

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

NASA Technical Memorandum 83328

(NASA-TM-83328) BASIC DATA MANIPULATION AND
DISPLAY SYSTEM (BDMADS) (NASA) 61 P
HC A04/MF A01 CSCL 09B

N83-19496

Unclas

G3/61 02956

BASIC Data Manipulation and Display System (BDMADS)

John R. Szuch
Lewis Research Center
Cleveland, Ohio



February 1983

NASA

CONTENTS

SUMMARY.....	1
INTRODUCTION.....	1
JET ENGINE APPLICATION.....	3
USING BDMADS.....	4
Cataloging Programs.....	4
Creating a Measurements File.....	5
Calculating Variables.....	7
Printing Out Results.....	8
Generating a DIF File for VisiCalc.....	9
JET ENGINE CALCULATION SUBROUTINE.....	9
CONCLUDING REMARKS.....	10
APPENDIXES.....	11
A - Source Listings of BDMADS Programs.....	11
B - Applesoft II - Extended Floating Point BASIC.....	40
C - Use of VisiCalc with BDMADS.....	44
REFERENCES.....	46

BASIC DATA MANIPULATION AND DISPLAY SYSTEM (BDMADS)

John R. Szuch

National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

SUMMARY

E-1575

BDMADS, a BASIC Data Manipulation and Display System, is a collection of software programs that run on an Apple II Plus personal computer. BDMADS provides a "user-friendly" environment for the engineer in which to perform scientific data processing. BDMADS permits a user to (1) type in sets of measurements at the keyboard using BDMADS' database management program, (2) store the measurements in a user-named disk file, (3) interactively manipulate (edit, search, sort, etc.) the data using the facilities of the database management program, (4) incorporate a user-defined calculation subroutine into the BDMADS calculation program, (5) perform all of the calculations on any or all of the sets of measurements, (6) obtain hardcopy printouts of measurements and calculated variables, and (7) interactively change the calculation equations and immediately obtain new results without recompiling. An additional feature of BDMADS is its ability to store results in disk files that are compatible with VisiCalc, the popular electronic spreadsheet program. This report is intended as a guide to users of the BDMADS software. The report describes the computer programs and their use. Jet engine performance calculations are used to illustrate the use of BDMADS. Source listings of the BDMADS programs are provided and should permit users to customize the programs for their particular applications.

INTRODUCTION

Engineers and scientists often face the task of performing a large number of complex operations (i.e. calculations) on an array of numbers. In general, these numbers (measurements) have to be processed to obtain another (perhaps larger) array of numbers (variables). Depending on the complexity of the required calculations and the multiplicity of data sets to be processed, the engineer may choose any of a number of approaches to getting the job done. If relatively few calculations are involved, he may opt to do the job with pencil and paper and a hand-held calculator. However, with this approach, he runs the risk of (1) making errors in the calculations, (2) inadequately documenting the process and the results, and (3) discovering, after the job is complete, that the measurements have been revised or the required calculations have been modified or expanded. Invariably, the manual calculations have to be repeated. If the calculations appear to be too complex for hand processing (e.g. iterations and table lookups), the engineer can, of course, write

ORIGINAL PAGE IS
OF POOR QUALITY

a computer program to do the processing. Often, one will write a program in a compiled language such as Fortran and run the program on a large mainframe computer. Unfortunately, despite the tremendous computing power available in a mainframe, this approach often is too time consuming because of the time spent (1) formatting the measurements, (2) writing, debugging, and compiling the program, and (3) waiting for the results to come back (turnaround time).

Recent advancements in personal computers, such as the Apple II Plus (ref. 1), have now made it possible for the engineer to access significant, scientific computing power at a very low cost and to have interactive, dedicated use of that computing power at his desk. Most of the popular personal computers have some form of interpretive BASIC as their programming language. This permits the engineer to make changes in his program and to immediately see the effects of those changes. Since most personal computer systems include a low-cost dot-matrix printer, documented, report-quality results can be obtained without delay. For these reasons, the personal computer is well suited for the scientific data processing task.

With this in mind, it was decided to design and develop a software package that would run on an Apple II Plus personal computer system and that would provide a "user-friendly" environment in which the engineer could perform scientific data processing. That software package is called BDMADS, for BASIC Data Manipulation and Display System. As the name implies, BDMADS is written, for the most part, in Applesoft BASIC (ref. 2). Two associated subroutines are written in 6502 machine language to speed up the processing.

BDMADS permits a user to (1) type in sets of measurements at the keyboard using BDMADS' database management program, (2) store the measurements in a user-named disk file, (3) interactively manipulate (edit, search, sort, etc.) the data using the facilities of the database management program, (4) incorporate a user-defined calculation subroutine (that must be programmed in Applesoft BASIC) into the BDMADS calculation program, (5) perform all of the calculations on any or all of the sets of measurements, (6) obtain hardcopy printouts of measurements and calculated variables, and (7) interactively change the calculation equations and immediately obtain new results without compiling. All of this can be accomplished with a minimum of programming by the user.

An additional feature of BDMADS is its ability to store results in disk files that are compatible with VisiCalc, the popular electronic spreadsheet program (ref. 3). This allows a user to transmit his measurement and variable data to VisiCalc and to use the power of VisiCalc to further manipulate and format the data for reports, etc. In a sense, BDMADS is intended to be a supplement to VisiCalc, providing the capabilities to do many of the operations that VisiCalc can't do (e.g. iteration and interpolation).

ORIGINAL PAGE IS OF POOR QUALITY

This report is intended to serve as a users' guide to BDMADS. A particular application (the calculation of jet engine aerothermodynamic characteristics based on engine test data) is described and the steps required to formulate both the measurements file and the application-specific calculation subroutine are detailed. The use of BDMADS is illustrated by leading the reader through a terminal session in which the measurement file is created and stored on disk, the BDMADS calculation program is run, a hardcopy print-out is generated, and a VisiCalc file is produced.

Since the user of BDMADS will have to create his own calculation subroutine, some rudimentary knowledge of Applesoft BASIC is required. An appendix to this report points out some of the important features of Applesoft BASIC. Hopefully, this will help a user to begin programming his own application subroutine without the need for further training. It is expected that more experienced BASIC programmers will eventually want to customize the BDMADS package for their particular needs.

JET ENGINE APPLICATION

The NASA Lewis Research Center makes extensive use of computer simulations of jet engines for studying engine dynamics and for evaluating proposed engine control schemes. In the course of developing a simulation model (i.e. mathematical representation) of a helicopter turboshaft engine, it became necessary to analyze the individual component models that made up the simulation. The objective was to see how well the models matched available engine test data. For the particular engine being studied, test measurements of twelve engine parameters were available at each of six different operating points (readings). Since the engine model could not be completely defined by only twelve measurements, it was necessary to adopt a modeling philosophy in which certain components were assumed to be correctly modeled and the remaining component models would be adjusted to satisfy the twelve measurements. The analysis required the calculation of thirty-six model parameters (variables) at each operating point. The following section will describe how BDMADS was used to perform those calculations.

Figure 1 shows a flow diagram of the turboshaft engine model. Table I lists the twelve measurements that were available from the engine tests. Table II lists the thirty-six variables that had to be calculated. The equations that had to be solved for each reading are listed in Table III. It is important to note that the solution of the set of equations required (1) a subroutine (PROCOM) to compute specific enthalpy of a gaseous fuel-air mixture from supplied values of gas temperature and fuel-air ratio, (2) a table lookup routine (FUN1) to interpolate tables of univariate function data, (3) the ability to effectively go into the PROCOM subroutine backwards (i.e. compute temperature from enthalpy and fuel-air ratio), and (4) an iterative loop to solve for turbine inlet pressure, given values of turbine exit pressure, turbine inlet temperature, turbine flow

ORIGINAL PAGE IS
OF POOR QUALITY

rate, and the turbine flow characteristic. The required subroutines were coded in Applesoft BASIC and are included in the BDMADS package.

USING BDMADS

Cataloging Programs

The easiest way of seeing what BDMADS is and what it can do is to start using the programs. The BDMADS software is stored on a 5.25 inch floppy diskette that has been formatted (initialized) for use with the Apple DOS 3.3 operating system (ref. 4). With the computer power "off", the user places the diskette in the disk drive and turns "on" the computer. The DOS, which resides on the BDMADS diskette, is "booted" into the computer memory and a greeting program HELLO is loaded and executed. HELLO presents a catalog of the disk contents on the monitor screen. Figure 2 shows what the BDMADS catalog should look like. The catalog indicates that there are 332 free sectors on the diskette. The programs/files listed in Figure 2 consume 164 out of a total of 496 sectors on the diskette. Each item listed in the catalog is of the form:

(* or no *) (A, B, or T) (N) (Name)

The presence of an asterisk denotes a locked (read-only) program or file. An "A" denotes an Applesoft BASIC program. A "B" denotes a binary (machine language) program or subroutine. A "T" denotes a text file (either random-access or sequential). The number, N represents the number of sectors used to store the program/file on the diskette.

The key programs in BDMADS are BDMADS.MAIN, BDMADS.CALC, and BDMADS.DBM. BDMADS.MAIN is, as its name implies, the main program that the user runs. It will, as needed, call BDMADS.DBM to perform database management functions or BDMADS.CALC to actually perform the user-defined calculations and obtain hard-copy print-outs of the results. BDMADS.DBM includes a "search" mode in which the user can find records (readings) in the measurement file by specifying a key word(s). To speed up the search, BDMADS makes use of the binary SEARCH routine (ref. 5). To compensate for the limited PRINT formatting facilities in Applesoft BASIC, the BDMADS.CALC program makes use of the binary PRINTUSR subroutine (ref. 6) to allow the user to more conveniently format the printer output. In general, the BDMADS.DBM program will produce two text files for storing the measurements. For the jet engine example, the measurements file is called ENGL.RAF (ENGL is the engine designation and RAF stands for random-access file). This file is used to store the actual measurement data. A second file is used to store information regarding the structure of the measurements file. In general, the structure file has the same name as the measurement file - appended with a control character (CNTRL-B) that doesn't show up in the catalog display. The structure file is locked. If the user decides, at some time, to change the structure (number of

measurements, name of measurements, maximum field width of the measurements), he can make use of a supplied, menu-driven FILE UTILITY program (ref. 7) to make those changes. Actual changes to values of the measurements, or addition of more readings, are made through the editing facilities of BDMADS.DBM. Finally, if the user opts to generate a VisiCalc file of the results, the file will be constructed by BDMADS.CALC and stored on the diskette. For the jet engine example, the file is called ENG1.DIF. The DIF stands for Data Interchange Format (ref. 8). The DIF was developed by Software Arts Inc. to allow VisiCalc files to be used with a variety of programs. Source listings for BDMADS.MAIN, BDMADS.CALC, BDMADS.DBM, and FILE UTILITY are provided in Appendix A.

Creating a Measurements File

The BDMADS.MAIN program is loaded and executed by issuing the RUN BDMADS.MAIN command. A title page is displayed on the screen and the user is prompted to "PRESS RETURN TO CONTINUE". After pressing the RETURN key, the user is presented with the following options:

1. CREATE/EDIT DATA FILE
2. PERFORM CALCULATIONS
3. CHANGE DATA FILE STRUCTURE
4. QUIT

To create a measurements file, the user selects option 1 followed by a RETURN. The program then reminds the user to "INSERT BDMADS.DBM AND DATA DISK IN DRIVE" and "PRESS RETURN TO CONTINUE". If all of the programs and data are stored on a single diskette (as in our example), the user merely presses RETURN. The BDMADS.DBM program is automatically loaded into memory and executed.

BDMADS.DBM also displays a title screen and prompts the user to "PRESS RETURN FOR MENU". The following menu of options is displayed and the user is asked to "ENTER CHOICE (BY NUMBER):".

1. INITIALIZE NEW FILE
2. DISPLAY FILE STRUCTURE
3. ENTER DATA AS RECORDS
4. DISPLAY/PRINT RECORDS
5. CHANGE/DELETE RECORDS
6. FIND DATA IN FILE
7. COMPUTE FIELD SUBTOTALS
8. PRINT REPORT
9. PRINT MAIL LABELS
10. END PROGRAM

It should be noted that BDMADS.DBM is a full-featured, database management program that is based on the A.I.M. program (ref. 9). As such, some of the options (e.g. 7, 8 and 9) probably won't be useful in the data processing application.

To initialize a new measurements file, the user selects option 1 and presses RETURN. If the disk drive is not in the default slot #6 or if drive #1 is not being used, the user is given the opportunity to specify the slot and drive numbers. The user is then asked to "ENTER NAME OF FILE". For the jet engine example, the file name, ENGI.RAF was entered. A screen form is displayed to the user on which the names of the fields (measurements) and the maximum field lengths can be specified. When prompted, the user types in the name for each field (e.g. WA2CX) and the length of each field (e.g. 7). The field length must be large enough to contain all of the digits and the decimal point. Table IV lists the measurements data for the jet engine example. The first field is reserved for a "key" that can be used for labeling each record (reading). For the jet engine example, the first field was named READING NO.. When the file structure definitions are complete, the user presses RETURN and the program returns to the menu. At this point, the user can select option 2 to look at the structure he's created. Again, he'll be asked to "ENTER NAME OF FILE". After typing ENGI.RAF and a RETURN, the user will be presented with the file structure, including the fixed record length computed by BDMADS.DBM from the specified field lengths. When the user presses RETURN, he'll again be returned to the menu.

To enter data into the measurements file, the user selects option 3 and types in the file name when prompted to do so. A screen form is displayed that shows the current record (reading) number. If this is a new file, the first record to be entered will be record number 1. BDMADS.DBM keeps track of how many records there are in the file. The names of the pre-defined fields are displayed with brackets indicating the allowable length of each entry. After typing in the values for all of the fields, the user will be given an opportunity to correct any mistakes in the entries. If the user responds to a request for data for the first field with a RETURN, the program assumes that the data entry is complete. By again pressing RETURN, the user is returned to the menu. At this point, the measurements file and its associated structure file will be stored on the diskette. By selecting option 10, the user can exit from BDMADS.DBM and have BDMADS.MAIN reloaded and reexecuted.

If, after creating a measurements file, the user decides to change the file structure (i.e. add a field, rename a field, or change a field length), he can do this by selecting option 3 in BDMADS.MAIN. This loads and executes the FILE UTILITY program. The user is prompted to enter the "NAME OF FILE TO RESTRUCTURE". If the disk drive/slot are different from the default positions, the user will have an opportunity to specify them.

The field names and field lengths for the file are displayed. The program instructs the user to use the ESC and RETURN keys to indicate where changes in the file structure are desired. Where changes are indicated, the user types in the new specification. To prevent unwanted changes in the data values, any new field lengths must be large enough to accommodate the data presently in the fields. After editing the existing file

structure, the user is asked "DO YOU WISH TO ADD A FIELD? Y/N". If the user responds by pressing Y, he will be asked to enter "FIELD NAME" and "LENGTH". The program then proceeds to restructure the data file and update the structure file. The user is informed when "NEW FILES ARE READY FOR USE" and reminded to "DON'T FORGET TO MAKE A BACKUP" of the data file. The BDMADS.MAIN program is then reloaded and reexecuted.

Calculating Variables

To perform a pre-defined set of calculations using the data in the measurements file, the user selects option 2 in BDMADS.MAIN. The user is then prompted to "INSERT BDMADS.CALC AND DATA DISK IN DRIVE" and "PRESS RETURN TO CONTINUE". Again, if all files are on one disk, the user merely has to press RETURN. The BDMADS.CALC program is then automatically loaded and executed. A title screen is displayed and the user is again prompted to "PRESS RETURN FOR CATALOG". After the catalog is displayed, the user is prompted to "ENTER NAME OF INPUT DATA FILE". For the jet engine example, the file name ENGI.RAF was entered. The file structure was then read off of the corresponding file and the following message was displayed:

THERE ARE 6 RECORDS IN ENGI.RAF

OPTIONS ARE: 1. CALC FOR ALL RECS.
2. CALC FOR BLOCK OF RECS.
3. CALC FOR SINGLE REC.
4. QUIT

ENTER CHOICE:

As the message indicates, the user has the option of performing the calculations for all of the records (readings), a contiguous block of records (e.g. 2 through 5), a single record, or none. If option 2 is selected, the user is prompted to "ENTER START REC. NO.:" and then to "ENTER END REC. NO.:". If option 3 is selected, the user is prompted to "ENTER REC. NO.:". After the records to be processed are specified, the corresponding measurements data are read from the file. The first field in each record (the key) is stored as an element in a string array, KY\$(I). The measurements are stored as elements in a two-dimensional array, ME(I,J) where I denotes the record and J denotes the field. After the data are read from the disk, the messages "INPUT DATA ARE IN MEMORY" and "CALCULATIONS NOW BEGIN" are displayed on the screen. For each specified record, a CALL is made to the user-defined calculation subroutine. As currently structured, BDMADS.CALC requires the user-supplied routine to start at statement 3000.

The JET ENGINE CALCULATION SUBROUTINE section describes the makeup of the user-supplied subroutine. In general, that subroutine is written in terms of the elements of the measurements array, ME(I,J) and the elements of a variables array, VA(I,K). The number of variables to be calculated for

each reading, K is user-defined. When all of the calculations are completed, the messages "CALCULATIONS DONE" and "WANT TO SEE RESULTS? Y/N" will appear on the screen.

Printing Out Results

If the user indicates that he wants to see the results of the calculations, he will be asked "WANT HARD-COPY? Y/N:". If the user answers "Y", he will be prompted to "ALIGN PAPER. TURN ON PRINTER". The turning on of most printers will set the top-of-form which can be utilized for multi-page printouts. Statements 1040-1290 in BDMADS.CALC control the printing out of results. As currently written, the program makes use of printer control commands supported by the Epson MX100 dot-matrix printer (ref. 10) and the Orange Micro GRAPPLER interface card (ref. 11). Other combinations of printer and interface card may require some modifications of the program. As written, the program asks the user the questions "WANT EMPHASIZED PRINT? Y/N:" and "WANT TITLE? Y/N:". If a title is requested, the user will be asked to "ENTER TITLE: " and "PRESS RETURN TO CONTINUE:".

Figure 3 shows a portion of the printout for the jet engine example. Note the ENGI MODEL VERIFICATION title and the structure of the printout as defined by statements 1040-1290. The first field (key) in each record (named READING NO. in the example) is printed, followed by a blank line. The remaining measurements are printed two-to-a-line with their names followed by their values. Another blank line is printed, followed by the calculated variables (names and values), again two-to-a-line. Logic is set up in the print control statements to form-feed after printing two records. The readability of the results is improved by having BDMADS.CALC call the machine language routine PRINTUSR. PRINTUSR allows the user to specify the field width and the number of decimal places when printing out the data. For example, PRINT USR(X)'11,5' will print the numerical value of X with a print field of 11 with 5 decimal places. Customization of the printout may be facilitated by noting the Applesoft BASIC I/O commands outlined in Appendix B.

If the user does not want a hard-copy printout, the results will be displayed on the video screen. For each reading to be listed, the key and the measurements are listed followed by a prompt to "PRESS SPACE BAR TO CONTINUE LIST". The user presses the space bar to display each calculated variable with the list scrolling off of the top of the screen. It is recommended that the user first list out results on the screen before obtaining a hard-copy. Occasionally, PRINTUSR will not print correctly the first time it is called. The problem is corrected by issuing the RUN command to rerun BDMADS.CALC.

When the listing of data is complete or when the hard-copy printout is finished, the user is asked "WANT RESULTS IN DIF FILE? Y/N:". The following section describes the generation of the DIF file. If a DIF file is not desired, the user is prompted to "INSERT MAIN DISK IN DRIVE" and "PRESS RETURN TO CONTINUE".

The BDMADS.MAIN program will be reloaded and reexecuted automatically.

Generating a DIF File for VisiCalc

If the user opts to generate a Data Interchange Format (DIF) file containing the measurements and variables, a catalog will be displayed on the screen and the user will be asked to "ENTER NAME OF DIF FILE:". For the jet engine example, the name ENGI.DIF was typed in. As the DIF file is constructed by BDMADS.CALC, its contents are displayed on the screen. When the file is finished, the messages "DIF FILE COMPLETE", "INSERT MAIN DISK IN DRIVE", and "PRESS RETURN TO CONTINUE" are displayed. The BDMADS.MAIN program is then reloaded and reexecuted. Appendix C illustrates the use of VisiCalc to read the DIF file and to reformat the data.

JET ENGINE CALCULATION SUBROUTINE

A user of BDMADS must provide his own calculation subroutine to process the data in the measurements file. The subroutine will be called for each requested reading. The subroutine must satisfy the following constraints: (1) the subroutine must start at statement 3000, (2) the subroutine must be written in Applesoft BASIC, (3) the subroutine must be written in terms of measurements ME(I,J) and variables VA(I,K), (4) symbols for all other variables used in the subroutine must not conflict with symbols used in the fixed portion of BDMADS.CALC, (5) the subroutine must end with a RETURN statement. A listing of BDMADS.CALC with the jet engine calculation subroutine is provided in Appendix A. Also provided is a cross-reference of symbols used in the program. The following paragraphs point out some of the salient features of the jet engine subroutine.

Several remarks, denoted by REM, are contained in the subroutine to enhance the readability of the code. An integer ID is used to branch around the statements 3050-3400 for all calls to the subroutine except the first. This avoids redimensioning of the arrays which is a "no-no" in interpretive BASIC. An array of nine constants is defined by statements 3050-3140. These constants correspond to the constants C contained in the equations listed in Table III. The ninth constant is not used in the calculations but was to be compared with a calculated variable, VA(I,36). The jet engine equations contain references to four table lookups (i.e. $Y = \text{FUN1}(N, X)$ where N is a curve number). The data for the four curves are contained in statements 3170-3290. The actual reading in and storing of the curve data is accomplished in statements 3300-3330.

The jet engine calculations are performed in statements 3360-3750. The number of variables to be computed is defined in statement 3360. To aid in the reading of statements, remarks are provided in statements 3370-3400 that relate ME array elements to actual measurement names. Following each statement in which a

variable is computed, a corresponding string variable is defined. This provides a label that is used when printing out the results. Table V defines the elements of the ME(I,J) and VA(I,K) arrays, in terms of their labels, for the jet engine example.

Statements 3470-3490 and 3620 contain calls to the PROCOM subroutine (GOSUB 160). Inputs T, FA, and PI are defined prior to each call. T represents the temperature in deg.R, FA represents the fuel-air ratio, and PI=0 indicates that the normalcalling mode is in effect. The output of the subroutine, H is assigned to the appropriate variable following the RETURN from the subroutine. Statements 3640 and 3700 contain inverse calls to PROCOM (i.e. enthalpy input). The inverse call (GOSUB 440) is denoted by PI=1. Statements 3500-3520 and 3740 contain calls to the FUN1 table lookup subroutine (GOSUB 500). The inputs to FUN1 are the curve number, NC and the input variable, XI. The output of the subroutine, YC is assigned to the appropriate variable following the RETURN from the subroutine. Statement 3740 also contains a call to the turbine iteration subroutine (GOSUB 600). The inputs to the subroutine are the squared critical speed ratio, TH, the flow rate, W, the back pressure, PB, and the curve number for the corrected flow-pressure ratio curve. The outputs of the subroutine are the turbine inlet pressure, P and the corrected flow, WC. Statement 3760 returns to the BDMADS.CALC program.

CONCLUDING REMARKS

The emergence of low-cost, powerful personal computers such as the Apple II Plus now makes it possible for engineers and scientists to interactively perform arithmetic data processing at their desks. Also, the development of software packages such as VisiCalc have made it possible for the non-programmer to interactively create, on the video screen, large, dynamic tables of equations and data and to view the effects of changes in any of the entries in the tables. The BDMADS package, described in this paper, was intended to compensate for some of the observed shortcomings of VisiCalc (e.g. lack of iteration and interpolation capabilities). The resultant package can be used alone or in conjunction with VisiCalc to provide a uniquely powerful data processing environment on the Apple II Plus computer system. While it is expected that commercial software packages will appear in the future that satisfy many of the needs of scientists and engineers, BDMADS, today, provides a flexible software system in which users can not only address their current data processing requirements but also have the ability to customize the package to meet their future needs.

ORIGINAL PAGE IS
OF POOR QUALITY

APPENDIX A - SOURCE LISTINGS OF BDMADS PROGRAMS

BDMADS.MAIN PROGRAM

```
10 REM BDMADS - MAIN PROGRAM
20 HOME : GOSUB 160: FLASH
30 VTAB 3: HTAB 17: PRINT "BDMADS": VTAB 4: HTAB 18: PRINT
  "MAIN": VTAB 5: HTAB 16: PRINT "PROGRAM": NORMAL
40 VTAB 20: HTAB 14: PRINT "BY J.R. SZUCH": VTAB 21: HTAB 1
  6: PRINT "OCT. 1982"
50 VTAB 22: HTAB 8: INVERSE : PRINT "PRESS RETURN TO CONTIN
  UE";: GET R$: PRINT R$: NORMAL
60 D$ = CHR$ (4): HOME
70 PRINT "OPTIONS ARE: 1. CREATE/EDIT DATA FILE": HTAB 14:
  PRINT "2. PERFORM CALCULATIONS ": HTAB 14: PRINT "3. CHANGE
  DATA FILE": HTAB 17: PRINT "STRUCTURE": HTAB 14: PRINT "4. Q
  UIT"
80 PRINT
90 INPUT "ENTER CHOICE: ";IC: HOME
100 ON IC GOTO 110,130,150,180: GOTO 80
110 PRINT "INSERT BDMADS.DBM AND DATA DISK IN DRIVE"
120 PRINT "PRESS RETURN TO CONTINUE";: GET R$: PRINT R$: PR
  INT D$"RUN BDMADS.DBM"
130 PRINT "INSERT BDMADS.CALC AND DATA DISK IN DRIVE"
140 PRINT "PRESS RETURN TO CONTINUE";: GET R$: PRINT R$: HO
  ME : PRINT D$"RUN BDMADS.CALC"
150 PRINT "INSERT FILE UTILITY DISK IN DRIVE": PRINT "PRESS
  RETURN TO CONTINUE";: GET R$: PRINT R$: PRINT D$"RUN FILE U
  TILITY"
160 VTAB 23: FOR K = 1 TO 39: PRINT "*";: NEXT K: PRINT : V
  TAB 1: FOR K = 1 TO 39: PRINT "*";: NEXT K: PRINT
170 FOR K = 2 TO 22: VTAB K: PRINT "*";: HTAB 39: PRINT "*"
  : NEXT K: RETURN
180 END
```

```

10 PRINT CHR$(4)"BLOAD SEARCH,A#0302"
20 POKE 1013,76: POKE 1014,2: POKE 1015,3
30 POKE 768,0: POKE 769,0
40 REM PLACE TITLE ON SCREEN
50 TEXT : HOME : GOSUB 290: FLASH
60 VTAB 3: HTAB 17: PRINT "BDMADS": VTAB 4: HTAB 16: PRINT
  "DATABASE": VTAB 5: HTAB 15: PRINT "MANAGEMENT": VTAB 6: HTA
  B 16: PRINT "PROGRAM": NORMAL
70 VTAB 20: HTAB 14: PRINT "BY J.R. SZUCH": VTAB 21: HTAB 1
  6: PRINT "OCT. 1982"
80 VTAB 22: HTAB 8: INVERSE : PRINT "PRESS RETURN FOR MENU"
  ;: GET R$: PRINT R$: NORMAL
90 HOME
100 GOTO 400
110 VTAB 23: CALL - 868: PRINT "CORRECT (Y/N) ";G$;: GET Y
  $: PRINT Y$: RETURN
120 PRINT "ENTER ";: INVERSE :CV = PEEK (37): GOSUB 130:P$
  (K) = Q$: RETURN
130 PRINT T$(K);: NORMAL : PRINT ": ";G$;: PRINT V1$; TAB( T
  V(K) + 3);V2$: VTAB (CV + 2): HTAB 2
140 IF K = 1 AND T$(1) = "RECORD NO." THEN Q$ = STR$(REC)
  + ".": GOTO 160
150 INPUT Q$
160 VTAB (CV + 2): HTAB 1: FOR I = 1 TO TV(K) + 2: PRINT "
  ";: NEXT I: PRINT " ": VTAB (CV + 1): HTAB ( LEN (T$(K)) + 9
  ): IF K = NF THEN CALL - 958
170 PRINT Q$: RETURN
180 PRINT D$"OPEN ";FILE$",L";RL
190 PRINT D$"WRITE";FILE$",R";RX
200 RETURN
210 PRINT "PRESS ";: INVERSE : PRINT "RETURN";: NORMAL : PR
  INT " TO QUIT ";: RETURN
220 VTAB 2: HTAB (20 - LEN (AA$) / 2): INVERSE : PRINT AA$
  : NORMAL
230 VTAB 4: HTAB 2: PRINT B$: VTAB 4: INVERSE : HTAB (20 -
  LEN (A1$) / 2): PRINT A1$: POKE 34,4: NORMAL : RETURN
240 PRINT D$"OPEN";FILE$",L";RL
250 PRINT D$"READ";FILE$",R";RX
260 RETURN
270 HOME : PRINT : FOR K = 1 TO NF: PRINT K;". ";T$(K);" -
  ";P$(K): NEXT K: RETURN
280 HOME : VTAB 7: PRINT : PRINT "ENTER IN CORRECT DATA:";G
  $;: INPUT P$(Z): RETURN
290 VTAB 23: FOR K = 1 TO 39: PRINT "*";: NEXT K: PRINT : V
  TAB 1: FOR K = 1 TO 39: PRINT "*";: NEXT K: PRINT
300 FOR K = 2 TO 22: VTAB K: PRINT "*";: HTAB 39: PRINT "*"
  : NEXT K: RETURN
310 HOME : PRINT D$"CATALOG,S"SX;","D"DX
320 PRINT : PRINT "ENTER FILE NAME: ";G$;: INPUT FILE$: IF
  LEN (FILE$) = 0 THEN CLEAR : GOTO 400
330 PRINT D$"OPEN";FILE$ + BC$
340 PRINT D$"READ";FILE$ + BC$
350 INPUT NF: INPUT RL
360 IF DM = 0 AND FM = 0 THEN DIM T$(2 * NF),TV(NF),P$(NF)
370 FOR K = 1 TO NF: INPUT T$(K): INPUT TV(K): NEXT K

```

```

380 PRINT D$ "CLOSE" FILE$ + BC$
390 RETURN
400 REM *** SUBROUTINES ABOVE
410 REM MENU
420 V1 = 10: DIM M$(V1): B$ = "
": REM 30 SPACES
430 BC$ = CHR$(2): D$ = CHR$(4): G$ = CHR$(7): H$ = CHR$(
(B): V1$ = CHR$(91): V2$ = CHR$(93)
440 TEXT: HOME: GOSUB 290
450 AA$ = "BDMADS.DBM": A1$ = "MENU": GOSUB 220: RESTORE
460 FOR K = 1 TO V1: READ M$(K): NEXT K
470 DATA "INITIALIZE NEW FILE", "DISPLAY FILE STRUCTURE", "E
NTER DATA AS RECORDS", "DISPLAY/PRINT RECORDS", "CHANGE/DELETE
RECORDS"
480 DATA "FIND DATA IN FILE", "COMPUTE FIELD SUBTOTALS", "PR
INT REPORT", "PRINT MAIL LABELS", "END PROGRAM"
490 FOR K = 1 TO V1: VTAB (K + 5): HTAB 8: PRINT K; ". "; M$(
K): NEXT K
500 VTAB 22: HTAB 4: PRINT "ENTER CHOICE (BY NUMBER): "; G$; A
H$: H$: NORMAL: INPUT Y$: Y = VAL (Y$): IF Y < 1 OR Y > V1
THEN 500
510 VTAB (Y + 5): HTAB 8: INVERSE: PRINT Y; ". "; M$(Y): NOR
MAL: FOR KK = 1 TO 400: NEXT KK
520 IF Y = 10 THEN 4400
530 SX = 6: DX = 1: POKE 34, 0: HOME: GOTO 580
540 TEXT: HOME: VTAB 6: CALL - 958: PRINT "ENTER SLOT #
FOR DATA DISKETTE: "; G$: GET SX$: PRINT SX$: SX = VAL (SX$)
550 IF SX < 1 OR SX > 7 THEN 540
560 VTAB 8: CALL - 868: PRINT "ENTER DRIVE # FOR DATA DISK
ETTE: "; G$: GET DX$: PRINT DX$: DX = VAL (DX$)
570 IF (DX - 1) * (DX - 2) < > 0 THEN 560
580 VTAB 10: PRINT "SLOT # = "; SX: PRINT "DRIVE # = "; DX: G
OSUB 110: IF Y$ = "N" THEN 540
590 IF Y$ < > "Y" THEN 540
600 POKE 768, SX: POKE 769, DX
610 HOME: PRINT "LOAD RECORDS DISK": PRINT "HIT RETURN TO
CONTINUE": INPUT Y$: ON Y GOTO 620, 4290, 1060, 1400, 1730, 2160,
3840, 2700, 2700
620 REM INITIALIZER SUBROUTINE
630 V = 16: REM V = NO. OF FIELD TITLES, EVEN NO.
640 DIM T$(V), T(2 * V), TV(2 * V), S1(V), S2(V), S3(V), S4(V), W(
V)
650 DM = 1
660 FOR K = 1 TO V: T$(K) = "": NEXT K: T$(0) = H$ + H$ + H$
+ LEFT$(B$, 4)
670 TEXT: HOME: PRINT "ENTER NAME OF FILE: "; G$: INPUT F
ILE$
680 HOME: INVERSE: HTAB 14: PRINT "INITIALIZER"
690 HTAB (20 - LEN (FILE$) / 2): PRINT FILE$: POKE 34, 3: N
ORMAL
700 HOME: PRINT "YOU WILL BE ENTERING IN DATA IN AN": PRIN
T "ORDERED FORMAT CONSISTING OF FIELDS.": PRINT "THE AVAILAB
LE NUMBER OF FIELDS IS "; V; "."
710 N = 0
720 VTAB 8

```



```

730 FOR K = 1 TO V / 2: PRINT K; ". "; T$(K); : HTAB (24): PR
INT K + V / 2". "; T$(K + V / 2): NEXT K: PRINT : PRINT
740 N = N + 1
750 VTAB 19: CALL - 958: PRINT "ENTER NAME FOR FIELD #"; N:
VTAB 20: PRINT "FIELD #1 MAY BE NAMED RECORD NO."; G$: VTAB
23: HTAB 10: GOSUB 210: VTAB 19: HTAB 25: INPUT " ": Z$: IF
N = 1 AND Z$ = "" THEN CLEAR : GOTO 400
760 IF Z$ = "" THEN 790
770 T(N) = N
780 T$(N) = Z$: GOTO 720
790 HOME : N = N - 1
800 IF INT (N / 2) = N / 2 THEN N2% = N / 2
810 IF INT (N / 2) < > N / 2 THEN N2% = N / 2 + 1
820 FOR K = 1 TO N2%
830 PRINT K; ". "; T$(T(K)); : HTAB (21): PRINT K + N2%; ". "T$
(T(K + N2%)): NEXT K
840 VTAB 22: CALL - 868: PRINT "CORRECT (Y/N) "; G$: GET Y
$: PRINT Y$: IF Y$ = "N" THEN 700
850 IF Y$ < > "Y" THEN 840
860 HOME : VTAB 4: PRINT "ENTER LENGTH VALUES FOR EACH FIEL
D:"
870 FOR K = 1 TO N
880 VTAB 6: CALL - 958: PRINT "FOR FIELD # "; K; " "; INVER
SE : PRINT T$(T(K)); G$: NORMAL : INPUT " "; TV(K): NEXT K
890 TEXT : HOME : INVERSE : HTAB (20 - LEN (A$) / 2): PRIN
T A$: PRINT : PRINT : POKE 34,3: NORMAL
900 RL = 0: PRINT "# FLD NAME FLD LENGTH"
910 PRINT "- - - - -"
920 FOR K = 1 TO N
930 PRINT K; ". "; : HTAB 5: PRINT T$(T(K)); : HTAB 25: PRINT
" "; TV(K): RL = RL + TV(K): NEXT K
940 VTAB 23: CALL - 868: PRINT "CORRECT (Y/N) "; G$: GET Y
$: PRINT Y$: IF Y$ = "N" THEN 860
950 IF Y$ < > "Y" THEN 940
960 PRINT D$"OPEN"; FILE$ + BC$; ", S"SX; ", D"DX
970 PRINT D$"WRITE"; FILE$ + BC$
980 PRINT N: PRINT RL: FOR K = 1 TO N: PRINT T$(T(K)): PRIN
T TV(K): NEXT K
990 PRINT D$"CLOSE"; FILE$ + BC$
1000 PRINT D$"LOCK"; FILE$ + BC$
1010 PRINT D$"OPEN"; FILE$, "L"; RL
1020 PRINT D$"WRITE"; FILE$, "R0"
1030 PRINT 0
1040 PRINT D$"CLOSE"
1050 CLEAR : GOTO 400
1060 REM DATA ENTRY SUBROUTINE
1070 A1$ = "DATA ENTRY ROUTINE"
1080 TEXT : HOME : GOSUB 220: HOME : GOSUB 310: RX = 0: GOSU
B 240
1090 INPUT REC: REC = REC + 1
1100 PRINT D$"CLOSE"; FILE$
1110 HOME : PRINT : PRINT "THIS WILL BE RECORD # "; REC
1120 VTAB 23: PRINT "RETURN ON "; : INVERSE : IF T$(1) = "RE
CORD NO." THEN PRINT T$(2); : GOTO 1140
1130 PRINT T$(1);
1140 NORMAL : PRINT " TO QUIT": VTAB 8

```

```

1150 FOR K = 1 TO NF
1160 GOSUB 120
1170 IF K = 2 AND T$(1) = "RECORD NO." AND LEN (P$(2)) = 0
    THEN 1290
1180 IF LEN (P$(1)) = 0 THEN 1290
1190 IF LEN (P$(K)) < = TV(K) THEN 1210
1200 PRINT "ENTRY TOO LONG";G$;G$: FOR KK = 1 TO 750: NEXT
    KK: VTAB (CV + 1): HTAB 1: CALL - 958: GOTO 1160
1210 NEXT K
1220 GOSUB 110
1230 IF Y$ = "N" THEN 1340
1240 IF Y$ < > "Y" THEN 1220
1250 RX = REC: GOSUB 180
1260 FOR KK = 1 TO NF: PRINT P$(KK): NEXT KK
1270 PRINT D$"CLOSE"
1280 REC = REC + 1: GOTO 1110
1290 REC = REC - 1:RX = 0
1300 GOSUB 180
1310 PRINT REC
1320 PRINT D$"CLOSE"
1330 CLEAR : GOTO 400
1340 GOSUB 270: PRINT NF + 1;". *ABORT*"
1350 VTAB 23: HTAB 10: GOSUB 210: PRINT : VTAB 21: PRINT "E
    NTER LINE # OF INCORRECT DATA: ";G$;: INPUT Z$:Z = VAL (Z$)
1360 IF Z$ = "" THEN 1250
1370 IF Z = NF + 1 THEN 1110
1380 IF Z < 1 OR Z > NF THEN 1340
1390 GOSUB 280: GOTO 1340
1400 REM DISPLAY/PRINT DATA
1410 A1$ = "DISPLAY/PRINT RECORDS"
1420 TEXT : HOME : GOSUB 220: HOME : GOSUB 310:RX = 0: GOSU
    B 240
1430 INPUT REC
1440 PRINT D$"CLOSE";FILE$
1450 HOME : VTAB 6: PRINT "ENTER IN PRINTER INFORMATION.":
    PRINT : PRINT "DO YOU HAVE A PRINTER (Y/N) ";G$;: GET Y$: PR
    INT Y$: IF Y$ = "N" THEN SLOT = 0: GOTO 1540
1460 IF Y$ < > "Y" THEN 1450
1470 VTAB 10: CALL - 868: PRINT "ENTER IN SLOT # ";G$;: GE
    T SL$: PRINT SL$:SLOT = VAL (SL$)
1480 IF Y$ = CHR$ (27) OR Y$ = CHR$ (32) THEN CLEAR : GO
    TO 400
1490 IF SLOT < 1 OR SLOT > 7 THEN PRINT G$;G$: GOTO 1470
1500 PRINT : PRINT "PLEASE TURN ON THE PRINTER.": PRINT "PR
    ESS ";: INVERSE : PRINT "RETURN";: NORMAL : PRINT " WHEN REA
    DY. ";G$;: INPUT Y$
1510 PRINT D$"PR#";SLOT
1520 PRINT
1530 PRINT D$"PR#0"
1540 HOME : VTAB 22: HTAB 10: GOSUB 210: VTAB 7: PRINT
1550 PRINT "VIEW(V) OR PRINT(P) DATA ";G$;: GET VP$: PRINT
    VP$: IF VP$ = CHR$ (13) THEN CLEAR : GOTO 400
1560 PRINT : CALL - 958: PRINT "ENTER RECORD # TO START WI
    TH ";G$;: INPUT START$::RX = VAL (START$)
1570 IF RX > REC THEN PRINT "ONLY ";REC;" RECORDS ON FILE!
    ";G$;G$: GOTO 1560

```

```

1580 IF RX < 1 THEN RX = 1
1590 GOSUB 240
1600 FOR K = 1 TO NF: INPUT P$(K): NEXT K
1610 IF VP$ = "P" THEN PRINT D$"PR#";SLOT: PRINT : PRINT "
RECORD # ";RX
1620 GOSUB 270
1630 PRINT D$"PR#0"
1640 PRINT D$"CLOSE";FILE$
1650 VTAB 21: INVERSE : PRINT "-->";: NORMAL : PRINT " = NE
XT ";: INVERSE : PRINT "<--";: NORMAL : PRINT " = LAST"; T
AB( 31);"RECORD #": VTAB 22: PRINT "PRESS ";: INVERSE : PRIN
T "ESC";: NORMAL : PRINT " FOR RECORD # "; TAB( 30);RX;" OF
";REC
1660 GOSUB 210: PRINT G$;: CALL - 868: GET Y$: PRINT Y$: I
F Y$ = CHR$(3) THEN END
1670 IF Y$ = CHR$(27) THEN VTAB 23: CALL - 868: PRINT "
ENTER RECORD NUMBER ";G$;: INPUT RX: GOTO 1570
1680 IF Y$ = CHR$(8) THEN RX = RX - 2: GOTO 1720
1690 IF Y$ = CHR$(13) THEN 1540
1700 IF Y$ = CHR$(21) OR Y$ = CHR$(32) THEN 1720
1710 GOTO 1650: REM ****
1720 RX = RX + 1: GOTO 1570
1730 REM CHANGE/DELETE DATA
1740 A1$ = "CHANGE/DELETE DATA"
1750 TEXT : HOME : GOSUB 220: HOME : GOSUB 310:RX = 0: GOSU
B 240
1760 INPUT REC
1770 PRINT D$"CLOSE";FILE$
1780 HOME : VTAB 23: HTAB 10: GOSUB 210: VTAB 5: PRINT : PR
INT "ENTER RECORD # TO CHANGE/DELETE ";G$;: INPUT RD$
1790 IF LEN (RD$) = 0 THEN CLEAR : GOTO 400
1800 RD = INT ( VAL (RD$)): IF RD < = 0 THEN 1780
1810 IF RD < = REC THEN 1830
1820 PRINT : PRINT "ONLY ";REC;" RECORDS ON FILE!": PRINT :
PRINT "PRESS RETURN TO CONTINUE.";G$;G$;: INPUT Y$: GOTO 17
80
1830 RX = RD: IF RX < = 0 THEN RX = 0
1840 GOSUB 240: FOR K = 1 TO NF: INPUT P$(K): NEXT K
1850 PRINT D$"CLOSE";FILE$
1860 GOSUB 270
1870 VTAB 22: CALL - 868: PRINT "CHANGE/DELETE RECORD # ";
RD;" (Y/N) ";: GET Y$: PRINT Y$: IF Y$ = "N" THEN 1780
1880 IF Y$ < > "Y" THEN 1870
1890 VTAB 22: CALL - 868: PRINT "CHANGE(C), DELETE(D), OR
ABORT(CR) ";G$;: GET Y$: PRINT Y$
1900 IF Y$ = CHR$(13) THEN 1780
1910 IF Y$ = "C" THEN 2060
1920 IF Y$ < > "D" THEN 1890
1930 VTAB 22: CALL - 868: PRINT "VERIFY DELETION (YES/NO):
";G$;: INPUT Y$: IF Y$ < > "YES" THEN 1780
1940 IF RD < > REC THEN 1980
1950 RX = 0:REC = REC - 1: GOSUB 190: PRINT REC
1960 PRINT D$;"CLOSE";FILE$
1970 GOTO 1780
1980 RX = REC: GOSUB 240: FOR K = 1 TO NF: INPUT P$(K): NEXT
K

```

```

1990 PRINT D$;"CLOSE";FILE$
2000 RX = RD: IF T$(1) = "RECORD NO." THEN P$(1) = STR$(RX
) + "."
2010 GOSUB 180: FOR K = 1 TO NF: PRINT P$(K): NEXT K
2020 PRINT D$;"CLOSE";FILE$
2030 REC = REC - 1:RX = 0: GOSUB 180: PRINT REC
2040 PRINT D$;"CLOSE";FILE$
2050 GOTO 1780
2060 REM CHANGE ROUTINE
2070 VTAB 23: HTAB 10: GOSUB 210: PRINT : VTAB 22: CALL -
868: PRINT "ENTER FIELD # TO CHANGE: ";G$;: INPUT Y$:Y = VA
L (Y$): IF Y = 0 THEN 2130
2080 IF INT (Y) < 1 OR INT (Y) > NF THEN 2070
2090 VTAB 22: CALL - 868: PRINT "ENTER: ";: INVERSE : PRIN
T G$;T$(Y);: NORMAL : INPUT P$(Y)
2100 IF LEN (P$(Y)) < = TV(Y) THEN 2120
2110 VTAB 23: CALL - 868: HTAB 10: PRINT "ENTRY TOO LONG!"
;G$;G$: FOR KK = 1 TO 500: NEXT KK: VTAB 23: HTAB 10: GOSUB
210: PRINT : GOTO 2090
2120 GOSUB 270: GOTO 2070
2130 RX = RD: GOSUB 180: FOR K = 1 TO NF: PRINT P$(K): NEXT
K
2140 PRINT D$;"CLOSE";FILE$
2150 GOTO 1780
2160 REM **FIND MODULE**
2170 A1$ = "FIND INFORMATION":B1$ = "SEARCH"
2180 TEXT : HOME : GOSUB 220: HOME : GOSUB 310:RX = 0: GOSU
B 240
2190 INPUT REC
2200 PRINT D$;"CLOSE "FILE$
2210 FY = 0
2220 HOME : PRINT : PRINT "WANT TO CREATE FILE": PRINT "WIT
H FOUND RECORDS (Y/N): ";G$;: GET Y$: PRINT Y$
2230 IF Y$ = "Y" THEN FY = 1: TEXT : HOME : PRINT "ENTER NA
ME OF NEW FILE: ";G$;: INPUT MFILE$: HOME : PRINT D$"OPEN"MF
ILE$ + BC$: PRINT D$"WRITE"MFILE$ + BC$: PRINT NF: PRINT RL:
FOR K = 1 TO NF: PRINT T$(K): PRINT TV(K): NEXT K
2240 IF Y$ = "Y" THEN PRINT D$"CLOSE"MFIL$ + BC$: PRINT D
$"LOCK"MFIL$ + BC$
2250 DIM SP$(NF),SERFLD(NF):N = 0
2260 NQ = 0
2270 HOME : PRINT : PRINT "THE FOLLOWING LINES ARE AVAILABL
E FOR THE ";B1$;":": PRINT : FOR K = 1 TO NF STEP 2: P
RINT K;". ";T$(K); TAB( 20);K + 1;". ";T$(K + 1): NEXT K
2280 FOR K = 1 TO NF: VTAB 20: HTAB 1: CALL - 958: VTAB 24
: HTAB 10: GOSUB 210
2290 NQ = NQ + 1
2300 VTAB 20: HTAB 1: CALL - 868: PRINT "ENTER CHOICE FOR
SEARCH FIELD # ";K;G$;: INPUT ": ";SERFLD$: IF SERFLD$ = ""
AND K = 1 THEN CLEAR : GOTO 400
2310 IF SERFLD$ = "" THEN NQ = NQ - 1:K = NF: GOTO 2360
2320 IF K = NF THEN NP = K
2330 SERFLD(K) = VAL (SERFLD$): IF SERFLD(K) < 1 OR SERFLD(
K) > NF THEN 2300

```

ORIGINAL PAGE IS
OF POOR QUALITY

**ORIGINAL PAGE 19
OF POOR QUALITY**

```

2340 VTAB 22: HTAB 1: CALL - 868: PRINT "ENTER SEARCH PARA
METER ";G$;: INPUT "":SP$(K):SP = LEN (SP$(K))
2350 IF SP > TV(SERFLD(K)) THEN VTAB 22: HTAB 1: CALL - 8
68: INVERSE : PRINT "SEARCH PARAMETER TOO LONG !";: NORMAL :
FOR II = 1 TO 750: NEXT II: GOTO 2340
2360 NEXT K: FOR II = 1 TO 300: NEXT II
2370 HOME : FOR K = 1 TO NQ: VTAB (2 * K + 6): HTAB 1: PRIN
T "SEARCH ";T$(SERFLD(K));" FOR ";SP$(K);: IF K < NQ THEN H
TAB ( PEEK (36) + 2): PRINT "AND";
2380 NEXT K: PRINT
2390 GOSUB 110: IF Y$ = "N" THEN 2270
2400 IF Y$ < > "Y" THEN 2390
2410 PRINT D$;"OPEN ";FILE$;","L";RL
2420 FOR K1 = 1 TO REC
2430 SRFL = 0
2440 PRINT D$;"READ ";FILE$;","R";K1
2450 FOR K = 1 TO NF: INPUT P$(K): NEXT K
2460 FOR K = J TO NQ
2470 & P$(SERFLD(K)),SP$(K)
2480 IF PEEK (26) = 0 THEN K = NQ:SRFL = 1
2490 NEXT K: IF SRFL = 1 THEN 2670
2500 PRINT D$;"CLOSE ";FILE$
2510 N = N + 1: HOME : GOSUB 270
2520 IF FY = 1 THEN VTAB 22: INVERSE : HTAB 1: PRINT "PRES
S RETURN TO SAVE FOUND RECORD": NORMAL : GET Y$: PRINT Y$: P
RINT D$"OPEN"MFIL$","L"RL: PRINT D$"WRITE"MFIL$","R": PRINT
N
2530 IF FY = 1 THEN PRINT D$"WRITE"MFIL$","R"N: FOR J = 1
TO NF: PRINT P$(J): NEXT J: PRINT D$"CLOSE"MFIL$
2540 VTAB 21: HTAB 1: CALL - 868: INVERSE : PRINT "-->";:
NORMAL : PRINT " = TO CONTINUE";: HTAB 31: PRINT "RECORD #";
: VTAB 22: HTAB 1: CALL - 868: INVERSE : PRINT "'P'";: NORM
AL : PRINT " = TO PRINT";: HTAB 31: PRINT K1;" OF ";REC;
2550 VTAB 23: HTAB 1: CALL - 868: INVERSE : PRINT "ESC";:
NORMAL : PRINT " TO QUIT";: GET Z$: PRINT Z$;: IF Z$ < > "P
" AND Z$ < > CHR$ (21) AND Z$ < > CHR$ (27) THEN 2550
2560 IF Z$ = CHR$ (21) THEN 2660
2570 IF Z$ = CHR$ (27) THEN K1 = REC: GOTO 2670
2580 HOME : IF SLOT < > 0 THEN 2620
2590 PRINT : PRINT "ENTER IN PRINTER SLOT #";G$;: INPUT SLO
T
2600 PRINT : PRINT "TURN ON PRINTER"
2610 VTAB 23: HTAB 5: INVERSE : PRINT "PRESS ANY KEY TO CON
TINUE";: NORMAL : GET Z$: PRINT Z$
2620 PRINT D$"PR#"SLOT
2630 PRINT : PRINT "RECORD #";K1: GOSUB 270
2640 PRINT D$"PR#0"
2650 GOTO 2540
2660 PRINT D$"OPEN "FILE$","L"RL
2670 NEXT K1
2680 PRINT D$"CLOSE "FILE$
2690 HOME : PRINT : PRINT "SEARCH COMPLETE": PRINT : PRINT
N;" RECORDS FOUND": VTAB 23: HTAB 10: GOSUB 210: GET Z$: PRI
NT Z$: CLEAR : GOTO 400
2700 REM PRINT REPORT OR LABELS
2710 A1$ = "PRINT REPORT":B1$ = "REPORT":C1$ = "COLUMN"

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

2720 IF Y = 9 THEN A1$ = "PRINT MAIL LABELS":B1$ = "LABEL":
    C1$ = "LINE"
2730 TEXT : HOME : GOSUB 220: HOME : GOSUB 310:RX = 0: GOSU
    B 240
2740 INPUT REC
2750 PRINT D$"CLOSE";FILE$
2760 T$(NF + 1) = H$ + H$ + H$ + H$ + B$: DIM TX$(2 * NF),TW
    (2 * NF): FOR K = 1 TO NF
2770 TX$(K) = "": IF Y = 9 THEN TX$(K) = CHR$ (13)
2780 NEXT K
2790 HOME : PRINT : PRINT "USING SAVED REPORT FORMAT? Y/N:
    "": GET Y$: PRINT Y$: HOME : IF Y$ = "N" THEN 2830
2800 IF Y$ < > "Y" THEN 2790
2810 PRINT : PRINT "ENTER NAME OF FORMAT": PRINT "PRESS RET
    URN FOR CATALOG";G$: INPUT "":FO$: IF FO$ = "" THEN HOME
    : PRINT D$"CATALOG": GOTO 2810
2820 GOSUB 4520: GOTO 3080
2830 HOME : CALL - 958:L = 0:LK = 0: PRINT : PRINT "THE FO
    LLOWING FIELDS ARE AVAILABLE FOR": PRINT "THE ";B1$;":": PRI
    NT : FOR K = 1 TO NF STEP 2:LK = LK + 1: VTAB (LK + 8): PRIN
    T K;". "; TAB( 5);T$(K); TAB( 21);K + 1;". "; TAB( 25);T$(K +
    1): NEXT K
2840 L = L + 1
2850 VTAB 23: HTAB 10: GOSUB 210: PRINT : VTAB 20: CALL -
    868: PRINT "ENTER FIELD # FOR ";B1$;" ";C1$;" ";L;G$: INPUT
    TV$: IF L = 1 AND LEN (TV$) = 0 THEN CLEAR : GOTO 400
2860 IF LEN (TV$) = 0 THEN 2900
2870 IF TV$ = CHR$ (21) OR TV$ = CHR$ (32) THEN CLEAR :
    GOTO 400
2880 TV = INT ( VAL (TV$)): IF TV < 1 OR TV > NF THEN 2850
2890 TW(L) = TV: GOTO 2840
2900 HOME :LC = L - 1: PRINT : PRINT "YOUR ";B1$;" WILL CON
    SIST OF:": PRINT
2910 FOR K = 1 TO LC: HTAB 10: PRINT K;". ";T$(TW(K)): NEXT
    K: PRINT
2920 VTAB 22: PRINT "IS THIS CORRECT (Y/N) ";G$: GET Y$: P
    RINT Y$: IF Y$ = "N" THEN 2830
2930 IF Y$ = CHR$ (21) OR Y$ = CHR$ (32) THEN CLEAR : GO
    TO 400
2940 IF Y$ < > "Y" THEN 2920
2950 IF Y = 8 THEN 3080
2960 VTAB 22: CALL - 868: PRINT "JOIN TOGETHER TWO LINES O
    R MORE (Y/N) ";G$: GET Y$: PRINT Y$: IF Y$ < > "Y" THEN 30
    80
2970 FOR K = 1 TO NF:TX$(K) = CHR$ (13): NEXT K
2980 VTAB 21: CALL - 958: PRINT "TO ENTRY #": VTAB 23: HT
    AB 10: GOSUB 210: PRINT
2990 VTAB 20: PRINT "JOIN ENTRY # ";G$: CALL - 868: INPUT
    LX$: IF LEN (LX$) = 0 THEN 3080
3000 REM
3010 VTAB 23: CALL - 958:LX = VAL (LX$): IF LX < 1 OR LX
    > LC - 1 THEN PRINT "INVALID!";G$:G$: GOTO 2980
3020 VTAB 21: HTAB 13: PRINT LX + 1
3030 PRINT "ENTER CONJUNCTION "; INVERSE : PRINT "(SPACE ,
    ; / & .)";G$: NORMAL : CALL - 868:TX$(LX) = "": REM NO
    THING INBETWEEN ""
3040 GET T1$: PRINT T1$;: IF T1$ = CHR$ (13) THEN 3060

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

3050 TX$(LX) = TX$(LX) + T1$: GOTO 3040
3060 VTAB 23: CALL - 868: PRINT T$(TW(LX));TX$(LX);T$(TW(L
X + 1)): FOR K = 1 TO 1000: NEXT K
3070 GOTO 2980
3080 HOME : PRINT : PRINT "THE ";B1$;" WILL LOOK LIKE THIS:
": PRINT
3090 FOR K = 1 TO LC: PRINT T$(TW(K));
3100 IF Y = 9 THEN PRINT TX$(K);: GOTO 3120
3110 PRINT " ";
3120 NEXT K
3130 PRINT
3140 VTAB 22: PRINT "IS THIS CORRECT (Y/N) ";G$;: GET Y$: P
RINT Y$: IF Y$ = "N" THEN 2830
3150 IF Y$ < > "Y" THEN 3140
3160 VTAB 19: CALL - 958: PRINT "DO YOU WANT TO SAVE THIS
FORMAT? ";G$;: GET Y$: PRINT Y$: IF Y$ = "N" THEN 3200
3170 IF Y$ < > "Y" THEN 3160
3180 PRINT "ENTER NAME OF FORMAT": PRINT "PRESS RETURN FOR
CATALOG";G$;: INPUT ":";FO$: IF FO$ = "" THEN HOME : PRINT
D$"CATALOG": GOTO 3180
3190 GOSUB 4420
3200 SRFL = 0: HOME : PRINT : PRINT "DO YOU WANT A SEARCH IN
A SPECIFIC FIELD?": GOSUB 110: IF Y$ = "N" THEN 3290
3210 IF Y$ < > "Y" THEN 3200
3220 SRFL = 1: HOME : PRINT : PRINT "THE FOLLOWING FIELDS AR
E AVAILABLE FOR THE SEARCH:": PRINT : FOR K = 1 TO NF STEP
2: PRINT K;". ";T$(K); TAB( 20);K + 1;". ";T$(K + 1): NEXT K
3230 VTAB 24: HTAB 5: GOSUB 210
3240 VTAB 20: HTAB 1: CALL - 868: PRINT "ENTER # FOR FIELD
TO SEARCH ";G$;: INPUT ":";SERFLD$: IF SERFLD$ = "" THEN SR
FL = 0: GOTO 3290
3250 SERFLD = VAL (SERFLD$): IF SERFLD < 1 OR SERFLD > NF TH
EN 3240
3260 VTAB 22: HTAB 1: CALL - 868: PRINT "ENTER SEARCH PARA
METER ";G$;: INPUT ":";SP$:SP = LEN (SP$): IF SP > TV(SERFL
D) THEN VTAB 22: CALL - 868: INVERSE : PRINT "SEARCH PARAM
ETER TOO LONG !";: NORMAL : FOR II = 1 TO 750: NEXT II: GOTO
3260
3270 FOR II = 1 TO 300: NEXT II: HOME : PRINT : PRINT "SEAR
CH ";T$(SERFLD);" FOR ";SP$: GOSUB 110: IF Y$ = "N" THEN 320
0
3280 IF Y$ < > "Y" THEN 3270
3290 HOME : PRINT : PRINT "WANT SORTED ";B1$;" (Y/N) ";G$;:
GET Y1$: PRINT Y1$: IF Y1$ < > "Y" THEN 3470
3300 DIM SRT$(REC),SO(REC)
3310 HOME : PRINT : PRINT "THE FOLLOWING FIELDS ARE AVAILAB
LE FOR THE SORT:": PRINT : FOR K = 1 TO NF STEP 2: PRINT K;"
. ";T$(K); TAB( 20);K + 1;". ";T$(K + 1): NEXT K

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

3320 PRINT : PRINT "SORT ON WHICH FIELD (BY NUMBER): ";G$;:
      INPUT SF$:SF = VAL (SF$): IF SF < 1 OR SF > NF THEN 3310
3330 PRINT : PRINT "***SORT ON ";: INVERSE : PRINT T$(SF):
      NORMAL
3340 PRINT : PRINT "ASCENDING OR DESCENDING SORT (A/D): ";G
      $;: GET S$: PRINT S$:S1 = (S$ = "A")
3350 PRINT D$:"OPEN"FILE$,L";RL
3360 FOR K = 1 TO REC:RX = K: GOSUB 250: IF SF = 1 THEN 338
      0
3370 FOR L = 1 TO SF - 1: INPUT Z$: NEXT L
3380 INPUT SRT$(K):SO(K) = K: NEXT K
3390 PRINT D$"CLOSE"FILE$
3400 M = 1
3410 M = 3 * M + 1: IF M < REC THEN 3410
3420 M = (M - 1) / 3: IF M < 1 THEN 3460
3430 FOR J = M + 1 TO REC:LL = J - M:SS$ = SRT$(J):S = SO(J
      )
3440 IF S1 = (SRT$(LL) > SS$) THEN SRT$(LL + M) = SRT$(LL):
      SO(LL + M) = SO(LL):LL = LL - M: IF LL > 0 THEN 3440
3450 SRT$(LL + M) = SS$:SO(LL + M) = S: NEXT J: GOTO 3420
3460 REM SORTED LIST COMPLETE
3470 HOME : PRINT : PRINT "ENTER IN PRINTER SLOT #";G$;: IN
      PUT SLOT
3480 IF Y = 8 THEN GOSUB 4590: GOTO 3520
3490 HOME : PRINT "ALIGN PAPER AND TURN ON PRINTER": PRINT
      "PRESS RETURN TO CONTINUE";: GET R$: PRINT R$: HOME
3500 PRINT "HOW MANY CARRIAGE RETURNS FROM ONE ";
3510 PRINT B1$;" TO NEXT: ";G$: INPUT CR
3520 PRINT D$"OPEN"FILE$,L";RL
3530 IF Y1$ < > "Y" THEN 3550
3540 FOR K1 = 1 TO REC:RX = SO(K1): GOTO 3640
3550 HOME : VTAB 23: HTAB 10: GOSUB 210: PRINT : VTAB 12: C
      ALL - 868: PRINT "ENTER RECORD # TO START WITH: ";G$;: INPU
      T RX$:RX = VAL (RX$): IF LEN (RX$) = 0 THEN CLEAR : GOTO
      400
3560 IF RX < = REC THEN RX = RX - 1: HOME : GOTO 3580
3570 PRINT "ONLY ";REC;" RECORDS ON FILE. ";G$:G$: FOR KK =
      1 TO 400: NEXT KK: GOTO 3490
3580 RX = RX + 1: IF RX < 1 THEN RX = 1
3590 IF RX < = REC THEN 3640
3600 PRINT D$"CLOSE"FILE$
3610 PRINT D$"PR#0"
3620 PRINT "END OF FILE!";G$:G$
3630 PRINT : PRINT "PRESS ";: INVERSE : PRINT "RETURN";: NO
      RMAL : PRINT " WHEN READY";: GET Y$: PRINT Y$: RUN 400
3640 GOSUB 250: FOR K = 1 TO NF: INPUT P$(K): NEXT K
3650 IF SRFL = 0 THEN 3680
3660 & P$(SERFLD),SP$
3670 IF PEEK (26) = 0 THEN 3820
3680 PRINT D$"PR#"SLOT
3690 PRINT CHR$ (9); CHR$ (1);
3700 IF Y = 9 THEN 3750
3710 PRINT CHR$ (27); CHR$ (68);: FOR K = 1 TO LC: PRINT
      CHR$ (TI(K));: NEXT K: PRINT CHR$ (0);: FOR K = 1 TO LC
3720 PRINT CHR$ (9);P$(TW(K));

```


ORIGINAL PAGE IS
OF POOR QUALITY

```
3730 NEXT K
3740 GOTO 3780
3750 FOR K = 1 TO LC
3760 PRINT P$(TW(K));TX$(K);
3770 NEXT K
3780 IF CR = 0 THEN 3810
3790 CX = CR: IF Y = 9 THEN CX = CR - 1
3800 FOR K = 1 TO CX: PRINT : NEXT K: IF Y = 9 THEN 3820
3810 PRINT D$;"PR#0"
3820 IF Y1$ < > "Y" THEN 3580
3830 NEXT K1: PRINT D$"PR#0": GOTO 3600
3840 REM **COMPUTE SUBTOTALS**
3850 A1$ = "COMPUTE SUBTOTALS":B1$ = "SUBTOTALS"
3860 TEXT : HOME : GOSUB 220: HOME : GOSUB 310:RX = 0: GOSU
  B 240
3870 INPUT REC
3880 PRINT D$"CLOSE"FILE$
3890 DIM SP$(NF),SERFLD(NF)
3900 HOME : PRINT : PRINT "THE FOLLOWING FIELDS ARE AVAILAB
  LE FOR THE ";B1$;":": PRINT : FOR K = 1 TO NF STEP 2: PRINT
  K;". ";T$(K); TAB( 20);K + 1;". ";T$(K + 1): NEXT K
3910 VTAB 24: HTAB 10: GOSUB 210
3920 VTAB 20: HTAB 1: CALL - 868: PRINT "ENTER # FOR FIELD
  TO SUBTOTAL ";G$;: INPUT "":ST$: IF ST$ = "" THEN CLEAR :
  GOTO 400
3930 ST = VAL (ST$): IF ST < 1 OR ST > NF THEN 3920
3940 FOR II = 1 TO 300: NEXT II:SRFL = 0
3950 HOME : PRINT : PRINT "WANT SUBTOTALS ON SPECIFIC RECOR
  DS ?";G$;: GET Y$: PRINT Y$;: IF Y$ = "N" THEN 4100
3960 IF Y$ < > "Y" THEN 3950
3970 SRFL = 1
3980 HOME : PRINT : PRINT "THE FOLLOWING FIELDS ARE AVAILAB
  LE FOR THE SEARCH:": PRINT : FOR K = 1 TO NF STEP 2: PRINT K
  ;". ";T$(K); TAB( 20);K + 1;". ";T$(K + 1): NEXT K
3990 FOR K = 1 TO NF: VTAB 20: HTAB 1: CALL - 958: VTAB 24
  : HTAB 10: GOSUB 210
4000 VTAB 20: HTAB 1: CALL - 868: PRINT "ENTER CHOICE FOR
  SEARCH FIELD # ";K;G$;: INPUT "":SERFLD$: IF SERFLD$ = ""
  AND K = 1 THEN SRFL = 0: GOTO 4100
4010 IF SERFLD$ = "" THEN NP = K - 1:K = NF: GOTO 4050
4020 SERFLD(K) = VAL (SERFLD$): IF SERFLD(K) < 1 OR SERFLD(
  K) > NF THEN 4000
4030 VTAB 22: HTAB 1: CALL - 868: PRINT "ENTER SEARCH PARA
  METER ";G$;: INPUT "":SP$(K):SP = LEN (SP$(K))
4040 IF SP > TV(SERFLD(K)) THEN VTAB 22: HTAB 1: CALL - 8
  68: INVERSE : PRINT "SEARCH PARAMETER TOO LONG !";: NORMAL :
  FOR II = 1 TO 750: NEXT II: GOTO 4030
4050 NEXT K: FOR II = 1 TO 300: NEXT II
4060 HOME : FOR K = 1 TO NP: VTAB (2 * K + 6): HTAB 1: PRIN
  T "SEARCH ";T$(SERFLD(K));" FOR ";SP$(K);: IF K < NP THEN H
  TAB ( PEEK (36) + 2): PRINT "AND";
4070 NEXT K: PRINT : GOSUB 110
4080 IF Y$ = "N" THEN 3980
4090 IF Y$ < > "Y" THEN 4080
4100 PRINT D$"OPEN "FILE$",L"RL
4110 FT = 0
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
4120 FOR K1 = 1 TO REC
4130 PRINT D$"READ "FILE$",R"K1
4140 IF SRFL = 1 THEN 4180
4150 PRINT D$"POSITION "FILE$",R"(ST - 1)
4160 PRINT D$"READ "FILE$
4170 INPUT P$(ST): GOTO 4240
4180 FOR K = 1 TO NF: INPUT P$(K): NEXT K
4190 FL = 0
4200 FOR K = 1 TO NP
4210 & P$(SERFLD(K)),SP$(K)
4220 IF PEEK (26) = 0 THEN K = NP:FL = 1
4230 NEXT K: IF FL = 1 THEN 4250
4240 FT = FT + VAL (P$(ST))
4250 NEXT K1
4260 PRINT D$"CLOSE "FILE$
4270 HOME : PRINT : PRINT "THE TOTAL FOR ";T$(ST);" IS ";FT
4280 VTAB 24: HTAB 10: GOSUB 210: GET Y$: PRINT Y$;: GOTO 3
900
4290 REM DISPLAY FILE STRUCTURE SUBROUTINE
4300 A1$ = "FILE STRUCTURE DISPLAY ROUTINE"
4310 TEXT : HOME : GOSUB 310
4320 HOME : VTAB 1: PRINT "FILENAME: ";FILE$
4330 VTAB 2: PRINT "RECORD LENGTH: ";RL
4340 VTAB 3: PRINT "NO. OF FIELDS: ";NF: PRINT
4350 VTAB 5: PRINT "FIELD #"; TAB( 10);"FIELD LENGTH"; TAB(
24);"FIELD NAME"
4360 VTAB 6: FOR K = 1 TO 39: PRINT "-";: NEXT K: PRINT "-"
4370 VTAB 7: FOR K = 1 TO NF: HTAB 3: PRINT K;". "; TAB( 15)
;TV(K); TAB( 24);T$(K): NEXT K
4380 VTAB 23: INVERSE : HTAB 10: PRINT "PRESS RETURN FOR ME
NU": NORMAL : GET Y$: PRINT Y$
4390 RUN 400
4400 REM RETURN TO MAIN PROGRAM
4410 FOR K = 1 TO 100: NEXT K: TEXT : HOME : PRINT "DON'T F
ORGET TO MAKE A BACKUP": PRINT "OF YOUR DATA.": FOR K = 1 TO
1000: NEXT K: PRINT D$" RUN BDMADS.MAIN"
4420 PRINT D$"OPEN "FO$
4430 PRINT D$"DELETE "FO$
4440 PRINT D$"OPEN "FO$
4450 PRINT D$"WRITE "FO$
4460 PRINT LC
4470 FOR K = 1 TO LC: IF TX$(K) = CHR$ (13) THEN TX$(K) =
CHR$ (19)
4480 PRINT TW(K): PRINT CHR$ (34);TX$(K): IF TX$(K) = CHR
$ (19) THEN TX$(K) = CHR$ (13)
4490 NEXT K
4500 PRINT D$"CLOSE "FO$
4510 RETURN
4520 PRINT D$"OPEN "FO$
4530 PRINT D$"READ "FO$
4540 INPUT LC
4550 FOR K = 1 TO LC: INPUT TW(K): INPUT TX$(K): IF TX$(K)
= CHR$ (19) THEN TX$(K) = CHR$ (13)
4560 NEXT K
4570 PRINT D$"CLOSE "FO$
4580 RETURN
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
4590 HOME : PRINT "USING SAVED PRINTER FORMAT? Y/N: "; GET
Y$: PRINT Y$: HOME
4600 IF Y$ = "Y" THEN 5170
4610 PRINT "USING WIDE PAPER? Y/N: "; GET Y$: PRINT Y$: HO
ME
4620 IF Y$ = "Y" THEN WP = 1
4630 PRINT "WANT TITLE ON TOP? Y/N: "; GET Y$: PRINT Y$: H
OME
4640 IF Y$ = "N" THEN 4760
4650 TZ = 1
4660 PRINT "ENTER TITLE: ": INPUT T$: HOME
4670 PRINT "WANT ENLARGED TITLE? Y/N: "; GET Y$: PRINT Y$:
HOME
4680 IF Y$ = "Y" THEN OT = 1
4690 PRINT "WANT SUBTITLE? Y/N: "; GET Y$: PRINT Y$: HOME
4700 IF Y$ = "N" THEN 4750
4710 SZ = 1
4720 PRINT "ENTER SUBTITLE: ": INPUT TS$: HOME
4730 PRINT "WANT ENLARGED SUBTITLE? Y/N: "; GET Y$: PRINT
Y$: HOME
4740 IF Y$ = "Y" THEN OS = 1
4750 PRINT "ENTER NO. OF BLANK LINES UNDER TITLES: ": INPUT
B1: HOME
4760 PRINT "WANT COMMENT LINES? Y/N: "; GET Y$: PRINT Y$:
HOME
4770 IF Y$ = "N" THEN 4890
4780 CZ = 1
4790 PRINT "WANT CONDENSED COMMENTS? Y/N: "; GET Y$: PRINT
Y$: HOME
4800 IF Y$ = "Y" THEN CC = 1
4810 PRINT "ENTER NO. OF COMMENT LINES: ": INPUT CL: HOME :
DIM NC(CL), C$(CL,5), TC(CL,5)
4820 FOR I = 1 TO CL
4830 PRINT "ENTER NO. OF COMMENTS FOR LINE "; I; ": ": INPUT
NC(I): PRINT
4840 FOR J = 1 TO NC(I)
4850 PRINT "ENTER COMMENT "; J; " FOR LINE "; I; ": ": INPUT C$(
I,J)
4860 PRINT "ENTER TAB FOR COMMENT "; J; " FOR LINE "; I; ": ":
INPUT TC(I,J)
4870 NEXT J: PRINT : NEXT I: HOME
4880 PRINT "ENTER NO. OF BLANK LINES UNDER COMMENTS: ": INP
UT B2: HOME
4890 PRINT "WANT CONDENSED LISTS? Y/N: "; GET Y$: PRINT Y$
: HOME
4900 IF Y$ = "Y" THEN CI = 1
4910 PRINT "WANT HEADINGS? Y/N: "; GET Y$: PRINT Y$: HOME
4920 IF Y$ = "N" THEN 4990
4930 HZ = 1: DIM H$(LC)
4940 DIM TI(LC)
4950 FOR I = 1 TO LC
4960 IF HZ = 1 THEN PRINT "ENTER HEADING FOR "; T$(TW(I)); "
LIST": INPUT H$(I)
4970 PRINT "ENTER TAB FOR "; T$(TW(I)); " LIST": INPUT TI(I)
4980 NEXT I: HOME
4990 PRINT "HOW MANY CARRIAGE RETURNS FROM END OF ONE ";
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
5000 PRINT "LINE TO NEXT";G$;: INPUT CR: HOME
5010 PRINT "WANT TO SAVE PRINTER FORMAT? Y/N: ";: GET Y$: P
RINT Y$: HOME
5020 IF Y$ = "N" THEN 5390
5030 PRINT "ENTER FILE NAME": PRINT "PRESS RETURN FOR CATAL
OG";G$;: INPUT "":FF$: IF FF$ = "" THEN HOME : PRINT D$"CA
TALOG": GOTO 5030
5040 HOME : PRINT "WRITING FILE ";FF$
5050 PRINT D$"OPEN "FF$: PRINT D$"DELETE "FF$: PRINT D$"OPE
N "FF$
5060 PRINT D$"WRITE "FF$
5070 PRINT WP: PRINT TZ: IF TZ = 0 THEN 5110
5080 PRINT OT: PRINT SZ: IF SZ = 0 THEN 5100
5090 PRINT OS
5100 PRINT B1
5110 PRINT CZ
5120 IF CZ = 1 THEN PRINT CC: PRINT CL: FOR I = 1 TO CL: P
RINT NC(I): FOR J = 1 TO NC(I): PRINT TC(I,J): NEXT J: NEXT
I: PRINT B2
5130 PRINT CI: PRINT HZ: PRINT LC: FOR I = 1 TO LC
5140 IF HZ = 1 THEN PRINT H$(I)
5150 PRINT TI(I): NEXT I: PRINT CR
5160 PRINT D$"CLOSE "FF$: HOME : PRINT "FILE "FF$;" COMPLET
E": GOTO 5390
5170 PRINT : PRINT "ENTER NAME OF FILE": PRINT "PRESS RETUR
N FOR CATALOG";G$: INPUT "":FF$: IF FF$ = "" THEN HOME : P
RINT D$"CATALOG": GOTO 5170
5180 HOME : PRINT "READING FILE "FF$: PRINT D$"OPEN "FF$
5190 PRINT D$"READ "FF$
5200 INPUT WP
5210 INPUT TZ
5220 IF TZ = 0 THEN 5280
5230 INPUT OT
5240 INPUT SZ
5250 IF SZ = 0 THEN 5270
5260 INPUT OS
5270 INPUT B1
5280 INPUT CZ
5290 IF CZ = 1 THEN INPUT CC: INPUT CL: FOR I = 1 TO CL: I
NPUT NC(I): FOR J = 1 TO NC(I): INPUT TC(I,J): NEXT J: NEXT
I: INPUT B2
5300 INPUT CI: INPUT HZ
5310 INPUT LC: FOR I = 1 TO LC
5320 IF HZ = 1 THEN INPUT H$(I)
5330 INPUT TI(I): NEXT I: INPUT CR
5340 PRINT D$"CLOSE "FF$: PRINT "FILE "FF$;" CONTENTS IN ME
MORY"
5350 FOR I = 1 TO 1000: NEXT I
5360 HOME : IF TZ = 1 THEN PRINT "ENTER TITLE: ": INPUT T$
: HOME
5370 IF SZ = 1 THEN PRINT "ENTER SUBTITLE: ": INPUT TS$: H
OME
5380 IF CZ = 1 THEN FOR I = 1 TO CL: FOR J = 1 TO NC(I): P
RINT "ENTER COMMENT ";J;" FOR LINE ";I;": ": INPUT C$(I,J):
NEXT J: PRINT : NEXT I: HOME
5390 REM
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
5400 PRINT "WANT EMPHASIZED PRINT? Y/N: "; GET Y$: PRINT Y
    $: HOME : IF Y$ = "Y" THEN EP = 1
5410 PRINT "ALIGN PAPER AND TURN ON PRINTER": PRINT "PRESS
    RETURN TO CONTINUE";: GET R$: PRINT R$: HOME
5420 PRINT D$"PR#"SLOT: PRINT CHR$ (27); CHR$ (78); CHR$ (
    3);
5430 IF EP < > 1 THEN 5440
5435 IF CC = 1 OR CI = 1 THEN PRINT CHR$ (27);"G";: GOTO
    5440
5437 PRINT CHR$ (27);"E";
5440 IF TZ = 0 THEN 5590
5450 O = OT: GOSUB 5490:E$ = T$: GOSUB 5540
5460 IF SZ = 0 THEN 5570
5470 O = OS: GOSUB 5490:E$ = TS$: GOSUB 5540: GOTO 5570
5480 REM CENTERING TITLES
5490 IF WP = 0 AND O = 0 THEN XL = 85
5500 IF WP = 0 AND O = 1 THEN XL = 42
5510 IF WP = 1 AND O = 0 THEN XL = 123
5520 IF WP = 1 AND O = 1 THEN XL = 63
5530 RETURN
5540 IF O = 1 THEN PRINT CHR$ (14);
5550 PRINT SPC( (XL - LEN (E$)) / 2);E$
5560 RETURN
5570 IF B1 = 0 THEN 5590
5580 FOR I = 1 TO B1: PRINT : NEXT I
5590 PRINT CHR$ (9); CHR$ (1);: IF CZ = 0 THEN 5680
5600 IF CC = 1 THEN PRINT CHR$ (15);
5620 FOR I = 1 TO CL
5630 PRINT CHR$ (27); CHR$ (68);: FOR J = 1 TO NC(I): PRIN
    T CHR$ (TC(I,J));: NEXT J: PRINT CHR$ (0);
5640 FOR J = 1 TO NC(I): PRINT CHR$ (9);C$(I,J);: NEXT J:
    PRINT
5650 NEXT I
5660 IF B2 = 0 THEN 5680
5670 FOR I = 1 TO B2: PRINT : NEXT I
5680 IF CI = 0 AND CC = 1 THEN PRINT CHR$ (18);
5690 IF CI = 1 AND CC = 0 THEN PRINT CHR$ (15);
5700 IF HZ = 0 THEN 5740
5710 PRINT CHR$ (27); CHR$ (68);: FOR I = 1 TO LC: PRINT
    CHR$ (TI(I));: NEXT I: PRINT CHR$ (0);
5720 FOR I = 1 TO LC: PRINT CHR$ (9);H$(I);: NEXT I: PRINT

5730 FOR I = 1 TO LC: PRINT CHR$ (9);: FOR J = 1 TO LEN (
    H$(I)): PRINT "-";: NEXT J: NEXT I: PRINT
5740 PRINT D$"PR#0"
5750 RETURN
```

```

10 REM FILE UTILITY PROGRAM
20 TEXT : HOME : VTAB 3: HTAB 17: PRINT "APPLE II"
30 VTAB 5: HTAB 8: PRINT "** FILE UTILITY PROGRAM **"
40 VTAB 8: INPUT "NAME OF FILE TO RESTRUCTURE? ";G$:H$ = G$
  + CHR$(2):S1 = 6:D1 = 1:S2 = 6:D2 = 1
50 IF ER = 1 THEN 80
60 VTAB 11: HTAB 6: PRINT "ORIGINAL IN SLOT #";S1: VTAB 12:
  HTAB 17: PRINT "DRIVE #";D1: VTAB 13: HTAB 5: PRINT "DUPLIC
  ATE IN SLOT #";S2: VTAB 14: HTAB 17: PRINT "DRIVE #";D2
70 GOTO 120
80 VTAB 11: HTAB 6: PRINT "ORIGINAL IN SLOT # ? ";: GET S1$
  : PRINT S1$:S1 = VAL (S1$): IF S1 < 1 OR S1 > 7 THEN 80
90 VTAB 12: HTAB 17: PRINT "DRIVE # ? ";: GET D1$: PRINT D1
  $:D1 = VAL (D1$): IF D1 < 1 OR D1 > 2 THEN 90
100 VTAB 13: HTAB 5: PRINT "DUPLICATE IN SLOT # ? ";: GET S
  2$: PRINT S2$:S2 = VAL (S2$): IF S2 < 1 OR S2 > 7 THEN 100
110 VTAB 14: HTAB 17: PRINT "DRIVE # ? ";: GET D2$: PRINT D
  2$:D2 = VAL (D2$): IF D2 < 1 OR D2 > 2 THEN 110
120 VTAB 17: HTAB 4: PRINT "VERIFY CORRECT (Y/N) ? ";: GET
  V$: PRINT V$: IF V$ = "Y" THEN 140
130 POKE 34,6:ER = 1: GOTO 40
140 REM
150 REM VARIABLES FOR NEW FILES
160 REM
170 D$ = CHR$(4)
180 PRINT D$"OPEN "H$;","S";S1;","D";D1
190 PRINT D$"READ "H$
200 INPUT X1: INPUT X2
210 DIM M$(X1 + 1),M(X1 + 1),P$(X1 + 1)
220 FOR J = 1 TO X1: INPUT M$(J): INPUT M(J)
230 NEXT
240 PRINT D$"CLOSE"
250 TEXT : HOME : VTAB 5: IF X1 = > 12 THEN VTAB 2
260 PRINT "THERE ARE ";X1;" FIELDS IN THIS FILE": PRINT "TH
  EIR NAMES/LENGTHS ARE:"
270 FOR J = 1 TO X1: VTAB 8 + J: IF X1 = > 12 THEN VTAB 4
  + J
280 HTAB 4: PRINT M$(J);
290 IF M(J) < 100 AND M(J) = > 10 THEN HTAB 20: GOTO 320
300 IF M(J) < 10 THEN HTAB 21: GOTO 320
310 HTAB 19
320 PRINT M(J): NEXT
330 VTAB 23: FLASH : PRINT "ESC";: NORMAL : PRINT " TO CHAN
  GE - ";: FLASH : PRINT "C/R";: NORMAL : PRINT " TO SELECT NE
  XT ITEM"
340 FOR J = 1 TO X1: VTAB 8 + J: IF X1 = > 12 THEN VTAB 4
  + J
350 HTAB 3: GET V$
360 IF V$ = CHR$(27) THEN INPUT M$(J): GOTO 390
370 IF V$ = CHR$(13) THEN 430
380 GOTO 350
390 VTAB 8 + J: HTAB 19: IF X1 = > 12 THEN VTAB 4 + J
400 INPUT M(J)
410 IF J = X1 THEN 500
420 NEXT

```

ORIGINAL PAGE IS
OF POOR QUALITY

```
430 VTAB 8 + J: HTAB 19: IF X1 = > 12 THEN VTAB 4 + J
440 GET V$
450 IF V$ = CHR$ (27) AND J = X1 THEN INPUT M(J): GOTO 50
0
460 IF V$ = CHR$ (13) AND J = X1 THEN 500
470 IF V$ = CHR$ (27) THEN INPUT M(J): NEXT
480 IF V$ = CHR$ (13) THEN NEXT
490 GOTO 430
500 IF X1 = 16 THEN 570
510 HOME : VTAB 8: PRINT "DO YOU WISH TO ADD A FIELD ? (Y/N
): "; GET V$: PRINT V$
520 IF V$ = "Y" THEN 550
530 IF V$ = "N" THEN 570
540 GOTO 500
550 VTAB 10: HTAB 5: INPUT "FIELD NAME: ";M$(X1 + 1)
560 VTAB 12: HTAB 9: INPUT "LENGTH: ";M(X1 + 1)
570 REM
580 REM SET UP NEW FILE
590 REM
600 TEXT : HOME : VTAB 10: HTAB 10: PRINT "(WORKING ON FILE
)": PRINT : HTAB 10: PRINT "(PLEASE STAND BY)"
610 X3 = X1
620 IF V$ = "Y" THEN X3 = X1 + 1
630 FOR J = 1 TO X3:T = T + M(J): NEXT : PRINT
640 F$ = G$:I$ = H$: IF S1 = S2 AND D1 = D2 THEN F$ = "FILE
XZXZXZ":I$ = F$ + CHR$ (2)
650 PRINT D$"OPEN "I$;" ,S";S2;" ,D";D2
660 PRINT D$"WRITE "I$
670 PRINT X3: PRINT T
680 FOR J = 1 TO X3: PRINT M$(J): PRINT M(J): NEXT
690 PRINT D$"CLOSE"
700 REM
710 REM READ DATA
720 REM
730 X = 0
740 PRINT D$"OPEN "G$;" ,L";X2;" ,S";S1;" ,D";D1
750 PRINT D$"READ "G$;" ,R";X
760 INPUT HH
770 PRINT D$"CLOSE "G$
780 REM
790 REM WRITE DATA FILE
800 REM
810 PRINT D$"OPEN "F$;" ,L";T;" ,S";S2;" ,D";D2
820 PRINT D$"WRITE "F$;" ,R";X
830 PRINT HH
840 PRINT D$"CLOSE "F$
850 REM
860 REM READ FILE INFO
870 REM
880 X = 1
890 PRINT D$"OPEN "G$;" ,L";X2;" ,S";S1;" ,D";D1
900 PRINT D$"READ "G$;" ,R";X
910 FOR K = 1 TO X1
920 INPUT P$(K)
930 NEXT K
940 PRINT D$"CLOSE "G$
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
950 REM
960 REM WRITE NEW FILE
970 REM
980 PRINT D$"OPEN "F$; ",L";T; ",S"S2: ",D";D2
990 PRINT D$"WRITE "F$; ",R";X
1000 FOR K = 1 TO X3
1010 PRINT P$(K)
1020 NEXT K
1030 PRINT D$"CLOSE "F$
1040 REM
1050 REM STEP X AND CHECK FOR END
1060 REM
1070 X = X + 1
1080 IF X = (HH + 1) THEN 1100
1090 GOTO 890
1100 REM
1110 REM CHECK FILE NAMES
1120 REM
1130 IF F$ = G$ THEN 1200
1140 PRINT D$"UNLOCK "H$
1150 PRINT D$"DELETE "H$
1160 PRINT D$"DELETE "G$
1170 PRINT D$"RENAME "I$; ", ";H$
1180 PRINT D$"RENAME "F$; ", ";G$
1190 PRINT D$"LOCK "H$
1200 HOME : VTAB 10: HTAB 5: PRINT "NEW FILES ARE READY TO
USE": PRINT : PRINT "DON'T FORGET TO MAKE A BACKUP": PRINT "
OF YOUR DATA.": FOR K = 1 TO 1000: NEXT K: PRINT CHR$ (4);"
RUN BDMADS.MAIN
```



```

10 REM   BDMADS - CALCULATION PROGRAM
20 HOME : GOSUB 120: FLASH
30 VTAB 3: HTAB 17: PRINT "BDMADS": VTAB 4: HTAB 14: PRINT
"CALCULATION": VTAB 5: HTAB 16: PRINT "PROGRAM": NORMAL
40 VTAB 20: HTAB 14: PRINT "BY J.R. SZUCH": VTAB 21: HTAB 1
6: PRINT "OCT. 1982"
50 VTAB 22: INVERSE : HTAB 8: PRINT "PRESS RETURN FOR CATAL
OG": GET R$: PRINT R$:
60 NORMAL
70 D$ = CHR$ (4): PRINT D$"BLOAD PRINTUSR,A$94A0": HOME
80 PRINT D$"CATALOG"
90 VTAB 20
100 PRINT "ENTER NAME OF INPUT DATA FILE": INPUT M$
110 HOME :BC$ = CHR$ (2): GOTO 720
120 VTAB 23: FOR K = 1 TO 39: PRINT "*":; NEXT K: PRINT : V
TAB 1: FOR K = 1 TO 39: PRINT "*":; NEXT K: PRINT
130 FOR K = 2 TO 22: VTAB K: PRINT "*":; HTAB 39: PRINT "*"
: NEXT K: RETURN
140 REM
150 REM
160 REM   PROCOM SUBROUTINE
170 T2 = T * T
180 IF T > = 2300. THEN 260
190 IF T > = 1200. THEN 230
200 CA = .24034 - 1.2464E - 5 * T + 2.0357E - 8 * T2
210 HA = 3.5214 + .22623 * T + 1.126E - 5 * T2
220 GOTO 280
230 CA = .20264 + 5.2545E - 5 * T - 7.6364E - 9 * T2
240 HA = 2.3733 + .22519 * T + 1.292E - 5 * T2
250 GOTO 280
260 CA = .23032 + 3.0024E - 5 * T - 3.0756E - 9 * T2
270 HA = - 37.404 + .25987 * T + 5.36E - 6 * T2
280 AM = 28.97 - .94619 * FA
290 R = 1.9862 / AM
300 TD = 3500. - T
310 CF = .93330 - 5.87E - 5 * TD - 3.27E - 8 * TD * TD
320 HF = - 132.2 + .50899 * T + 6.18E - 5 * T2
330 H = (HA + HF * FA) / (1. + FA)
340 CP = (CA + CF * FA) / (1. + FA)
350 CV = CP - R
360 GM = CP / CV
370 IF PI = 0 THEN 410
380 IF ABS (H - HD) < = .0001 * HD THEN 410
390 T = T + (HD - H) / .25
400 GOTO 170
410 RETURN
420 REM
430 REM
440 REM   PROCOM INVERSE SUBROUTINE
450 TD = HD / .25
460 GOSUB 170
470 RETURN
480 REM
490 REM
500 REM   FUN1 INTERPOLATION SUBROUTINE

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

510 M = N(NC): FOR L = 1 TO M
520 IF XI < X(L,NC) THEN K = L - 1: GOTO 550
530 NEXT L: K = L - 1: YC = Y(K,NC) + (XI - X(K,NC)) / (X(K,N
C) - X(K - 1,NC)) * (Y(K,NC) - Y(K - 1,NC))
540 GOTO 570
550 IF K = 0 THEN K = 1
560 YC = Y(K,NC) + (XI - X(K,NC)) / (X(K + 1,NC) - X(K,NC))
* (Y(K + 1,NC) - Y(K,NC))
570 RETURN
580 REM
590 REM
600 REM TURBINE ITERATION SUBROUTINE
610 PR = .5
620 XI = PR: P = PB / PR
630 GOSUB 510
640 WC = YC
650 WP = WC * P / SQR (TH)
660 IF ABS (W - WP) < = .001 * W THEN 690
670 PR = PR - (W - WP) * .05
680 GOTO 620
690 RETURN
700 REM
710 REM
720 REM DATA INPUT STARTS HERE
730 PRINT D$"OPEN "M$ + BC$
740 PRINT D$"READ "M$ + BC$
750 INPUT NM: INPUT RL: NM = NM - 1
760 DIM MN$(2 * NM), ML(NM): INPUT KN$: INPUT KL
770 FOR K = 1 TO NM: INPUT MN$(K): INPUT ML(K): NEXT K
780 PRINT D$"CLOSE "M$ + BC$
790 PRINT D$"OPEN "M$",L"RL
800 PRINT D$"READ "M$",R"
810 INPUT REC: DIM ME$(REC,NM), ME(REC,NM), KY$(REC)
820 PRINT D$"CLOSE "M$
830 PRINT "THERE ARE "REC" RECORDS IN "M$: PRINT
840 PRINT "OPTIONS ARE: 1. CALC FOR ALL RECS.": HTAB 14: PR
INT "2. CALC FOR BLOCK OF RECS.": HTAB 14: PRINT "3. CALC FO
R SINGLE REC.": HTAB 14: PRINT "4. QUIT": PRINT
850 INPUT "ENTER CHOICE: "; IC
860 HOME
870 ON IC GOTO 880,890,910,1650
880 BR = 1: ER = REC: GOTO 920
890 INPUT "ENTER START REC.NO.: "; BR
900 INPUT "ENTER END REC.NO.: "; ER: GOTO 920
910 INPUT "ENTER REC.NO.: "; RX: BR = RX: ER = RX
920 FOR I = BR TO ER
930 PRINT D$"OPEN "M$",L"RL: PRINT D$"READ "M$",R" I: INPUT
KY$(I)
940 FOR J = 1 TO NM
950 INPUT ME$(I,J): ME(I,J) = VAL (ME$(I,J))
960 NEXT J: NEXT I
970 PRINT D$"CLOSE "M$
980 HOME
990 PRINT "INPUT DATA ARE IN MEMORY": PRINT : PRINT "CALCUL
ATIONS NOW BEGIN"

```

```

1000 FOR I = BR TO ER: GOSUB 3000: NEXT I: PRINT : PRINT "C
ALCULATIONS DONE": PRINT
1010 PRINT "WANT TO SEE RESULTS? Y/N: ";: GET Y$: PRINT Y$:
IF Y$ = "N" THEN 1390
1020 PRINT : PRINT "WANT HARD-COPY? Y/N: ";: GET Y$: PRINT
Y$: IF Y$ = "N" THEN HOME : GOTO 1320
1030 PRINT : PRINT "ALIGN PAPER. TURN ON PRINTER"
1040 PRINT : INPUT "WANT EMPHASIZED PRINT? Y/N: ";EP$
1050 PRINT : INPUT "WANT TITLE? Y/N: ";Y$
1060 IF Y$ = "Y" THEN PRINT : INPUT "ENTER TITLE: ";TI$
1070 PRINT
1080 PRINT "PRESS RETURN TO CONTINUE": GET R$: PRINT R$:
1090 PRINT D$"BRUN PRINTUSR,A$94A0": HOME : PRINT USR (NM)
'3,1': HOME : PRINT D$"BRUN PRINTUSR,A$94A0"
1100 HOME : PRINT D$"PR#1"
1110 A$ = CHR$ (1):I$ = CHR$ (9):E$ = CHR$ (27)
1115 IF EP$ = "Y" THEN PRINT CHR$ (27); CHR$ (69);
1120 IF Y$ = "Y" THEN TS = (40 - LEN (TI$)) / 2: PRINT CH
R$ (14); SPC( TS);TI$: PRINT
1130 RPP = 2:T1 = 28:T2 = 43:T3 = 65: REM TABS
1140 G = 11:H = 5
1150 PRINT I$"6L";: PRINT I$"80N";: PRINT I$"75R";: PRINT I
$"60P";
1160 PRINT I$A$;
1180 PRINT E$; CHR$ (68); CHR$ (T1); CHR$ (T2); CHR$ (T3);
CHR$ (0);
1190 FOR I = BR TO ER:C = C + 1: IF C = RPP + 1 THEN PRINT
CHR$ (12):C = 1
1200 PRINT KN$;" = ";KY$(I): PRINT
1210 FOR J = 1 TO NM - 1 STEP 2
1220 PRINT MN$(J);I$; USR (ME(I,J))'11,5'I$;MN$(J + 1);I$;
USR (ME(I,J + 1))'11,5'
1230 NEXT J
1240 PRINT
1250 FOR J = 1 TO NV - 1 STEP 2
1260 PRINT VA$(J);I$; USR (VA(I,J))'11,5'I$;VA$(J + 1);I$;
USR (VA(I,J + 1))'11,5'
1270 NEXT J
1280 PRINT
1290 NEXT I
1300 PRINT D$"PR#0"
1310 PRINT : PRINT "PRINTING DONE": GOTO 1390
1320 PRINT D$"BRUN PRINTUSR,A$94A0": PRINT USR (NM)'3,1':
HOME : PRINT D$"BRUN PRINTUSR,A$94A0": FOR I = BR TO ER
1330 HOME : PRINT KN$;" = ";KY$(I): PRINT
1340 FOR J = 1 TO NM: PRINT MN$(J), USR (ME(I,J))'11,5'
1350 NEXT J: IF I = 1 THEN VTAB 23: PRINT "PRESS SPACE BAR
TO CONTINUE LIST": POKE 35,22
1360 VTAB NM + 4: FOR J = 1 TO NV: GET SP$: PRINT SP$;: HTA
B 1: PRINT VA$(J), USR (VA(I,J))'11,5'
1370 NEXT J
1380 NEXT I: POKE 35,24: VTAB 23: CALL - 958: PRINT "LISTI
NG DONE"
1390 PRINT : PRINT "WANT RESULTS IN DIF FILE? Y/N :";: GET
Y$: PRINT Y$
1400 IF Y$ = "N" THEN HOME : PRINT "INSERT MAIN DISK IN DR
IVE": PRINT "PRESS RETURN TO CONTINUE": GET R$: PRINT R$: PR
INT D$"RUN BDMADS.MAIN"

```

ORIGINAL PAGE IS
OF POOR QUALITY

```
1410 IF Y$ < > "Y" THEN 1390
1420 REM DATA STORAGE STARTS HERE
1430 PRINT D$"MON 0"
1440 HOME : PRINT D$"CATALOG"
1450 VTAB 20: PRINT "ENTER NAME OF DIF FILE: ": INPUT V$: H
OME
1460 PRINT D$"OPEN"V$: PRINT D$"DELETE"V$: PRINT D$"OPEN"V$
1470 VN = NM + NV + 2: REM NO. OF VECTORS(ROWS)
1480 TN = ER - BR + 2: REM NO. OF TUPLES(COLUMNS)
1490 GOSUB 1680
1500 T = - 1:V = 0:S$ = "BOT": GOSUB 1750
1510 T = 1:V = 0:S$ = "": GOSUB 1750
1520 FOR J = 1 TO NM:T = 1:V = 0:S$ = MN$(J): GOSUB 1750: N
EXT J
1530 T = 1:V = 0:S$ = "": GOSUB 1750
1540 FOR J = 1 TO NV:T = 1:V = 0:S$ = VA$(J): GOSUB 1750: N
EXT J
1550 FOR I = BR TO ER
1560 T = - 1:V = 0:S$ = "BOT": GOSUB 1750
1570 T = 1:V = 0:S$ = KY$(I): GOSUB 1750
1580 FOR J = 1 TO NM:T = 0:V = ME(I,J):S$ = "V": GOSUB 1750
: NEXT J
1590 T = 1:V = 0:S$ = "": GOSUB 1750
1600 FOR J = 1 TO NV:T = 0:V = VA(I,J):S$ = "V": GOSUB 1750
: NEXT J
1610 NEXT I
1620 T = - 1:V = 0:S$ = "EOD": GOSUB 1750
1630 PRINT D$"CLOSE "V$
1640 PRINT "DIF FILE COMPLETE"
1650 PRINT
1660 PRINT "INSERT MAIN DISK IN DRIVE": PRINT "PRESS RETURN
TO CONTINUE";: GET R$: PRINT R$: HOME : PRINT D$"RUN BDMADS
.MAIN"
1670 END
1680 PRINT D$"WRITE "V$
1690 PRINT "TABLE": PRINT "0,1": GOSUB 1740
1700 PRINT "VECTORS": PRINT "0,";VN: GOSUB 1740
1710 PRINT "TUPLES": PRINT "0,";TN: GOSUB 1740
1720 PRINT "DATA": PRINT "0,0": GOSUB 1740
1730 RETURN
1740 PRINT CHR$(34): CHR$(34): RETURN
1750 PRINT (;",,";V
1760 PRINT S$
1770 RETURN
1780 REM
1790 REM
3000 REM USER-SUPPLIED CALCULATIONS
3010 REM ENGINE 'ENG1' EXAMPLE
3020 REM
3030 REM CONSTANTS
3040 ID = ID + 1: IF ID > 1 THEN 3410
3050 NC = 9: DIM C(NC)
3060 C(1) = .956: REM PS3/F3 RATIO
3070 C(2) = .0876: REM KWGTRB
3080 C(3) = .0025: REM OUTFLOW BLEED COEFF
3090 C(4) = .7826: REM COOLING BLEED COEFF
```

```

3100 C(5) = 7429.35: REM FT-LB-RPM/PPS-BTU/LBM
3110 C(6) = 1.8326E - 3: REM THETA SLOPE
3120 C(7) = .0856: REM THETA INTERCEPT
3130 C(8) = 450.: REM DESIGN NG/100
3140 C(9) = .0678: REM KDPBRN BASELINE
3150 REM
3160 REM TABLES
3170 DIM X(14,8),Y(14,8),N(8)
3180 REM CURVE 2 - B1 BLEED
3190 DATA 14, 65.,78.6,80.,81.25,82.5,83.12,83.75,84.37,8
5.,86.25,86.87,87.5,88.12,105.
3200 DATA .109,.109,.1062,.1015,.0945,.088,.078,.066,.05,.
026,.016,.009,.0,.0
3210 REM CURVE 4 - WXQ2 BLEED
3220 DATA 4, 3.,8.3,10.,12.
3230 DATA .0851,.0846,.0815,.0779
3240 REM CURVE 7 - POWER TURBINE FLOW
3250 DATA 12, .3,.35,.4,.45,.5,.55,.6,.65,.7,.75,.8,.85
3260 DATA .372,.372,.3705,.367,.3625,.357,.351,.34,.325,.3
06,.285,.255
3270 REM CURVE 8 - B2 BLEED
3280 DATA 4, 0.,3.,4.98,12.
3290 DATA .01057,.01057,.009,.009
3300 READ N(2): FOR J = 1 TO N(2): READ X(J,2): NEXT J: FOR
J = 1 TO N(2): READ Y(J,2): NEXT J
3310 READ N(4): FOR J = 1 TO N(4): READ X(J,4): NEXT J: FOR
J = 1 TO N(4): READ Y(J,4): NEXT J
3320 READ N(7): FOR J = 1 TO N(7): READ X(J,7): NEXT J: FOR
J = 1 TO N(7): READ Y(J,7): NEXT J
3330 READ N(8): FOR J = 1 TO N(8): READ X(J,8): NEXT J: FOR
J = 1 TO N(8): READ Y(J,8): NEXT J
3340 REM
3350 REM CALCULATIONS START HERE
3360 NV = 36: DIM VA(REC,NV),VA$(NV)
3370 REM ME(1)=WA2CX ME(2)=WFPH ME(3)=P2
3380 REM ME(4)=PS3 ME(5)=P49 ME(6)=T2
3390 REM ME(7)=T25 ME(8)=T3X:ME(9)=T45
3400 REM ME(10)=NG ME(11)=NP ME(12)=TORQLD
3410 VA(I,1) = ME(I,4) / ME(I,3):VA$(1) = "PS3Q2"
3420 VA(I,2) = ME(I,10) / SQR (ME(I,6) / 518.67) / C(8):VA$(
(2) = "PCNGC"
3430 VA(I,3) = ME(I,8) / ME(I,6):VA$(3) = "T3Q2"
3440 VA(I,4) = ME(I,1) * ME(I,3) / SQR (ME(I,6) / 518.67) /
14.696:VA$(4) = "WA2"
3450 VA(I,5) = ME(I,2) / 3600.:VA$(5) = "WF"
3460 VA(I,6) = ME(I,4) / C(1):VA$(6) = "P3"
3470 T = ME(I,6):FA = 0:PI = 0: GOSUB 160:VA(I,7) = H:VA$(7)
= "H2"
3480 T = ME(I,7): GOSUB 160:VA(I,8) = H:VA$(8) = "H25"
3490 T = ME(I,8): GOSUB 160:VA(I,9) = H:VA$(9) = "H3"
3500 NC = 2:XI = VA(I,2):IT = 0: GOSUB 500:VA(I,10) = YC:VA$(
(10) = "B1"
3510 NC = 8:XI = ME(I,1): GOSUB 500:VA(I,11) = YC:VA$(11) =
"B2"

```

```

3520 NC = 4:XI = ME(I,1): GOSUB 500:VA(I,12) = YC:VA$(12) =
"WXQ2"
3530 VA(I,13) = VA(I,4) * (1. - VA(I,10) - VA(I,11)):VA$(13)
= "WA3"
3540 VA(I,14) = VA(I,13) - VA(I,4) * (VA(I,12) + C(3)):VA$(1
4) = "WA31"
3550 VA(I,15) = VA(I,14) + VA(I,5):VA$(15) = "W41"
3560 VA(I,16) = VA(I,15) + C(4) * VA(I,12) * VA(I,4):VA$(16)
= "W45"
3570 VA(I,17) = C(5) * (VA(I,13) * (VA(I,9) - VA(I,8)) + VA(
I,4) * (VA(I,8) - VA(I,7))) / ME(I,10):VA$(17) = "TORQC"
3580 VA(I,18) = VA(I,5) / (VA(I,16) - VA(I,5)):VA$(18) = "FA
R45"
3590 VA(I,19) = ME(I,12) * ME(I,11) / C(5) / VA(I,16):VA$(19
) = "DH45"
3600 VA(I,20) = VA(I,17) * ME(I,10) / C(5) / VA(I,15):VA$(20
) = "DH41"
3610 VA(I,21) = VA(I,5) / VA(I,14):VA$(21) = "FAR41"
3620 T = ME(I,9):FA = VA(I,18): GOSUB 160:VA(I,22) = H:VA$(2
2) = "H45"
3630 VA(I,23) = VA(I,22) - VA(I,19):VA$(23) = "H49"
3640 HD = VA(I,23):FA = VA(I,18):PI = 1: GOSUB 440:VA(I,24)
= T:VA$(24) = "T49"
3650 VA(I,25) = VA(I,16) * (VA(I,22) - VA(I,8)) / VA(I,15) +
VA(I,8):VA$(25) = "H44"
3660 VA(I,26) = VA(I,22) / VA(I,25):VA$(26) = "H45Q4"
3670 VA(I,27) = VA(I,25) + VA(I,20):VA$(27) = "H41"
3680 VA(I,28) = C(6) * ME(I,9) + C(7):VA$(28) = "THTA45"
3690 VA(I,29) = VA(I,19) / VA(I,28):VA$(29) = "DHQTH5"
3700 HD = VA(I,27):FA = VA(I,21): GOSUB 440:VA(I,30) = T:VA$
(30) = "T41"
3710 VA(I,31) = C(6) * VA(I,30) + C(7):VA$(31) = "THTA41"
3720 VA(I,32) = VA(I,15) * SQR (VA(I,31)) / C(2):VA$(32) =
"P41"
3730 VA(I,33) = VA(I,20) / VA(I,31):VA$(33) = "DHQTH4"
3740 TH = VA(I,28):W = VA(I,16):PB = ME(I,5):NC = 7: GOSUB 6
00:VA(I,34) = P:VA(I,35) = WC:VA$(34) = "P45":VA$(35) = "W45
C"
3750 VA(I,36) = (VA(I,6) * VA(I,6) - VA(I,6) * VA(I,32)) / V
A(I,14) / VA(I,14) / ME(I,8):VA$(36) = "KDPBNR"
3760 RETURN

```

note - Variables referenced in lines 10-1790 are not user-defined

A\$ 1110, 1160
 AM 280, 290
 BC\$ 110, 730, 740, 780
 BR 880, 890, 910, 920, 1000, 1190, 1320, 1480, 1550
 C 1190
 CA 200, 230, 260, 340
 CF 310, 340
 CP 340, 350, 360
 CV 350, 360
 C(3050, 3060, 3070, 3080, 3090, 3100, 3110, 3120, 3130, 3140,
 3420, 3460, 3540, 3560, 3570, 3590, 3600, 3680, 3710, 3720
 D\$ 70, 80, 730, 740, 780, 790, 800, 820, 930, 970, 1090, 1100,
 1300, 1320, 1400, 1430, 1440, 1460, 1630, 1660, 1680
 E 200, 210, 230, 240, 260, 270, 310, 320, 3110
 E\$ 1110, 1180
 EP\$ 1040, 1170
 ER 880, 900, 910, 920, 1000, 1190, 1320, 1480, 1550
 FA 280, 330, 340, 3470, 3620, 3640, 3700
 G 1140
 GM 360
 H 330, 380, 390, 1140, 3470, 3480, 3490, 3620
 HA 210, 240, 270, 330
 HD 380, 390, 450, 3640, 3700
 HF 320, 330
 I 920, 930, 950, 960, 1000, 1190, 1200, 1220, 1260, 1290, 132
 0, 1330, 1340, 1350, 1360, 1380, 1550, 1570, 1580, 1600, 1610, 34
 10, 3420, 3430, 3440, 3450, 3460, 3470, 3480, 3490, 3500, 3510, 3
 520, 3530, 3540, 3550, 3560, 3570, 3580, 3590, 3600, 3610, 3620,
 3630, 3640, 3650, 3660, 3670, 3680, 3690, 3700, 3710, 3720, 3730,
 3740, 3750
 I\$ 1110, 1150, 1160, 1220, 1260
 IC 850, 870
 ID 3040
 IT 3500
 J 940, 950, 960, 1210, 1220, 1230, 1250, 1260, 1270, 1340, 13
 50, 1360, 1370, 1520, 1540, 1580, 1600, 3300, 3310, 3320, 3330
 K 120, 130, 520, 530, 550, 560, 770
 KL 760
 KN\$ 760, 1200, 1330
 KY\$(810, 930, 1200, 1330, 1570
 L 510, 520, 530
 M 510
 M\$ 100, 730, 740, 780, 790, 800, 820, 830, 930, 970
 ME(810, 950, 1220, 1340, 1580, 3410, 3420, 3430, 3440, 3450, 3
 460, 3470, 3480, 3490, 3510, 3520, 3570, 3590, 3600, 3620, 3680,
 3740, 3750
 ME\$(810, 950
 ML(760, 770
 MN\$(760, 770, 1220, 1340, 1520
 NC 510, 520, 530, 560, 3050, 3500, 3510, 3520, 3740
 NM 750, 760, 770, 810, 940, 1210, 1340, 1360, 1470, 1520, 1580
 NV 1250, 1360, 1470, 1540, 1600, 3360
 N(510, 3170, 3300, 3310, 3320, 3330
 P 620, 650, 3740

PB 620, 3740
 PI 370, 3470, 3640
 PR 610, 620, 670
 R 290, 350
 R\$ 50, 1080, 1400, 1660
 RE 810, 830, 880, 3360
 RL 750, 790, 930
 RP 1130, 1190
 RX 910
 S\$ 1500, 1510, 1520, 1530, 1540, 1560, 1570, 1580, 1590, 1600,
 1620, 1760
 SP\$ 1360
 T 170, 180, 190, 200, 210, 230, 240, 260, 270, 300, 320, 300,
 1500, 1510, 1520, 1530, 1540, 1560, 1570, 1580, 1590, 1600, 1620
 , 1750, 3470, 3480, 3490, 3620, 3640, 3700
 T1 1130, 1180
 T2 170, 200, 210, 230, 240, 260, 270, 320, 1130, 1180
 T3 1130, 1180
 TD 300, 310, 450
 TH 650, 3740
 TI\$ 1060, 1120
 TN 1480, 1710
 TS 1120
 V 1500, 1510, 1520, 1530, 1540, 1560, 1570, 1580, 1590, 1600,
 1620, 1750
 V\$ 1450, 1460, 1630, 1680
 VA(1260, 1360, 1600, 3360, 3410, 3420, 3430, 3440, 3450, 3460,
 3470, 3480, 3490, 3500, 3510, 3520, 3530, 3540, 3550, 3560, 3570
 , 3580, 3590, 3600, 3610, 3620, 3630, 3640, 3650, 3660, 3670, 368
 0, 3690, 3700, 3710, 3720, 3730, 3740, 3750
 VA\$(1260, 1360, 1540, 3360, 3410, 3420, 3430, 3440, 3450, 3460,
 3470, 3480, 3490, 3500, 3510, 3520, 3530, 3540, 3550, 3560, 3570
 , 3580, 3590, 3600, 3610, 3620, 3630, 3640, 3650, 3660, 3670, 368
 0, 3690, 3700, 3710, 3720, 3730, 3740, 3750
 VN 1470, 1700
 W 660, 670, 3740
 WC 640, 650, 3740
 WP 650, 660, 670
 XI 520, 530, 560, 620, 3500, 3510, 3520
 X(520, 530, 560, 3170, 3300, 3310, 3320, 3330
 Y\$ 1010, 1020, 1050, 1060, 1120, 1390, 1400, 1410
 YC 530, 560, 640, 3500, 3510, 3520
 Y(530, 560, 3170, 3300, 3310, 3320, 3330

ORIGINAL PAGE IS
 OF POOR QUALITY

SEARCH HEX CODE

*302.371

ORIGINAL PAGE IS
OF POOR QUALITY

0302- 20 E3 DF 85 06 84
0308- 07 A0 00 84 1A B1 06 85
0310- 18 C8 B1 06 48 C8 B1 06
0318- 85 07 68 85 06 20 BE DE
0320- 20 E3 DF 85 08 84 09 A0
0328- 00 B1 08 85 19 C8 B1 08
0330- 48 C8 B1 08 85 09 68 85
0338- 08 A5 18 F0 1E C5 19 90
0340- 1A A5 19 F0 16 A5 18 38
0348- E5 19 85 1B A0 00 B1 06
0350- D1 08 D0 08 C8 C4 19 90
0358- F5 E6 1A 60 E6 1A A5 1B
0360- C5 1A 90 08 E6 06 D0 02
0368- E6 07 D0 E0 A9 00 85 1A
0370- F0 E9

*94A0.95F4

**ORIGINAL PAGE IS
OF POOR QUALITY**

94A0- 20 2E FB BA BD 00 01 85
 94A8- 74 85 0C BD FF 00 85 73
 94B0- 85 0B A9 4C 85 0A 38 A5
 94B8- 73 E9 02 85 73 A5 74 E9
 94C0- 00 85 74 18 A5 0B 69 2F
 94C8- 85 0B A5 0C 69 00 85 0C
 94D0- 60 A0 03 A2 0C 20 2B EB
 94D8- 20 B7 00 38 E9 27 D0 2D
 94E0- 20 B1 00 20 67 DD 20 FB
 94E8- E6 8A 8D 09 03 20 4C E7
 94F0- 8A 8D 0A 03 A9 27 20 C0
 94F8- DE AD 09 03 38 E9 02 B0
 9500- 04 20 F2 E6 60 AD 09 03
 9508- 38 E9 15 B0 F4 A0 03 A9
 9510- 0C 20 F9 EA A9 00 8D 08
 9518- 03 A5 A2 10 08 A9 01 8D
 9520- 08 03 20 AF EB AD 0A 03
 9528- F0 0B 8D 0B 03 20 39 EA
 9530- CE 0B 03 D0 F8 20 A0 E7
 9538- 20 23 EC A9 00 8D 0B 03
 9540- A0 03 A2 0C 20 2B EB 20
 9548- 55 EA 20 23 EC A0 03 A2
 9550- 11 20 2B EB 20 39 EA A0
 9558- 03 A9 0C 20 A7 E7 20 F2
 9560- EB A5 A1 09 B0 AE 0B 03
 9568- 9D 16 03 A0 03 A9 11 20
 9570- F9 EA AE 0B 03 E8 8A CD
 9578- 0A 03 F0 1E EE 0B 03 AE
 9580- 09 03 CA 8A CD 0B 03 B0
 9588- B7 20 82 EB F0 1B AE 09
 9590- 03 A9 AA 9D 15 03 A9 00
 9598- F0 45 EE 0B 03 AE 0B 03
 95A0- A9 AE 9D 16 03 A9 00 F0
 95A8- D3 AE 0A 03 F0 01 EB 8E
 95B0- 0B 03 AE 09 03 CA BD 16
 95B8- 03 38 E9 B0 D0 0B A9 A0
 95C0- 9D 16 03 CA EC 0B 03 D0
 95C8- ED AD 0B 03 F0 11 E8 8E
 95D0- 0B 03 AD 09 03 CD 0B 03
 95D8- F0 B4 A9 AD 9D 16 03 AE
 95E0- 09 03 CA BD 16 03 20 ED
 95E8- FD CA D0 F7 AD 16 03 29
 95F0- 0F 20 93 EB 60

APPENDIX B - APPLESOFT II - (EXTENDED FLOATING POINT BASIC

Many engineers and scientists might not be familiar with Applesoft BASIC. As a result, many potential users of BDMADS might shy away, knowing that they would have to program their calculations in Applesoft. Rather than expect them to obtain formal BASIC training, it was decided to include a brief summary of Applesoft features in this paper. It is hoped that this Appendix will permit a user to begin writing Applesoft routines. As a user gains more experience with the language, he will be able to write his routines more efficiently, making use of all of the power of the language (ref. 2).

The following paragraphs describe salient features of Applesoft BASIC:

There are three types of simple variables in Applesoft - real (9.99999999E+37), integer (32767), and string (0 to 255 characters long). Only the first two characters in the variable name are significant. Integers are denoted by a trailing % (e.g. A%). Strings are denoted by a trailing \$. One, two, and three-dimensional arrays are permitted with the array size limited by available memory. String values are enclosed in quotes (e.g. "ABC").

The usual algebraic operators are available. They are:

= assignment (LET is optional)
- negation
* multiplication
/ division
+ addition
- subtraction

In addition, exponentiation is denoted by ^ (e.g. X^2).

The following relational and logical operators are available:

= equal
<> not equal
< less than
> greater than
<= less than or equal
>= greater than or equal

NOT logical "NOT"
AND logical "AND"
OR logical "OR"

Relational and logical expressions have value 1 if true and 0 if false. Relational operators can also be used to compare strings.

All variables are global. There are no COMMON blocks in

Applesoft.

Branches to subroutines are indicated by GOSUB N where N is the first line number in the subroutine. Subroutines must end with a RETURN statement. The RETURN will cause the next executable statement following the GOSUB to be executed. Statements can be combined under a single statement number by separating the statements with a colon (:).

The following math functions are available:

SIN(X)	sine of X radians
COS(X)	cosine of X radians
TAN(X)	tangent of X radians
ATN(X)	arctangent of X, in radians
INT(X)	largest integer less than or equal to X
RND(1)	random number (from 0 to .999999999)
RND(0)	returns last random number again
SGN(X)	-1 if X < 0, 0 if X = 0, 1 if X > 0
ABS(X)	absolute value of X
SQR(X)	positive square root of X
EXP(X)	e (2.718289) to the power X
LOG(X)	natural logarithm of X

A user may define a function for later use in his program. For example,

```
10 DEF FN A(X)=X+23/X
.
.
50 Y=FNA(4)
```

In this case, Y would take on the value 9.75.

Applesoft provides a number of commands for controlling the flow of a program. These include:

GOTO N	branch to line N
IF..THEN..	If the expression following the IF is true, then the statements following the THEN are executed. Otherwise, the next numbered line is executed.
FOR X=1 TO 20 STEP 4..NEXT X	Executes all statements between the FOR statement and the corresponding NEXT. In this case, X takes on the values 1,5,9,13,17,21. As soon as X exceeds the specified final value (20), execution continues after the NEXT. If not specified, the step size is 1.
ON X GOTO A,B,C..	Branches to the Xth line number in the list. If X exceeds the number of items in the list, execution continues at the next statement.
ON X GOSUB A,B,C..	Branches to the subroutine at the Xth line number in the list.

ORIGINAL PAGE IS
OF POOR QUALITY

ONERR GOTO N

Subsequent errors cause a branch to an error-handling routine at line N instead of displaying a system message and halting the program.

The following input/output commands are available in Applesoft:

INPUT A	puts ? on screen and waits for user to type a value for A.
INPUT "XYZ";A	prints XYZ on screen and waits for user to type a value for A
GET A	waits for user to type a one-character value for A.
DATA 5,"Y",10..	establishes a list of data elements that can be read. In this case, a real number (5), a string value (Y), and another real number (10) are defined.
READ X,A\$,Z	assigns next 3 data elements to variables X, A\$, and Z.
RESTORE	starts READING again from first data element in program.
PRINT A\$;X;Y	prints value of string variable A\$, immediately followed by values of real variables X and Y.
PRINT X,Y,Z	prints values of variables X, Y, and Z in three fixed tab fields.
PR#N	sends output to peripheral device in slot N instead of screen (slot 0). Usually, PR#1 will send output to printer and PR#6 will send output to disk.

The following array and string commands and functions are available in Applesoft:

DIM A(X,Y,Z)	sets maximum subscripts for A. Reserves (X+1)(Y+1)(Z+1) real elements starting at A(0,0,0). The array can be dimensioned anywhere in the program, preceding the first reference to the array, but must not be redimensioned.
LEN(A\$)	returns number of characters in string A\$.
STR\$(X)	returns numeric value of X, converted to a string.
VAL(A\$)	returns A\$, up to first non-numeric character, as a number.
CHR\$(X)	returns ASCII character whose code is X.
ASC(A\$)	returns ASCII code for first character of A\$.
LEFT\$(A\$,X)	returns leftmost X characters of A\$.

ORIGINAL PAGE IS
OF POOR QUALITY

RIGHT\$(A\$,X)	returns rightmost X characters of A\$.
MID\$(A\$,X,Y)	returns Y characters of A\$ starting at Xth character.
A\$+B\$	concatenates strings A\$ and B\$.

The following editing and format-related commands are available in Applesoft:

LIST	lists entire program.
LIST X,Y	lists lines X through Y.
DEL X,Y	deletes lines X through Y.
REM..	remark; is ignored by program.
VTAB Y	moves cursor to line Y (1 to 24).
HTAB X	moves cursor to position X in current line (1 to 40).
TAB(X)	in PRINT statement, moves cursor to position X (1 to 40).
POS(O)	returns current horizontal position of cursor.
SPC(X)	in PRINT statement, puts X spaces between last and next item.
HOME	clears screen, puts cursor at top, left (1,1).
CLEAR	resets all variables to zero.
FLASH	sets subsequent computer output to flashing.
INVERSE	sets subsequent computer output to black on white.
NORMAL	turns off flashing and/or inverse output.
SPEED=X	sets character output rate (0 to 255).
esc A	moves cursor one space right.
esc B	moves cursor one space left.
esc C	moves cursor one space down.
esc D	moves cursor one space up.
right arrow	enters character under cursor and moves cursor one space right.
left arrow	deletes character under cursor and moves cursor one space left.
ctrl X	Cancels line currently being typed.

APPENDIX C - USE OF VISICALC WITH BDMADS

Current versions of the electronic spreadsheet program, VisiCalc provide for reading of data files that conform to Software Arts Inc.'s Data Interchange Format (DIF). The BDMADS.CALC program is designed to generate DIF files containing keys, measurements, variables, and their names. This Appendix illustrates how VisiCalc commands can be used to improve the readability of the jet engine data contained in the file ENG1.DIF. VisiCalc also provides the capabilities to interactively perform additional calculations on any or all of the data. However, no attempt will be made at this time to demonstrate those capabilities. The purpose of this Appendix is merely to show how BDMADS can communicate data to VisiCalc and how a few simple VisiCalc commands can be used to improve the readability of the data.

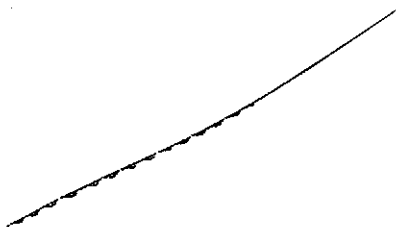
VisiCalc is loaded and executed by booting the VisiCalc program diskette. After VisiCalc is loaded, it displays a blank spreadsheet on the screen and awaits commands from the user. To load a DIF file from a diskette in the drive, the user enters the compound command /S#L followed by the file name. After typing the file name, followed by a RETURN, the user is asked whether the data should be entered into the table by row or column. Because of the way the DIF file is constructed by BDMADS.CALC, the user should press R for row. The DIF file will be loaded and the filled-in spreadsheet will be displayed. For a discussion of VisiCalc and its commands, the reader is referred to the VisiCalc User's Manual (reference 3).

Figure 4 shows the jet engine data as it should appear in the spreadsheet after loading of ENG1.DIF. The VisiCalc column headings (A,B,C,...BK) and row headings (1,2,3,...254) have been omitted from the figure. One of the powerful capabilities of VisiCalc is its facility to easily reformat the display. For example, to obtain the modified display, shown in figure 5, one merely has to issue a series of commands at the keyboard. These include positioning of the cursor over a particular row or column and issuing the commands /IR to insert a blank row or /IC to insert a blank column. The command /GC6 changes the column width from the default width of 8 characters to 6 characters. VisiCalc automatically rounds off the entries as the column width is changed. The original data are not lost in this process and the user can experiment with various column widths to get the desired results. The labels READING and CALCULATED VALUES were merely typed in after positioning the cursor. Since VisiCalc allows for 254 rows in the table, additional calculations can be performed with the equations written in terms of the coordinates of the entries in the table.

To obtain a hard-copy printout of the spreadsheet, the user positions the cursor at the upper left corner of the area to be printed (A1 in our example) and issues the commands /P. VisiCalc prompts for the slot number of the printer interface card (1) followed by a RETURN. The user then types a minus sign (-)

followed by the location of the lower right corner of the area to be printed (H55) and a RETURN. The minus sign suppresses a line feed normally issued by VisiCalc following the printing of a line. It should be noted that VisiCalc also allows for the output of special printer control characters to print emphasized characters, condensed characters, etc.

ORIGINAL PAGE IS
OF POOR QUALITY



REFERENCES

1. Espinosa, C. : Apple II Reference Manual. Apple Computer Inc., 1979.
2. Applesoft II Basic Programming Reference Manual. Apple Computer Inc., 1978.
3. Bricklin, D. and Frankston, B. : VisiCalc Computer Software Program. Personal Software Inc., 1979.
4. Apple II - The DOS Manual. Apple Computer Inc., 1980.
5. Reynolds, L. : Applesoft Sub-string Search Function. Call A.P.P.L.E., Jan. 1981, pp.26+.
6. Baldwin, J.R. : Print USRing. NIBBLE, vol.3 no.6, 1982, pp.165-166.
7. Brickley, T.C. : A.I.M. File Utility. NIBBLE, vol.2 no.8, 1981, pp.101+.
8. Programmer's Guide to the Data Interchange Format. Technical Note SATN-18, Software Arts Inc., 1980.
9. Weinstock, M.D. : Apple A.I.M.. NIBBLE, vol.1 no.7, 1980, pp.9+.
10. Dettterich, B.C. : Epson MX100. Call A.P.P.L.E., Feb. 1982, pp.45+.
11. The GRAPPLER Operator's Manual. Orange Micro Inc., 1982.

Table I - Jet Engine Measurements

WA2CX	compressor inlet corrected flow	lbm/sec
WFPH	fuel flow	lbm/hr
P2	compressor inlet total pressure	psia
PS3	compressor exit static pressure	psia
P49	power turbine exit total pressure	psia
T2	compressor inlet total temperature	deg.R
T25	compressor bleed total temperature	deg.R
T3X	compressor exit total temperature	deg.R
T45	power turbine inlet tot. temp.	deg.R
NG	compressor speed	rpm
NP	power turbine speed	rpm
TORQLD	load torque	ft-lbf

Table II - Jet Engine Calculated Variables

ORIGINAL PAGE IS
OF POOR QUALITY

PS3Q2	compressor pressure ratio	
PCNGC	compressor corrected speed	percent
T3Q2	compressor temperature ratio	
WA2	compressor inlet airflow	lbm/sec
WF	fuel flow	lbm/sec
P3	compressor inlet total pressure	psia
H2	compressor inlet specific enthalpy	BTU/lbm
H25	compressor bleed specific enthalpy	BTU/lbm
H3	compressor exit specific enthalpy	BTU/lbm
B1	speed-sensitive bleed coefficient	
B2	flow-sensitive bleed coefficient	
WXQ2	turbine cooling bleed coefficient	
WA3	compressor exit airflow	lbm/sec
WA31	combustor inlet airflow	lbm/sec
W41	gas generator turbine inlet flow	lbm/sec
W45	power turbine inlet flow	lbm/sec
TORQC	compressor torque	ft-lbf
FAR45	power turbine inlet fuel-air ratio	
DH45	power turbine enthalpy drop	BTU/lbm
DH41	gas generator turbine enth.drop	BTU/lbm
FAR41	gas generator inlet fuel-air ratio	
H49	power turbine exit specific enth.	BTU/lbm
H45	power turbine inlet specific enth.	BTU/lbm
T49	power turbine exit total temp.	deg.R
H44	gas generator turbine exit specific enthalpy	BTU/lbm
H45Q5	ratio of specific enthalpies at stations 45 and 44	
H41	gas generator turbine inlet specific enthalpy	BTU/lbm
THTA45	squared critical velocity ratio at station 45	
DHQTH5	power turbine equivalent enthalpy drop	BTU/lbm
T41	gas generator turbine inlet total temperature	deg.R
THTA41	squared critical velocity ratio at station 41	
P41	gas generator turbine inlet total pressure	psia
DHQTH4	gas generator equivalent enthalpy drop	BTU/lbm
P45	power turbine inlet total pressure	psia
W45C	power turbine equivalent flow	lbm/sec
KDPBNR	combustor pressure drop coeff.	$\text{lb}^2\text{-sec}^2/\text{lbm}^2\text{-in}^4\text{-deg.R}$

Table III - Jet Engine Equations

PS3Q2	=	PS3/P2
PCNGC	=	NG / (T2/518.67) ^{1/2} / C ₈
T3Q2	=	T3X/T2
WA2	=	WA2C P2 / (T2/518.67) ^{1/2} / 14.696
WF	=	WFPH/3600
P3	=	PS3/C ₁
H2	=	PROCOM(T2, 0.0)
H25	=	PROCOM(T25, 0.0)
H3	=	PROCOM(T3, 0.0)
B1	=	FUN1(2, PCNGC)
B2	=	FUN1(8, WA2C)
WXQ2	=	FUN1(4, WA2C)
WA3	=	WA2(1-B1-B2)
WA31	=	WA3-WA2(WXQ2+C ₃)
W41	=	WA31+WF
W45	=	W41+C ₄ WXQ2 WA2
TORQ45	=	C ₅ [WA3(H3-H25)+WA2(H25-H2)] / NG
FAR45	=	WF / (W45-WF)
DH45	=	TORQ45 NP / C ₅ / W45
DH41	=	TORQ41 NG / C ₅ / W41
FAR41	=	WF / WA31
H45	=	PROCOM(T45, FAR45)
H49	=	H45-DH45
T49	=	PROCOM(H49, FAR45)
H44	=	W45(H45-H25) / W41+H25

ORIGINAL PAGE IS
OF POOR QUALITY

Table III - continued

$$H45Q4 = H45/H44$$

$$H41 = H44+DH41$$

$$THTA45 = C_6 T45+C_7$$

$$DHQTH5 = DH45/THTA45$$

$$T41 = \text{PROCOM}^{-1}(H41, FAR41)$$

$$THTA41 = C_6 T41+C_7$$

$$P41 = W41(THTA41)^{1/2} / C_2$$

$$DHQTH4 = DH41/THTA41$$

P45	}	P49Q4 = P49/P45
		W45C = FUN1(7, P49Q4)
W45C		W45 = W45C P45/THTA45 ^{1/2}

Table IV - Jet Engine Measurements

READING NO.	1	2	3	4	5	6
WA2CX	3.2676	5.329	6.3298	7.1391	7.9384	8.8772
WFPH	140.00	297.24	372.03	458.44	560.59	694.38
P2	14.366	14.175	14.161	14.091	14.021	13.920
PS3	50.100	105.20	130.60	150.30	172.20	200.40
P49	14.368	14.426	14.457	14.596	14.631	14.719
T2	516.71	515.59	508.34	507.99	507.18	507.17
T25	648.37	743.11	750.26	802.30	861.20	880.51
T3X	842.10	1014.7	1080.6	1115.4	1180.1	1235.4
T45	1400.3	1571.6	1615.3	1710.1	1850.4	1950.2
NG	29470.	37770.	39005.	39873.	41500.	42673.
NP	10995.	19995.	19995.	19995.	19995.	19995.
TORQLD	30.080	90.050	148.26	206.47	274.25	360.79

Table V - Array Definitions for Jet Engine Calculations

ME(I,1)	=	WA2CX			
ME(I,2)	=	WFPH			
ME(I,3)	=	P2			
ME(I,4)	=	PS3			
ME(I,5)	=	P49			
ME(I,6)	=	T2			
ME(I,7)	=	T25			
ME(I,8)	=	T3X			
ME(I,9)	=	T45			
ME(I,10)	=	NG			
ME(I,11)	=	NP			
ME(I,12)	=	TORQLD			
VA(I,1)	=	PS3Q2	VA(I,19)	=	DH45
VA(I,2)	=	PCNGC	VA(I,20)	=	DH41
VA(I,3)	=	T3Q2	VA(I,21)	=	FAR41
VA(I,4)	=	WA2	VA(I,22)	=	H45
VA(I,5)	=	WF	VA(I,23)	=	H49
VA(I,6)	=	P3	VA(I,24)	=	T49
VA(I,7)	=	H2	VA(I,25)	=	H44
VA(I,8)	=	H25	VA(I,26)	=	H45Q4
VA(I,9)	=	H3	VA(I,27)	=	H41
VA(I,10)	=	B1	VA(I,28)	=	THTA45
VA(I,11)	=	B2	VA(I,29)	=	DHQTH5
VA(I,12)	=	WXQ2	VA(I,30)	=	T41
VA(I,13)	=	WA3	VA(I,31)	=	THTA41
VA(I,14)	=	WA31	VA(I,32)	=	P41
VA(I,15)	=	W41	VA(I,33)	=	DHQTH4
VA(I,16)	=	W45	VA(I,34)	=	P45
VA(I,17)	=	TORQC	VA(I,35)	=	W45C
VA(I,18)	=	FAR45	VA(I,36)	=	KDPBNR

ORIGINAL PAGE IS
OF POOR QUALITY

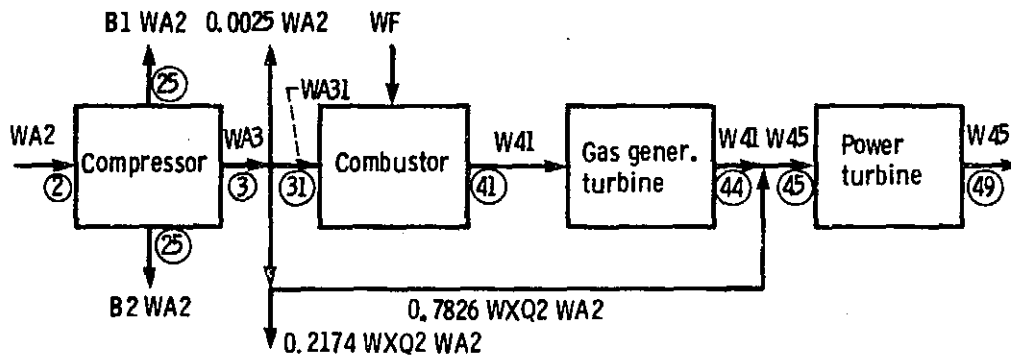


Figure 1. - Jet engine flow diagram.

BASIC DATA MANIPULATION AND DISPLAY
SYSTEM FOR A 48K APPLE II WITH DOS 3.3

FREECAT IS INSTALLED

DISK VOLUME 254 332 FREE SECTORS

```
*A 005 HELLO
*A 013 FILE UTILITY
*A 005 BDMADS.MAIN
*A 033 BDMADS.CALC
*B 002 SEARCH
*T 002 ENG1.RAF
  T 004 ENG1.RAF
  T 019 ENG1.DIF
*B 003 PRINTUSR
*A 078 BDMADS.DBM
```

Figure 2. - BDMADS catalog.

ENG1 MODEL VERIFICATION

READING NO. = 1

WA2CX	3.26760	WFPH	140.00000
P2	14.36600	PS3	50.10000
P49	14.36800	T2	516.71000
T25	648.37000	T3X	842.10000
T45	1400.30000	NG	29470.00001
NP	10995.00000	TDRQLD	30.08000
PS3Q2	3.48740	PCNQC	65.61298
T3Q2	1.62973	WA2	3.20028
WF	0.03889	P3	52.40586
H2	123.42300	H25	154.93567
H3	202.01451	B1	0.10900
B2	0.01036	WXQ2	0.08507
WA3	2.81830	WA31	2.53804
W41	2.57693	W45	2.79000
TDRQC	58.87303	FAR45	0.01414
DH45	15.95579	DH41	90.62413
FAR41	0.01532	H45	348.04040
H49	332.08461	T49	1340.39580
H44	364.00726	H45Q4	0.95614
H41	454.63139	THTA45	2.65179
DHQTH5	6.01699	T41	1789.12178
THTA41	3.36434	P41	53.95702
DHQTH4	26.93664	P45	17.21203
W45C	0.26414	KDPBNR	-0.01499

READING NO. = 2

WA2CX	5.32900	WFPH	297.24000
P2	14.17500	PS3	105.20000
P49	14.42600	T2	515.59000
T25	743.11000	T3X	1014.70000
T45	1571.60000	NG	37770.00001
NP	19995.00000	TDRQLD	90.05000
PS3Q2	7.42152	PCNQC	84.18366
T3Q2	1.96804	WA2	5.15541
WF	0.08257	P3	110.04184
H2	123.15661	H25	177.85309
H3	244.67046	B1	0.06961
B2	0.00900	WXQ2	0.08488
WA3	4.75016	WA31	4.29968
W41	4.38224	W45	4.72470
TDRQC	117.89706	FAR45	0.01779
DH45	51.29556	DH41	136.77370
FAR41	0.01920	H45	395.74587
H49	344.45031	T49	1382.11979
H44	412.77357	H45Q4	0.95875
H41	549.54727	THTA45	2.96571
DHQTH5	17.29619	T41	2115.77770
THTA41	3.96297	P41	99.58709
DHQTH4	34.51289	P45	23.42538
W45C	0.34752	KDPBNR	0.06133

Figure 3. - Jet engine printout.

READING NO. = 3

**ORIGINAL PAGE IS
OF POOR QUALITY**

WA2CX	6.32980	WFPH	372.03000
P2	14.16100	PS3	130.60000
P49	14.45700	T2	508.34000
T25	750.26000	T3X	1080.60000
T45	1615.30000	NG	39005.00001
NP	19995.00000	TORQLD	148.26000
PS3Q2	9.22251	PCNGC	87.55404
T3Q2	2.12574	WA2	6.16103
WF	0.10334	P3	136.61088
H2	121.43285	H25	179.59086
H3	261.13380	B1	0.00822
B2	0.00900	WXQ2	0.08479
WA3	6.05496	WA31	5.51719
W41	5.62053	W45	6.02934
TORQC	162.29193	FAR45	0.01744
DH45	66.17971	DH41	151.59639
FAR41	0.01873	H45	407.57017
H49	341.39045	T49	1371.11961
H44	424.15207	H45Q4	0.96091
H41	575.74845	THTA45	3.04580
DHQTH5	21.72820	T41	2207.17978
THTA41	4.13048	P41	130.39880
DHQTH4	36.70190	P45	29.00463
W45C	0.36264	KDPBNR	0.02580

READING NO. = 4

WA2CX	7.13910	WFPH	458.44000
P2	14.09100	PS3	150.30000
P49	14.59600	T2	507.99000
T25	802.30000	T3X	1115.40000
T45	1710.10000	NG	39873.00001
NP	19995.00000	TORQLD	206.47000
PS3Q2	10.66638	PCNGC	89.53326
T3Q2	2.19571	WA2	6.91678
WF	0.12734	P3	157.21757
H2	121.34966	H25	192.27363
H3	269.86710	B1	0.00000
B2	0.00900	WXQ2	0.08471
WA3	6.85453	WA31	6.25132
W41	6.37867	W45	6.83721
TORQC	190.50508	FAR45	0.01898
DH45	81.27349	DH41	160.28938
FAR41	0.02037	H45	434.44978
H49	353.17629	T49	1413.13838
H44	451.85893	H45Q4	0.96147
H41	612.14831	THTA45	3.21953
DHQTH5	25.24390	T41	2327.22534
THTA41	4.35047	P41	151.87771
DHQTH4	36.84413	P45	33.34428
W45C	0.36786	KDPBNR	0.01926

READING NO. = 5

WA2CX	7.93840	WFPH	560.59000
P2	14.02100	PS3	172.20000
P49	14.63100	T2	507.18000
T25	861.20000	T3X	1180.10000
T45	1850.40000	NG	41500.00001
NP	19995.00000	TORQLD	274.25000
PS3Q2	12.28158	PCNGC	93.26100
T3Q2	2.32679	WA2	7.65909
WF	0.15572	P3	180.12552
H2	121.15716	H25	206.70183
H3	286.17650	B1	0.00000
B2	0.00900	WXQ2	0.08463
WA3	7.59016	WA31	6.92279
W41	7.07851	W45	7.58581
TORQC	225.28298	FAR45	0.02096
DH45	97.30055	DH41	177.78032
FAR41	0.02249	H45	474.75583
H49	377.45528	T49	1500.00366
H44	493.96651	H45Q4	0.96111
H41	671.74683	THTA45	3.47664
DHQTH5	27.98693	T41	2522.33366
THTA41	4.70803	F41	175.33048
DHQTH4	37.76110	F45	38.12709
W45C	0.37099	KDPBNR	0.01527

READING NO. = 6

WA2CX	8.87720	WFPH	694.38000
P2	13.92000	PS3	200.40000
P49	14.71900	T2	507.17000
T25	880.51000	T3X	1235.40000
T45	1950.20000	NG	42673.00001
NP	19995.00000	TORQLD	360.79000
PS3Q2	14.39655	PCNGC	95.89798
T3Q2	2.43587	WA2	8.50325
WF	0.19288	P3	209.62343
H2	121.15478	H25	211.44903
H3	300.29170	B1	0.00000
B2	0.00900	WXQ2	0.08355
WA3	8.42672	WA31	7.69504
W41	7.88792	W45	8.44390
TORQC	264.01269	FAR45	0.02338
DH45	114.99583	DH41	192.24925
FAR41	0.02507	H45	504.49257
H49	389.49674	T49	1540.52947
H44	525.14768	H45Q4	0.96067
H41	717.39694	THTA45	3.65954
DHQTH5	31.42360	T41	2666.45529
THTA41	4.97215	F41	200.78456
DHQTH4	38.66525	F45	43.45021
W45C	0.37200	KDPBNR	0.02533

ORIGINAL PAGE IS
OF POOR QUALITY

	1	2	3	4	5	6
WA2CX	3.2676	5.329	6.3298	7.1391	7.9384	8.877200
WFPH	140	297.24	372.03	458.44	560.59	694.38
P2	14.366	14.175	14.161	14.091	14.021	13.92
PS3	50.1	105.2	130.6	150.3	172.2	200.4
P49	14.368	14.426	14.457	14.596	14.631	14.719
T2	516.71	515.59	508.34	507.99	507.18	507.17
T25	648.37	743.11	750.26	802.3	861.2	880.51
T3X	842.1	1014.7	1080.6	1115.4	1180.1	1235.4
T45	1400.3	1571.6	1615.3	1710.1	1850.4	1950.2
NG	29470	37770	39005	39873	41500	42673
NP	10995	19995	19995	19995	19995	19995
TORQLD	30.08	90.05	148.26	206.47	274.25	360.79
PSQ2	3.487401	7.421517	9.222513	10.66638	12.28158	14.39655
PCNGC	65.61298	84.18366	87.55404	89.53326	93.26100	95.89798
T3Q2	1.629734	1.968037	2.125743	2.195713	2.326787	2.435870
WA2	3.200278	5.155407	6.161028	6.916783	7.659093	8.503249
WF	.0388889	.0825667	.1033417	.1273444	.1557194	.1928833
P3	52.40586	110.0418	136.6109	157.2176	180.1255	209.6234
H2	123.4230	123.1566	121.4329	121.3497	121.1572	121.1548
H25	154.9357	177.8531	179.5909	192.2736	206.7018	211.4490
H3	202.0145	244.6705	261.1338	269.8671	286.1765	300.2917
B1	.109	.0696066	.0082155	0	0	0
B2	.0103578	.009	.009	.009	.009	.009
WXQ2	.0850748	.0848803	.0847859	.0847095	.0846341	.0835475
WA3	2.818300	4.750158	6.054963	6.854531	7.590161	8.426719
WA31	2.538036	4.299677	5.517192	6.251322	6.922793	7.695037
W41	2.576925	4.382244	5.620534	6.378667	7.078512	7.887920
W45	2.789998	4.724704	6.029339	6.837206	7.585809	8.443898
TORQC	58.87303	117.8971	162.2919	190.5051	225.2830	264.0127
FAR45	.0141357	.0177863	.0174387	.0189787	.0209579	.0233769
DH45	15.95579	51.29556	66.17971	81.27349	97.30055	114.9958
DH41	90.62413	136.7737	151.5964	160.2894	177.7803	192.2493
FAR41	.0153224	.0192030	.0187308	.0203708	.0224937	.0250659
H45	348.0404	395.7459	407.5702	434.4498	474.7558	504.4926
H49	332.0846	344.4503	341.3905	353.1763	377.4553	389.4967
T49	1340.396	1382.120	1371.120	1413.138	1500.004	1540.529
H44	364.0073	412.7736	424.1521	451.8589	493.9665	525.1477
H45Q4	.9561359	.9587481	.9609058	.9614722	.9611093	.9606680
H41	454.6314	549.5473	575.7485	612.1483	671.7468	717.3969
THTA45	2.651790	2.965714	3.045799	3.219529	3.476643	3.659537
DHQTH5	6.016989	17.29619	21.72820	25.24390	27.98693	31.42360
T41	1789.122	2115.778	2207.180	2327.225	2522.334	2666.455
THTA41	3.364345	3.962974	4.130478	4.350473	4.708029	4.972146
P41	53.95702	99.58709	130.3988	151.8777	175.3305	200.7846
DHQTH4	26.93664	34.51289	36.70190	36.84413	37.76110	38.66525
P45	17.21203	23.42538	29.00463	33.34428	38.12709	43.45021
W45C	.2641411	.3475179	.3626406	.3678585	.3709877	.372
KDPBNR	-.014986	.0613285	.0258001	.0192600	.0152717	.0253284

Figure 4. - VisiCalc table as loaded from DJI file.

READING

1 2 3 4 5 6

MEASUREMENTS

WA2CX	3.2676	5.329	6.3298	7.1391	7.9384	8.8772
WFFH	140	297.24	372.03	458.44	560.59	694.38
P2	14.366	14.175	14.161	14.091	14.021	13.92
PS3	50.1	105.2	130.6	150.3	172.2	200.4
P49	14.368	14.426	14.457	14.596	14.631	14.719
T2	516.71	515.59	508.34	507.99	507.18	507.17
T25	648.37	743.11	750.26	802.3	861.2	880.51
T3X	842.1	1014.7	1080.6	1115.4	1180.1	1235.4
T45	1400.3	1571.6	1615.3	1710.1	1850.4	1950.2
NG	29470	37770	39005	39873	41500	42673
NP	10995	19995	19995	19995	19995	19995
TDRQLD	30.08	90.05	148.26	206.47	274.25	360.79

CALCULATED VALUES

PS3Q2	3.4874	7.4215	9.2225	10.666	12.282	14.397
PCNGC	65.613	84.184	87.554	89.533	93.261	95.898
T3Q2	1.6297	1.9680	2.1257	2.1957	2.3268	2.4359
WA2	3.2003	5.1554	6.1610	6.9168	7.6591	8.5032
WF	.03889	.08257	.10334	.12734	.15572	.19288
F3	52.406	110.04	136.61	157.22	180.13	209.62
H2	123.42	123.16	121.43	121.35	121.16	121.15
H25	154.94	177.85	179.59	192.27	206.70	211.45
H3	202.01	244.67	261.13	269.67	286.18	300.29
B1	.109	.06961	.00822	0	0	0
B2	.01036	.009	.009	.009	.009	.009
WXQ2	.08507	.08488	.08479	.08471	.08463	.08355
WA3	2.8183	4.7502	6.0550	6.8545	7.5902	8.4267
WA31	2.5380	4.2997	5.5172	6.2513	6.9228	7.6950
W41	2.5769	4.3822	5.6205	6.3787	7.0785	7.8879
W45	2.7900	4.7247	6.0293	6.8372	7.5858	8.4439
TDRQC	58.873	117.90	162.29	190.51	225.28	264.01
FAR45	.01414	.01779	.01744	.01898	.02096	.02338
DH45	15.956	51.296	66.180	81.273	97.301	115.00
DH41	90.624	136.77	151.60	160.29	177.78	192.25
FAR41	.01532	.01920	.01873	.02037	.02249	.02507
H45	348.04	395.75	407.57	434.45	474.76	504.49
H49	332.08	344.45	341.39	353.18	377.46	389.50
T49	1340.4	1382.1	1371.1	1413.1	1500.0	1540.5
H44	364.01	412.77	424.15	451.86	493.97	525.15
H45Q4	.95614	.95875	.96091	.96147	.96111	.96067
H41	454.63	549.55	575.75	612.15	671.75	717.40
THTA45	2.6518	2.9657	3.0458	3.2195	3.4766	3.6595
DHQTH5	6.0170	17.296	21.728	25.244	27.987	31.424
T41	1789.1	2115.8	2207.2	2327.2	2522.3	2666.5
THTA41	3.3643	3.9630	4.1305	4.3505	4.7080	4.9721
P41	53.957	99.587	130.40	151.88	175.33	200.78
DHQTH4	26.937	34.513	36.702	36.844	37.761	38.665
P45	17.212	23.425	29.005	33.344	38.127	43.450
W45C	.26414	.34752	.36264	.36786	.37099	.372
KDFBNR	-.0150	.06133	.02580	.01926	.01527	.02533

ORIGINAL PAGE IS
OF POOR QUALITY.

Figure 5. - Modified VisiCalc table.

1. Report No. NASA TM-83328	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle BASIC DATA MANIPULATION AND DISPLAY SYSTEM (BDMADS)		5. Report Date February 1983	6. Performing Organization Code 505-32-6B
		8. Performing Organization Report No. E-1575	
7. Author(s) John R. Szuch		10. Work Unit No.	
9. Performing Organization Name and Address National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135		11. Contract or Grant No.	
		13. Type of Report and Period Covered Technical Memorandum	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		14. Sponsoring Agency Code	
		15. Supplementary Notes	
16. Abstract <p>BDMADS, a BASIC Data Manipulation and Display System, is a collection of software programs that run on an Apple II Plus personal computer. BDMADS provides a "user-friendly" environment for the engineer in which to perform scientific data processing. This report is intended as a guide to users of the BDMADS software. The report describes the computer programs and their use. Jet engine performance calculations are used to illustrate the use of BDMADS. Source listings of the BDMADS programs are provided and should permit users to customize the programs for their particular applications.</p>			
17. Key Words (Suggested by Author(s)) Microcomputer Software Data processing BASIC		18. Distribution Statement Unclassified - unlimited STAR Category 61	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages	22. Price*

* For sale by the National Technical Information Service, Springfield, Virginia 22161