

DATA BASE 1

SHAMMAS SOFTWARE SERVICES
1533F HONEY GROVE DRIVE
RICHMOND, VIRGINIA 23229

CASSETTE MANUAL

DISCLAIMER OF ALL WARRANTY AND LIABILITY

The author of the program makes no warranties, either expessed or implied, with respect to this manula or with respect to the software decribed in this manual, its quality, performance, merchantability or fitness for any particular purpose. DATA BASE 1 is sold "As Is". While every effort is made to make DATA BASE 1 versatile, in no event will Shammass Software Services and/or the author be liable for direct, indirect, incidental or consequential damages resulting from any defect in the software.

This manual is copyright 1982 by :

SHAMMAS SOFTWARE SERVICES

1533F Honey Grove Drive

Richmond, Virginia 23229

TABLE OF CONTENTS

FORWARD	1
INTRODUCTION TO DATA BASES	2
SMALL NUMERIC FILES	3
NEW FILES	4
ADD RECORD	6
CHANGE RECORD	8
DELETE RECORD	8
INSPECT RECORD	9
LOAD DATA	11
SAVE DATA	11
PRINT DATA	11
MEMORY CONFIGURATION	13
 LARGE NUMERIC FILES	 14
NEW FILE	15
OPEN FILE	17
ADD RECORD	18
DELETE RECORD	19
SEARCH RECORD	19
PRINT RECORD	20
EDIT RECORD	20
MERGE RECORD	22

MEMORY CONFIGURATION	28
MASS MEDIA FILE CONFIGURATION	29
ALPHA/NUMERIC FILES	30
NEW FORMAT	31
PRINT FORMAT	33
NEW FILE	35
OPEN FILE	36
EDIT-ADD RECORD	37
DELETE RECORD	43
MERGE RECORD	43
SEARCH RECORD	44
MEMORY CONFIGURATION	47
FORMAT FILE CONFIGURATION	48
DATA FILE CONFIGURATION	49
TABLE 1	50
TABLE 2	51

FORWARD

DATA BASE 1 is a package of programs that perform data management operations for the Hewlett-Packard HP41C/V handheld computer and its digital cassette drive.

This manual deals with the cassette version of revision 2.0 and consists of two major parts:

- (1) Strictly numeric data : This consists of a "small" numeric file program and a group of other programs that tackle "large" numeric files.
- (2) Alpha/numeric data : This consists of a group of programs ready to process more popular types of data, such as business records.

SYSTEM REQUIREMENTS:

- (1) HP41CV or HP41C with a quad memory module.
- (2) The HP-IL interface module (82160A).
- (3) The digital cassette drive (82161A).
- (4) The Extended Function/Memory Module (82180A).
- (5) One Extended Memory module (82181A).

INTRODUCTION TO DATA BASES

Before the user starts to use the DATA BASE 1 package we would like to present the following brief explanation about just what a data base is.

Data bases are to information as banks are to money. A bank is an organization that manages and safeguards our money. Through account numbers we are able to perform a variety of transactions in an orderly fashion. A "sloppy" bank will soon find itself out of business! Data bases are similar. They consist of collections of programs that essentially process our data in an orderly fashion which allows us to, on a later date, retrieve, alter, delete and change any part of the information saved (with no bad surprises, hopefully). Moreover, data bases pave the way for other non-data management software to access the stored data and carry out a multitude of calculations. An example is the regression software available from Shammis Software Services that use files created by DATA BASE 1.

Having mentioned that data bases operate in an orderly and systematic way, we come now to define the terms for the building blocks of data structure. Perhaps the following example will be best. Consider a mailing list (something we should all be familiar with) which is composed of at least a name, street address, city, state and zip code. For any person to send or receive mail the above items of information must be available. We will call each item a "field". A collection of fields will compose a "record". Each field is merely participating to define a unit piece of data, the record. A collection of records is a "file". A set of files makes a data base! Thus an entire mailing list is a file of records dealing with names and addresses (the fields). Mailing lists are not the ONLY data bases. This is what DATA BASE 1 will present: the ability to create and manipulate our own particular data base.

SMALL NUMERIC FILES

The first part of DATA BASE 1 is a numeric filing program for a small amount of numeric data. This is when all the data can fit in the HP41CV's memory (we also call it RAM).

When it comes to numeric data bases, we are essentially storing a "table" of numbers consisting of "rows" and "columns". The rows are the observations or data points as we will call them. The columns are the variables involved for which they will be noted as X1, X2, X3, etc. In data bases terminology the observations (i.e. rows) are the records and each of the columns is a data field.

The features of the small numeric data base portion are:

- (1) Creation of new files.
- (2) Adding, deleting, inspecting and changing records.
- (3) Inspecting and changing a particular field.
- (4) Menu driven programs.

USERS' INSTRUCTIONS

Please note that for the following:

- (1) A "R/S*" means to omit the [R/S] command if a printer is used.
- (2) When the quotes are used we are implying keying in the ALPHA mode.
- (3) Do not hesitate to use the catalog command (CAT 1) to inspect the programs residing in memory.

To load and run this portion of DATA BASE 1 carry out:

Perform a MASTER CLEAR by holding down the back arrow key and the pressing the [ON] key. The message "MEMORY LOST" should appear. Load the small numeric data base set-up program, SGET:

INPUT : "SGET"
FUNCTION : [XEQ] "READA"

The loaded program will perform initialization steps that will end up with storing the option programs in the Extended Memory. The program will show the following menu;

```
1/NEW FILE
2/ADD REC
3/CHANGE REC
4/DELETE REC
5/INSPCT REC
6/LOAD DATA
7/SAVE DATA
8/PRINT DATA
9/PEEK FIELD
10/QUIT
```

SELECT BY NUMBER

IT IS IMPORTANT TO POINT OUT THAT FOR EACH OPTION THE FOLLOWING HOLDS TRUE:

- TWO KEYS ARE OF SPECIAL USE, NAMELY THE [A] (SIGMA+) KEY THAT WILL ALLOW THE USER TO REDO OR RESTART THE CHOSEN OPTION, AND THE [E] (LN) KEY TO "EXIT" TO THE MENU.
- THE TWO KEYS ABOVE ARE ACTIVE WHETHER THE USER WISHES TO INTERRUPT THE PROGRAM BY PRESSING THE [R/S] KEY OR THE CHOSEN OPTION IS ALLOWED TO COMPLETE ITS TASK!

Here is what happens for every option in the menu:

(1) NEW FILE : This will enable the user to create a new file on the cassette with the ability to specify the maximum number of

records as well as the number of those to be entered soon after file creation.

(1.1) The first message that appears is,

DISPLAY : MAX DATA?

INPUT : enter maximum number of records to be entered in all and fits the available memory size, ND.

FUNCTION : [R/S]

(1.2) The program will ask for the number of variables (fields).

DISPLAY : NO. VARS?

INPUT : Enter the number of fields, NV.

FUNCTION : [R/S]

(1.3) The filename is required next.

DISPLAY : FILENAME?

INPUT : filename.

FUNCTION : [R/S]

(1.4) The program will ask for the accompanying status filename. A status file will enable the user to reset himself to the conditions allowing him proper data entry, especially after using other software that may alter the machine's status. Input here is optional and none is usually needed if the user does not understand clearly what status files do. If no input is made the program will resort to composing a status filename by chaining the data filename to an "S".

DISPLAY : STATUS FILE?

INPUT : (optional) status filename.

FUNCTION : [R/S]

DISPLAY : an audible beep.

As an example, we want to create a new data file, named "SML", of maximum ten records and three fields (variables). We will enter three records only from the table:

X1	X2	X3
-----	-----	-----
111	111	111
222	22	222
3	333	333

Here is what we do, taking things from ground zero.

- Perform a MASTER CLEAR

FUNCTION : [back arrow] and [ON] keys
 DISPLAY : MEMORY LOST

- Load the set-up program, SGET:

INPUT : "SGET"
 FUNCTION : [XEQ] "READA"

- When the menu appears select option number one. This is a good time to set the display format to one decimal.

DISPLAY : SELECT BY NUMBER
 FUNCTION : [shift] "FIX" 1
 INPUT : 1

- We will tell the program to create a file with ten records maximum and three fields, naming the file SML.

DISPLAY : MAX DATA?
 INPUT : 10
 FUNCTION : [R/S]
 DISPLAY : NO. VARS?
 INPUT : 3
 FUNCTION : [R/S]

DISPLAY : FILENAME?
 INPUT : "SML"
 FUNCTION : [R/S]
 DISPLAY : STATUS FILES?
 FUNCTION : [R/S]

The file SML has been created with an accompanying status file SSML. Now we will exit to the menu by pressing the [E] key.

FUNCTION : [E]

(2)ADD REC : This option allows the user to add records.

(2.1) If the created file is full with data the following "FILE

FULL" message will appear. Otherwise a loop will begin to prompt for the data entry, one field at a time.

DISPLAY : $X_{i,j}$?

INPUT : $X(i,j)$ (ith field and jth record)

FUNCTION : [R/S]

The loop end will be signaled with an audible beep.

Returning to our example, we will select option number two this time.

DISPLAY : SELECT BY NUMBER

INPUT : 2

FUNCTION : [R/S]

DISPLAY : $X_{1,1}$?

INPUT : 111

FUNCTION : [R/S]

DISPLAY : $X_{2,1}$?

INPUT : 111

FUNCTION : [R/S]

DISPLAY : $X_{3,1}$?

INPUT : 111

FUNCTION : [R/S]

FUNCTION : [A]

DISPLAY : $X_{1,2}$?

INPUT : 222

FUNCTION : [R/S]

DISPLAY : $X_{2,2}$?

INPUT : 22

FUNCTION : [R/S]

DISPLAY : $X_{2,3}$?

INPUT : 222

FUNCTION : [R/S]

FUNCTION : [A]

DISPLAY : $X_{1,3}$?

INPUT : 3

FUNCTION : [R/S]

DISPLAY : $X_{2,3}$?

INPUT : 333

FUNCTION : [R/S]

DISPLAY : X3,3?
INPUT : 333
FUNCTION : [R/S]

Notice that we have used the [A] (or Again Key) to repeat this option twice. Now we use the [E] key to exit to the menu.

FUNCTION : [E]

(3) CHANGE REC : To change a record we use this option to zoom on the desired record.

(3.1) The program will ask for the data element location,

DISPLAY : CHANGE RECORD NO. ?

INPUT : record number .

FUNCTION : [R/S]

(3.2) A loop will start to show the "old" values of each field in the record concerned and give an opportunity to either alter or keep each current field value.

DISPLAY : $X_i, i = (X_i, j)$

INPUT : (optional) new value for $X(i, j)$.

FUNCTION : [R/S]

The loop will end with a beep.

(4) DELETE REC : To delete one or more consecutive records. The program will check if the deleted records extend to the last one. If so then only the number of records is altered, otherwise the program will copy an appropriate number of records from the last portion of the file to replace the deleted ones.

DISPLAY : DELETE FROM REC NO.?

INPUT : Enter first record number to be deleted.

FUNCTION : [R/S]

DISPLAY : TO REC NO. ?

INPUT : Enter last record number. If no entry is made, one record is deleted, the one specified above.

FUNCTION : [R/S]

The deletion will end with a beep.

(5) INSPCT REC : To inspect (i.e. view) one or more records.

DISPLAY : INSPECT REC?

INPUT : (first record number)+(last record number)/1000

FUNCTION : [R/S]

A loop will begin to view/print the records.

DISPLAY : FILE : (filename)

FUNCTION : [R/S]*

DISPLAY : X1,1 = (X1,1)

FUNCTION : [R/S]

DISPLAY : X2,1 = (X2,1)

FUNCTION : [R/S]

DISPLAY :
.....

Let us return to our example. We will proceed to view the data.
We select option number five from the menu.

- We will display records one to three. The steps below will
assume that a printer is used. Use the [R/S] key to view the
displayed results in sequence if no printer is used.

DISPLAY : INSPECT REC?

INPUT : 1.003

FUNCTION : [R/S]

DISPLAY : FILE : SML

DISPLAY : X1,1 =111.0

X2,1 =111.0

X3,1 =111.0

X1,2 =222.0

X2,2 =22.0

X3,2 =222.0

X1,3 =3.0

X2,3 =333.0

X3,3 =333.0

- At the end of the loop we exit to the main menu.

FUNCTION : [E]

- We want to delete record number one. We will select option four.

```
DISPLAY : DELETE FROM REC NO. ?  
INPUT : 1  
FUNCTION : [R/S]  
DISPLAY : TO REC NO. ?  
FUNCTION : [R/S]      <---- No entry is made
```

- A beep will signal the loop end. We will exit to the menu.

```
FUNCTION : [E]
```

- We will choose to change record two. Select option three from the menu.

```
DISPLAY : CHANGE RECORD NO. ?  
INPUT : 2  
FUNCTION : [R/S]
```

- We want to make record number two:

```
X(1,2)= 1  
X(2,2)= 2  
X(3,2)= 3
```

```
DISPLAY : X1,2 ?  
INPUT : 1  
FUNCTION : [R/S]  
DISPLAY : X2,2 ?  
INPUT : 2  
FUNCTION : [R/S]  
DISPLAY : X3,2 ?  
INPUT : 3  
FUNCTION : [R/S]
```

- A beep will signal the end of loop. We will exit to the menu.

```
FUNCTION : [E]
```

- We want to examine the two remaining records. Select option five from the menu. Use the [R/S] to view the displayed results if no printer is used.

```
DISPLAY : INSPECT REC?  
INPUT : 1.002  
FUNCTION : [R/S]
```

```
DISPLAY : FILE : SML
```

```
DISPLAY : X1,1= 3.0
```

X2,1= 333.0

X3,1= 333.0

X1,2= 1.0

X2,2= 2.0

X3,2= 3.0

- Exit to the main menu.

FUNCTION : [E]

(6) LOAD DATA : To load data from the cassette. This simply requires the filename.

DISPLAY : FILENAME?

INPUT : filename.

FUNCTION : [R/S]

DISPLAY : NDATA= (# of records)

FUNCTION : [R/S]

DISPLAY : NVAR=(# of variables or fields)

FUNCTION : [R/S]

DISPLAY : Audible beep. The data is loaded now.

(7) SAVE DATA : To save the data from the memory to the cassette. This consolidates all the new entries and changes done during the session. No user interaction is needed. A beep will signal successful data storage.

Back to our example. Select option seven and watch the cassette drive go to work. Exit back the the main menu.

FUNCTION : [E]

(8) PRINT DATA : This option is aimed at using the printer to obtain a printout of the data plus some file information.

PRINTOUT : R00= filename
 R01= ND (# of records)
 R02= NV (# of variables)
 R03= 9
 R04= Display code

R05=...
.....

(9) PEEK FIELD : To peek at and poke values in specific data elements. The latter is located by its record and field numbers.

DISPLAY : CHNG X(I,J)?

INPUT : field number.

FUNCTION : ENTER^

INPUT : record number.

FUNCTION : [R/S]

DISPLAY : OLD= (Xi,j)

INPUT : (optional) new value for X(I,J).

FUNCTION : [R/S]

DISPLAY : Audible beep.

(10) Quit : To exit the samll numeric data base menu.

DISPLAY : BYE

To end the example we will select option ten.

The user should not erase the file created for the example shown, because the data stored will be reused in the next section.

MEMORY CONFIGURATION

REC #	CONTENT
-----	-----
00	Data filename.
01	# of records, ND
02	# of variables, NV
03	Register storage shift, 9
04	Display code
05-07	Used
08	Max. # of records.
09	Used
10	X(1,1)
11	X(2,1)
12	etc...

LARGE NUMERIC FILES

This is the second portion of DATA BASE 1. it is similar to the small numeric files routines, except it takes into account the fact that the calculator's memory (RAM) will be able to store only a portion of the entire information. A new feature added is the ability to merge/append data files. This will allow the expansion of files to meet new needs, making the initial selection of the file size less critical! The merge capability includes small numeric data files too.

USERS' INSTRUCTIONS

Please note that for the following:

- (1) A "R/S*" means to omit the [R/S] command if a printer is used.
- (2) When the quotes are used we are implying keying in the ALPHA mode.
- (3) Do not hesitate to use the catalog command (CAT 1) to inspect the programs residing in memory.

To load and run this portion of DATA BASE 1 carry out:

Perform a MASTER CLEAR by holding down the back arrow key and the pressing the [ON] key. The message "MEMORY LOST" should appear. Load the small numeric data base set-up program, LGET:

INPUT : "LGET"
FUNCTION : [XEQ] "READA"

The loaded program will perform initialization steps that will end up with storing the option programs in the Extended Memory. The program will show the following menu;

```
1/NEW FILE
2/OPEN FILE
3/ADD REC
4/DELETE REC
5/SRCH REC
6/PRINT REC
7/EDIT REC
8/MERGE REC
9/QUIT
```

SELECT BY NUMBER

IT IS IMPORTANT TO POINT OUT THAT FOR EACH OPTION THE FOLLOWING HOLDS TRUE:

- TWO KEYS ARE OF SPECIAL USE, NAMELY THE [A] (SIGMA+) KEY THAT WILL ALLOW THE USER TO REDO OR RESTART THE CHOSEN OPTION, AND THE [E] (LN) KEY TO "EXIT" TO THE MENU.
- THE TWO KEYS ABOVE ARE ACTIVE WHETHER THE USER WISHES TO INTERRUPT THE PROGRAM BY PRESSING THE [R/S] KEY OR THE CHOSEN OPTION IS ALLOWED TO COMPLETE ITS TASK!

Here is what happens for every option in the menu:

(1) NEW FILE : This will enable the user to create a new file on the cassette with the ability to specify the maximum number of

records as well as the number of those to be entered soon after file creation.

(1.1) The first message that appears is,

DISPLAY : MAX DATA?

INPUT : enter maximum number of records to be entered in all and fits the available memory size, ND.

FUNCTION : [R/S]

(1.2) The program will ask for the number of variables (fields).

DISPLAY : NO. VARS?

INPUT : Enter the number of fields, NV.

FUNCTION : [R/S]

(1.3) The filename is required next.

DISPLAY : FILENAME?

INPUT : filename.

FUNCTION : [R/S]

(1.4) The program will ask for the accompanying status filename. A status file will enable the user to reset himself to the conditions allowing him proper data entry, especially after using other software that may alter the machine's status. Input here is optional and none is usually needed if the user does not understand clearly what status files do. If no input is made the program will resort to composing a status filename by chaining the data filename to an "S".

DISPLAY : STATUS FILE?

INPUT : (optional) status filename.

FUNCTION : [R/S]

DISPLAY : an audible beep.

As an example, we want to create a data file MDM that will hold fifty records at the most. Each record will have three fields. The memory partition SIZE will be set as one hundred (100). This tells the program how many records the available memory can hold. Let us start from ground zero.

- Perform a MASTER CLEAR

FUNCTION : [back arrow] and [ON] keys
DISPLAY : MEMORY LOST

- Load the set-up program, LGET:

INPUT : "LGET"
FUNCTION : [XEQ] "READA"

- When the menu appears select option number one. This is a good time to set the display format to five decimals.

DISPLAY : SELECT BY NUMBER
FUNCTION : [shift] "FIX" 5
INPUT : 1
FUNCTION : [R/S]
DISPLAY : MAX DATA?
INPUT : 50
FUNCTION : [R/S]
DISPLAY : NO. VAR?
INPUT : 3
FUNCTION : [R/S]
DISPLAY : FILENAME?
INPUT : MDM
FUNCTION : [R/S]
DISPLAY : STATUS FILE?
FUNCTION : [R/S]

Note that for the last prompt we entered nothing, telling the program to compose the status filename as SMDM. An audible beep will signal the creation of a new file.

- Exit the file creation option back to the menu.

FUNCTION : [E]

(2) OPEN FILE : This option allows us to "open" an existing file (new or old) for input/output operations. This step allows us to activate data retrieval from any file. It is sufficient to supply the filename in order to open it.

DISPLAY : FILENAME?

INPUT : filename.

FUNCTION : [R/S]

DISPLAY : NDATA= (# of file records)

FUNCTION : [R/S]

DISPLAY : NVAR= (# of variables or fields)

FUNCTION : [R/S]

DISPLAY : beep.

(3) ADD REC : This option allows the addition of records. Should it be selected while the data file is full a "NOP : FULL" message will indicate that it is full house! The user will also notice that during a repeated data entry the cassette drive will operate to transfer a data "batch" from memory to create space for more data. Make sure that you exit the option by pressing the [E] key to insure that any residual data is also sent to mass storage!

(3.1) A loop will begin to prompt the data entry.

DISPLAY : Xi,j?

INPUT : X(i,j)

FUNCTION : [R/S]

(3.2) The loop will end with a beep. Use the [A] key to add more data.

Continuing our example we want to open the file MDM and add one record of data.

- Select option number two to open the file.

DISPLAY : SELECT BY NUMBER

INPUT : 2

FUNCTION : [R/S]

DISPLAY : FILENAME?

INPUT : MDM

FUNCTION : [R/S]

- The file is now open. Let us return to the menu.

FUNCTION : [E]

- Select the file additon option (i.e. option number three). We want to enter the following data:

X(1,1) = 25.36

X(2,1) = 65.36

X(3,1) = 11.25

INPUT : 3

FUNCTION : [R/S]
DISPLAY : X1,1?
INPUT : 25.36
FUNCTION : [R/S]
DISPLAY : X2,1?
INPUT : 65.36
FUNCTION : [R/S]
DISPLAY : X3,1?
INPUT : 11.25
FUNCTION : [R/S]

- Exit the option to the menu.

FUNCTION : [E]

(4) DELETE REC : This allows the user to cancel one or more records in the opened file.

DISPLAY : DELETE FROM REC?

INPUT : The first record to be deleted.

FUNCTION : [R/S]

DISPLAY : TO REC?

INPUT : The last record to be deleted.

FUNCTION : [R/S]

The deletion will end by an audible beep.

(5) SRCH REC : To look for a record, this option allows the search for a record containing a field with a particular value.

DISPLAY : VAR NO. ?

INPUT : Enter the field (i.e. variable) number that will act as the key to the search.

FUNCTION : [R/S]

DISPLAY : VALUE?

INPUT : Enter the comparison value desired.

FUNCTION : [R/S]

If a record is found to meet the search criteria it is displayed. All records are searched. Use the [R/S] to stop the program and

the [E] key to exit back to the menu.

```
DISPLAY : Xn,1= (Xn,1)
          FUNCTION : [R/S]*
          Xn,2= (Xn,2)
          FUNCTION : [R/S]*
          .....
          .....
```

For additional search press the [A] key.

(6) PRINT REC : To view/print a range of records.

DISPLAY : PRINT FROM REC NO. ?

INPUT : Enter first record printed/viewed.

FUNCTION : [R/S]

DISPLAY : TO REC NO. ?

INPUT : Enter the last record printed/viewed.

FUNCTION : [R/S]

```
DISPLAY : FILE : (filename)
FUNCTION : [R/S]*
DISPLAY : X(1,first rec) =...
FUNCTION : [R/S]*
DISPLAY : X(2,first rec) =...
FUNCTION : [R/S]*
```

Use the [A] key to print another range of records.

(7) EDIT REC : This is the most powerful option, because it allows us to edit the stored data. Choosing this option will cause another menu (or a sub-menu if you like) to appear:

```
1/CHNG REC/FILE
2/INSPCT REC/FILE
3/INSP/CHNG 1 REC/MEM
4/CHNG REC/MEM
5/SAVE DATA
6/GOTO LMENU
```

SELECT BY NUMBER

Before we explain what each option does we would like to point out that, as the reader may notice, some of the options are similar and might seem as duplicates. The similar options

perform the same tasks, basically. The difference is the location of the data edited. Having loaded a record in memory we may want to examine the record over and over again, so no input/output operation is needed.

(7.1) CHNG REC/FILE : To change records present in mass storage.

DISPLAY : FIRST REC NO.?

INPUT : First record number.

FUNCTION : [R/S]

DISPLAY : LAST REC NO.?

INPUT : Last record number.

FUNCTION : [R/S]

Note that the record range specified should be small enough to fit the available memory, otherwise the program will load from the first record to that which memory allows.

DISPLAY : FILE : (filename)

FUNCTION : [R/S]*

A double nested loop will start to show all the fields for the record range specified.

DISPLAY : $X_{i,j} = (X_j, i)$

INPUT : (optional) New value for $X(i, j)$.

FUNCTION : [R/S]

A beep will signal the end of the loops.

(7.2) INSP REC/FILE : To inspect records in the file. This option is very similar to the above. The difference is that data is only viewed. If no printer is used the [R/S] key is to be pressed to continue viewing the data.

(7.3) INSP/CHNG 1 REC/MEM : To inspect/change one record that is residing in the memory of the HP41C.

DISPLAY : REC NO.?

INPUT : Enter record number.

FUNCTION : [R/S]

If the record sought is not in the memory a "NOT IN MEMORY" message will appear. The rest of the display is similar to option number one.

(7.4) CNHG REC/MEM : To change the records residing in memory. This is similar to option number one with the exception that no data is read from mass storage.

(7.5) INSPCT REC/MEM : To inspect records residing in memory. This is similar to option number two with the exception that no data is read from mass storage.

(7.6) SAVE DATA : This step is needed to save the records residing in memory after their successful editing. No messages will appear. An audible beep will signal the termination of the storage step.

(7.7) GOTO LMENU : This is to exit from the sub-menu to the main menu.

(8) MERGE REC : To merge the records of the current opened file with other records from another file. The program will check if the additional data can fit in the opened file. If not, a "TOO BIG" message will indicate that you cannot merge. The remedy is to create a new empty file and merge the records from the other files. The two file must contain the same number of fields!

DISPLAY : MERGE FROM REC?

INPUT : First record to be merged from the "other" file.

FUNCTION : [R/S]

DISPLAY : TO REC NO.?

INPUT : Last record in the merge range.

FUNCTION : [R/S]

DISPLAY : FILENAME?

INPUT : filename of the other file.

FUNCTION : [R/S]

A beep will signal the completion of the merge.

(9) Quit : To exit the LMENU.

DISPLAY : BYE

Having introduced all the options in LMENU, we now turn to our example.

- Let us merge the only record in file MDM with the two records from file SML. Select option number eight.

```
DISPLAY : SELECT BY NUMBER
INPUT : 8
FUNCTION : [R/S]
DISPLAY : MERGE FROM REC NO.?
INPUT : 1
FUNCTION : [R/S]
DISPLAY : TO REC NO.?
INPUT : 2
FUNCTION : [R/S]
DISPLAY : FILENAME?
INPUT : SML
FUNCTION : [R/S]
DISPLAY : an audible beep
```

- Exit the merge option.

```
FUNCTION : [E]
```

- Let us print the three records present in file MDM. Select option number six.

```
INPUT : 6
FUNCTION : [R/S]
DISPLAY : PRINT FROM REC NO.?
INPUT : 1
FUNCTION : [R/S]
DISPLAY : TO REC NO.?
INPUT : 3
FUNCTION : [R/S]
```

```
DISPLAY : FILE : MDM
```

```
FUNCTION : [R/S]*
DISPLAY : X1,1= 25.36000
FUNCTION : [R/S]*
DISPLAY : X2,1= 65.36000
FUNCTION : [R/S]*
DISPLAY : X3,1= 11.25000
FUNCTION : [R/S]*
DISPLAY : X1,2= 3.00000
FUNCTION : [R/S]*
DISPLAY : X2,2= 333.00000
FUNCTION : [R/S]*
DISPLAY : X3,2= 333.00000
```

FUNCTION : [R/S]*
DISPLAY : X1,3= 1.00000
FUNCTION : [R/S]*
DISPLAY : X2,3= 2.00000
FUNCTION : [R/S]*
DISPLAY : X3,3= 3.00000
FUNCTION : [R/S]*
DISPLAY : audible beep.

- Exit to the main menu.

FUNCTION : [E]

- Let us delete the third record. Select option number four.

INPUT : 4
FUNCTION : [R/S]
DISPLAY : DELETE FROM REC NO.?
INPUT : 3
FUNCTION : [R/S]
DISPLAY : TO REC NO.?
INPUT : 3
FUNCTION : [R/S]
DISPLAY : audible beep.

- Exit to the menu.

FUNCTION : [E]

- Let us add a record. Select option number three. Use the following record:

X(1,3)= 111 , X(2,3)=22.5 , X(3,3)=3.141

INPUT : 3
FUNCTION : [R/S]
DISPLAY : X1,3?
INPUT : 111
FUNCTION : [R/S]
DISPLAY : X2,3?
INPUT : 22.5
FUNCTION : [R/S]
DISPLAY : X3,3?
INPUT : 3.141
FUNCTION : [R/S]
DISPLAY : audible beep.

- Exit to the menu.

FUNCTION : [E]

- Select the edition option, number seven.

INPUT : 7
FUNCTION : [R/S]

The editing submenu appears (the following assumes a printer is used, otherwise press the [R/S] key).

DISPLAY : EDITING SYB-MENU
1/CHNG REC/FILE
2/INSPCT REC/FILE
3/INSP/CHNG 1 REC/MEM
4/CHNG REC/MEM
5/INSPCT REC/MEM
6/SAVE DATA
7/GOTO LMENU

SELECT BY NUMBER

- We will alter the second record. Select option number three and use the following data:

X(1,2)= 65 , X(2,2)= 25 , X(3,2)= 55

INPUT : 3
FUNCTION : [R/S]
DISPLAY : REC NO.?
INPUT : 2
FUNCTION : [R/S]
DISPLAY : X1= 3.00000
INPUT : 65
FUNCTION : [R/S]
DISPLAY : X2= 333.00000
INPUT : 25
FUNCTION : [R/S]
DISPLAY : X3= 333.00000
FUNCTION : [R/S]

- Let us go back to the sub-menu.

FUNCTION : [E] or [A]

- Now we want to inspect records one to three. We know that they reside in memory. Select option number five. The printout will simulate the use of a printer. Press the [R/S] key if none is used to view the display in the sequence shown.

INPUT : 5
FUNCTION : [R/S]
DISPLAY : FILE : MDM

DISPLAY : X1,1= 25.36000

```
X2,1= 65.36000
X3,1= 11.25000

X1,2= 65.00000
X2,2= 25.00000
X3,2= 55.00000

X1,3= 111.00000
X2,3= 22.50000
X3,3= 3.14100
```

- Exit to the sub-menu.

FUNCTION : [E]

- Now is the time to save the updated data in the memory.
Select option number six.

```
INPUT : 6
FUNCTION : [R/S]
DISPLAY : an audible beep.
FUNCTION : [E]
```

- Exit the editing submenu altogether. Select option number seven.

```
INPUT : 7
FUNCTION : [R/S]
```

- We want to search for the record that has an X2 of 25. Select option number eight in the main menu.

```
INPUT : 8
FUNCTION : [R/S]
DISPLAY : VAR NO. ?
INPUT : 2           for X2
FUNCTION : [R/S]
DISPLAY : VALUE?
INPUT : 25
FUNCTION : [R/S]
```

The program will search and print the following:

```
DISPLAY : REC 2
FUNCTION : [R/S]*
DISPLAY : X1,2= 65.00000
FUNCTION : [R/S]*
DISPLAY : X2,2= 25.00000
FUNCTION : [R/S]*
DISPLAY : X3,2= 55.00000
```

- Exit the search option and the menu itself. For the latter select option number nine.

FUNCTION : [E]

menu

DISPLAY : SELECT BY NUMBER

INPUT : 9

FUNCTION : [R/S]

DISPLAY : BYE

This ends the example.

MEMORY CONFIGURATION

REG #	CONTENT
00	Filename.
01	# of records, ND.
02	# of fields/record, NV.
03	Data location shift, 14.
04	Display format code.
05	RAM capacity for # of records.
06	Used
..
14	Used
15	X(1,1) <--- indices may change
16	X(2,1) according to loaded
17	Etc. portions.

MASS MEDIA FILE COFIGURATION

REC #	CONTENT
-----	-----
00	Status filename.
01	Max. # of records +(RAM capacity)/1000
02	# of records, ND.
03	# of variables (fields/record), NV.
04	X(1,1)
05	X(2,1)
06	Etc.

ALPHA/NUMERIC FILES

This portion of DATA BASE 1 is the most powerful. The user is able to store both alpha data as well as the numeric type. The involvement of alpha data types merits a special attention. While a user can save any real number in ONE memory register, this may not hold true for alpha data. More than one register may be involved in saving an alpha data type, resembling somewhat a train of wagons. Due to the latter fact we must now specify how many registers at most will be used to contain information for a data field, for each field. This is better known as "formatting". The formatting step is the first step involved to define the nature of the data we want to store. It is like creating a "mold" to produce mass production plastic items. Since the format information reside in a separate file we can refer to it and reuse it with many files that follow the same format, of course. To the user, formatting is the step where we teach the DATA BASE 1 program to adopt his terminology during prompting for data transactions.

USERS' INSTRUCTIONS

Please note that for the following:

- (1) A "R/S*" means to omit the [R/S] command if a printer is used.
- (2) When the quotes are used we are implying keying in the ALPHA mode.
- (3) Do not hesitate to use the catalog command (CAT 1) to inspect the programs residing in memory.

To load and run this portion of DATA BASE 1 carry out:

Perform a MASTER CLEAR by holding down the back arrow key and the pressing the [ON] key. The message "MEMORY LOST" should appear. Load the small numeric data base set-up program, AGET:

INPUT : "AGET"
FUNCTION : [XEQ] "READA"

The loaded program will perform initialization steps that will end up with storing the option programs in the Extended Memory. The program will show the following menu;

```
1/NEW FORMAT
2/PRINT FORMAT
3/NEW FILE
4/OPEN FILE
5/EDIT-ADD REC
6/DELETE REC
7/MERGE REC
8/SRCH REC
9/QUIT
```

SELECT BY NUMBER

IT IS IMPORTANT TO POINT OUT THAT FOR EACH OPTION THE FOLLOWING HOLDS TRUE:

- TWO KEYS ARE OF SPECIAL USE, NAMELY THE [A] (SIGMA+) KEY THAT WILL ALLOW THE USER TO REDO OR RESTART THE CHOSEN OPTION, AND THE [E] (LN) KEY TO "EXIT" TO THE MENU.
- THE TWO KEYS ABOVE ARE ACTIVE WHETHER THE USER WISHES TO INTERRUPT THE PROGRAM BY PRESSING THE [R/S] KEY OR THE CHOSEN OPTION IS ALLOWED TO COMPLETE ITS TASK!

Here is what happens for every option in the menu:

(1) NEW FORMAT : This option allows the user to define a new format file for a new application.

DISPLAY : FIELDS?

INPUT : Enter number of fields/record.

FUNCTION : [R/S]

A loop will start to accept input for fields information.

DISPLAY : FIELD i

INPUT : field name (up to 12 characters).

FUNCTION : [R/S]

DISPLAY : FLD i SIZE?

INPUT : Field size (in registers). If it is a numeric field enter as negative number (use the [CHS] Key).

FUNCTION : [R/S]

We would like to point out that to calculate the number of registers/field divide the maximum number of letters by six and add one if the division's remainder is NOT ZERO. The result is equal to the number of registers. It is also possible, to save format space, to have a numeric field with more than one number, such as having a measured item's maximum, minimum and average!

When the loop end the program will ask for the format filename.

DISPLAY : FILENAME?

INPUT : format filename.

FUNCTION : [R/S]

As an example we want to set up a format for the data concerning the famous PPC calculator/computer club chapters throughout the country. We want to have the data in three fields such that:

- 1) Field # 1 has the chapter's name of twelve letters maximum (i.e. two registers).
- 2) Field # 2 has two numeric "sub-fields". The first number is the membership size in that chapter, while the second represent the number of chapter members who own digital cassette drives!
- 3) Field # 3 has the chapter coordinator's name with twelve letters maximum (i.e. two registers).

Table 1 has the listing of the above data. The format filename is CHP.

Let us take matters from ground zero.

- Reform a MASTER CLEAR!

FUNCTION : [back arrow] and [ON] keys
DISPLAY : MEMORY LOST

- Load the set-up program, AGET:

INPUT : "AGET"
FUNCTION : [XEQ] "READA"

- When the menu appears select option number one. This is a good time to set the display format to five decimals.

DISPLAY : SELECT BY NUMBER
FUNCTION : [shift] "FIX" 5
INPUT : 1
FUNCTION : [R/S]
DISPLAY : FIELDS?
INPUT : 3
FUNCTION : [R/S]
DISPLAY : FIELD1
INPUT : CHAPTER NAME
FUNCTION : [R/S]
DISPLAY : FLD1 SZE?
INPUT : 2
FUNCTION : [R/S]
DISPLAY : FIELD2
INPUT : MEMBERS
FUNCTION : [R/S]
DISPLAY : FLD2 SZE?
INPUT : -2 (i.e. [2] [CHS])
FUNCTION : [R/S]
DISPLAY : FIELD3
INPUT : COORDINATOR
FUNCTION : [R/S]
DISPLAY : FLD3 SZE?
INPUT : 2
FUNCTION : [R/S]
DISPLAY : FILENAME?
INPUT : CHP
FUNCTION : [R/S]

- The format file will be created now. Exit this option back to the menu.

FUNCTION : [E]

(2) PRINT FORMAT : This option allows the user to review the format stored. All we need here is the format filename!

DISPLAY : FORM. NAME?

INPUT : Format filename.

FUNCTION : [R/S]

The program will list the format data.

DISPLAY : FORMAT FILE : (filename)

FUNCTION : [R/S]*

DISPLAY : F = (# of fields/record)

FUNCTION : [R/S]*

A loop will start to display the specifications of each field.

DISPLAY : FIELD i

FUNCTION : [R/S]*

DISPLAY : NAME : (field (i) name)

FUNCTION : [R/S]*

DISPLAY : SIZE : (field size in registers)

FUNCTION : [R/S]*

DISPLAY : TYPE : ALPHA or NUMERIC

FUNCTION : [R/S]*

At the end of the above loop the total field size is displayed.

DISPLAY : TOTAL SIZE : (Total size)

FUNCTION : [R/S]*

Going back to our example, we select option number two.

DISPLAY : SELECT BY NUMBER

INPUT : 2

FUNCTION : [R/S]

DISPLAY : FORM. NAME?

INPUT : CHP

FUNCTION : [R/S]

- The following is a simulated printout. Press the [R/S] key

after every display is no printer is used.

DISPLAY : FORMAT FILE : CHP

F = 3

-----FIELD 1
NAME : CHAPTER NAME
SIZE : 2
TYPE : ALPHA

-----FIELD 2
NAME : MEMBERS
SIZE : 2
TYPE : NUMERIC

-----FIELD 3
NAME : COORDINATOR
SIZE : 2
TYPE : ALPHA

TOTAL SIZE = 6

- Exit this option back to the menu.

FUNCTION : [E]

(3) NEW FILE : Having defined a format file we can now proceed via this option to create a new data file. We need to know the maximum number of records. As opposed to the numeric data files we need not supply (or re-supply, actually) the number of fields per record, because all this information comes from the format file that we will specify.

DISPLAY : FORMAT?

INPUT : The appropriate format filename.

FUNCTION : [R/S]

DISPLAY : MAX RECS?

INPUT : Maximum number of records.

FUNCTION : [R/S]

DISPLAY : FILENAME?

INPUT : Data filename.

FUNCTION : [R/S]

The program will ask for the accompanying status filename. A status file will enable the user to reset himself to the conditions allowing him proper data entry, especially after using other software that may alter the machine's status. Input here is optional and none is usually needed if the user does not understand clearly what status files do. If no input is made the program will resort to composing a status filename by chaining the data filename to an "S".

DISPLAY : STATUS?

INPUT : (optional) status filename.

FUNCTION : [R/S]

DISPLAY : an audible beep.

Back to our example. Using the format filename CHP we want to create a data file, call it CHAT, of ten records maximum and an accompanying status file SCHAT. Select option number three.

```
INPUT : 3
FUNCTION : [R/S]
DISPLAY : FORMAT?
INPUT : CHP
FUNCTION : [R/S]
DISPLAY : MAX. RECS?
INPUT : 10
FUNCTION : [R/S]
DISPLAY : FILENAME?
INPUT : CHAT
FUNCTION : [R/S]
DISPLAY : STATUS
INPUT : SCHAT
FUNCTION : [R/S]
```

- Exit back to the menu.

```
FUNCTION : [E]
```

(4) OPEN FILE : While formatting and creating new data files have been considered, none of these steps actually lets the user access the data file that is properly set up. We can only perform data transactions if we "open" a data file! This option allows us to do just that.

A user may notice that the menu does not include an option for

closing files. This is done implicitly when we establish a new format, view an existing format and create a new file. In addition, any alteration of the memory contents would damage the set-up. Opening a file at the beginning of a session is a good practice!

DISPLAY : FILENAME?

INPUT : Data filename.

FUNCTION : [R/S]

DISPLAY : Audible beep to confirm.

Let us now open the file we created earlier. Select option number four.

INPUT : 4
FUNCTION : [R/S]
DISPLAY : FILENAME?
INPUT : CHAT
FUNCTION : [R/S]

- Exit to the menu.

FUNCTION : [E]

(5) EDIT-ADD REC : This option will allow the user to add records and edit them for updates or error corrections. Included in the option is a number of routines put together to manipulate data. An editin sub-menu will appear (Use the [R/S] to display the menu option in sequence when a printer is not used):

EDIT SUB-MENU

1/ADD REC
2/CHNG REC/MEM
3/CHNG REC
4/INSPCT REC/MEM
5/INSPCT REC
6/INSPCT 1 REC
7/GOTO AMENU

SELECT BY NUMBER

Here are the sub-menu operations:

(5.1) ADD REC : This option adds a record to the data file. A "FILE FULL" message appears if the number of records in the file have reached the maximum value.

DISPLAY : REC i

FUNCTION : [R/S]*

A loop will start to show the field names and prompt for data entry. If the field is of type alpha a "???" message will be prompted. If it is of numeric type an "Xi,j,k" prompt will appear to inform the user the "sub-field", field and record numbers to be entered. For the HP41C/V, the alpha display is equivalent to four data registers (i.e. it takes 4 registers to save the contents of an entire alpha display). Should the field size be set as greater than four for an alpha data type (NOT NECESSARILY MULTIPLES OF FOUR!), the user should input a portion enough to fill the alpha register (i.e. 24 letters) and then press the [R/S] key. The program will prompt for the rest of the alpha input.

DISPLAY : *field name

FUNCTION : [R/S]*

DISPLAY : ??? (If alpha)

INPUT : Alpha data.

FUNCTION : [R/S]

or DISPLAY : Xi,j,k?

INPUT : Numeric data.

FUNCTION : [R/S]

DISPLAY : Audible beep.

Now let us focus our attention on the example in progress. Table 2 contains the data to be added. Select option number five from the main menu and then option number one from the edit sub-menu.

INPUT : 5

FUNCTION : [R/S]

menu display (may need [R/S] to view)

DISPLAY : SELECT BY NUMBER

INPUT : 1

FUNCTION : [R/S]

DISPLAY : REC 1

FUNCTION : [R/S]*

DISPLAY : *CHAPTER NAME
FUNCTION : [R/S]*
DISPLAY : ???
INPUT : WASH D.C.
FUNCTION : [R/S]
DISPLAY : *MEMBERS
FUNCTION : [R/S]*
DISPLAY : X1,2,1?
INPUT : 250
FUNCTION : [R/S]
DISPLAY : X2,2,1?
INPUT : 50
FUNCTION : [R/S]
DISPLAY : *COORDINATOR
FUNCTION : [R/S]*
DISPLAY : ???
INPUT : W. M. KOLB
FUNCTION : [R/S]
DISPLAY : Audible beep.

- Use the [A] (the [A]gain key) to add record two.

FUNCTION : [A]
DISPLAY : REC 2
FUNCTION : [R/S]*
DISPLAY : *CHAPTER NAME
FUNCTION : [R/S]*
DISPLAY : ???
INPUT : WASHINGTON
FUNCTION : [R/S]
DISPLAY : *MEMBERS
FUNCTION : [R/S]*
DISPLAY : X1,2,2?
INPUT : 300
FUNCTION : [R/S]
DISPLAY : X2,2,2?
INPUT : 40
FUNCTION : [R/S]
DISPLAY : *COORDINATOR
FUNCTION : [R/S]*
DISPLAY : ???
INPUT : B. MOORE
FUNCTION : [R/S]

- Use the [A] key to enter the third record.

FUNCTION : [A]
DISPLAY : REC 3
FUNCTION : [R/S]*
DISPLAY : *CHAPTER NAME
FUNCTION : [R/S]*
DISPLAY : ???

```

INPUT : UTAH
FUNCTION : [R/S]
DISPLAY : *MEMBERS
FUNCTION : [R/S]*
DISPLAY : X1,2,3?
INPUT : 185
FUNCTION : [R/S]
DISPLAY : X2,2,3?
INPUT : 30
FUNCTION : [R/S]
DISPLAY : *COORDINATOR
FUNCTION : [R/S]*
DISPLAY : ???
INPUT : C. L. DENNIS
FUNCTION : [R/S]

```

- Use the [A] key to enter the fourth record.

```

FUNCTION : [A]
DISPLAY : REC 4
FUNCTION : [R/S]*
DISPLAY : *CHAPTER NAME
FUNCTION : [R/S]*
DISPLAY : ???
INPUT : KANSAS
FUNCTION : [R/S]
DISPLAY : *MEMBERS
FUNCTION : [R/S]*
DISPLAY : X1,2,4?
INPUT : 400
FUNCTION : [R/S]
DISPLAY : X2,2,4?
INPUT : 36
FUNCTION : [R/S]
DISPLAY : *COORDINATOR
FUNCTION : [R/S]*
DISPLAY : ???
INPUT : C. BABBAGE
FUNCTION : [R/S]

```

- Exit this option to the edit sub-menu.

```

FUNCTION : [E]

```

(5.2) CHNG REC/MEM : To change the last record dealt with which is still residing in memory. DATA BASE 1 keeps only one alpha record in the memory due to space limitations. In this option the program displays the record's contents, field by field, with the ability to enter the updated data.

A loop will show the record in the following way:

DISPLAY : *field name

FUNCTION : [R/S]*

DISPLAY : field contents

INPUT : (optional) any updated data.

FUNCTION : [R/S]

The record is then updated in the mass storage.

Going back to our example, we want to change the coordinator's name from C. BABBAGE to C. HARRIS. Fortunately this happens to be in record four, the last one dealt with, residing in memory.

```
DISPLAY : SELECT BY NUMBER
INPUT : 2
FUNCTION : [R/S]
DISPLAY : REC 4
FUNCTION : [R/S]*
DISPLAY : *CHAPTER NAME
FUNCTION : [R/S]*
DISPLAY : KANSAS
FUNCTION : [R/S]
DISPLAY : *MEMBERS
FUNCTION : [R/S]*
DISPLAY : 400.00000
FUNCTION : [R/S]*
DISPLAY : 36.00000
FUNCTION : [R/S]
DISPLAY : X1,2,4?
FUNCTION : [R/S] (No input needed!)
DISPLAY : X2,2,4?
FUNCTION : [R/S] (No input needed!)
DISPLAY : *COORDINATOR
FUNCTION : [R/S]*
DISPLAY : C. BABBAGE
FUNCTION : [R/S]*
INPUT : C. HARRIS (The only input needed)
```

- Exit to the edit sub-menu.

FUNCTION : [E]

(5.3) CHNG REC : This is similar to the above option except that we will be dealing with a stored record.

(5.4) INSPCT REC/MEM : This option serves to view the record currently in memory. The display prompts are similar to that

above.

(5.5) INSPCT REC : This is similar to the above and is used to view one or more records from mass storage. Using the printer one can list his records.

(5.6) INSPCT 1 REC : Option used to inspect one record only.

(5.7) GOTO AMENU : This option takes the user back to the main menu.

Let us use option number (5.5) to list all four records in the opened file. We will simulate a printout! Use [R/S] if no printer is connected.

```
INPUT : 5
FUNCTION : [R/S]
DISPLAY : -----REC 1

      *CHAPTER NAME
      WASH. D.C.

      *MEMBERS
           250.00000 ***
           50.00000 ***

      *COORDINATOR
      W. M. KOLB

      -----REC 2

      *CHAPTER NAME
      WASHINGTON

      *MEMBERS
           300.00000 ***
           40.00000 ***

      *COORDINATOR
      B. MOORE

      -----REC 3

      *CHAPTER NAME
      UTAH

      *MEMBERS
           105.00000 ***
           30.00000 ***

      *COORDINATOR
```

C. L. DENNIS

-----REC 4

*CHAPTER NAME
KANSAS

*MEMBERS
400.00000 ***
36.00000 ***

- Exit to the edit sub-menu.

FUNCTION : [E]

(6) DELETE REC : This option allows to delete a record. One record is deleted at a time to avoid accidental loss of a large data amount if missused.

DISPLAY : DELETE REC NO.?

INPUT : Record number to be deleted.

FUNCTION : [R/S]

DISPLAY : Audible beep.

(7) MERGE REC : This option allows the user to merge data from another file if:

- 1) Both files are created under the same format.
- 2) There is enough space in the opened file for the merge.

DISPLAY : FILENAME?

INPUT : The name for the "other" (unopened) file.

FUNCTION : [R/S]

DISPLAY : FIRST REC NO.?

INPUT : First record number for the other file.

FUNCTION : [R/S]

DISPLAY : LAST REC NO.?

INPUT : Last record in the other file.

FUNCTION : [R/S]

A "FORMAT MISMATCH" message will appear if the format filenames in both files undergoing the merge are not identical in name. A "TOO BIG" message will signal space deficiency.

(8) SRCH REC : This option allows the user to search for any record in the opened file that matches a particular data. Any field or "sub-field" can be used. The program scans the entire file. It is worthwhile to point out that to reduce search time one can ask to match the leftmost part of the field (in case of alpha data).

DISPLAY : FIELD NO.?

INPUT : Enter the field number.

FUNCTION : [R/S]

DISPLAY : FIELD IS: field name

At this stage if the user finds out that he entered the wrong field number, simply press the [A] key and redo.

FUNCTION : [R/S] (To confirm the correctness of data)

For alpha type data.

DISPLAY : MIN. REGS NEEDED TO SRCH?

INPUT : Enter the minimum number of registers that will be used to match the leftmost portion of the field in question.

FUNCTION : [R/S]

DISPLAY : SEARCH FOR?

INPUT : search data.

FUNCTION : [R/S]

The search will begin and the records found will be indicated by number.

DISPLAY : REC i (record number)

FUNCTION : [R/S]

For numeric data.

DISPLAY : SUB-FIELD NO.?

INPUT : Enter subfield number.

FUNCTION : [R/S]

DISPLAY : VALUE?

INPUT : Numeric value to be matched.

FUNCTION : [R/S]

The search will produce the list of record numbers that fit the search criteria.

(9) Quit : To exit the menu.

We return one final time to our example.

- Search for the record that has C. HARRIS as a chapter coordinator. The latter is field number three. Select option number eight.

```
INPUT : 8
FUNCTION : [R/S]
DISPLAY : FIELD NO.?
INPUT : 3
FUNCTION : [R/S]
DISPLAY : FIELD IS: COORDINATOR
FUNCTION : [R/S] (As an O.K.)
DISPLAY : MIN. REGS NEEDED TO SRCH
INPUT : 2
FUNCTION : [R/S]
DISPLAY : SEARCH FOR?
INPUT : C. HARRIS
FUNCTION : [R/S]
DISPLAY : RECORD 4
FUNCTION : [R/S]
DISPLAY : Audible beep.
```

- Let us search again for the record with members in a chapter that own fifty cassette drives. In this case it is sub-field number two in field number two.

```
FUNCTION : [A]
DISPLAY : FIELD NO.?
INPUT : 2
FUNCTION : [R/S]
DISPLAY : FIELD IS: MEMBERS
FUNCTION : [R/S]
DISPLAY : SUB-FIELD NO.?
INPUT : 2
FUNCTION : [R/S]
```

DISPLAY : VALUE?
INPUT : 50
FUNCTION : [R/S]
DISPLAY : RECORD 1
FUNCTION : [R/S]
DISPLAY : Audible beep.

- Exit the search option to the menu.

FUNCTION : [E]

- Let us delete record number four. Select option number six.

INPUT : 6
DISPLAY : DELETE REC NO.?
INPUT : 4
FUNCTION : [R/S]
DISPLAY : Audible beep.
FUNCTION : [E]

This concludes our example.

MEMORY CONFIGURATION

REG #	CONTENT
00	Data filename.
01	# of fields/record, F.
02	Maximum # of records.
03	# of actual records.
04	Used.
05	Storage shift for format, A.
06	Storage shift for data location, D.
07	Sum of field sizes, S.
08	Used.
..
A+1	Field # 1 name, part 1.
A+2	Field # 1 name, part 2.
A+3	Field # 1 size.
...
A+3(F-1)	Field # F name, part 2.
A+3F	Field # F size.
....
D+1	R(1,1,k)
D+2	R(2,1,k)
....

Note: R(i,j,k) denotes sub-field (i) of field (j) in record (j).

FORMAT FILE CONFIGURATION

REC #	CONTENT
-----	-----
00	# of fields/record, F.
01	Field # 1 name, part 1.
02	Field # 1 name, part 2.
03	Field # 1 size
04	Field # 2 name, part 1.
..
3F	Field # F size.
3F+1	Sum of sizes, S.

DATA FILE CONFIGURATION

REC #	CONTENT
-----	-----
00	Format filename.
01	Status filename.
02	# of fields/record, F.
03	Maximum # of records.
04	Actula # of records.
05	R(1,1,1)
06	R(2,1,1)
..

Table 1

FORMAT FILE SPECIFICATIONS FOR EXAMPLE

FIELD # 1

NAME : CHAPTER NAME

SIZE : 2

TYPE : ALPHA

FIELD # 2

NAME : MEMBERS

SIZE : 2

TYPE : NUMERIC

FIELD # 3

NAME : COORDINATOR

SIZE : 2

TYPE : ALPHA

TOTAL SIZE : 6

FORMAT FILENAME : CHP

Table 2
SAMPLE DATA

Chapter name -----	Members -----		Coordinator -----
Wash D.C.	250	50	W. M. KOLB
Washington	300	40	B. MOORE
Utah	185	30	C. L. DENNIS
Kansas	400	36	C. BABBAGE

Filename : CHAT

Maximum records : 10

