NFPO +1
      READ (102'N) K
      IF (K) 7010, 7010, 7005
7005  WRITE (5, 97005) K, I
      IC = IC + 1
97005 FORMAT (' ACCOUNT'I6' ID='I5)
7010  CONTINUE
      WRITE (5,97012)
      READ (105'1)-NKIDS,I
      WRITE (105'1) IC,I
97012 FORMAT (1H1)
7012  CALL DATSW (14, I)
      GO TO (7012, 101), I

C     ------------------------------------------------------------------

C     ------------------------------------------------------------------

C     GENERAL USER ERROR ROUTINE

9999  IF (NERR) 998, 998, 999
998   NERR = 1
999   READ (104'NERR) LENG, (LIST(I), I = 1, 79)
      WRITE (NEG, 99999) IRED, (LIST(I), I = 1, LENG),IBLK, NL
      M(MO, LINE) = 8
      GO TO 1001

99999 FORMAT ('*** ',80A1)
C-------T E M P O R A R Y-----------------------------------------------
1900  PAUSE
      GO TO 6200
      END
// DUP
*DELETE              SIMBA
*STORE       WS  UA  SIMBA

// JOB    0003 0008 0007
// XEQ SIMBA    3
*FILES(100,LEX,0008),(102,CASE,0008),(103,SIMAC,0008),(104,SIMER,0008)
*FILES(105,SIMCT,0008),(106,SIMAT,0008),(201,DOP1,0008),(202,DOP2,0008)
*FILES(101,LABEL,0008),(107,SIMDL,0008),(300,SCRAP,0008)
```

```
// JOB
// FOR
* LIST SOURCE PROGRAM
*ONE WORD INTEGERS

C      THIS SIMBAD SUBROUTINE SCANS THE 'LARR' ARRAY FOR
C      WORDS WHOSE VALUE IS BETWEEN N1 AND N2 (INCLUSIVE).
C      NUM IS SET TO THE NUMBER OF QUALIFYING WORDS.  NVAL IS
C      SET TO THE VALUE OF THE FIRST QUALIFYING WORD.

       SUBROUTINE FIND (LARR, N1, N2, NUM, NVAL)
       DIMENSION LARR(50)
       NVAL = 0
       NUM = 0
       IF (LARR(1)) 5,5,1
 1     K = LARR(1)+1
       DO 4 I=2,K
       IF (LARR(I)-N1) 4,3,2
 2     IF (LARR(I)-N2) 3,3,4
 3     NUM = NUM+1
       IF (NUM-1) 4, 31, 4
 31    NVAL = LARR(I)
 4     CONTINUE
5      RETURN
       END
// DUP
*DELETE              FIND
*STORE       WS  UA  FIND
```



```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
*TRANSFER TRACE
*ARITHMETIC TRACE
       SUBROUTINE LABL (IX,K,N,IOUT,LABEL,NDUM,ICHAR)
       DIMENSION NDUM(60),IOUT(115),LABEL(79,11,3)
       NDUM(1) = LABEL(IX,K,1)
       NDUM(2) = LABEL(IX,K,2)
       NDUM(3) = LABEL(IX,K,3)
       CALL A3A1 (NDUM,1,3,IOUT,N,ICHAR)
       RETURN
       END
// DUP
*DELETE              LABL
*STORE       WS  UA  LABL
```

PAGE 1 SELECT CASES WITH CORRECT ACCOUNT NUMBER FROM DISK

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
*TRANSFER TRACE
*ARITHMETIC TRACE
      SUBROUTINE RKID (NKIDS,ITEM,LINE,NAC,NFK,NLI,NER,NFPO,LCT,LSB,NRRA
     *,NALT)
      DIMENSION ITEM(80,2),LSB(20,2)
      NER = 1
      NR = NFK - NFPO + 1
4     IF (NKIDS) 1,1,2
1     NER = 2
12    RETURN
2     IF(LSB(NALT,LINE)-1) 5,5,6
6     NRRA = NRRA + 1
      READ (300'NRRA) K,(ITEM(I,LINE),I=2,63)
      IF (NRRA - 2289) 20,1,1
20    LCT = LCT + 1
      GO TO 12
5     READ (102'NR) (ITEM(I,LINE),I=1,NLI)
      NFK = NFK + 1
      NR = NFK - NFPO + 1
      IF (ITEM(1,LINE)) 2,2,3
3     NKIDS = NKIDS - 1
      IF (ITEM(1,LINE)-NAC) 4,10,4
10    LCT = LCT + 1
      GO TO 12
      END

// DUP
*DELETE              RKID
*STORE      WS  UA   RKID
```


PAGE 1 PROGRAM TO CREATE SIMBAD ERROR MESSAGE FILE ON DISK

```
// JOB    0003 0008
// DUP
*DELETE              SIMER
*STOREDATA  WS  UA  SIMER 00080008  0008
// FOR
*ONE WORD INTEGERS
*IOCS(CARD,1403 PRINTER,DISK)
**SIMBAD PROGRAM TO CREATE DIAGNOSTIC MESSAGE 'SIMER' FILE ON DISK
      DIMENSION IN(80)
      DEFINE FILE 104(32, 80, U, ITEMP)

      N = 1
      WRITE (5,902)
100   READ (2, 900) (IN(I), I = 1, 80)
      IF (IN(1) - 16448) 101, 5, 101
101   I = 81
1     I = I - 1
      IF (IN(I) - 16448) 3, 2, 3
2     IF (I) 100, 100, 1
3     WRITE (104'N) I, (IN(J), J = 1, 79)
      WRITE (5, 901) N, I, (IN(J), J = 1, 79)
      N = N +1
      GO TO 100
5     WRITE (5, 902)
      CALL EXIT

900   FORMAT (80A1)
901   FORMAT ('0ERROR',I3,' LENGTH',I3,'       MESSAGE     ',79A1)
902   FORMAT (1H1)
      END
// XEQ          1
*FILES(104,SIMER,0008)
I CAN'T MAKE ANY SENSE OUT OF YOUR LAST STATEMENT
ONLY 1 VALUE NUMBER MAY BE INCLUDED IN THIS STATEMENT
THIS STATEMENT NEEDS 1 ITEM ONLY, AS WELL AS A VALUE
NO VALID I.D. SPECIFIED
YOU ARE ATTEMPTING TO USE AN I.D. NUMBER NOT ASSIGNED TO YOU
YOU MAY NOT DELETE AN I.D. WHILE IN TEST MODE
YOU MAY NOT DELETE MORE THAN 1 I.D. NUMBER AT A TIME
YOU MAY NOT CREATE NEW I.D. NUMBERS WHILE IN TEST MODE
THERE IS NO MORE ROOM ON THE DISK FOR NEW I.D. NUMBERS
(NEVER HAPPEN)
DEFINITIONS MUST INCLUDE AT LEAST 1 ITEM NUMBER
TO REVIEW YOUR DEFINITIONS, STATE ITEM NUMBERS BUT NO VALUES
CLEAR EITHER ENTIRE MATRIX OR JUST ONE ITEM AT A TIME
(NEVER HAPPEN)
THERE ARE NO DATA IN THE SPECIFIED CASE FILE
SUMMARY REQUEST NEEDS RESTATEMENT OF DEFINED ITEMS
ACCOUNT CONTAINS NO DATA FOR ITEMS SPECIFIED
SPECIFY THE I.D. NUMBER YOU WANT RESTORED
UNABLE TO RESTORE THE SPECIFIED I.D.
YOU MAY NOT DELETE OR RESTORE WHILE IN TEST MODE
NO DATA ARE IN SUMMARY MATRIX. DEFINE YOUR SUMMARY
YOU MUST BE IN TEST MODE TO USE ALTERNATE DATA BASE
```

PAGE 2 PROGRAM TO CREATE SIMBAD ERROR MESSAGE FILE ON DISK

INVALID ALTERNATE DATE I.D.

PAGE 1 SUBROUTINE TO LOAD SPECIFIED CASE ITEM

```
// JOB
// FOR
*      TRANSFER TRACE
*      ARITHMETIC TRACE
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
**S I M B A D   'L O A D'   S U B R O U T I N E
      SUBROUTINE LOAD(ID,LSB,LINE,NCACC,NCID,IMOD,ITEM,     NLI,NFPO,IERR
     *)
      DIMENSION LSB(20, 2), ITEM(80, 2)

      IERR = 0

      IF (ID - LSB(NCID, LINE)) 2, 10, 2
2     IF (LSB(IMOD, LINE) -2) 6, 4, 6
4     NREC = LSB(NCID , LINE) - NFPO +1
      WRITE (102'NREC) (ITEM(I,LINE), I = 1, NLI)
      LSB(IMOD, LINE) = 1

6     NREC = ID - NFPO +1
      READ (102'NREC) (ITEM(I, LINE), I=1,NLI)
      ITEMP = ITEM(1, LINE)
      IF (ITEMP) 8, 12, 8
8     IF (ITEMP - LSB(NCACC, LINE))14, 9, 14
9     LSB(NCID, LINE) = ID
10    RETURN
12    IERR = 1
13    LSB(NCID, LINE) = 0
      RETURN
14    IERR = 2
      GO TO 13
      END
// DUP
*DELETE              LOAD
*STORE       WS  UA  LOAD
```

```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS

C       SIMBAD SUBROUTINE CALLED WHEN ERROR CODE IS RETURNED
C       BY TCU

        SUBROUTINE TCUER(M0,M1,L)
        COMMON IOTCU(200)
        NEG=-1
        NL=21*256

C       IGNORE OPEN ERROR
        IF(M0-1)7,41,42

C       WRITE ERROR

 41     GO TO (56,51,53,56,56),M1

C       READ ERROR

 42     GO TO (56,51,52,54,55),M1

C       DISCONNECT

 51     M0=-1
        WRITE (1,109) L
C       CHANGE THE ABOVE LATER FOR IMPLIED LOG OUT
 109    FORMAT(/'LINE',I2,' CLOSED'/)
        GO TO 7

C       USER BREAK WHILE READING

 52     WRITE (NEG,104) NL,NL,NL
 104    FORMAT(A1,'STATEMENT CANCELLED',A1,'TRY AGAIN',A1)
        GO TO 7

C       USER BREAK WHILE WRITING

 53     WRITE(NEG,105) NL,NL,NL
 105    FORMAT(A1,'TRY AGAIN',A1,A1)
        GO TO 7

C       LENGTH ERROR

 54     WRITE (NEG,106)NL,NL,NL
 106    FORMAT(A1,'STATEMENT TOO LONG',A1,'TRY AGAIN',A1)
        GO TO 7

C       PARITY ERROR

 55     WRITE (NEG,107) NL,NL,NL
 107    FORMAT(A1,'MACHINE ERROR',A1,'PLEASE RE-TYPE',A1)
```



```
        GO TO 7

C       SYSTEM ERROR

 56     IF(M0-1)57,58,59

 57     WRITE(1,888)
 888    FORMAT(//'SOMETHING TERRIBLE JUST HAPPENED'//)
        CALL EXIT
 58     WRITE (1,108) NL,M1,L,NL
 108    FORMAT (A1,'PROG ERR CODE',I2,' ON LINE ',I2,A1)
        GO TO 7
 59     WRITE (1,110) NL,M1,L,NL
 110    FORMAT(A1,'PROG ER CODE ',I2,' WHILE READING LINE ',I2,A1)
 7      RETURN
        END
```

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM

C      SUBROUTINE TO PROCESS INPUT STRING FOR LINE L
C      AND BUILD TWO OUTPUT ARRAYS.
C      THE INPUT ARRAYS ARE LA (LEXICON ARRAY) AND LCA (THE
C      CORRESPONDING LEXICON CODE ARRAY).
C      WHEN A WORD IN THE INPUT STRING, AS GIVEN IN LA, IS
C      FOUND IN THE LEXICON, ITS CODE IS PLACED IN LCOA (LEXICON
C      CODE OUTPUT ARRAY.. ANY INTEGERS FOUND IN LA
C      ARE PLACED IN IOA (INTEGER OUTPUT ARRAY)

      SUBROUTINE SCAN (LIST,NCT,LA,LCA,LCOA,IOA,LOA)
      DIMENSION LA(50), LCA(50), LCOA(10), IOA(20), LOA(30), LIST(181)
      DIMENSION NTEMP(6), ICHAR(40)
C---- 40 CHARACTER 'ICHAR' ARRAY FOR A3 CONVERSION ----------------------
      DATA ICHAR /
     --14016,-11968,-11712,-11456,-10944,-11200,-14784,-15040,-7360,
     --6848,-16064,-10688,-7616,-7104,-10432,-9920,-6592,-14272,
     --15296,-15552,-15808,-14528,-10176,-6336,-6080,-5824,16448,-4032,
     --3776,-3520,-3264,-3008,-2752,-2496,-2240,-1984,-1728,19264,
     -27456,20032/
C-------------------------------------------------------------------------

      IOA(1) = 0
      LOA(1) = 0
      IIOA = 2
      ILOA = 2
      IF = 1
      IB = 1
C     ------------------------------------------------
  500 IF (LIST (IB)) 5011,5011,501
 5011 IF (LIST(IB) + 4032) 502,503,503
  501 IB = IB + 1
      IF (NCT-IB) 600,600,500
  502 IF(LIST(IB)+16064)5021,5022,5022
 5021 LIST(IB) = LIST(IB)+16384
 5022 MODE = 1
      GO TO 504
  503 MODE = 2
  504 IF = IB
  505 IB = IB + 1
      IF (NCT-IB+1) 506, 506, 507
  506 ISW1 = 1
      GO TO (508,509),MODE
  508 NTEMP(2) = 16448
      NTEMP(3) = 16448
      J = IB - IF
      DO 5081 I = 1,J
      K=IF+I-1
 5081 NTEMP(I) = LIST(K)
      CALL A1A3 (NTEMP, 1, 3, LOA, ILOA, ICHAR)
```



```
      LOA(1) = LOA(1) + 1
      ILOA = ILOA + 1
      GO TO (501,503),ISW1
  509 J = IB - IF
      DO 5091 I=1,J
      K=IF+I-1
 5091 NTEMP(I)=LIST(K)
      CALL MASI (NTEMP,1,J,I)
       IOA(IIOA)=I
      IOA(1) = IOA(1) + 1
      IIOA = IIOA + 1
      GO TO (500,502),ISW1
  507 IF (LIST(IB)) 5071,5071,506
 5071 IF (LIST(IB) + 4032) 510,511,511
  511 GO TO (512,505),MODE
  512 ISW1 = 2
      GO TO 508
  510 IF(LIST(IB)+16064)5101,5102,5102
 5101 LIST(IB) = LIST(IB)+16384
 5102 GO TO (514,513),MODE
  513 ISW1 = 2
      GO TO 509
  514 IF (IB - IF - 2) 505,515,505
  515 DO 5151 I=1,3
      K=IF+I-1
 5151 NTEMP(I)=LIST(K)
      CALL A1A3 (NTEMP, 1, 3, LOA, ILOA, ICHAR)
      LOA(1) = LOA(1) + 1
      ILOA=ILOA + 1
  516 IB = IB + 1
      IF (NCT-IB+1) 600, 600, 517
  517 IF(LIST(IB))5171,501,501
 5171 IF(LIST(IB) + 4032) 516,503,503
  600 LCOA(1) = 0
      K = 2
      I = 2
       GO TO 6041
  601 J = 2
  602 IF (LOA(I)-LA(J)) 606,603,606
  603 LCOA(K) =LCA(J)
      LCOA(1) = LCOA(1) + 1
      K = K + 1
  604 I = I + 1
 6041 IF(LOA(1)-I+2)601,605,601
  605 RETURN
  606 J = J + 1
      IF (J - LCA(1) - 2) 602,604,602
      END
// DUP
*DELETE              SCAN
*STORE      WS  UA   SCAN
```

PAGE 1 SUBROUTINE TO STORE LINT OF TEXT FOR DELAYED PRINT

```
// JOB
// FOR
**    S I M B A D   D E L A Y   S U B R O U T I N E
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM

      SUBROUTINE DELAY (M, MO, M1, M2, LINE, LIST, LSB, NRC, LENG)
      DIMENSION M(125, 2), LIST(181), LSB(20,2)

      NEG = -1
      M(M2, LINE) = -1
      CALL TCU (3, LINE, M(M1, LINE))
      M(M2, LINE) = LENG
      READ (NEG, 1) LIST
      CALL PACK (LIST, 1, 120, LIST, 121)
      ITEMP = LINE+200
      I = LSB(NRC, LINE)
      WRITE (ITEMP'I) M(M2, LINE), (LIST(N), N = 121, 181)
      LSB(NRC, LINE) = LSB(NRC, LINE) +1
      M(MO, LINE) = 5
      RETURN
1     FORMAT (120A1)
      END
// DUP
*DELETE              DELAY
*STORE      WS  UA  DELAY
```


PAGE 1 PROGRAM TO CREATE SIMBAD LEXICON FILES

```
// JOB    0003 0008
// FOR
*IOCS (DISK,1403 PRINTER, CARD,TYPEWRITER)
*ONE WORD INTEGERS
      DIMENSION LCA(50), LA(50), IN(9), ICHAR(40)
      DEFINE FILE 100(2,50,U,I)
       DATA LA /50*0/
       DATA LCA /50*0/

C---- 40 CHARACTER 'ICHAR' ARRAY FOR A3 CONVERSION ----------------------
      DATA ICHAR /
     --14016,-11968,-11712,-11456,-10944,-11200,-14784,-15040,-7360,
     --6848,-16064,-10688,-7616,-7104,-10432,-9920,-6592,-14272,
     --15296,-15552,-15808,-14528,-10176,-6336,-6080,-5824,16448,-4032,
     --3776,-3520,-3264,-3008,-2752,-2496,-2240,-1984,-1728,19264,
     -27456,20032/
C-------------------------------------------------------------------------
      I = 2
      WRITE (5,777)
 777   FORMAT(1H1,'LISTING OF SIMBAD LEXICON AND CODES')
 1    READ(2,100) (IN(J),J=1,9),LCA(I)
      IF(LCA(I)-9999) 2,3,2
 2    CALL A1A3 (IN,1,3,LA,I,ICHAR)
      WRITE (5,102) (IN(J),J=1,9),LCA(I)
      I = I+1
      GO TO 1
    3 LA(1) = I -1
      LCA(1) = I-1
      I = 1
      WRITE (100'1) LA
      WRITE (100'2) LCA
      WRITE(5,555)
 555  FORMAT (1H1)
 100  FORMAT (9A1,I4)
 102  FORMAT (1HO,9A1,2X,I4)
      CALL EXIT
      END
// XEQ          1
*FILES(100,LEX,0008)
MESSAGE  0001
NEW      0002
N        0002
TEST     0003
DELETE   0004
D        0004
REMOVE   0004
END      0005
OFF      0005
DONE     0005
FINISHED 0005
SUMMARY  0006
ACCOUNT  0006
STATUS   0007
NPR      0008
```

PAGE 2 PROGRAM TO CREATE SIMBAD LEXICON FILES

```
RECOVER   0009
RESTORE   0009
DEFINE    0013
CLEAR     0014
FILE      0015
MAIL      0100
PO        0101
PROBATION0101
ALT       0102
EXC       0103
          9999
```


PAGE 1 LABEL MAINT. PROGRAM AND LABELS

```
// JOB
// FOR
**SIMBAD PROGRAM TO CREATE, UPDATE AND PRINT LABEL MATRIX
*     ONE WORD INTEGERS
*     LIST SOURCE PROGRAM
*     IOCS(DISK,CARD,TYPEWRITER,1403 PRINTER)
      DIMENSION NAR(79,11,3), IN(18), INP(6), LINE(120), ICHAR(40)
      DEFINE FILE 1(36, 79, U, N)
C     ------------------------------------------------------------------

      DATA ICHAR /
     --14016,-11968,-11712,-11456,-10944,-11200,-14784,-15040,-7360,
     --6848,-16064,-10688,-7616,-7104,-10432,-9920,-6592,-14272,
     --15296,-15552,-15808,-14528,-10176,-6336,-6080,-5824,16448,-4032,
     --3776,-3520,-3264,-3008,-2752,-2496,-2240,-1984,-1728,19264,
     -24640,20032/
      DATA IN/18*16448/

C     ------------------------------------------------------------------

      WRITE (5, 901)
      WRITE (1,969)
969   FORMAT(///'DATA SWITCH 1'/'UP FOR NEW'/'DOWN TO UPDATE'/)
      PAUSE 1
      IMAX = 79
      JMAX = 11
      LMAX = 3
      LL = 18
      L = 0
      LFILE = 1
      NBL = 16448
      MMM = 19264
      CALL DATSW (1,I)
      GO TO (4, 2), I
2     WRITE (1, 92)
92    FORMAT (/'UPDATING LABEL MATRIX'/)
      N = 1
      DO 3 J = 1, JMAX
      DO 3 K = 1, LMAX
3     READ (LFILE'N) (NAR(I, J, K), I = 1, IMAX)
      GO TO 10

4     WRITE (1, 94)
94    FORMAT (/'CREATING NEW LABEL MATRIX'/)
      CALL A1A3 (IN, 1, 9, INP, 1, ICHAR)
      DO 1 I = 1, IMAX
      DO 1 J = 1, JMAX
      DO 1 K = 1, LMAX
1     NAR(I, J, K) = INP(K)

C     ------------------------------------------------------------------

C     READ IN LABEL CARDS AND PLACE IN ARRAY
```

```
10    READ (2, 91) I, J, (IN(K), K = 1, LL)

C     CHANGE AMPERSAND TO PLUS SIGN

      DO 1101 K = 1, LL
      IF (IN(K)-20544) 1101, 11, 1101
11    IN(K) = 20032
1101  CONTINUE

C     RIGHT JUSTIFY 9 CHARACTER LABEL FIELD IF J IS NOT 1

      K = (LL/2)
      IF (J-1) 1200, 1200, 1104

1104  IF (IN(K) - 16448) 1200, 1106, 1200
1106  K = K-1
      IF (IN(K) - 16448) 1110, 1108, 1110

1108  IF (K-1) 1200, 1200, 1106
1110  KK = LL/2
1112  IN(KK) = IN(K)
      IN(K) = 16448
      KK = KK-1
      K = K-1
      IF (K) 1200, 1200, 1112
1200  CONTINUE


91    FORMAT (2I2,18A1)

      IF (J-2) 14, 12, 12
12    J = J+1
14    IF (I) 99, 99, 15
15    IF (J) 99, 99, 20
20    IF (I - IMAX) 25, 25, 99
25    IF(J-JMAX  ) 30, 30, 99
30    CALL A1A3 (IN, 1, LL, INP, 1, ICHAR)

      IF (J-1) 31, 34, 31
31    DO 32 L = 1, LMAX
32    NAR(I, J, L) = INP(L)
      GO TO 10
34    DO 35 L = 1, 3
35    NAR(I, 1, L) = INP(L)
      NAR(I, 2, 1) = INP(4)
      NAR(I, 2, 2) = INP(5)
      NAR(I, 2, 3) = INP(6)
      GO TO 10

C     ------------------------------------------------------------

C     ILLEGAL COORDINATES OR END OF DECK

99    IF(I+J-199)105,500,105
```



```
C     ILLEGAL

105   WRITE (1,9105) I, J, (IN(K), K = 1, LL)
9105  FORMAT (/'BYPASSED ILLEGAL LABEL CARD'2I3,9A1/)
      GO TO 10

C     ------------------------------------------------------------

C     PRINT THE LABEL MATRIX

500   CALL FILL(LINE, 1, 120, NBL)
      DO 600 I = 1, IMAX
      DO 550 J = 1, JMAX

      M = (10*J)-6
      DO 502 L = 1, LMAX
502   INP(L) = NAR(I , J, L)
      CALL A3A1 (INP, 1, LMAX, IN, 1, ICHAR)
550   CALL MOVE (IN, 1, LL, LINE, M)
600   WRITE (5, 9600) I, (LINE(K), K = 1, 112), I
9600  FORMAT (1H ,I2,1X,112A1,I3)
901   FORMAT (1H1,45X'LISTING OF LABEL MATRIX')
      WRITE (5, 90)

      N = 1
      DO 950 J = 1, JMAX
      DO 950 K = 1, LMAX
950   WRITE (LFILE'N) (NAR(I, J, K), I = 1, IMAX)
      WRITE (1, 9950)
9950  FORMAT (/'MATRIX WRITTEN ON DISK'/)
      CALL EXIT
90    FORMAT (1H1)
      END

// DUP
*DELETE              LMUP
*STORE      WS  UA  LMUP

// JOB    0003 0008
// XEQ LMUP     1
*FILES(1,LABEL,8)
0101AGE
010211-BELOW
010312
010413
010514
010615
010716-ABOVE
0201SEX
0202MALE
0203FEMALE
0301ETHNIC GROUP
0302MEX-AMER
```

```
0303BLACK
0304OTHERS
0401LIVING ARRANGEMENT
0402NAT PRNTS
0403ONLY MOTH
0404ONLY FATH
0405PAR-STPPR
0406OTHER
0501TOTAL IN HOUSEHOLD
05021-2
05033-5
05046-7
05058+
0601SIBLING POSITION
0602FIRST
0603SECOND
0604THIRD
0605FOURTH
0606FIFTH
0607SIXTH+
0701OTH FAM W RECORD
0702YES
0703NO
0801NO IN FAM W RECORD
0802NONE
08031
08042
08053
08064+
0901FAMILY INCOME
0902TO 2999
09033000-4999
09045000-6999
09057000-8999
09069K-10999
090711K-12999
090813000+
1001HEAD OF HOUSE SEX
1002MALE
1003FEMALE
1101EDUC-HOUSE HEA
11021-3 YRS
11034-7 YRS
11048-11 YRS
110512 YRS
110613-14 YRS
110715-16 YRS
110817+ YRS
1201MOVES LAST 5 YEARS
1202NONE
12031
12042
12053-4
12065
```



```
12076+
1301GRADE IN SCHOOL
13025-BELOW
13036TH
13047TH
13058TH
13069TH
130710-ABOVE
1401PROPER GRADE
1402AHEAD
1403PROPER
14041 YR BACK
14051-2 BACK
14062+ BACK
1501ABSENCE THIS YEAR
15020-2 DAYS
15033-7 DAYS
15048-14 DAYS
150515-24 DYS
150625+ DAYS
1601ABSENCE LAST YEAR
16020-2 DAYS
16033-7 DAYS
16048-14 DAYS
160515-24DYS
160625+ DAYS
1701I.Q.
1702BELOW 80
170380 - 89
170490 - 109
1705110 - 119
1706120+
1801NO. PRIOR OFFENSES
1802NONE
18031
18042
18053
18064-5
18076+
1901NO. JUV. OFFENSES
1902NONE
19031
19042+
2001CUMUL SERIOUSNESS
2002MINOR
20032
20043
20054
20065
2007VRY SERUS
2101AGE AT 1ST OFFENSE
2102UNDER 10
210310 YRS
210411 YRS
```

210512 YRS
210613 YRS
210714 YRS
210815+ YRS
2201COMPANIONS 1ST OFF
2202NONE
22031
22042
22053+
2301TIME 1ST-THIS OFF
23020-8 MO
23039-16 MO
230417-24 MO
230525-33 MO
230634-49 MO
230750+ MO
2401TIME LAST-THIS OFF
24020-4 MO
24035-8 MO
24049-14 MO
240515-21 MO
240622+ MO
2501REFERRAL HISTORY
2502YES
2503NO
2601PLACED OUT OF HOME
2602YES
2603NO
2701EVER A WARD
2702YES
2703NO
2801PRESENTLY A WARD
2802YES
2803NO
2901REFERRAL REASON
2902ROBRY-AS
2903BURG-GTFT
2904PETTY TFT
2905ILLEG SEX
2906NARCOTICS
2907PROB VIOL
2908INCORGBLE
2909TRUANCY
29100THER
3001REFERL SERIOUSNESS
3002MINOR
30032
30043
30054
3006VRY SERUS
3101REFERRED BY
3102POLICE
3103SHERIFF
3104SCHOOL

3105PARENTS
31060THERS
3201COMPANIONS STDYOFF
3202NONE
32031
32042
32053
32064+
3301DETAINED
3302YES
3303NO
3401LEGTH OF DETENTION
34021 DAY
34032 DAYS
34043-5 DAYS
34056-16 DAYS
340617-21 DYS
340722+ DAYS
3501TIME REFERRAL-DISP
35020-2 DAYS
35033-12 DAYS
350413-18 DYS
350519-26 DYS
350627-31 DYS
350732-43 DYS
350844+ DAYS
3601PETITION REQUESTED
3602YES
3603NO
3701PETITION SUBMITTED
3702YES
3703NO
3801DISPOSITION AWARD
3802DISM-INTK
3803ISUP-AGEN
3804FSUP CTD
3805DISM-CORT
3806ISUP-CORT
3807FSUP-CORT
3901DISPOSITION REC
3902DSMS-INTK
3903INFSUP
3904FOR SUP
4001DISPOSITION AWARD
4002DISMISSED
4003INF SUP
4004FOR SUP
4101NATURE OF DISP
4102-POS -TRT
4103-POS +TRT
4104+POS -TRT
4105+POS +TRT
4201DISP. AWARDED-REC.
4202SAME

```
4203DIFFERENT
4301CHARGE PLEA
4302ADMITTED
4303DENIED
4401PLACEMENT AWARDED
4402OUT HOME
4403IN HOME
4501PLACEMENT REC
4502OUT OF HM
4503OWN HOME
4601PLACMNT AWARD-REC.
4602SAME
4603DIFFERENT
4701PROB OFFICER IN CT
47021
47032+
4801AGE OF PROB OFFICR
480222-24 YRS
480325-26 YRS
480427-29 YRS
480530-33 YRS
480634-37 YRS
480738-42 YRS
480842+
4901SEX OF PROB OFFICR
4902MALE
4903FEMALE
5001MARITAL STATUS-PO
5002SINGL-WID
5003MARR ONCE
5004MARR 2+
5005SEP-DIVOR
5101NO CHLD OF PO
5102NONE
51031
51042
51053
51064+
5201EXPERIENCE OF PO
52020-1 YEAR
52032 YEARS
52043 YEARS
52054 YEARS
52065-6 YEARS
52077+ YEARS
5301CONTACTS MADE
5302YES
5303NO
5401NUMBER OF CONTACTS
54021-5
54036-11
540412-21
540522+
5501TIME DISP-1ST CONT
```



```
5502PRIOR-9DY
550310-31 DAYS
550432+ DAYS
5601AVERAGE LENGTH CONT
56021-15 MINS
560316-20 MIN
560421-25 MIN
560526+ MINS
5701CTTS OUTSD WK HRS
5702YES
5703NO
5801PROPN. CONT IN HOME
5802NONE
5803TO HALF
5804OVER HALF
5901EMPH ON OBTAIN INF
5902NONE
5903LOW
5904AVERAGE
5905HIGH
6001EMPH ON VIEW SELF
6002NONE
6003LOW
6004AVERAGE
6005HIGH
6101EMPH ON APRROVAL
6102NONE
6103LOW
6104AVERAGE
6105HIGH
6106VERY HIGH
6201EMPH ON DISAPROVAL
6202NONE
6203LOW
6204AVERAGE
6205HIGH
6301PROBABILITY
6302
9999
```

D. THE SIMBAD REFERENCE MANUAL

Attached to this report is the first version of a reference manual which has been designed to assist those who are operating the SIMBAD system, particularly beginners, in the use of the computer.

It describes in detail, and with parallel illustration, both the various operations that may be performed by the user and the computer's response to them.

Within the limitations of the designated variables and values, which are listed in Section XII of the Manual and which have been selected on the dual basis of essential information about delinquent minors and of evidenced predictive utility to the decision-making process, the system offers a wide capability in performance of the following main functions:

1. Storing and Retrieving Case File Data

The user is shown:

How to set up contact with the computer.

How to establish a new case file for a minor.

How to add or change data in an existing case file.

How to obtain a view of all data stored in a case file.

How to obtain a view of specific data stored in a case file.

How to delete a case file from the records.

2. Obtaining Summaries and Lists

The user is shown:

How to obtain the total number of all the department's minors whose cases are on file.

How to obtain a list of all the case numbers of these minors.

How to obtain the number of all minors falling within a certain category specified by the user.

How to obtain a list of all the relevant case numbers of minors falling within the above specified category.

Since there are a total of 62 variables with a component 256 values, any of which may be used, in any combination, for the purpose of defining the category specified, it will readily be appreciated that the system allows an extremely wide range of summarizing capability.

3. Asking questions

The user is shown:

How to ask a question about the likelihood that an actual delinquent minor will recidivate, or that his behavior will get better, remain the same, or get worse, based on the information on file about him.

How to ask the above questions about an actual case, based on additional, hypothetical assumptions made about him, e.g., "What is the probability of this minor's recidivism if he were not to live with his mother only?"

How to ask the above questions about a hypothetical minor, based on the data you choose to feed into the computer about him.

How to ask the above questions about a minor, either real or hypothetical, in the light of various dispositional and placement alternatives, e.g., "What is the probability that this minor's behavior will or will not improve if he is dismissed, or is placed under informal supervision, or is placed under formal supervision?"

As with summaries, it will be evident that such questions, all of which may be based on any of the 62 variables with their 256 values, used in any combinations desired, allow of great flexibility and variety of output distribution. Their value as a potential means of assistance to the decision-making process will be obvious.

The SIMBAD system also offers a mailing and a message service. Other features, such as those which are built in to insure protection of data, including the use of the Test Mode of operation which allows the user freedom of operation without risk of changing data on file, are described in the Introduction to the Manual.

IMPLEMENTATION PLAN FOR SIMBAD

GENERAL PRINCIPLES

As a computer system, SIMBAD is presently in experimental operation at the University of Southern California. That the basic concept underlying SIMBAD is feasible, has already been demonstrated in the laboratory. What is needed now is a test of SIMBAD as an operating system within the actual context of probation work. In this section, a program to implement the SIMBAD system will be outlined, with emphasis placed on evaluating its effectiveness in the day-to-day work of probation.

Implementation will require that several IBM 2741 remote user communication terminals be physically located in probation offices in the Southern California area, allowing these offices to have real time access to the IBM 1130 computer on which SIMBAD is currently operating. It is hoped that the user terminals can be placed in many different probation offices so as to obtain as wide a practical experience as possible with the use of the system.

Two major aspects of SIMBAD need operational evaluation: (1) the extent to which it improves the record-keeping and administrative efficiency of a probation office, and (2) the extent to which it aids the probation office in making decisions about individual youngsters. The first of these can be evaluated in a relatively straightforward fashion. The system is specifically designed to increase the ease and efficiency of record-keeping. Data are readily stored and retrieved in the computer's filing system. The question of how useful this is revolves around whether the increase in efficiency justifies the added cost of the time-shared computer and rental of the user's communication terminal. This judgment can be made by supervisory personnel. One particular aspect of SIMBAD, we feel confident, will be extremely useful to any probation office and will more

than justify the cost: this is the capability of requesting statistical summaries of the current status of a department's case load, broken down into specific categories at the request of the user. For example, a supervisor could request the computer to list the number of cases currently be being handled by each officer under his supervision and request these ca cases to be categorized by sex, age, type of offence, home status of the juvenile offender, etc. Since all probation departments are required to keep fairly extensive statistical records of their operation, this component of SIMBAD provides a means of automating that function. With very little programming modification, monthly or quarterly reports of statistical information could be generated by the computer.

The second major aspect of SIMBAD, that of aid to probation decisions on the disposition and treatment of individual delinquents, will require a more extensive and systematic evaluation since the issues here are complex and much more important. The most pressing need is to evaluate the effectiveness of the prediction model used to reduce the decision uncertainty of the decision-maker, or to be more precise, to help him make decisions about probation alternatives that, hopefully, will have the highest probability of "success." This prediction capability is based on a conditional probability model[1] developed on data that is now about four years old, and which needs to be cross-validated on new data before it real effectiveness can be demonstrated. An extremely important phase of any implementation program would be to test the usefulness of the prediction model by selecting a sample of cases from several probation departments and using the model to predict the various disposition-treatment alternatives those cases theoretically should be given. Predictions would therefore be based on the variables that the model presently considers to be useful. The predictions could then be compared with what actually happened, i.e., with the disposition-treatment categories to which the cases were actually assigned.

[1]The technical details of the juvenile probation research project and the SIMBAD system are given in: McEachern, Taylor, and Newman, American Behavioral Scientist, XI, 3, Jan.-Feb., 1968 and McEachern and Newman, Journal of Research on Crime and Delinquency, June-July 1969 (in press).

The implementation of SIMBAD should also enable a determination of what is the most relevant information for probation decisions. In developing SIMBAD, we made the plausible assumption that the decision-maker (probation officer and/or juvenile court judge) uses all available relevant information in coming to his decision. As a step in this direction, and as a result of an extensive analysis of what variables might be useful in predicting future delinquent behavior, several variables have been grouped into information categories (cf. Section XII). However, these are considered only prototypical and may not prove to be the most useful in operational context. We take the viewpoint that the question of what information is most relevant is up to the judgment of experienced probation officers and/or juvenile court judges. Furthermore, if such officers are provided with a computerized system such as SIMBAD, which allows easy collection, processing, and retrieval of information, then proper research studies, conducted in the actual day-to-day work of probation, should soon yield an answer to the question of what is the most relevant information.

Finally, we should mention that the operational implementation of SIMBAD should also enable a critical evaluation and possible modification of the criteria used in developing the prediction model for aiding probation decision-making. One assumption underlying the development of the prediction model is that, in making decisions, the probation officer is vitally concerned with the consequences to the youngster of the various actions taken with respect to him. Two criteria were assumed to influence the decision-maker, and both of these concerned the youngster's future behavior: (1) the number of offenses subsequent to the probation decision (recidivism), and (2) a rating on a behavior "improvement-deterioration" scale which essentially described the youngster's behavior after probation disposition as "getting worse," "staying the same," or "improving." It was further assumed that the decision-maker chooses that disposition-treatment alternative which, other things being equal, increases the probability that the youngster will improve or that he will keep out of trouble, i.e., reduces the

probability of recidivism. However, it is not always true or obvious that the best decision alternative is the one with the "highest probability of success." Other criteria may well be important such as the protection of society. We may need to weigh the balance and select, for example, that treatment which minimizes loss to society. This injects into the decision situation the important notion of utility or subjective value of the consequences of a decision. It is entirely feasible for probation officers, in consultation with juvenile court judges and/or parents, to make such value judgments and thus introduce more relevant criteria into the situation.

SPECIFIC PLANS

At the present time, SIMBAD is operating with two IBM 2741 remote user communication terminals connected to the IBM 1130 computer. There is the capability of adding two more terminals for a total of four. In order to introduce SIMBAD to probation departments, the following steps will be taken:

(1) A probation department will be selected for participation in SIMBAD trial implementation from each of four different counties in Southern California. Appropriate supervisory personnel from these departments will be trained in the operation of SIMBAD, using the terminals presently located at the Public Systems Research Institute of the University of Southern California. With the use of the SIMBAD Reference Manual, this initial training can be accomplished in a day or two. However, since not all probation offices operate in the same manner, this training period will also be used to adapt and revise the reference manual so that it is appropriate for each participating department. During this training period, the supervisory personnel will also, in consultation with SIMBAD personnel, specify how they would like to have the SIMBAD system integrated into their normal daily activities. Questions that need to be resolved are: the number and type of probation officers who should have access to the SIMBAD system; and

the extent that it should be used in processing current case loads. It is our present opinion that as many experienced probation officers as possible should have access to the system; but this may not be feasible in an operational context. Also at this time, an evaluation and rating form will be developed to enable an on-going evaluative review of the usefulness of the SIMBAD system to probation officers.

(2) A terminal will be located in each participating probation department, and the supervisory personnel, along with SIMBAD personnel, will introduce the system to probation officers and train them in its use. Each terminal has access to the IBM 1130, and when SIMBAD is in operation, all the functions described in the reference manual will be available to the user. Log records will be established to gather information on such matters as use time, number of questions asked, number of statistical summaries requested, delay in response time, etc. These will serve to provide estimates of the actual operating effectiveness of the user terminals.

(3) A systematic evaluation of SIMBAD as an aid to probation administration and probation decision-making will be conducted. As mentioned previously, it is the latter of these functions that is crucial. The two basic problems are: (a) cross validation of the prediction model, which will also entail updating the data files and, (b) evaluating and improving, if necessary, the criteria of the "effects of probation."

Cross Validation

Cross validation of the prediction model will be carried out in two ways:

(a) Each department will select from its files a sample of cases that can be described as closely as possible by means of the information variables describing the present SIMBAD sample of prior cases

(Section XII). Each case, of course, can also be described according to what disposition-treatment was given to that case. The SIMBAD prediction model can be used to assign the probabilities of "success" on the criterion, given the background information known about the case and the disposition-treatment alternative. A comparison of these will enable an evaluation of how well the model agrees with those actions actually taken on a sample of cases from each department.

(b) Each department will also sample, from its present case load and new referrals, two comparable sub-samples, one of which will be processed through using the SIMBAD system as an aid, and the other processed in the conventional fashion. Detailed follow-up records will be planned for these cases, and a periodic check, perhaps monthly if this proves feasible, will be made on the progress of each case. This will permit a more direct test of the efficiency of the SIMBAD model, since its predictions are made on the criteria of recidivism and the "behavior improvement-deterioration" scale. It is thus necessary to gather information on just how well individual youngsters fare when being aided, or not aided, by the SIMBAD system. It is assumed that SIMBAD will improve the efficiency of probation decision-making; and the gathering of these data will enable an empirical check on this assumption.

During this cross validation phase, it is also intended to revise the information variables currently being used by SIMBAD. At this time, it appears that the best way to do this is to form a panel of experienced probation officers in each department and have them go over the complete list of variables, deleting those that are not considered relevant. At the same time, those variables considered important and relevant can be added to the list. Procedures can then be set up for incorporating these new

variables into the prediction model, which will require obtaining frequency data on the variables relevant to the criteria, and/or having experts estimate the probabilities of each new information variable given each criterion category, using methods developed by Professor Ward Edwards and his students at the University of Michigan.* The procedures will enable a revision of the data base of SIMBAD and, hopefully, make it more relevant to probation departments.

Criterion Evaluation

In conjunction with the program mentioned above, we feel there is a strong need to take a critical look at the criterion presently being used to measure the effects of probation.

One of the motives behind the development of SIMBAD was to demonstrate that the modern theory of human decision-making, if coupled with computer technology, could provide a means for improving probation decision-making. It may seem that such decisions are beyond the scope of a formal decision analysis, since they depend heavily on the intuition and the subjective judgment of the decision-maker. However, the contemporary theory of decision recognizes that almost all important decision problems require human judgment, and proposes that such judgments be directly introduced into the formal analysis of a decision problem. In particular, it is proposed that a panel consisting of expert probation officers, juvenile court judges and law enforcement experts, might be able to make explicit value judgments (utility) about the consequences to the juvenile and/or to society, of various probation-decision alternatives. These value judgments could be used to augment the traditional criteria of recidivism and of the behavior "improvement-deterioration" scale presently being used by SIMBAD.

*Edwards, W., Nonconservative Probation Information Processing Systems. Report ESD-TR-66-404, Dec. 1966 (b). (Engineering Psychology Laboratory, Institute of Science and Technology, University of Michigan, Ann Arbor.)

(4) As a final step, which can be initiated after several months of operational experience with the use of SIMBAD, a critical review of the actual computer technology which has been developed for SIMBAD will be made. We do not expect that the current programming and/or hardward configuration of SIMBAD is the "best" for actual operational use. Many limitations of the system will be revealed with experience, and we fully expect that these can be corrected by the use of more sophisticated programming techniques and by the use of more flexible user terminals. Moreover, the IBM 1130 machine is not designed for very extensive time-sharing, and if SIMBAD proves to be operationally feasible, it will be necessary to investigate the possibility of expanding the system for ultimate use in all probation offices.

REFERENCES

Edwards, W., Nor.conservative Probation Information Processing Systems. Report ESD-TR-66-404, Dec. 1966 (b). (Engineering Psychology Laboratory, Institute of Science and Technology, University of Michigan, Ann Arbor).

Hill, W.M., Stoller, F.H., and Straub, Constance I. Group Therapy for Social Impact: Innovation in Leadership Training. American Behavioral Scientist, XI (1), Sept.-Oct., 1967.

McEachern, A.W. (ed.) Views of Authority: Probationers & Probation Officers. Los Angeles: Youth Studies Center, University of California, 1961.

McEachern and Newman, J.R., Journal of Research on Crime and Delinquency, June-July, 1969 (in press).

McEachern, A.W., and Taylor, E.M., Positional Authority and Delinquent Behavior. Los Angeles: Youth Studies Center, University of Southern California, 1966.

McEachern, A W., Taylor, E.M., and Newman, J.R., American Behavioral Scientist, XI, 3, Jan.-Feb., 1968.

Wilkins, L.T., Information and Decisions Regarding Offenders. Paper presented at the National Symposium on Law Enforcement Science and Technology, Chicago, 1967.

SIMBAD

REFERENCE MANUAL

A Guide for Agencies Participating in the SIMBAD Computer System to Aid Probation Departments in Data Analysis and in Decision-making.

by

Joan Hounsfield

The work preceding and included in the compilation of this manual has been supported in part by Public Health Service Grant MH-06597 from the National Institute of Mental Health; Grants 168 and 359 from the Office of Law Enforcement Assistance; and grants from the Ford Foundation.

Public Systems Research Institute

University of Southern California

1969

FOREWORD

The SIMBAD system was developed to enable experimentation in the use of a computer as a guide to probation decision-making. It is a device for the convenient storage and speedy retrieval of large volumes of data about delinquent minors. It is also, we hope, a useful predictive tool, capable, through comparison of a single case with a great many others, of indicating what is likely to happen in such a case on the basis of what has already happened to others like him in the past.

This application of the empirical approach to decision-making has never before been attempted in the field of probation. SIMBAD, therefore, like all prototypes, must be put to trial before either its limitations or its usefulness can be assessed. We rely on user evaluation to correct any shortcomings and to structure its most serviceable form.

SIMBAD has been conceived as a supplement to, not as a usurper of human wisdom. While we do not believe that dealing with people can ever be reduced to a rigid science, those who must make decisions about the lives of other human beings must make use of any assistance that research and past experiences can lend them. As a tool for probation workers, computerized statistical analysis can be useful with those cases about which the least is known: it can also be useful with those cases which, because of the size of a worker's caseload, must perforce be handled in routine fashion.

The data on which the calculations for the mathematical model were based were obtained through a major research study into the juvenile probation system which is described in some detail in the American Behavioral Scientist, Volume XI, Number 3, January-February, 1968. For this study, data were collected on 2,290 juveniles referred to eight participating Southern California probation departments in October and November 1963. The basic data collected consisted of complete background information on all juveniles referred to these probation departments for delinquent acts over a two-month period, including personal characteristics, delinquent history, school experience,

socio-economic status, family history and structure, reasons for referral, detention history, the court process and initial disposition and placement. All cases were followed for a period of one year, during which time records were kept of any subsequent referral and of any treatment contacts. Information on the probation officers' characteristics, official positions and caseloads was also obtained.

It is from these data that the variables and values appearing in Section XII of the Manual were eventually selected and from which has been developed the mathematical model of the probation process which is the SIMBAD system. The model is based on certain aspects of the rapidly developing field of Bayesian statistics and in particular on the use of Bayes' theorem as a basis of making classification and treatment decisions about individuals.

As the system comes to be used by participating probation departments, it is hoped that the new data constantly being fed into the system will enable it to reflect a true process of change through automatic updating, incrementation and evaluation.

A. W. McEachern
Director
Public Systems Research Institute

ACKNOWLEDGMENTS

This Manual and the operations it sets out to describe represent the completion of a significant phase in the development of an experimental model which was blueprinted for production four years ago.

In seeking a means whereby new technology might be applied to the practice of probation, the Public Systems Research Institute (then the Youth Studies Center) of the University of Southern California set out to develop mathematical models of the probation process and to build them into a computerized system which could be used by probation departments for data analysis and decision-making. Construction of this pilot model, known as SIMBAD, is now complete.

This achievement marks an end but also a beginning. A mechanical "means" has been found, according to plan; the "end" will hopefully be reached after it is set into motion, tested and adjusted to user requirements.

It is with gratitude that we acknowledge the assistance of those institutions that have furnished financial support of both the research on which SIMBAD is based and the construction of the model: namely, the National Institute of Mental Health, the Office of Law Enforcement Assistance and the Ford Foundation. The research findings obtained as a result of this assistance have a far wider significance than can possibly be visible from this publication.

Acknowledgment is also due to all the staff of P.S.R.I. who have worked on the SIMBAD project. The compiler of this Manual is personally grateful for the assistance received from many of them and for the freedom to draw on material previously published in reports of the SIMBAD project:

Alexander W. McEachern Director, P.S.R.I.	Program Director
Edward M. Taylor Research Associate	Co-Director
J. Robert Newman	Statistical-Mathematical Consultant
Joe E. Keefe	Senior Programmer and System Analyst
Frederick F. Sinatra	Senior Programmer and System Analyst
Ann Strong	Programmer
Richard W. Hixson	Computer Programmer-Operator
Elizabeth Hughes	Assistant Programmer
Ann E. Ashford	Research Assistant
Sandra Bell	Research Assistant
Jean E. Braun	Research Assistant
Sridaran Sivasubramanian	Research Assistant
Belinda M. Smith	Research Assistant
Diane A. Capito	Project Secretary
Edna Huffsteter	Project Secretary
Jewel E. Henderson	Secretary
Alcester S. Peers	Secretary

TABLE OF CONTENTS

SIMBAD

"SIMULATION AS A BASIS FOR SOCIAL AGENTS' DECISIONS"

INTRODUCTION

I WHAT IS SIMBAD?

SIMBAD is an experimental operating computer system devised to introduce new knowledge and technology into the practice of probation by providing participating probation departments with a tool to aid them in data analysis and in decision making.

Each probation department participating in the SIMBAD program has installed on its own premises a remote use communication terminal linked to a digital computer facility based at the University of Southern California, Public Systems Research Institute. By communication with the computer, the department is able to:

(1) Store and retrieve information such as would be in a minor's case file, i.e., data relative to his person, his environment, his history within the judicial process, his contact with probation authorities, etc.;

(2) Ask questions about the probability of improvement of a minor's behavior or his potential recidivism, particularly under various dispositional and placement alternatives;

(3) Obtain statistical summaries of the current status of a department's case load, broken down into specified categories.

II THE SIMBAD PROGRAM

The word SIMBAD stands for "Simulation as a Basis for Social Agents' Decisions." The "knowledge" which has been fed into the computer is a condensed summary of past experiences gained at all points in the probation process, and the system may therefore be seen as a simulation of the probation process itself.

Initially, the computer has been programmed with a large body of data obtained through major research into juvenile probation undertaken by the University of Southern California. Based on what has happened in the past, the computer is able to furnish probability estimates of the success of any future disposition and treatment decisions that may have to be made at any points in the probation process. As the program develops, with participating departments constantly feeding new data into the computer as they

make use of it, the data bank on which the computer operates will constantly be updated and incremented. The program thus allows the user to base his decisions on the best available empirical knowledge, while leaving him free to exploit his own experience in relation to the idiosyncrasies of an individual case.

III THE COMPUTER AND ITS TERMINALS

SIMBAD is operated on an IBM 1130 digital computer located in the Public Systems Research Institute of the University of Southern California. This computer may be linked to any number of IBM 2741 remote use communication terminals, each based at a participating probation department which will thereby gain real-time access to the computer. These terminals, the main input and output devices of the system, consist of typewriters linked to the computer through the telephone lines.

IV SECURITY FEATURES OF THE SYSTEM

1. To gain access to the computer, the user must first identify himself by typing his department's Account Number. This insures protection of data, since no other department can obtain a response to this Account Number.
2. The user may then type a number of statements, questions or commands, causing the computer to respond to each with a teletyped reply acknowledging receipt of the information or furnishing a response to the question or command. The response affords the user a means of visibly checking the accuracy of his operations as he performs them. Moreover, if he makes an error, the computer will draw his attention to it by typing a gentle remonstrance such as is illustrated in Section XI.
3. The computer allots an identification number to each case in the user's Account. These I.D.'s cannot be used by any other participating agency.
4. A further security feature, of particular interest to new users, is the Test Mode of operation described in Section C, under which they are able to experiment without any risk of permanently destroying or changing data on file.
5. All information received by the computer is automatically stored in its filing system unless the user instructs otherwise. The user is therefore at all times in complete control of what case information he will keep stored in the computer.

V CLASSIFICATION AND SPECIFICATION OF DATA USED IN THE PROGRAM

Each piece of information a user wishes to store or receive about a minor is capable of definition by the combination of two numbers which may be found in the List of Variables and Values in Section XII following:

1. The Item Number	Denotes the class or category of the information. Items are variables which have been selected on the basis of (a) essential information about minors, and (b) evidenced predictive utility to the decision making process.
2. The Value Number	Denotes the specific "compartment" of the above category within which the information falls. Values are mutually exclusive, and their total is exhaustive of the category containing them.

The computer operates on these two numbers and on two further numbers: the department's Account Number and the minor's Identification Number. It also operates on certain command key-words and letters that constitute the computer's "language."

VI THE COMPUTER'S LANGUAGE

The language used by the SIMBAD system has been designed in such a way that the user does not need to remember, or have reference to, a complicated code. The few key words from which the computer takes direction have been chosen because they are likely words that an average person typing simple directions might automatically use.

A list of these words follows in Section IX. All are shown here in capitals for easy visibility, but all may be typed in lower case letters. The computer operates on the first three letters of a word only (in the case of two words, NEW and DELete, it will also operate on the initial n or d only).

No syntax is required - the key word triggers action wherever it comes in the sentence. This allows the user to type a statement in whatever order is most natural to him: the computer will scan the statement, sift out the

key words and numbers and discard the rest as irrelevant. For example, if a user, wishing to extablish a new case file, were to type:

Please give me a new case number, or
I have a new minor to report on, or
New, or
n

in each case the computer would operate identically (by issuing a new identity number for a minor).

Similarly, if the user wished to delete a case file from the records, he might type, with identical results:

Delete case 503 please
503 to be deleted
Remove case 503 from the file
d 503

VII THE OPERATING MANUAL

The manual is divided into 7 main operating sections, as shown in the Table of Contents, each section demonstrating operations and computer responses for a different purpose.

At the risk of seeming over-repetitive, operating instructions are described in the simplest language and in minute detail, so that a user consulting the manual at any operational stage should have minimal need to refer to what has gone before.

A beginner, however, will be well advised (a) to practice operations in the approximate order shown (since, in Section E, for example, summaries have been arranged in an order that demonstrates the ease of transition from one summary to another); and (b) where applicable, to practice under Test Mode conditions as described in Section C.

As he practices, he may become aware of certain apparent redundancies (such as appear in the List of Variables at Items 48-51). If so, he is reminded that this manual is merely a trial version and that, before full implementation of the program is possible, it will be necessary to conduct full discussions with users to discover their difficulties and implement their needs. Users are therefore invited to make note of their problems with this experimental model and volunteer suggestions as to how best it can be made to serve them.

VIII DEFINITION OF TERMS USED IN THE MANUAL

1. I.D.) FILE) CASE) — Used synonymously to indicate a delinquent minor or his case file. An I.D. NUMBER is the identification number allotted to this file. The command FILe is also used when asking to see all data stored for a specified I.D. (Section B.6.)

2. ACCOUNT — The total I.D.'s filed by a probation department with the computer. An ACCOUNT NUMBER is the identification number of the participating department.

3. ITEM) VARIABLE) CATEGORY) — Used synonymously to indicate the class of data as set out in Section XII (List of Variables and Values). An ITEM NUMBER is the reference number of the Item.

4. VALUE — A specific within the category or Item. A VALUE NUMBER is the reference number of the Value as set out in Section XII (List of Variables and Values). A zero is never used as a reference number. A zero in a "Value" position denotes absence of information, not variable measurement.

5. STATEMENT — One complete operating unit of information or command fed into the computer. This can consist, according to operations performed, of elements sometimes expressed singly and sometimes in combination. It may consist of a simple key-word or letter (n), or a combination of word(s) and number(s) (DELete 503), or a combination of numbers (Item Number and Value Number(s)), or simply Item Numbers alone, but never Value Numbers alone. Each statement must be made as a separate "line," i.e., the return key must be used after each.

6. UPDATING — A change in the Value of a datum filed in the computer (as in Section B.4.), i.e., an addition to, or change of, the information contained in a minor's file.

7. SUMMARY DEFINITION MATRIX — To understand the operational relevance of this term, Section E. (Summaries) should be consulted.

The following elementary description may be of assistance to beginning users:

STEP a. of a Summary process defines the nature of the desired summary by identifying the elements (Items and Values) to be included in it. This operation sets up in the computer a Summary Definition Matrix, or pattern from which the summary to be displayed is generated.

When the Matrix is clear (before elements have been defined), it can be envisioned as similar to a bookkeeper's 10-columnar ledger, with a column at left (for Item entries) and 10 columns for spread of Value entries, all 10 columns containing zeros.

As the user states his definition, the computer enters the Item at left and places a digit 1 in whichever columns describe the values that have been defined (column 1=Value 1; Column 6=Value 6, etc.). All other columns retain the zero.

STEP b. of the Summary is the Summary Request.
In a Summary by Inclusion (Section E.1.) the computer will now look only at the Values where a 1 is displayed in the Matrix for those Items enumerated in the Summary Request:

In a Summary by Exclusion, (Section E.2.) the computer will do so after reversing the positions of all zeros and 1's for those items enumerated in the Summary Request. After completion of the summary, the zeros and 1's will return to their prior positions.

IX SUMMARY OF KEY WORDS & OPERATIONS

PURPOSE OF OPERATION		OPERATION In capitals=words to be typed (First 3 letters suffice) In lower case=relevant numbers to be typed	ALTERNATE KEYWORDS
A. SETTING UP CONTACT WITH THE COMPUTER Typewriter ON; TALK BUTTON; DIAL your PHONE NO.; DATA BUTTON; HANG UP; RETURN KEY			
B. STORING & RETRIEVING INFORMATION ABOUT A CASE			
1. ESTABLISH A NEW CASE FILE (new I.D.)		NEW	N
2. DELETE A CASE FILE from your Account		DELete i.d.	D REMove
3. RESTORE A CASE FILE erroneously deleted		REStore i.d.	
4. UPDATE A CASE FILE with new information	•	i.d. item value	
5. SEE SPECIFIC DATA in a case file	•	i.d. item(s)	
6. SEE ALL DATA in a case file	•	i.d. FILe	
C. OPERATING IN TEST MODE			
1. BEGIN Test Mode		TESt	
2. END Test Mode		TESt OFF	DONe END FINished
D. ASKING QUESTIONS			
1. ABOUT RECIDIVISM (200 series questions)	•	i.d. 200 or 2item	
2. ABOUT IMPROVEMENT (300 series questions)	•	i.d. 300 or 3item	
E. OBTAINING SUMMARIES OF YOUR ACCOUNT			
1. Step a. DEFINE TERMS of desired summary		DEFine item value(s)	
2. Step b. SEE SUMMARY defined values included	•	SUMmary item(s)	
or SEE SUMMARY defined values excluded	•	SUMmary EXClude item(s)	
3. REVIEW CONTENT of Summary Definition Matrix		DEFine item(s)	
4. CLEAR SUMMARY MATRIX of incompatible data		CLEar	
5. SEE LIST of all I.D.'s in Account	•	ACCount	
6. SEE LIST of classified I.D.'s	•	DEFine item value(s) ACCount	
F. REQUESTING SERVICES			
1. HAVE PRINTED DATA MAILED (operations marked•)		(Add word MAIl to statement)	
2. DISPLAY TYPED MESSAGE at computer center		MESsage (text of message)	
G. ENDING CONTACT WITH THE COMPUTER		OFF	DONe END FINished

X SOME BASIC DIRECTIONS TO THE BEGINNER

1. When typing numbers, use the actual numerals on your typewriter:

   (a) Do not use lower case letter l in place of digit 1.
   (b) Do not use upper case letter O in place of digit 0.

2. When typing words, you may use lower case letters throughout. The capitalization used in the Manual is for easy visibility and to indicate those elements of each word that are essential to computer operation.

3. Always type a space between numbers and between numbers and words: e.g., between an I.D. number and an Item number; an Item number and a Value number; between the command DELete and an I.D., etc.

4. Always depress the return key on completion of every statement.

5. There is no need to repeat an I.D. number, once you have stated it, for subsequent operations you perform about that case. The computer will continue to operate on the I.D. until you either:

   (a) state a different I.D. number, or
   (b) end contact with the computer by "signing off."

6. If you perform an operation incorrectly, the computer will immediately respond by typing a message informing you of your error. A list of such responses is shown in Section XI. Directions for correcting typing errors are shown in paragraphs b. and c. of that Section.

XI IF YOU MAKE AN ERROR

a. "The Computer Talks Back." (Some responses you may receive when operating incorrectly.)

1. ACCOUNT CONTAINS NO DATA FOR ITEMS SPECIFIED
2. DEFINITIONS MUST INCLUDE AT LEAST ONE ITEM NUMBER
3. I CAN'T MAKE ANY SENSE OUT OF YOUR LAST STATEMENT
4. MAXIMUM VALUE FOR ITEM ___ IS ___
5. NO DATA ARE IN SUMMARY MATRIX. DEFINE YOUR SUMMARY
6. NO VALID I.D. SPECIFIED
7. ONLY ONE VALUE NUMBER MAY BE INCLUDED IN THIS STATEMENT
8. SPECIFY THE I.D. NUMBER YOU WANT RESTORED
9. THERE ARE NO DATA IN THE SPECIFIED CASE FILE
10. THIS STATEMENT NEEDS 1 ITEM ONLY AS WELL AS A VALUE
11. TO REVIEW YOUR DEFINITION STATE ITEM NUMBERS BUT NO VALUES
12. UNABLE TO RESTORE THE SPECIFIED I.D.
13. YOU ARE ATTEMPTING TO USE AN I.D. NUMBER NOT ASSIGNED TO YOU
14. YOU MAY NOT DELETE OR RESTORE WHILE IN TEST MODE
15. YOU MAY NOT CREATE NEW I.D.'S WHILE IN TEST MODE
16. YOU MAY NOT DELETE AN I.D. WHILE IN TEST MODE
17. YOU MAY NOT DELETE MORE THAN ONE I.D. NUMBER AT A TIME

b. If you Detect your Error while Typing

Backspace to the exact location of your error and type over it the correct letters or digits.

c. If you Detect your Error while Computer is Responding

Press ATTENTION KEY on your typewriter. This will cause the computer to interrupt its response. Retype your statement.

a. Explanation of your Error

1. Your Account has no case files that contain data on any of the Items specified in your Summary Definition.
2. You failed to specify any Item Number in your definition.
3. You failed to state key-words or numbers necessary for operation. You may have omitted a key-word or given an incorrect combination of numbers, etc.
4. You specified a Value Number that does not exist on the List of Variables and Values (Section XII).
5. The Summary Definition Matrix is clear. You have not assembled any data into it by defining Items and Values (Sections VIII 7, E.1., E.2.).
6. You have given the computer no I.D. on which to operate (See B.4.b.). This response may occur in any operation involving an individual minor.
7. You have attached more than one Value to an Item number within the same statement (Sections B.1., B.4.).
8. You failed to state the I.D. to be restored, or you erroneously specified more than one I.D. in the same statement (Section B.3.).
9. This response may occur when asking a question. Computer is unable to answer question for lack of data about the minor (Section D.).
10. You have stated a Value but no Item, or else more than one Item (Sections B.1., B.4.).
11. You need to type simply DEFine and the Item Number(s) for which you wish to see the data. The computer will display all Values (Sections E.1., E.2.).
12. Since deleting the I.D. to be restored, you have made a subsequent deletion or else "signed off." (Section B.3.).
13. (Section IV (3)).
14. (Section C.).
15. (Section C.).
16. (Section C.).
17. You tried to delete 2 or more I.D.'s in one statement, instead of separately. (Section B.2.).

XI. SUMMARY OF KEY WORDS & OPERATIONS

PURPOSE OF OPERATION		OPERATION
A. GETTING IN CONTACT WITH THE COMPUTER Typewriter ON; TALK BUTTON; DIAL your PHONE NO.; DATA BUTTON; HANG UP; RETURN KEY		In capitals=words to be typed (First 3 letters suffice) In lower case=relevant numbers to be typed
B. STORING & RETRIEVING INFORMATION ABOUT A CASE		
1. ESTABLISH A NEW CASE FILE (new I.D.)		NEW
2. DELETE A CASE FILE from your Account		DElete i.d.
3. RESTORE A CASE FILE erroneously deleted		REStore i.d.
4. UPDATE A CASE FILE with new information	•	i.d. item value
5. SEE SPECIFIC DATA in a case file	•	i.d. item(s)
6. SEE ALL DATA in a case file	•	i.d. FIle
C. OPERATING IN TEST MODE		
1. BEGIN Test Mode		TESt
2. END Test Mode		TESt OFF
D. ASKING QUESTIONS		
1. ABOUT RECIDIVISM (200 series questions)	•	i.d. 200 or 2item
2. ABOUT IMPROVEMENT (300 series questions)	•	i.d. 300 or 3item
E. OBTAINING SUMMARIES OF YOUR ACCOUNT		
1. Step a. DEFINE TERMS of desired summary		DEFine item value(s)
2. Step b. SEE SUMMARY defined values included	•	SUMmary item(s)
or SEE SUMMARY defined values excluded	•	SUMmary EXClude item(s)
3. REVIEW CONTENT of Summary Definition Matrix		DEFine item(s)
4. CLEAR SUMMARY MATRIX of incompatible data		CLEar
5. SEE LIST of all I.D.'s in Account	•	ACCount
6. SEE LIST of classified I.D.'s	•	DEFine item value(s) ACCount
F. REQUESTING SERVICES		
1. HAVE PRINTED DATA MAILED (operations marked•)		(Add word MAIL to statement)
2. DISPLAY TYPED MESSAGE at computer center		MESsage (text of message)
G. ENDING CONTACT WITH THE COMPUTER		OFF

XII LIST OF VARIABLES & VALUES

Item No. / Value No.

PERSONAL

11 AGE
- 1 = 11 or below
- 2 = 12 years
- 3 = 13 years
- 4 = 14 years
- 5 = 15 years
- 6 = 16 or above

12 SEX
- 1 = Male
- 2 = Female

13 ETHNIC GROUP
- 1 = Mexican-American
- 2 = Black
- 3 = All others

HOUSEHOLD

14 MINOR'S LIVING ARRANGEMENTS
- 1 = With both natural parents
- 2 = With mother only
- 3 = With father only
- 4 = With parent & step-parent
- 5 = Other

15 TOTAL PERSONS IN HOUSEHOLD
- 1 = 1 or 2
- 2 = 3 - 5
- 3 = 6 - 7
- 4 = 8 or more

16 MINOR'S SIBLING POSITION IN HOUSEHOLD
- 1 = First child
- 2 = Second
- 3 = Third
- 4 = Fourth
- 5 = Fifth
- 6 = Sixth or upward

17 OTHER MEMBERS WITH CRIME/DELINQ. RECORD?
- 1 = Yes
- 2 = No

18 TOTAL OTHERS IN HOUSEHOLD WITH RECORD
- 1 = None
- 2 = 1
- 3 = 2
- 4 = 3
- 5 = 4 or more

19 ANNUAL FAMILY INCOME
- 1 = To $ 2,999
- 2 = $ 3,000 - $ 4,999
- 3 = $ 5,000 - $ 6,999
- 4 = $ 7,000 - $ 8,999
- 5 = $ 9,000 - $10,999
- 6 = $11,000 - $12,999
- 7 = $13,000 or more

20 HEAD OF HOUSEHOLD - SEX
- 1 = Male
- 2 = Female

21 HEAD OF HOUSEHOLD - EDUCATION
- 1 = 1 - 3 years
- 2 = 4 - 7 years
- 3 = 8 - 11 years
- 4 = 12 years
- 5 = 13 - 14 years
- 6 = 15 - 16 years
- 7 = 17 or more years

22 MOVES OF RESIDENCE IN LAST 5 YEARS
- 1 = None
- 2 = 1
- 3 = 2
- 4 = 3 or 4
- 5 = 5
- 6 = 6 or more

EDUCATION AND INTELLIGENCE

23 SCHOOL GRADE
- 1 = 5th or lower grade
- 2 = 6th
- 3 = 7th
- 4 = 8th
- 5 = 9th
- 6 = 10th or higher grade

24 PROPER GRADE FOR AGE?
- 1 = Ahead
- 2 = Proper
- 3 = 1 year behind
- 4 = 1 - 2 years behind
- 5 = 2 or more years behind

25 ABSENCE THIS SCHOOL YEAR
- 1 = 0 - 2 days
- 2 = 3 - 7 days
- 3 = 8 - 14 days
- 4 = 15 - 24 days
- 5 = 25 or more days

26 ABSENCE LAST SCHOOL YEAR
- 1 = 0 - 2 days
- 2 = 3 - 7 days
- 3 = 8 - 14 days
- 4 = 15 - 24 days
- 5 = 25 or more days

27 MINOR'S I.Q.
- 1 = below 80
- 2 = 80 - 89
- 3 = 90 - 109
- 4 = 110 - 119
- 5 = 120 or above

PREVIOUS RECORD

28 NUMBER OF PRIOR OFFENSES
- 1 = None
- 2 = 1
- 3 = 2
- 4 = 3
- 5 = 4 or 5
- 6 = 6 or more

29 NUMBER OF PURELY JUVENILE OFFENSES
- 1 = None
- 2 = 1
- 3 = 2 or more

30 SERIOUSNESS OF MINOR'S CUMULATIVE RECORD
- 1 = Minor
- 2 =
- 3 =
- 4 =
- 5 =
- 6 = Very serious

31 FIRST OFFENSE: MINOR'S AGE AT
- 1 = Under 10 years
- 2 = 10 years
- 3 = 11 years
- 4 = 12 years
- 5 = 13 years
- 6 = 14 years
- 7 = 15 or older

32 FIRST OFFENSE: NUMBER COMPANIONS
- 1 = None
- 2 = 1
- 3 = 2
- 4 = 3 or more

33 TIME BETWEEN FIRST AND CURRENT OFFENSE
- 1 = 0 - 8 months
- 2 = 9 - 16 months
- 3 = 17 - 24 months
- 4 = 25 - 33 months
- 5 = 34 - 49 months
- 6 = 50 or more months

34 TIME BETWEEN LAST PRIOR AND CURRENT OFFENSE
- 1 = 0 - 4 months
- 2 = 5 - 8 months
- 3 = 9 - 14 months
- 4 = 15 - 21 months
- 5 = 22 or more months

35 HAS MINOR A REFERRAL HISTORY?
- 1 = Yes
- 2 = No

36 MINOR EVER PLACED OUT OF HIS HOME?
- 1 = Yes
- 2 = No

37 WAS MINOR EVER A WARD?
- 1 = Yes
- 2 = No

38 IS MINOR PRESENTLY A WARD?
- 1 = Yes
- 2 = No

CURRENT OFFENSE

39 REASON FOR REFERRAL
- 1 = Robbery or assault
- 2 = Burglary or grand theft
- 3 = Petty theft, auto theft, forgery
- 4 = Illegal sex acts
- 5 = Narcotics, glue sniffing, traffic
- 6 = Probation violation
- 7 = Incorrigibility, runaway
- 8 = Truancy
- 9 = Other

40 SERIOUSNESS OF REFERRAL OFFENSE
- 1 = Minor
- 2 =
- 3 =
- 4 =
- 5 = Very serious

41 REFERRAL AGENT
- 1 = Police
- 2 = Sheriff
- 3 = School
- 4 = Parent(s)
- 5 = Other

42 NUMBER OF COMPANIONS
- 1 = None
- 2 = 1
- 3 = 2
- 4 = 3
- 5 = 4 or more

43 MINOR DETAINED?
- 1 = Yes
- 2 = No

44 LENGTH OF DETENTION
- 1 = 1 day
- 2 = 2 days
- 3 = 3 - 5 days
- 4 = 6 - 16 days
- 5 = 17 - 21 days
- 6 = 22 or more days

45 TIME BETWEEN REFERRAL AND DISPOSITION
- 1 = 0 - 2 days
- 2 = 3 - 12 days
- 3 = 13 - 18 days
- 4 = 19 - 26 days
- 5 = 27 - 31 days
- 6 = 32 - 43 days
- 7 = 44 or more days

DISPOSITION AND PLACEMENT

46 PETITION REQUESTED?
- 1 = Yes
- 2 = No

47 PETITION SUBMITTED?
- 1 = Yes
- 2 = No

48 DISPOSITION AWARDED (ALL ALTERNATIVES)
- 1 = Dismiss at intake
- 2 = Informal supervision, prob/other agent
- 3 = Formal supervision continued
- 4 = Dismissed at court
- 5 = Informal supervision (court)
- 6 = Formal supervision (court)

49 DISPOSITION RECOMMENDED
- 1 = Dismiss at intake
- 2 = Informal supervision
- 3 = Formal supervision

50 DISPOSITION AWARDED
- 1 = Dismiss at intake
- 2 = Informal supervision
- 3 = Formal supervision

51 NATURE OF DISPOSITION
- 1 = No position — No treatment
- 2 = No position — Treatment
- 3 = Position —— No treatment
- 4 = Position —— Treatment

52 DISPOSITION AWARDED AS RECOMMENDED?
- 1 = Same
- 2 = Different

53 CHARGE PLEA
- 1 = Admits charge
- 2 = Denies charge

54 PLACEMENT AWARDED
- 1 = Outside own home
- 2 In own home

55 PLACEMENT RECOMMENDED
- 1 = Outside own home
- 2 = In own home

56 PLACEMENT AWARDED AS RECOMMENDED?
- 1 = Same
- 2 = Different

57 NUMBER PROBATION OFFICERS IN COURT
- 1 = One
- 2 = Two or more

PROBATION OFFICER

58 AGE OF PROBATION OFFICER
- 1 = 22 - 24 years
- 2 = 25 - 26 years
- 3 = 27 - 29 years
- 4 = 30 - 33 years
- 5 = 34 - 37 years
- 6 = 38 - 42 years
- 7 = 42 or older

59 SEX OF PROBATION OFFICER
- 1 = Male
- 2 = Female

60 MARITAL STATUS
- 1 = Never married/widow
- 2 = In first marriage
- 3 = In second (or further) marriage
- 4 = Now separated or divorced

61 CHILDREN
- 1 = None
- 2 = 1
- 3 = 2
- 4 = 3
- 5 = 4 or more

62 EXPERIENCE
- 1 = 0 - 1 year
- 2 = 2 years
- 3 = 3 years
- 4 = 4 years
- 5 = 5 or 6 years
- 6 = 7 or more years

PROBATION OFFICER CONTACTS

63 ANY CONTACTS MADE WITH MINOR?
- 1 = Yes
- 2 = No

64 NUMBER OF CONTACTS MADE IN 12 MONTHS
- 1 = 1 - 5
- 2 = 6 - 11
- 3 = 12 - 21
- 4 = 22 or more

65 TIME FIRST CONTACT MADE
- 1 = Before - 9 days after, disposition
- 2 = 10 - 31 days after disposition
- 3 = 32 or more days after disposition

66 AVERAGE LENGTH OF CONTACT
- 1 = 1 - 15 minutes
- 2 = 16 - 20 minutes
- 3 = 21 - 25 minutes
- 4 = 26 or more minutes

67 ANY CONTACTS OUTSIDE WORKING HOURS?
- 1 = Yes
- 2 = No

68 PROPORTION OF CONTACTS IN MINOR'S HOME
- 1 = None
- 2 = Half or less
- 3 = Over half

CONTACT EMPHASIS

69 ON OBTAINING INFORMATION
- 1 = None
- 2 = Low
- 3 = Average
- 4 = High

70 ON GIVING MINOR A VIEW OF HIMSELF
- 1 = None
- 2 = Low
- 3 = Average
- 4 = High

71 ON EXPRESSING APPROVAL
- 1 = None
- 2 = Low
- 3 = Average
- 4 = High
- 5 = Very high

72 ON EXPRESSING DISAPPROVAL
- 1 = None
- 2 = Low
- 3 = Average
- 4 = High

S I M B A D
"Simulation as a Basis for Social Agents' Decisions"

A computer system
to aid Probation Departments in data analysis and decision-making

Public Systems Research Institute
University of Southern California
Los Angeles

A. SETTING UP CONTACT

A. SETTING UP CONTACT WITH THE COMPUTER

A.1. TO SIGN ON

Switch your typewriter ON

Press TALK button on your telephone: wait for dial tone

Dial your telephone number: wait for high-pitched ringing tone

Press DATA button on telephone: hang up

Press return key

a. Type your ACCOUNT NUMBER immediately after the response

b. Press return key

YOUR TERMINAL IS NOW READY TO GIVE AND RECEIVE INFORMATION ACCORDING TO YOUR INSTRUCTIONS

COMPUTER RESPONSE

A.1.a. PLEASE TYPE IN YOUR SIMBAD ACCOUNT NUMBER 1000

A.1.b. SIMBAD READY 6/19/69 1138

A.1.a Note that the *RETURN* key must be depressed at the end of every complete statement made by you: it is not used before typing in your Account number since you have not yet made a statement.

A.1.b The computer records the date and the time of day (11.38).

B. STORING AND RETRIEVING INFORMATION ABOUT A CASE

B.1. TO ESTABLISH A NEW CASE FILE (I.D.) FOR A MINOR

*
New

a. Type NEW or N. (This requests the computer to allot a new identity number: a new case number or minor's file number)

Press return key

item value

b. From the List of Variables and Values (Section XII) find the ITEM number and the VALUE number of the information you wish to store. These two numbers, used together, define the information. Type these 2 numbers (with a space between)

Press return key

item value
item value
item value

c. Any amount of multiple data can be stored in this way, provided that each entry is made as a separate statement (i.e., on a separate line, the return key being used after each). Each statement must consist of only one Item number and only one Value number.

* The CODES shown in left margin are visual descriptions of the statement required for each operation:

CAPITAL LETTERS indicate words to be typed by the user. The first three letters suffice. (In the case of the words NEW and DELete, the initial n or d suffices.) Though shown here in capitals for ease of recognition, all words may be typed in lower case.

LOWER CASE LETTERS indicate that the relevant numbers must be typed (e.g., the minor's I.D. number, an Item number or a Value number).

COMPUTER RESPONSE

B.1.a.
```
-n
NEW I.D. NUMBER IS    548
-
```

B.1.b.
```
-11 1
U    548   11 1 WAS  0
-
```

B.1.c.
```
-12 2
U    548   12 2 WAS  0
-35 2
U    548   35 2 WAS  0
-16 4
U    548   16 4 WAS  0
-39 8
U    548   39 8 WAS  0
-
```

B.1.a The "hyphen" at left indicates computer is ready for the next command. It also serves to identify those lines typed by the terminal's user.

B.1.b U stands for "Updating" and indicates that a datum has been changed in value. This is further emphasized by the subsequent statement of what the value was before change.
The statement reads: Case 548: "Age is 11 or under. Age was previously unknown."
The numbers shown in the statement are, in order: I.D. (case) number; Item (variable) number; new Value number; previous Value number.

B.1.c. The statements read: Case 548: Is a female
Has no record of prior offense
Is the 4th child of the family
Has been referred for truancy

B.2. TO DELETE A CASE FILE FROM YOUR ACCOUNT (when the minor is no longer your concern, e.g., his case file is closed).

(Beginners, experimenting with the use of the computer, should first establish a new (fictitious) I.D. (as in Section B.1.) and then delete it.)

Delete i.d.

Type DELete)	or	and the I.D. number (with space between)
D)	or	
REMove)		
		Press return key

Only one I.D. can be deleted at a time (in a single statement).

NOTE: The DELete procedure may be applied solely to the deletion of I.D.'s (case files) in the Account. The deletion of Items and Values is done by Updating (Section B.4.).

B.3. TO RESTORE A CASE FILE ERRONEOUSLY DELETED

Beginners, experimenting with the use of the computer, should first establish a new (fictitious) I.D. (as in Section B.1.), delete it (as in Section B.2.) and then restore it.

REStore i.d.

a. Type REStore and the I.D. number (with space between)
Press return key

b. Note that it may not always be possible for the computer to restore the contents of a deleted case file under the same I.D. number.

c. The file will be restored only provided that, subsequent to its deletion,

1. No deletion of any other I.D. has been made, and
2. Contact with the computer has not been disconnected by "signing off."

If a deleted file cannot be restored for the above reasons, a new I.D. will have to be requested (Section B.1.) and the lost data will have to be fed back into the computer under the new I.D. number.

Only one I.D. can be restored at a time (in a single statement).

NOTE: The REStore procedure may be applied solely to the restoration of I.D.'s (case files) in the Account. The restoration of deleted Items and Values is done by Updating (Section B.4.).

COMPUTER RESPONSE

B.2.
```
-delete 607
D   607
```

B.3.a.
```
-restore 607
CASE   607 RESTORED.  NEW I.D. IS   607
```

B.3.b.
```
CASE   607 RESTORED.  NEW I.D. IS   601
-
```

B.3.c.
```
-res 607
*** UNABLE TO RESTORE THE SPECIFIED I.D.
```

B.2. The computer responds, confirming deletion of the file, as requested.
B.3.a. The computer confirms the file has been restored under the same I.D. number.
B.3.b. The computer confirms the file has been restored, but under a different I.D. number.
B.3.c. The computer has been unable to restore the contents of the case file.

B.4. TO UPDATE A CASE FILE WITH NEW INFORMATION (add or change data).

WARNING: Beginners "practicing" should perform this operation in TEST MODE (Section C) to avoid real change of data.

From the List of Variables and Values (Section XII) find the ITEM number and the VALUE number of the information you wish to store.

i.d. item value

a. Type the minor's I.D., the ITEM number and the VALUE number (with a space between each number)

Press return key

i.d. item value
item value
item value...

b. Any Number of Changes can be made in this way as long as statements (each consisting of only 1 Item number and 1 Value number) are made separately, the return key being used after each. It is not necessary to repeat the I.D. in subsequent statements about the same minor. Once an I.D. has been stated, the computer operates on it until a different I.D. has been stated.

item 0

c. To Delete an Entire Item from an Updating Procedure

Type the Item number and 0 (with a space between)

Press return key

In this way you "zero out" a previously made statement now found to be unverifiable or otherwise entered in error.

NOTE that the DELete and REStore procedures described in Section B.2. are never applied to deletion and restoration of Items or Values, but are used solely for deleting I.D.'s (case files) from the Account. Items and Values are deleted or restored by updating, as above.

COMPUTER RESPONSE

B.4.a.
```
-570 19 3
U   570  19 3 WAS  0
```

B.4.b.
```
-39 3
U   570  39 3 WAS  7
-41 1
U   570  41 1 WAS  2
-42 4
U   570  42 4 WAS  2
-28 2
U   570  28 2 WAS  1
```

B.4.c.
```
-38 0
U   570  38 0 WAS  1
```

B.4.a. The statement reads: Case 570: Annual family income is between $5,000 - $6,999.

B.4.b. The statements read: Case 570: Referred for petty theft
Referred by police
Had 3 companions on offense
Has 1 prior offense

The response also shows that this minor has had one previous referral (by the Sheriff's department). He was then referred as a runaway and had one companion.

B.4.c. The statement reads: Case 570: It is not now known whether or not the minor is a ward of court. He was previously recorded as such.

Note that a zero merely indicates that the computer has no information about an item. A zero is never used as a number for variable measurement.

B.5. TO SEE SPECIFIC DATA IN A CASE FILE

i.d. item(s)

a. Type minor's I.D. (unless you are already operating on it (see Section B.4.b.)) and the ITEM number(s) for which you wish to see the Value (with spaces between)

Press return key

b. Several data may be examined simultaneously, the relevant Item numbers being typed consecutively.

B.6. TO SEE ALL DATA STORED IN A CASE FILE

i.d. FILe

Type minor's I.D. and the word FILe (with space between)

Press return key

COMPUTER RESPONSE

B.5.a.

```
-546 63
   546  63 1
```

B.5.b.

```
-546 64 65 66 71 72 58
   546  64 1
   546  65 1
   546  66 3
   546  71 5
   546  72 1
   546  58 7
```

B.6.

```
-570 file

DATA IN FILE OF CASE   570
11      4                47      2
12      2                48      2
13      3                49      2
14      2                50      1
15      2                51      2
16      2                52      1
17      2                53      2
18      2                54      1
19      1                55      2
20      2                56      2
21      2                58      1
22      5                59      2
23      4                60      2
24      1                61      1
25      3                62      1
27      2                63      1
28      1                64      3
39      7                65      3
40      4                66      3
41      2                67      2
42      2                68      2
43      1                69      4
44      1                70      4
45      2                71      3
46      2                72      4
```

B.5.a. The statement reads: Case 546: Probation officer has contacted minor.

B.5.b. The statements read: Case 546: Total number of contacts has been 1 - 5, the first being in the period extending from before disposition to 9 days after it. Average length of contact 21-25 mins., emphasis very high on giving approval. Age of probation officer 42 or older.

B.6. Case 570: Column 1 of print-out shows the Item numbers (variables). Column 2 of print-out shows the Value numbers.

C. OPERATING IN TEST MODE

The function of the Test Mode feature of SIMBAD is to permit operations that will not effect any permanent change of data filed in the computer.

(a) For Experimenting in Operational Training

Users becoming acquainted with the system can experiment without fear of inadvertently destroying or changing filed data by first typing the word TESt. All subsequent operations then cause the computer to respond as it would in normal mode except that no permanent changes are made in the data filed.

All operations may be performed in Test Mode except:

Section B.1. Establishing a new case file
Section B.2. Deleting a case file
Section B.3. Restoring a deleted case file

Operations which should always be performed in Test Mode by experimenters are:

Section B.4 Updating (changing or adding data to a file)
Section D. Questions, if updating procedures are involved

The reason for this is obvious: if performed in normal mode, the updating action would cause data on file to be actually changed.

(b) For Asking Hypothetical Questions

In Test Mode the user can ask hypothetical questions about an actual case, since he can (temporarily) add whatever data he wishes to the minor's case history (by updating it) and then ask what would be the probability of recidivism or behavioral improvement of a minor with such a history. (See Section D for the details about asking Questions).

C.1. TO BEGIN TEST MODE

TESt	Type TESt Press return key Type any desired operations except B.1., B.2. or B.3.

C.2. TO END TEST MODE

TESt OFF	Type TESt OFF or TESt DONe or TESt FINished or END TESt. Press return key

COMPUTER RESPONSE

```
-n
*** YOU MAY NOT CREATE NEW I.D. NUMBERS WHILE IN TEST MODE

-del 570
*** YOU MAY NOT DELETE AN I.D. WHILE IN TEST MODE

-res 570
*** YOU MAY NOT DELETE OR RESTORE WHILE IN TEST MODE
```

C.1. -tes
YOU ARE NOW IN TEST MODE

C.2. -test off
YOU ARE NOW IN NORMAL MODE

D. ASKING QUESTIONS

The flexibility of the system in furnishing response to a wide variety of questions is a most important feature of its role as aid to the decision maker. Questions may be asked:

About a delinquent minor, either real or hypothetical;

For purposes either of classification diagnosis or of aiding decisions about disposition, treatment and placement.

THE RESPONSE to any such questions will be displayed in one of two forms, as specified by the user. He may ask to see it in terms of either:

1. The likelihood of recidivism or non-recidivism, or
2. The likelihood that behavior will get worse, remain the same, or get better

In each case the probability will be expressed as a decimal number between .0 and 1.00.

HOW TO FORMULATE A QUESTION

QUESTION 200 asks to see the output in terms of Recidivism.

QUESTION 300 asks to see the output in terms of Behavioral Improvement.

Each of the Items on the List of Variables and Values (Section XII) also automatically becomes a Question Number if a digit 2 or a digit 3 is placed in the "hundreds" position before it: a 2 denotes that the question is to be answered in terms of Recidivism; a 3 denotes it is to be answered in terms of Improvement, e.g., Item 12 (age) becomes Question 212 (asking for an analysis of the likelihood of recidivism by age category).

Inherent, therefore, in every such Question Number are two elements:

A definition of the form of the response (denoted by digit 2 or 3);

A statement of the general category of concern (denoted by Item Number) which will form the basis for distribution of the analysis.

Questions about Recidivism are referred to as "200 series questions"

Questions about Improvement are referred to as "300 series questions"

D.1. TO ASK A QUESTION ABOUT RECIDIVISM (200 SERIES)

(a) For Classification Probability of a Specific Delinquent Minor

Simply link your Question number to the minor's I.D. number. It is assumed, of course, that some data (at least one Item and its Value) have been filed under this I.D.

Example (i) You wish to learn what is the probability that minor 555 will recidivate, based only on the information that has been filed about him.

i.d. 200

Type 555 (I.D. number) and 200 (Question number)

Press return key

Example (ii) You wish to learn the likelihood of this minor's recidivism analysed by the number of other household members having a record of crime or delinquency.

i.d. 2item

Type 555 (I.D. number) and 218 (Question number based on Item 18 (number of household with record)

Press return key

Note that the form of this question already allows the introduction of a "hypothetical" element, on which many changes may be rung. However, the asking of further hypothetical questions about an actual I.D. MUST BE DONE IN TEST MODE (Section C.), as demonstrated in the next example. Questions based on hypothetical data naturally involve Updating (Section B.4.). If asked in Normal Mode, they would cause the actual data on file to be altered.

COMPUTER RESPONSE

Example (i)

```
-555 200
QUESTION 200 CASE   555
PROBABILITY          NON-R     RECID
                     0.551     0.448
```

Example (ii)

```
-555 218
QUESTION 218 CASE   555
NO IN FAM W RECORD    NON-R     RECID
NONE                  0.599     0.400
1                     0.556     0.443
2                     0.524     0.475
3                     0.478     0.521
4+                    0.466     0.533
```

Example (i) In case 555, a 16 year old white female referred for probation violation, the likelihood of non-recidivism is seen to be greater than the likelihood of recidivism, based on the information filed about her.

Example (ii) The likelihood of non-recidivism will increase if it can be verified that the household in which she lives includes no other members with records of crime or delinquency; the likelihood will diminish if more than one such member has a record.

(b) For Classification Probability of a Specific Delinquent Minor, The Question Based on Some Hypothetical Elements.

Example (iii) You wish to learn what would happen if minor 555 were in proper school grade for her age and had no record of school absence, either this year or last year.

TESt i.d. item value item value... 200 or 2item	Type TESt (Return key to be used after each statement) Type I.D. (Item 24–school grade) Value (2=proper) Type Item (25–absence this year) Value (1=none) Type Item (26=absence last year) Value (1=none) Type 200

You may continue asking questions about this I.D. by varying your Question numbers or introducing other data by Updating (Section B.4.). The resulting output may then be compared with that obtained by using the minor's real data (Example (i))

Example (iv) You wish to see the influence of living arrangements on this minor's recidivism.

Type 214 (Item 14=living arrangements) Press return key

Example (v) You wish to see the influence of expressions of approval by the minor's probation officer.

Type 271 (Item 71=expressing approval) Press return key

(c) For Classification Probability of a Hypothetical Minor

Since Test Mode does not allow the creation of a new I.D., such questions must be asked in NORMAL MODE.

Example (vi) You wish to learn the likelihood of recidivism of boys aged over 16 who have a long record of prior offense and live in a household containing other delinquents.

n item value item value... 200 or 2item	Type n (Return key to be used after each statement) Type Item (11=age) Value (6=over 16) Type Item (12=sex) Value (1=male) Type Item (17=family record) Value (1=yes) Type Item (28=prior offenses) Value (6=6 or more) Type Item (39=referral reason) Value (2=burglary) Type 200

After you have finished asking questions about this I.D., be sure to delete it (Section B.2.), since, being fictitious, it must not remain on file.

COMPUTER RESPONSE

Example (iii)

```
-test
YOU ARE NOW IN TEST MODE
-555 24 2
U    555   24 2 WAS  0
-25 1
U    555   25 1 WAS  0
-26 1
U    555   26 1 WAS  0
-200
QUESTION 200 CASE    555
PROBABILITY              NON-R     RECID
                         0.696     0.303
```

Example (iv)

```
-214
QUESTION 214 CASE    555
LIVING ARRANGEMENT       NON-R     RECID
          NAT PRNTS      0.710     0.289
          ONLY MOTH      0.673     0.326
          ONLY FATH      0.633     0.366
          PAR-STPPR      0.692     0.307
              OTHER      0.731     0.268
```

Example (v)

```
-271
QUESTION 271 CASE    555
EMPH ON APPROVAL         NON-R     RECID
               NONE      0.687     0.312
                LOW      0.584     0.415
            AVERAGE      0.641     0.358
               HIGH      0.664     0.335
          VERY HIGH      0.799     0.200
```

Example (vi)

```
-n
NEW I.D. NUMBER IS    622
-11 6
U    622   11 6 WAS  0
-12 1
U    622   12 1 WAS  0
-17 1
U    622   17 1 WAS  0
-28 6
U    622   28 6 WAS  0
-39 2
U    622   39 2 WAS  0
-200
QUESTION 200 CASE    622
PROBABILITY              NON-R     RECID
                         0.399     0.600

-delete 622
D    622
```

Example (iii) The probability of non-recidivism is increased considerably in comparison with that based on the minor's real data (Example (1).

Example (iv) This response indicates that minor 555 is less likely to recidivate if she lives with two parents rather than one.

(The somewhat high probability shown at "Other" (.731) was based on relatively few cases; moreover, this category is a "catch all" for all living arrangements other than those itemized.)

(d) For Aiding Decisions about Disposition, Placement and Treatment

Probably the most valuable feature of the question-answer mechanism of the system lies in its ability to assist in the formation of decisions that have to be made about the disposition, placement and treatment of delinquent minors, since it is able to display the probabilities of success or failure of all dispositional and placement alternatives.

On the List of Variables and Values (Section XII), several Items will be seen to be concerned with disposition and placement. Among these are those concerned with past awards:

> Items 48 and 50 (dispositional award, shown in varying formats);
> Item 51 (nature of disposition and placement, with/without treatment);
> Item 54 (placement award).
> Stated as Recidivism Questions, these are Questions 248, 250, 251 and 254.

Any of the operations done in previous examples may be performed using the above numbers as operative question numbers. The output will then show an analysis of the probability of recidivism under all of the various dispositional and placement alternatives inherent in your question.

Example (vii) You wish to learn how the probability of minor 555's recidivism would be affected by the nature of the dispositional or placement award.

i.d. 248 or 250 etc.	Type I.D. (555) and 248 or 250 or 251 or 254 Press return key

Example (viii) You wish to learn what would happen if Case 555 were detained 7 days, were then placed under informal supervision and were seen weekly by the probation officer.

TESt i.d. item value item value... 248 or 250 etc.	Type TESt (Return key to be used after each statement) Type I.D. Item (44 detention) Value (4=7 days) Type Item (64=contact) Value (4=22 or more yearly) Type 250 (or other dispositional question)

It is similarly possible to ask questions such as 249, which would result in a response showing what dispositions have previously been recommended for such cases.

COMPUTER RESPONSE

Example (vii)

```
-555 248
QUESTION 248 CASE   555
DISPOSITION AWARD      NON-R     RECID
            DISM-INFF  0.565     0.434
            ISUP-AGEN  0.599     0.400
             FSUP CTD  0.375     0.624
            DISM-COPT  0.617     0.382
            ISUP-COPT  0.541     0.458
            FSUP-COPT  0.536     0.463
-555 250
QUESTION 250 CASE   555
DISPOSITION AWARD      NON-R     RECID
            DISMISSED  0.578     0.421
              INF SUP  0.501     0.458
              FOR SUP  0.524     0.475
-555 251
QUESTION 251 CASE   555
NATURE OF DISP         NON-R     RECID
            -POS -TRT  0.640     0.359
            -POS +TRT  0.468     0.531
            +POS -TRT  0.639     0.360
            +POS +TRT  0.491     0.508
-555 254
QUESTION 254 CASE   555
PLACEMENT AWARDED      NON-R     RECID
              OUT HOME 0.550     0.449
               IN HOME 0.573     0.426
```

Example (viii)

```
-test
YOU ARE NOW IN TEST MODE
-555 44 4
U   555  44 4 WAS  0
-64 4
U   555  64 4 WAS  0
-250
QUESTION 250 CASE   555
DISPOSITION AWARD      NON-R     RECID
            DISMISSED  0.255     0.744
              INF SUP  0.227     0.772
              FOR SUP  0.215     0.784
-
```

Example (vii) To demonstrate output format of the various dispositional questions, Questions 248, 250, 251 and 254 are here consecutively asked, based upon the minor's real data. It will be noted that the highest probability is for dismissal.

Example (viii) The response, after updating, now indicates that, for this minor, detention and supervision would increase the likelihood of recidivism.

D.2. TO ASK A QUESTION ABOUT BEHAVIORAL IMPROVEMENT

Such questions are asked in the same way as the Recidivism Questions illustrated in Section D.1., except that Question 300 is used instead of Question 200; digit 3 is attached to Item numbers instead of digit 2.

Since all operations demonstrated in Examples (i) through (viii) may be performed using a 300 series question in place of a 200 series question, they will not here be described in length. The following example will serve to illustrate the format in which the response to such behavioral questions is displayed:

Example (ix) Suppose you are setting up a new case file for a boy aged 14, Anglo, referred for a narcotics offense, having 3 prior offenses, whose I.Q. is 90.

You wish to learn the probability that his behavior will improve, remain the same, or deteriorate if he is dismissed, or is placed under informal supervision, or is placed under formal supervision.

```
n
item value
item value...
348 or 350 etc.
```

Type n (Return key to be used after each statement)

Type Item (12=sex) Value (1=male)

Type Item (11=age) Value (4=14 years)

Type Item (13=ethnic) Value (3=Anglo)

Type Item (39=offense) Value (5=narcotics)

Type Item (27=I.Q.) Value (3=90)

Type 350 (Item 50=disposition award) (or Questions 348, 351, 354, etc., according to format desired)

Example (x) If hypothetical elements are introduced, you must go into TEST MODE (Section C.) to avoid real change of data.

You now wish to ask whether, if the minor were made a ward of Court, he would be better placed in his own home or outside it.

```
TESt
item value...
354
```

Type TESt (Return key to be used after each statement)

Type Item (38=presently a ward) Value (1=yes)

Type 354 (Item 54=placement awarded)

COMPUTER RESPONSE

Example (ix)

```
-n
NEW I.D. NUMBER IS    622
-12 1
U   622  12 1 WAS  0
-11 4
U   622  11 4 WAS  0
-13 3
U   622  13 3 WAS  0
-39 5
U   622  39 5 WAS  0
-27 3
U   622  27 3 WAS  0
-350
QUESTION 350 CASE   622
DISPOSITION AWARD      WORSE     SAME     BETTER
            DISMISSED  0.533     0.263    0.202
              INF SUP  0.486     0.251    0.262
              FOR SUP  0.335     0.150    0.513
-348
QUESTION 348 CASE   622
DISPOSITION AWARD      WORSE     SAME     BETTER
            DISM-INTK  0.556     0.259    0.183
            ISUP-AGFN  0.505     0.277    0.216
             FSUP CTD  0.291     0.099    0.609
            DISM-CORT  0.436     0.249    0.314
            ISUP-CORT  0.486     0.251    0.262
            FSUP-CORT  0.339     0.154    0.505
```

Example (x)

```
-test
YOU ARE NOW IN TEST MODE
-38 1
U   622  38 1 WAS  0
-354
QUESTION 354 CASE   622
PLACEMENT AWARDED      WORSE     SAME     BETTER
            OUT HOME   0.365     0.178    0.455
             IN HOME   0.200     0.076    0.723
```

Example (ix) The behavior of Case 622 is most likely to improve if he is placed under formal supervision of the court.

The response to Question 348 is here shown as a further example of the format of computer output.

Example (x) Minor 622, however, is seemingly better placed in his own home than out of it.

E. OBTAINING SUMMARIES OF YOUR ACCOUNT

The computer displays not only case file data but also classified summaries and lists of your Account's total case-load. The nature of the summary is dictated entirely by the user: he defines its terms by stating the Items and Values to be considered. This defining process sets up in the computer a SUMMARY DEFINITION MATRIX (Section VIII 7) from which the summary table is generated and displayed on command. By simply repeating this process, changing Items and Values as required, you can specify multi-variate distributions and obtain any statistics you choose about your Account.

NOTE, however, that the system, as currently augmented, allows no more than three Items in any one summary request. There is no limit to the number of Values that may be specified: in direct contrast to updating procedures, you may, in a Summary Definition, specify several Values in the same statement.

NOTE also Section E.4., "Clearing the Summary Definition Matrix."

MAILING SERVICE. The computer cannot respond instantaneously to a summary request: it requires a few minutes to scan the files and assemble data into the Summary Definition Matrix. Use of the mailing service (Section F.1.) will save time.

Summaries may be requested by two methods:

1. Summary by INCLUSION of Values (Section E.1.)
2. Summary by EXCLUSION of Values (Section E.2.)

E.1.

TO OBTAIN A SUMMARY BY INCLUSION

TWO SEPARATE ACTIONS are required:

STEP a. Define the Terms of the Desired Summary

Type a statement, or series of statements (up to 3), each including the word DEFine, the appropriate Item number and its Value(s) to be included:

STEP b. Request Display of the Summary

Type the word SUMmary and Item number(s) (but not the Values) specified in your Step a.

E.1.

Example (i) A simple summary involving only one Item.

```
DEFine item value(s)
SUMmary item
```

You wish to learn how many of your cases are in grades 8 and 9.

(Return key to be used after each statement)

a. Type DEFine Item (23=grade) Values (4 & 5=8th & 9th)

b. Type SUMmary Item (23)

Press return key

COMPUTER RESPONSE

Example (i)

a. -def 23 4 5

b. -sum 23
SUMMARY OF GRADE IN SCHOOL

8TH	9TH
617	589

TOTAL CASES IN SUMMARY 1206

E.1.

Example (ii) A summary involving simultaneous consideration of three Items.

DEFine item value(s)
DEFine item value(s)
DEFine item value(s)
SUMmary item item item

You wish to learn how many boys in your Account, who were referred for petty theft, admitted the charge, and how many denied it.

(Return key to be used after each statement)

a. Type DEFine Item (12=sex) Value (1=male)
   Type DEFine Item (39=referral) Value (3=petty theft)
   Type DEFine Item (53=plea) Values (1 & 2=admit & deny)
b. Type SUMmary Items (12 39 53)

E.1.

Example (iii) To pass to a subsequent summary, you need change only those previously defined elements that will be incompatible with your new summary. The data you have already assembled in the Summary Definition Matrix will remain there until you change them (Section VIII 7).

Having just completed the operations of Example (ii), you now wish to see the same information about girls instead of boys.

(Return key to be used after each statement)

a. Type DEFine Item (12=sex) Value (2=female)
b. Type SUMmary Items (12 39 53)

COMPUTER RESPONSE

Example (ii)

a. -def 12 1

-def 39 3

-def 53 1 2

b. -sum 12 39 53
SUMMARY OF SEX BY REFERRAL REASON BY CHARGE PLEA

ADMITTED

	MALE
PETTY TFT	188

DENIED

	MALE
PETTY TFT	275

TOTAL CASES IN SUMMARY 463

Example (iii)

a. -def 12 2

b. -sum 12 39 53
SUMMARY OF SEX BY REFERRAL REASON BY CHARGE PLEA

ADMITTED

	FEMALE
PETTY TFT	35

DENIED

	FEMALE
PETTY TFT	44

TOTAL CASES IN SUMMARY 79

Example (iii)
b. Items 12 39 & 53 appear in the Summary Request, but only Item 12 appears in the definition, since Items 39 & 53 were defined in the preceding summary and the definition still remains in the Summary Matrix.

E.1.

Example (iv) In Example (iii) only a Value was changed. You may also change Items (bearing in mind that only 3 may simultaneously be considered). To exclude a previously defined Item from your summary (without substituting it by another Item), simply avoid mentioning the Item in your Summary Request at Step b. If, however, you wish to substitute the Item by another, the latter must be defined in Step a. as well as stated in Step b.

You now wish to exclude the sex factor from your summary and to learn how many minors in the group under consideration are known to have a previous referral history.

(Return key to be used after each statement)

a. Type DEFine Item (35=history) Values (1 & 2=Yes and No)

b. Type SUMmary Items (39 53 35)

COMPUTER RESPONSE

Example (iv)

```
-def 35 1 2

-sum 39 53 35
SUMMARY OF REFERRAL REASON     BY CHARGE PLEA          BY REFERRAL HISTORY

                        YES

                PETTY   TFT
ADMITTED                56
DENIED                  145

                        NO

                PETTY   TFT
ADMITTED                26
DENIED                  57

TOTAL CASES IN SUMMARY   284
```

Example (iv) The new Item for consideration (35=referral history) is defined.

The Summary Request contains the new Item 35 but not the excluded Item 12 (sex).

E.2. SUMMARY BY EXCLUSION OF VALUES.

In the previous examples of Summary by Inclusion, all pertinent Values were stated in the definition and were positively included in the summary. An alternative method allows you the option of defining Values to be excluded, in which case you use the command SUMmary EXClude in your Summary Request.

If, in the Items to be summarized, the Values you wish to see are more numerous than those you do not wish to see, it may be more convenient to exclude the latter rather than include the former. This method, where appropriate, saves time and results in a less bulky print-out. It has the advantage of increased flexibility in the following two ways: (1) in contrast with the Inclusion Method, you may (as in Example (v) following) request a summary based on Items only, with no Values defined, thus bypassing the entire defining operation of Step a; (2) if your summary does involve a definition (an Item with Values specified for exclusion), you may add to your Summary Request at Step b. further Items not included in the preceding definition. Such added Items will then be summarized in all their Values since you have defined none for exclusion, i.e., the Summary Definition Matrix shows zeros in the Value positions for these Items (Section VIII 7).

E.2.
Example (v) A summary bypassing the definition process since it is based on 2 Items only, no Values defined.

You wish to learn the school grades of all minors in your Account, and which minors are in a grade proper to their age level.

```
SUMmary EXClude item item
```

b. Type SUMmary EXClude Item (23=grade) Item (24=proper)
Press return key

COMPUTER RESPONSE

Example (v)

```
-sum exc 23 24
SUMMARY OF GRADE IN SCHOOL     BY PROPER GRADE

              5-BELOW    6TH    7TH    8TH    9TH   10-ABOVE
    AHEAD          23     15     33     89     82        148
   PROPER          31     16     38     94     98        126
1 YR BACK          29     23     57     78    119         87
  1-2 BACK         59     50    106    190    212         40
  2+ BACK          45     50     89    166     78          0

TOTAL CASES IN SUMMARY  2271
```

Example (v) There is no Step a. in this summary. The definition process has been bypassed since no Values are to be excluded.

E.2. TO OBTAIN A SUMMARY BY EXCLUSION

STEP a. If your Summary is to Exclude any Values, Define the Terms of the Desired Summary

Type a statement, or series of statements (up to 3), each including the word DEFine, the appropriate Item number and its Value(s) to be excluded:

STEP b. Request Display of the Summary

Type the words SUMmary EXClude and the pertinent Item number(s) (but not the Values) whether the Items have been specified in Step a. or not (and provided there are no more than 3 Items in total)

E.2.
Example (vi) A summary based on 2 Items with Values excluded.

```
DEFine item value(s)
DEFine item value(s)
SUMmary EXClude item item
```

You wish to learn the number of Mexican-Americans in your Account in all referral categories except truancy, incorrigibility and "Other" categories.

(Return key to be used after each statement)

a. Type DEFine Item (13=ethnic) Values excluded (2 & 3=non-Mex)
   Type DEFine Item (39=referral) Values excluded (7, 8 & 9)
b. Type SUMmary EXClude Items (13 39)

E.2
Example (vii) Illustrates the redefinition of an Item by change of Value at Step a. and the addition of a third, undefined Item at Step b.

Having just completed the above summary, you now wish to see the number of black minors (instead of Mexican) in the same referral categories, and how many of these have a referral history.

(Return key to be used after each statement)

a. Type DEFine Item (13=ethnic) Values excluded (1 & 3=non-black)
b. Type SUMmary EXClude Items (39 13 35)

COMPUTER RESPONSE

Example (vi)

```
-def 13 2 3

-def 39 7 8 9

-sum exc 13 39
SUMMARY OF ETHNIC GROUP        BY REFERRAL REASON

              MEX-AMER
 ROBRY-AS           17
BURG-GTFT           34
PETTY TFT          106
ILLEG SEX           12
NARCOTICS           21
PROB VIOL            8

TOTAL CASES IN SUMMARY    198
```

Example (vii)

```
-def 13 1 3

-sum exc 39 13 35
SUMMARY OF REFERRAL REASON     BY ETHNIC GROUP        BY REFERRAL HISTORY

                              YES

        ROBRY-AS BURG-GTFT PETTY TFT ILLEG SEX NARCOTICS PROB VIOL
BLACK          6         8        23         1         2         4

                              NO

        ROBRY-AS BURG-GTFT PETTY TFT ILLEG SEX NARCOTICS PROB VIOL
BLACK          9         3         5         0         0         0

TOTAL CASES IN SUMMARY     61
```

Examples (vi) and (vii) Note the difference in format between these two summaries. The Item first mentioned in your Summary Request determines the format and forms the basis of the distribution of the analysis. Items may be stated in whatever order you desire to produce whatever format suits your filing system best.

E.3. TO REVIEW THE CONTENT OF THE SUMMARY DEFINITION MATRIX

If, while defining a summary either by Inclusion or by Exclusion, you forget what data you have assembled in the Summary Definition Matrix during a preceding operation, it is a simple matter to obtain a review of defined data.

DEFine item item	Type DEFine and all Item numbers for which you wish to see the Values Press return key

The computer responds by displaying a digit 1 in whichever columnar positions describe the Values specified (i.e., Column 1=Value 1; Column 6=Value 6, etc.) Four examples of this review may be seen in the Computer Response opposite (see also Section VIII 7). Such a review can be made at any point in the summary defining process without interrupting it.

E.4. TO CLEAR THE SUMMARY DEFINITION MATRIX

Before defining and requesting a summary, it may be necessary to ascertain that the Summary Definition Matrix is clear, i.e., that it does not contain elements assembled during some previous operation that may be incompatible with the summary now to be made.

CLEar	Type CLEar Press return key

This action clears all data out of the Summary Definition Matrix, i.e., sets all digits in it again at zero (Section VIII 7).

The omission of a previously defined Item from a Summary Request (Step b.), as described in Example (iv), insures that the Item will not be taken into account in the summary but does not actually clear that Item from the Matrix.

Note that, in a Summary by Exclusion, the exclusion of all Values of a defined Item does not exclude that Item. For example, if in your first summary you have defined Item 12 (sex) Value 1 (male) for exclusion, and then in your next summary you wish to eliminate the sex factor entirely, you may not do so by defining Item 12 Value 2 (female) also for exclusion. To do so would automatically reinstate Value 1 (Section VIII 7).

COMPUTER RESPONSE

1. -def 14 1

   -def 12 1

   -sum exc 14 12
   SUMMARY OF LIVING ARRANGEMENT BY SEX

	ONLY MOTH	ONLY FATH	PAR-STPPR	OTHER
FEMALE	122	19	133	57

   TOTAL CASES IN SUMMARY 331

2. -def 12 14
   12) 1 0 0 0 0 0 0 0 0 0
   14) 1 0 0 0 0 0 0 0 0 0

3. -def 12 2

   -def 12

4. 12) 0 1 0 0 0 0 0 0 0 0

5. -sum exc 14
   SUMMARY OF LIVING ARRANGEMENT

	ONLY MOTH	ONLY FATH	PAR-STPPR	OTHER
	526	100	467	152

   TOTAL CASES IN SUMMARY 1245

6. -clear
   SUMMARY MATRIX CLEARED

7. -def 14
   14) 0 0 0 0 0 0 0 0 0 0

THE ABOVE COMBINES OPERATIONS DESCRIBED IN SECTIONS E.2., E.3. & E.4.

E.2. 1. SUMMARY BY EXCLUSION: You wish to learn how many girls in your Account are not living with their natural parents.

E.3. 2. REVIEW OF MATRIX: Note Item 12 displays digit 1 in the position of Value 1 (=males excluded).

E.4. 3. AN ERROR: You try to eliminate the factor of sex entirely by erroneously defining Value 2 also for exclusion.

E.3. 4. REVIEW OF MATRIX: Digit 1 is now shown in the position of Value 2. If proceeded with, the summary would now display the living arrangements of boys only.

E.4. 5. A CORRECT SUMMARY: To obtain a summary eliminating the sex factor entirely, simply omit Item 12 from your Summary Request.

E.4. 6. CLEARING THE MATRIX: The response confirms that the Summary Matrix is clear.

E.3. 7. REVIEW OF MATRIX: All Value positions are now set at zero.

E.5. TO SEE A LIST (AND TOTAL NUMBER) OF ALL I.D.'s IN YOUR ACCOUNT

To find these numbers the computer has to scan the Summary Definition Matrix just as it does for a numerical summary. It is therefore most important that the Matrix is first cleared.

As also with numerical summaries, the computer may require a few minutes to perform the operation. Use of the mailing service (Section F.1.) may therefore be appropriate.

ACCount	Type CLEar Press return key Type ACCount Press return key

E.6. TO SEE A LIST OF ALL I.D.'s CLASSIFIED BY ITEMS AND VALUES

You may wish to learn which minors in your Account fall within certain defined classifications. To obtain a list of such I.D.'s, the procedure is identical with that of Section E.5. above, except that, having first cleared the Matrix, you then define the terms of the required classification just as you did in Step a. of a Summary by Inclusion (Section E.1.). There is no restriction on the number of Items and Values that may be defined.

If, for example, you wish to see the I.D. numbers of all minors in your Account who are males and who have been referred for illegal sex acts:

DEFine item value(s) DEFine item value(s) ACCount	Type CLEar (Return key to be used after each statement) Type DEFine Item (12=sex) Value (1=male) Type DEFine Item (39=referral) Value (4=illegal sex) Type ACCount

The computer will then compile the I.D. numbers of all minors in your Account who have in common all elements of your definition.

COMPUTER RESPONSE

```
-clear
SUMMARY MATRIX CLEARED
-account
 FILES FOR ACCOUNT    1000 WHICH MATCH THE SUMMARY MATRIX

   548    550    555    571    572    573    574    575    576    577

   578    579    580    581    582

TOTAL CASES IN SUMMARY     15
-
```

```
-clear
SUMMARY MATRIX CLEARED
-def 12 1

-def 39 4

-acc
 FILES FOR ACCOUNT    1000 WHICH MATCH THE SUMMARY MATRIX

   550    571    576    580

TOTAL CASES IN SUMMARY      4
-
```

F. REQUESTING SERVICES

F.1. TO REQUEST PRINTED DATA TO BE MAILED TO THE USER

The response of the computer to simple storage and retrieval requests is almost instantaneous. However, in responding to more complex operations, such as summaries and lengthy questions, the computer may require a few minutes to search for, compile and print out the data.

If you wish to save time, you may request the print-out to be mailed to you from Computer Center, rather than have it displayed on your own terminal. The print-out time required using the Mailing Service is somewhat less than it would be otherwise.

- - - MAIl	Add the word MAIl to your statement (in the case of summaries, add it to your final statement, i.e., at Step b.)

This mailing service is not restricted to summaries and questions: it may also be appropriately used for the operations described in Sections B.4. (updating), B.5. and B.6. (seeing data in a case file).

COMPUTER RESPONSE

F.1.a.
```
-cle
SUMMARY MATRIX CLEARED
-def 64 1 2 3 4

-def 13 1

-def 12 1

-sum 64 13 12 mail
OUTPUT HAS BEEN PRINTED AND WILL BE SENT
```

F.1.b.
```
-def 13 2 3

-def 12 2

-sum exc 64 13 12 mail
OUTPUT HAS BEEN PRINTED AND WILL BE SENT
```

F.1.c.

OUTPUT FOR ACCOUNT 1000 DATE 6/19/69 TIME 1355

SUMMARY OF NUMBER OF CONTACTS BY ETHNIC GROUP BY SEX

MALE

	1-5	6-11	12-21	22+
MEX-AMER	**98**	**76**	**26**	**5**

TOTAL CASES IN SUMMARY 205

THE SAME SUMMARY REQUESTED (a) BY INCLUSION (b) BY EXCLUSION BOTH REQUESTED TO BE MAILED

You wish to learn the number of contacts made by probation officers with Mexican American boys in your Account.

F.1.a. SUMMARY BY INCLUSION:
Item 64 (number contacts) Values 1 2 3 4 (all Values) included
Item 13 (ethnic group) Value 1 (Mex-Amer.) included
Item 12 (sex) Value 1 (male) included
(Note computer's confirmation of the acceptance of the mailing order)

F.1.b. SUMMARY BY EXCLUSION:
Item 13 (ethnic group) Values 2 3 (black & others) excluded
Item 12 (sex) Value 2 (female) excluded
(Note that Item 64 (contacts) is not defined since all values are requested)

F.1.c. COMPUTER RESPONSE:
The summary (identical for both above methods) will be mailed to the user in this form.

F.2. TO HAVE A TYPED MESSAGE DISPLAYED AT COMPUTER CENTER

MESSage (text)	Type MESsage and the text of the message you wish to send Press return key

COMPUTER RESPONSE

a. -message have not received summaries requested mailed on may 5
MESSAGE SENT

b. FROM USER 1000,LINE 1 AT 1132

'MESSAGE HAVE NOT RECEIVED SUMMARIES REQUESTED MAILED ON MAY 5

a. The computer confirms acceptance of your message on your terminal.

b. Actual message as displayed at computer center.

G. ENDING CONTACT WITH THE COMPUTER

G.1. TO SIGN OFF

OFF

Type OFF) or
DONe) or
END) or
FINished)

Press return key

Switch your typewriter OFF

THE TELEPHONE IS NOW AUTOMATICALLY DISCONNECTED.

THE LINE IS FREE FOR SOMEONE ELSE TO USE.

COMPUTER RESPONSE

-off

OFF AT 1359

13.59 = The time of day when computer contact ended.

INDEX

(References are to page numbers)

END