NASA Technical Memorandum 86550

Search and Retrieval of Office Files Using dBASE III

William L. Breazeale and Charlotte R. Talley

AUGUST 1986

NASA

NASA Technical Memorandum 86550

Search and Retrieval of Office Files Using dBASE III

William L. Breazeale and Charlotte R. Talley

George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama

NASA
National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

1986

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

TECHNICAL MEMORANDUM

SEARCH AND RETRIEVAL OF OFFICE FILES USING dBASE III

PART I: THE USER'S GUIDE

1. INTRODUCTION

This document describes a way to automate the office file retrieval process using a commercially available software package. The document is divided into three parts — Part I, The User's Guide; Part II, The Technical Details; and Part III, Other Facts. Part I is written for the potential user and assumes that user is not conversant with computer terminology or software. This part describes the file retrieval problem, discusses the approach to solving the problem, illustrates the simplicity of using menus to lead the user, and lists some conclusions based on the authors' experience. Part II is directed toward the computer literate individual and assumes some familiarity with database management systems. It also includes technical details sufficient to serve as a guide to creating similar programs for use in other offices. The rationale for selecting dBASE III is described, the methodology is discussed, the types of search routines are explained, and some comments/suggestions are included. Part III consists of the Bibliography, the Glossary, and a printout of the complete system of programs and databases used in the authors' office.

2. THE PROBLEM: RETRIEVAL OF OFFICE FILES

All offices, regardless of their size or function, maintain some type of file system which can range from a quite rudimentary system with a small number of files to a sophisticated system containing a large number of files. The file system and the type and amount of information on file will vary with each office depending on the office's needs (as determined by the needs of the office manager) and the preference and ability of the person or persons charged with the responsibility of "maintaining the files."

A. The Traditional Way

In a typical office, filing is usually done in the following way. Someone (usually the secretary) is assigned the responsibility for "filing." Using a file guide such as NHB 1441.1 A "NASA Records Disposition Handbook" and their own preferences, this person establishes the categories or subjects under which office data will be filed. File folders are made; file cabinet drawers are set aside; and the various documents, correspondence, purchase orders, etc., are categorized and placed in their appropriate file folders and cabinet drawers. Assuming that a logical, systematic file system has been implemented, what could possibly be difficult about retrieving information from this system? A request for certain information is made, and the "file specialist" goes to the file cabinet, looks through a few folders (perhaps only one) that might contain the desired information. When the requested information is found, it is then given to the requester. As simple as this process sounds, there are at least three potential problem areas that hamper the retrieval process.

First, the person doing the filing has to decide what category to file an item under. Many times there is no clear cut category under which to file an item. This dilemma can be eased by "cross-filing," but over-use of this technique can increase file space significantly and could possibly make retrieval of an item even more difficult. Also over a period of time the person doing the filing may not be consistent in determining the category into which to file a given item, and this inconsistency could cause problems in retrieval.

The second area, which is even more of a problem, is who retrieves the data when the person that does the filing is not present or is unavailable? All the difficulties mentioned in the previous paragraph are now compounded by having someone unfamiliar with the "thought process" that determined under what category to file an item (and perhaps even unfamiliar with the filing system) searching for the needed data.

The third problem area involves the time interval between the filing of a specific item and the request for retrieval of this item. In general, the longer this interval gets, the less specific information the requester remembers about the item he desires, and therefore there is more difficulty in retrieving this item.

B. A New Way

Obviously the difficulties mentioned in the previous section could be eased considerably if there were available to all office staff a simple, easy-to-use technique (i.e., computer program) for determining the exact file folder that contained the desired item of interest. Because office files are simply a means of archiving or storing data in a systematic, logical way, they are in essence a database. With the advent of personal computers within most offices and the availability of excellent database management software, file search and data retrieval can be made quite easy. One approach is to use one of the available database management systems to establish an easy-to-use computer program that will rapidly search for and display the location of requested data.

3. APPROACH

Remember that the objective we are trying to achieve is to develop an easy-to-use computer program that will quickly and accurately retrieve the data that we are seeking. We are not looking for a better way to file, nor are we replacing the normal file system with a computer. What we are going to do is let the computer do the "looking" rather than having to stand at a file cabinet "thumbing through" a series of folders.

A. Scope The Problem

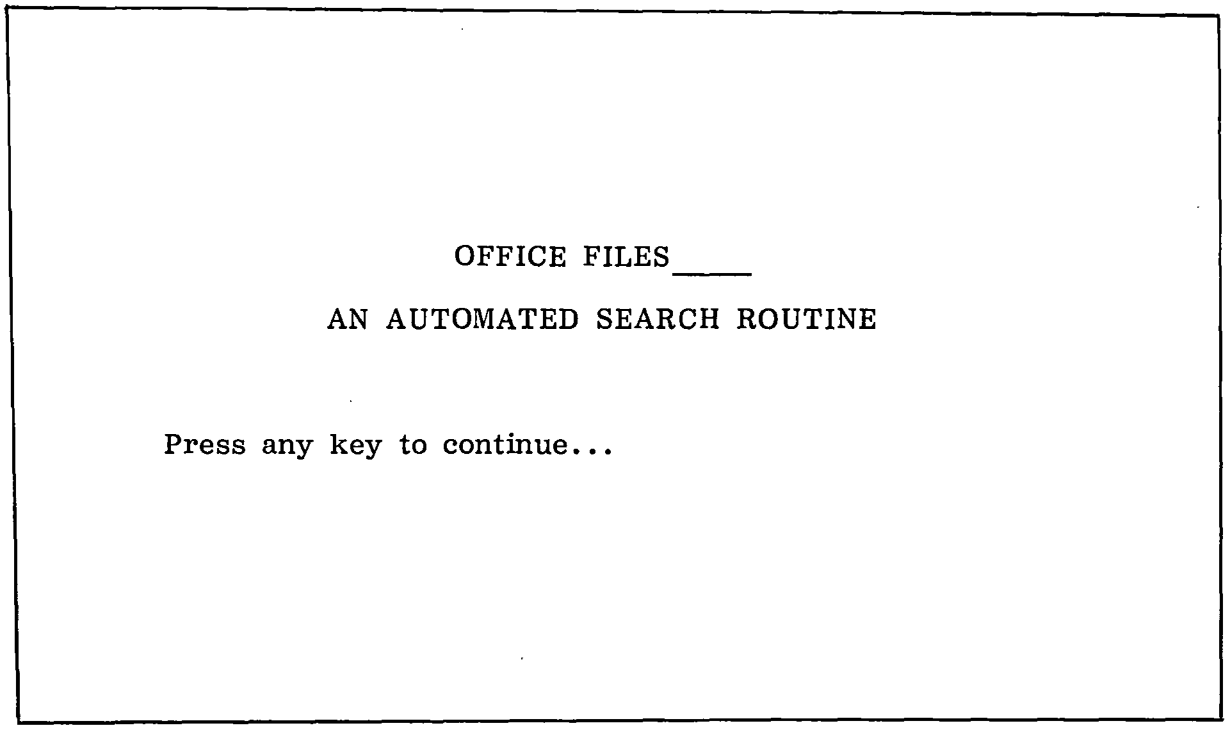
Earlier we stated that office files are simply a large database or a collection of smaller databases. Basically what we will do is synopsize certain facts from our office files and enter it into a computer data base. We will then develop an easy-to-use computer program that will search our computer database and tell us exactly where (i.e., what file folder or in some cases what floppy disk) the data we are seeking is located.

Before proceeding any further, let's define what is meant by "easy-to-use." Simply stated it means that the user will follow simple menus, cues, or prompts from the computer screen, and he/she will not need to know very much about computers and absolutely nothing about the application software used to create the program. Also the user will not need to know anything about the actual file system. However, the menus, cues, and prompts will be based on the file system, and the next section will illustrate this dependency.

B. Follow the Menus

This section contains a series of "screens" that the user will see when using our search program. Each screen cotnains instructions telling the user what to do next to continue the search or what to do to terminate the process. To initiate the search program (assuming that dBase III and the search program are "loaded" into the computer) the user simply types "DO OFMAIN" (in any combination of caps and/or lower case) and presses the "enter" key. This entry results in the display on the monitor of the "screen" shown in Figure 1 which is a sign-on screen.

This screen informs the user that the automated search routine is in use and prompts the user to "Press any key to continue...". When the user presses a key the next screen (Fig. 2) is displayed. The "Main Menu" is now on display, and the user is asked to enter the number corresponding to the category which is most likely to contain the information for which the user is looking. The categories shown in Figure 2 correspond to the file categories in the authors' office. This menu can be "tailor made" to fit the needs of any office, and the techniques for doing this task are discussed in part II of this report. The "box" at the end of the last line appears as reverse video on the computer monitor. In this example the user has selected the Travel file by entering "04". If any entry other than the numbers corresponding to the file categories is made, an error message appears saying "Invalid Entry - Press any key to continue...". When a key is pressed, the Main Menu is redisplayed so the user can try again.

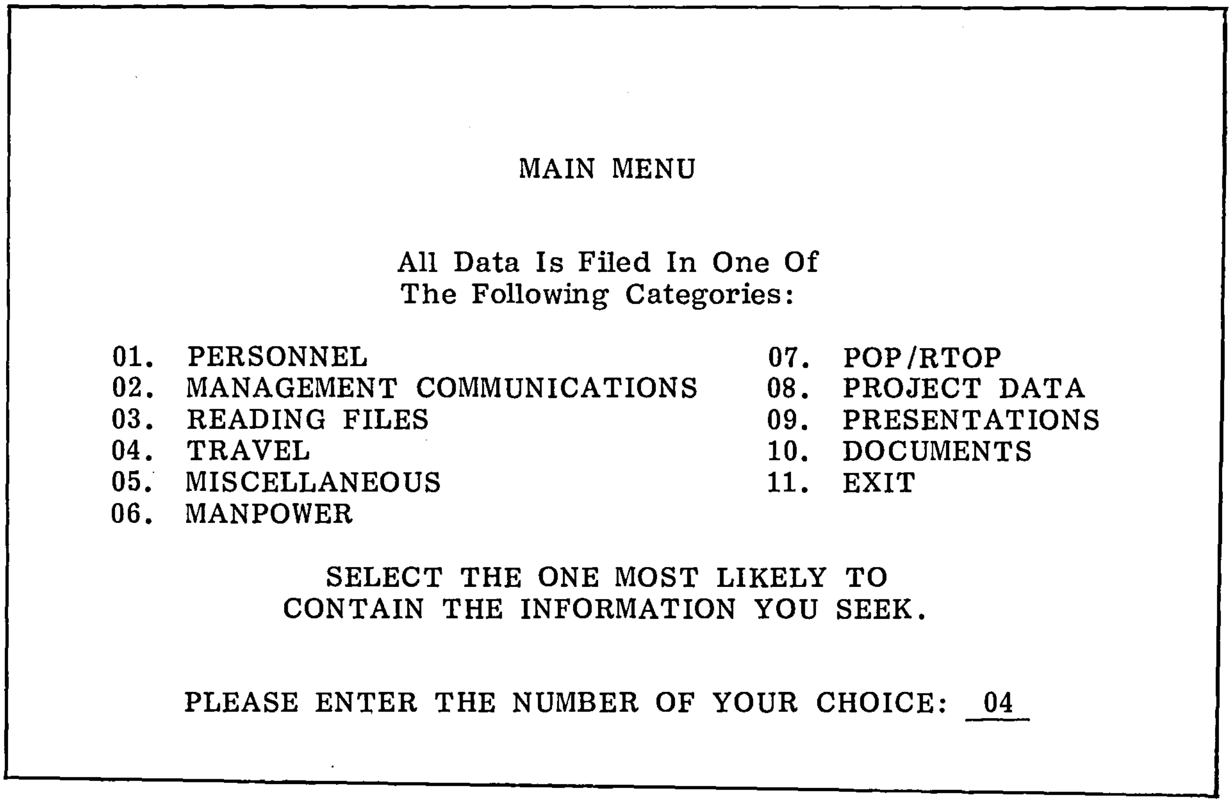


OFFICE FILES____

AN AUTOMATED SEARCH ROUTINE

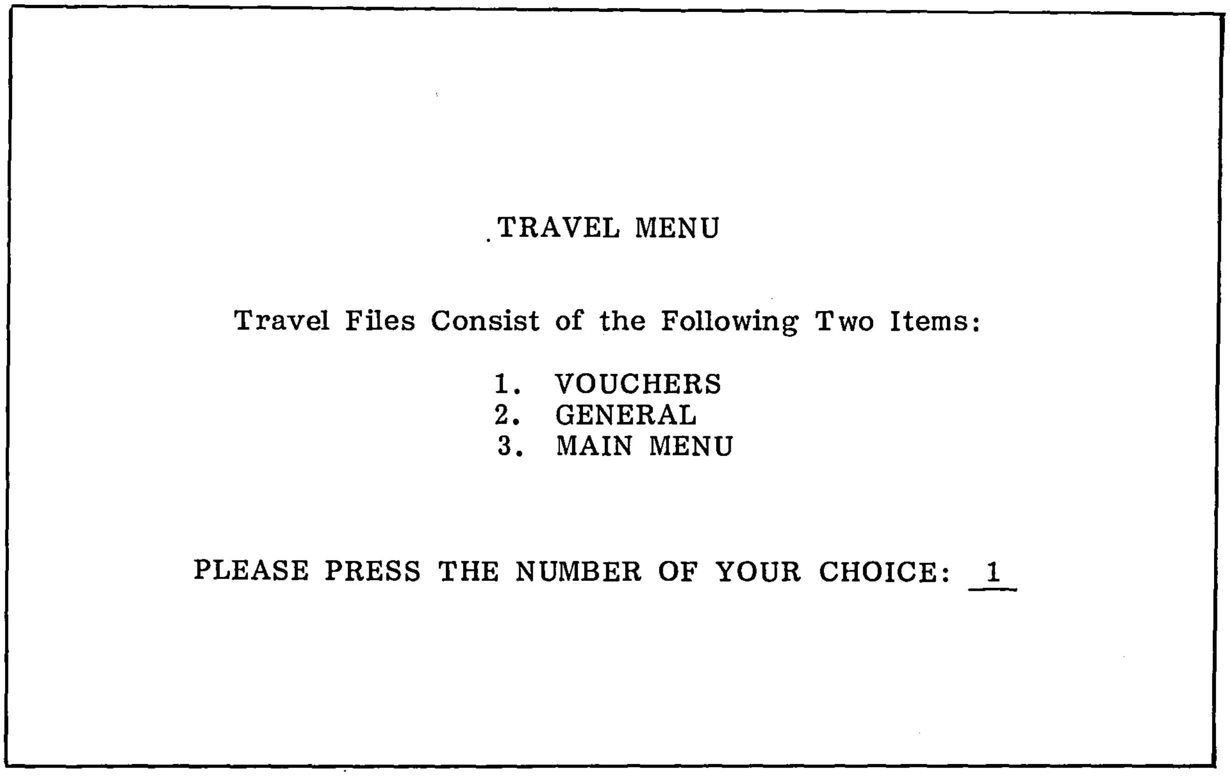
Press any key to continue...

Figure 1. Sign-on screen.



```
                            MAIN MENU

                    All Data Is Filed In One Of
                    The Following Categories:

   01.  PERSONNEL                               07.  POP/RTOP
   02.  MANAGEMENT COMMUNICATIONS               08.  PROJECT DATA
   03.  READING FILES                           09.  PRESENTATIONS
   04.  TRAVEL                                  10.  DOCUMENTS
   05.  MISCELLANEOUS                           11.  EXIT
   06.  MANPOWER

                  SELECT THE ONE MOST LIKELY TO
               CONTAIN THE INFORMATION YOU SEEK.

      PLEASE ENTER THE NUMBER OF YOUR CHOICE: _04_
```

Figure 2. Main Menu.

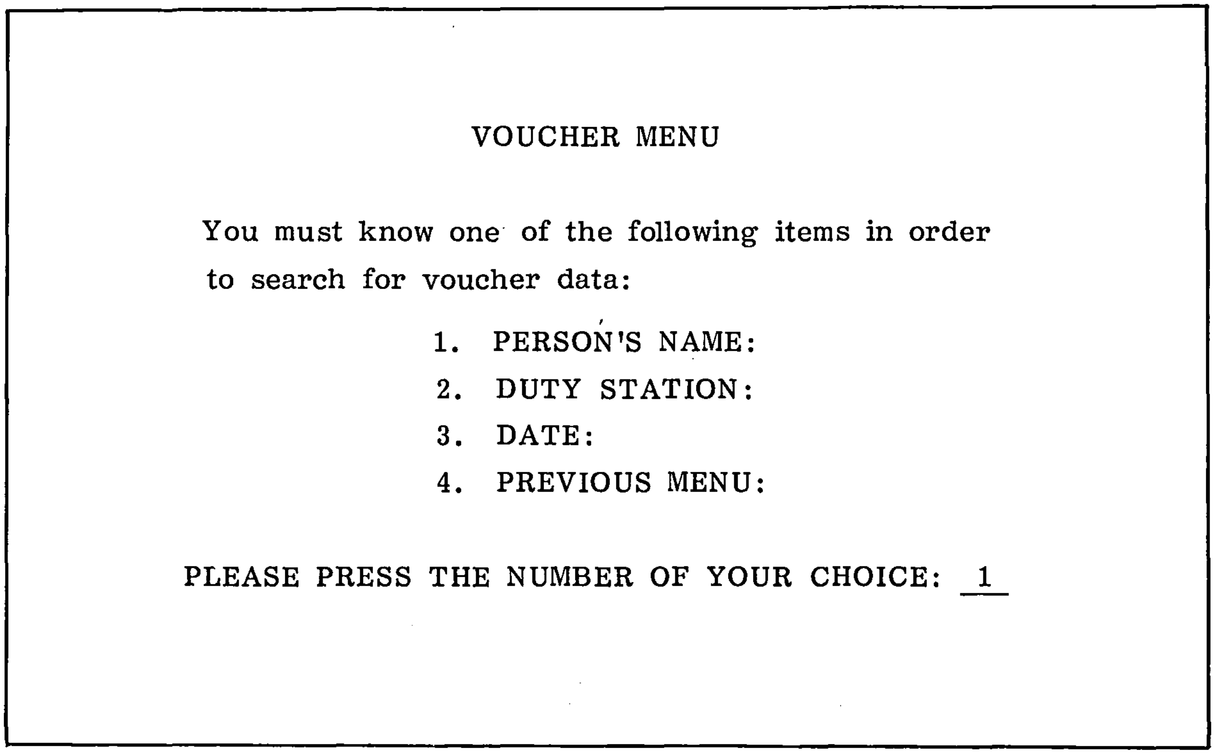


```
                          TRAVEL MENU

       Travel Files Consist of the Following Two Items:

                        1.  VOUCHERS
                        2.  GENERAL
                        3.  MAIN MENU

      PLEASE PRESS THE NUMBER OF YOUR CHOICE: _1_
```

Figure 3. Travel Menu.

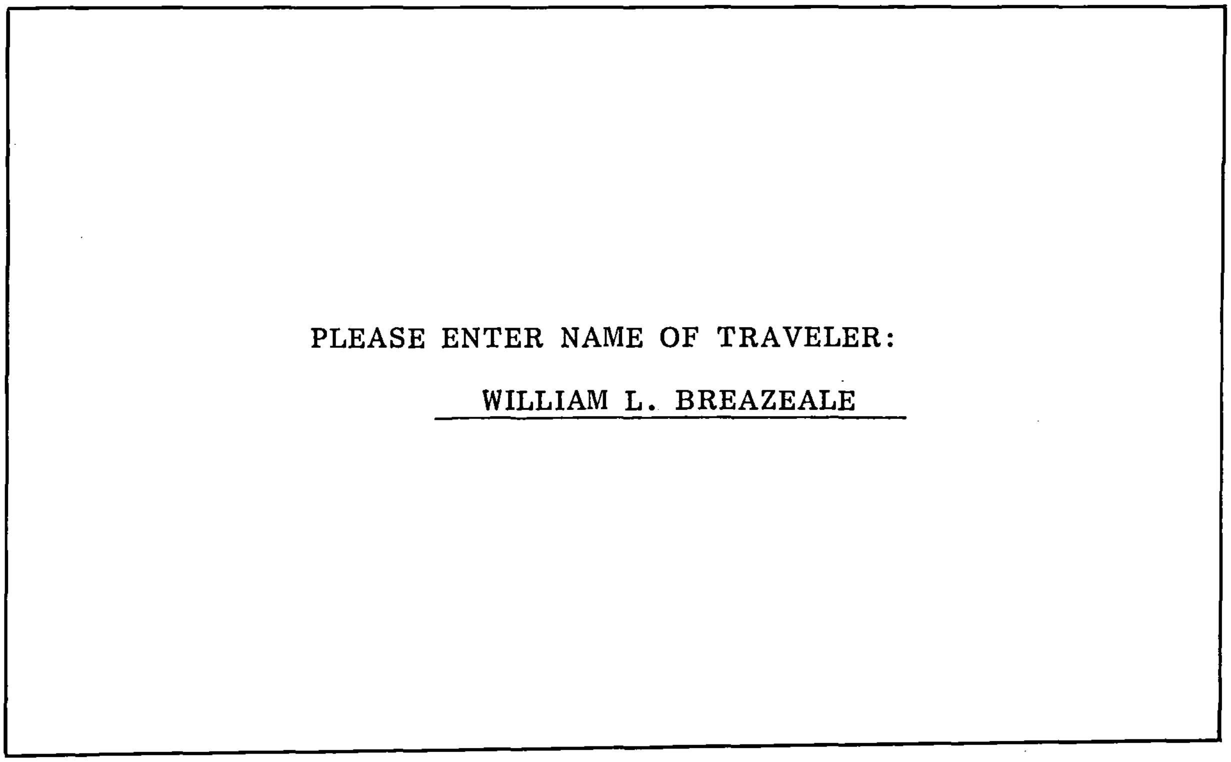
Once the Travel category has been selected, the screen in Figure 3 is displayed informing the user that this category is subdivided into "Vouchers" and "General." A third choice allows the user to return to the Main Menu. In this example, the user has selected "Vouchers" by entering "1." Once again any entry other than "1," "2," or "3" will result in the error message discussed in conjunction with the Main Menu. By entering "1," the screen in Figure 4 is displayed. The user is informed that to search for voucher data, the "Person's Name," "Duty Station," or "Date" must be known. A fourth choice is to return to the previous menu. Once again an improper entry results in an error message. By entering "1," the screen in Figure 5 is displayed, asking the user to enter the name of the traveler. With the entry of the traveller's name, the screen in Figure 6 is displayed telling the traveler's duty station, beginning date of travel, and location (file folder D-2) of that person's vouchers. When a key is pressed, the voucher menu is displayed again, and the user has the option of looking for more voucher data or returning via the Travel Menu to the Main Menu. Figure 7 shows the Voucher Menu, and the user has entered "2" indicating that "Duty Station" has been selected. Figure 8 shows that "Palo Alto, CA" has been entered as the "Duty Station," and Figure 9 shows a list of travelers to Palo Alto. If a duty station to which no one had traveled was entered, a message would appear on the screen stating that no travel to that duty station had occurred. Once again by pressing any key as instructed by the screen in Figure 9, the Voucher Menu (Fig. 4) is displayed again. By entering "4" the Travel Menu (Fig. 3) is displayed, and then by entering "3" the Main Menu (Fig. 2) is displayed. At this point the user can select other file categories to search or can exit from the program by entering "11."

Hopefully, the illustrated "search" of the Travel file has demonstrated the program's ease-of-use. The search of the other file categories is accomplished in a similar fashion.



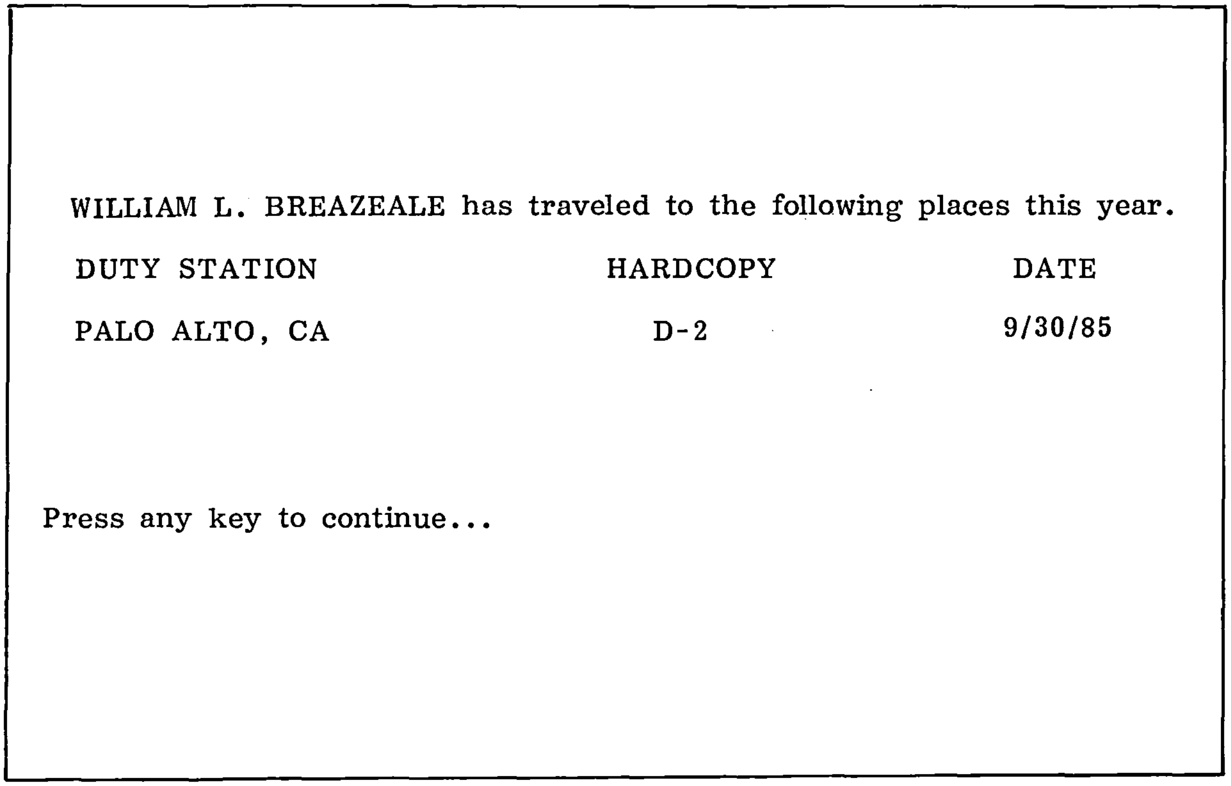
```
                         VOUCHER MENU

          You must know one of the following items in order
          to search for voucher data:

                      1.  PERSON'S NAME:
                      2.  DUTY STATION:
                      3.  DATE:
                      4.  PREVIOUS MENU:

        PLEASE PRESS THE NUMBER OF YOUR CHOICE:  1
```

Figure 4. Voucher Menu.



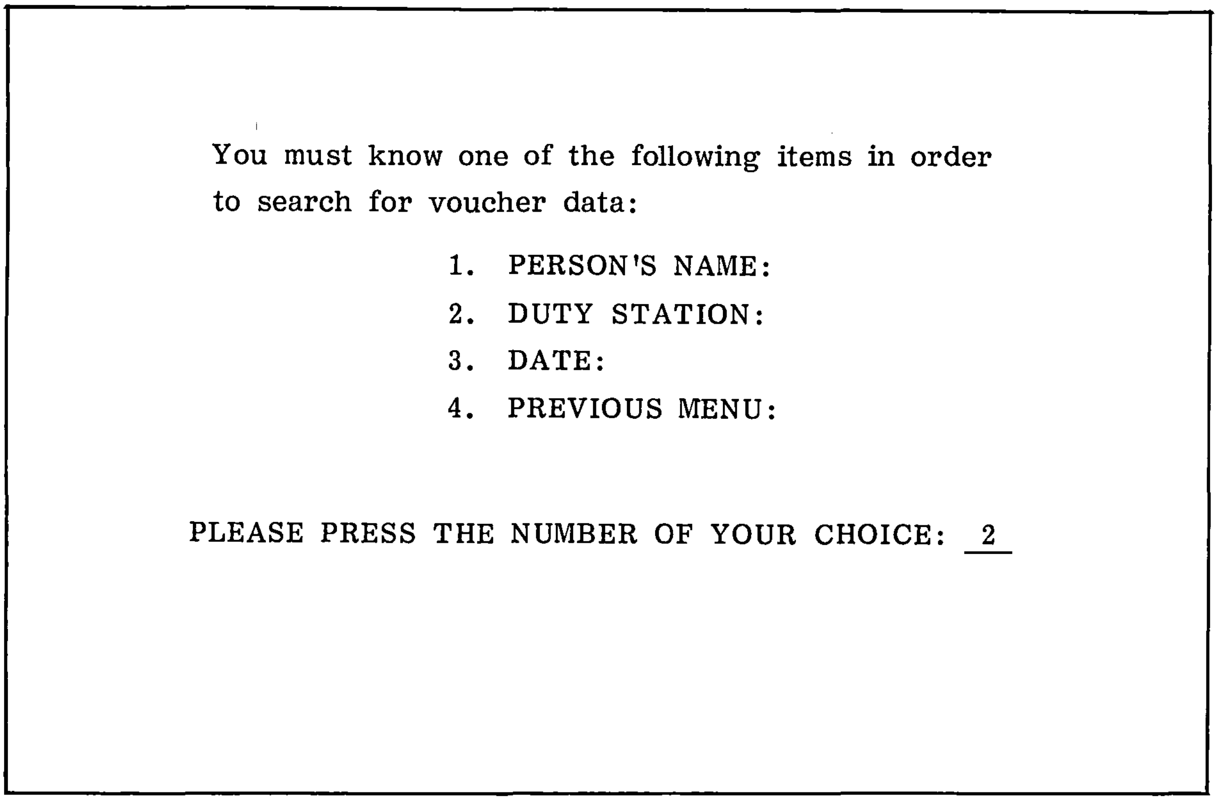
```
PLEASE ENTER NAME OF TRAVELER:
      WILLIAM L. BREAZEALE
```

Figure 5. Traveller's Name Entry.



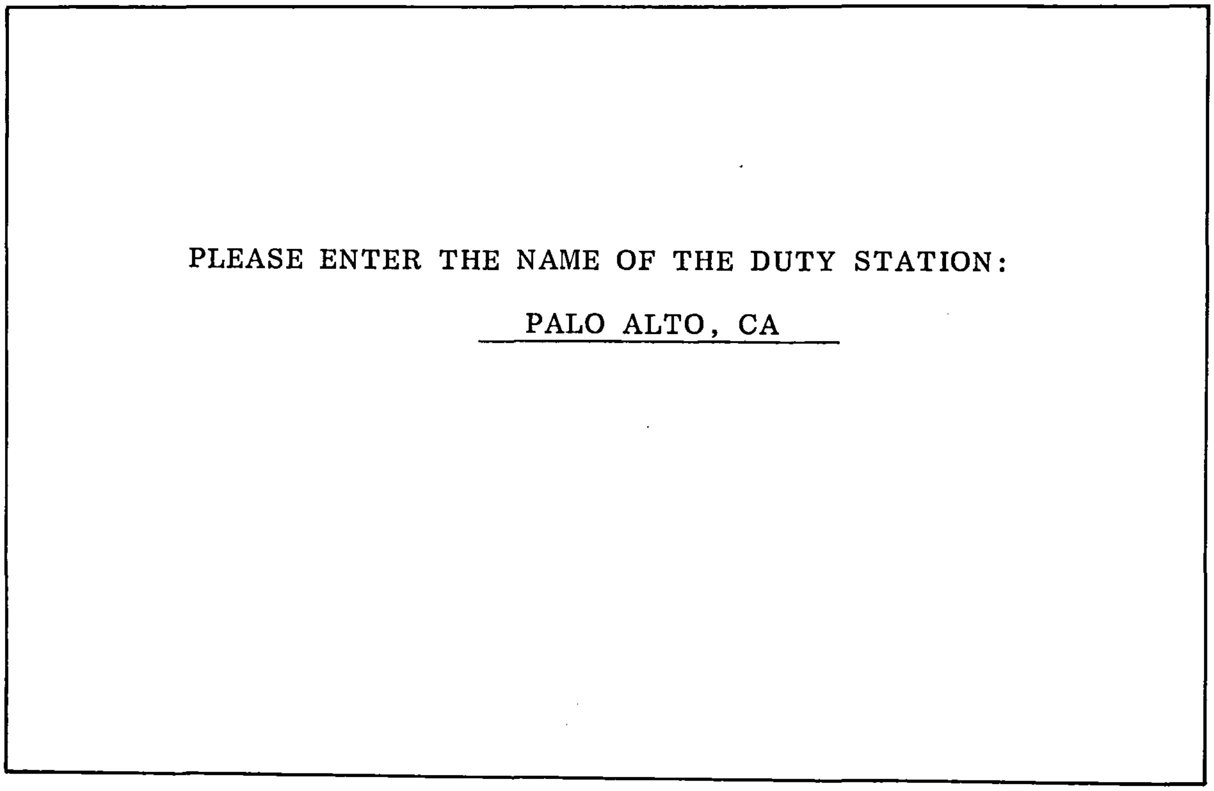
```
 WILLIAM L. BREAZEALE has traveled to the following places this year.

 DUTY STATION                    HARDCOPY               DATE

 PALO ALTO, CA                     D-2                 9/30/85



Press any key to continue...
```

Figure 6. Travel Destinations.



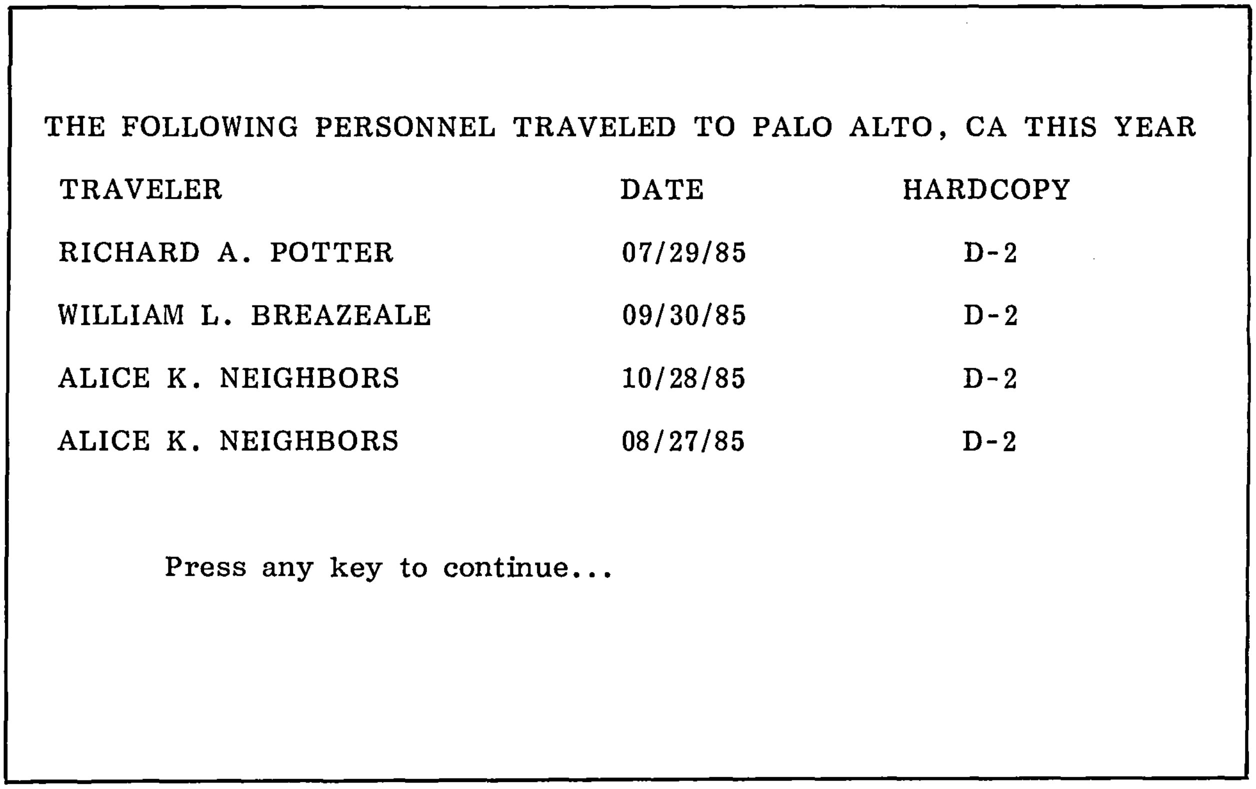
```
You must know one of the following items in order
to search for voucher data:

            1.  PERSON'S NAME:
            2.  DUTY STATION:
            3.  DATE:
            4.  PREVIOUS MENU:

PLEASE PRESS THE NUMBER OF YOUR CHOICE:  2
```

Figure 7. Voucher Menu.



```
PLEASE ENTER THE NAME OF THE DUTY STATION:
               PALO ALTO, CA
```

Figure 8. Duty Station Entry.



```
THE FOLLOWING PERSONNEL TRAVELED TO PALO ALTO, CA THIS YEAR

 TRAVELER                          DATE          HARDCOPY

 RICHARD A. POTTER                 07/29/85          D-2

 WILLIAM L. BREAZEALE              09/30/85          D-2

 ALICE K. NEIGHBORS                10/28/85          D-2

 ALICE K. NEIGHBORS                08/27/85          D-2


       Press any key to continue...
```

Figure 9. Duty Station List.

4. CONCLUSIONS

The search routine (OFMAIN.PRG) described in this report was completed in November 1985, and at the time this report was written it had been in use for about five months. The OFFMAIN.PRG has proven to be very useful and also quite easy to use. The authors are in a small project office where each member of the office has a personal computer which makes the files readily accessible to everyone. Therefore, when the secretary is out, files are easily retrievable by anyone in the office. The program should be very useful in a larger office where the file system would be much larger.

PART II: THE TECHNICAL DETAILS

1. WHY dBASE III

The selection of dBASE III was based on some very simple reasons — it was already available within the office; it has an easy to use programming language that is very powerful; it has rather modest hardware requirements; and it is capable of handling a wide range of databases. There was no attempt made to evaluate other database management systems which also could have accomplished our desired objectives, and consequently this document should not be construed as advocating the use of dBASE III over other systems. Also this document is not intended to be a tutorial on the use of dBASE III. For specific information on dBASE III capabilities, hardware requirements, and instructions, the reader should consule the dBASE III manual and/or various articles in the open literature.

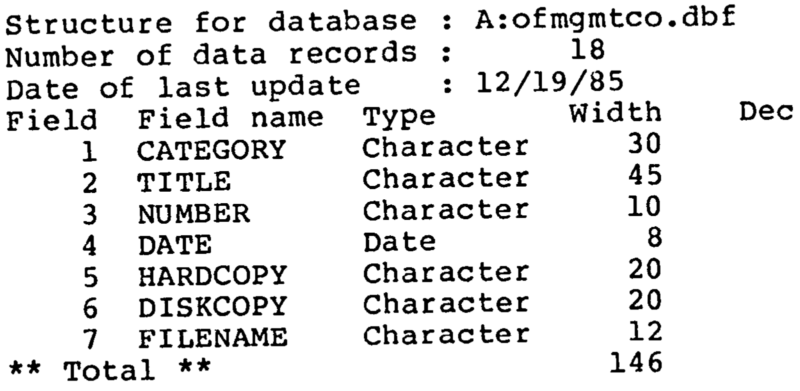
2. METHODOLOGY

The first step in "automating" the file search is to review the categories or subjects currently used for filing office data/material to be sure they are suitable and satisfy the office's filing needs. If changes are needed or appear to be needed, make them. There is no "ideal" number of file categories, but somewhere around 10 to 12 should handle most offices' needs.

A. What Size Databases

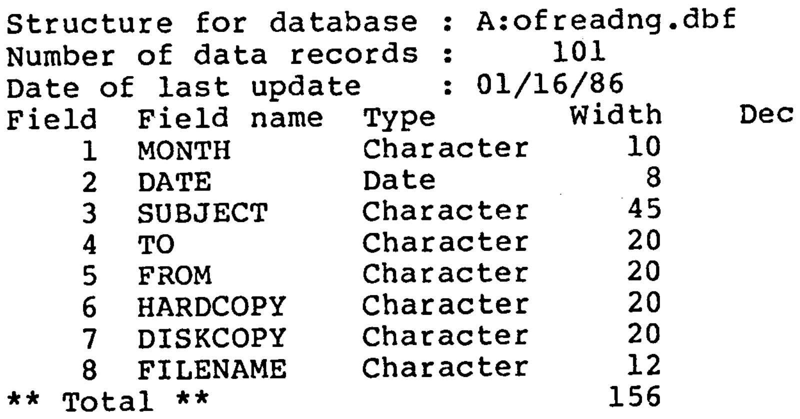
When using a database management system, there is an initial tendency to make a few database files with lots of fields in them. To characterize the wide range of information you will want to put in the database files will require lots of fields if you limit yourself to a few database files. A much better, more efficient approach is to have more files with fewer fields in each file. This approach will increase operating speed, reduce the chances of errors during data input, and allow the creation of tailor-made databases. Our recommendation is to create a database for each category or subject in the office file system and give it a name that is indicative of the subject it covers, e.g., a database for the office travel files might be given the name OFTRAVEL.DBF.

The next step is to select for each file category the information that will go into the database. Remember we are not trying to create a database that duplicates all the information in the file itself, but rather we are creating a database with sufficient information about an item to allow us to find the exact location (in the files) of that item. For example, Figure 10 shows the structure of the database (ofmgtco.dbf) for the Management Communications files. Each record (there are 18 records) has 7 fields containing the type of data indicated in the Figure. Thus, there are 126 fields of information in this database. Now look at Figure 11 which is the structure for the reading files data base (ofreadng.dbf). It has 8 fields, with four of these fields identical to those in the management communications database. Since there are 101 records, this database contains 808 fields of information. If these two databases were combined, 11 fields would now be required on each record to contain the desired information and the same total number of records (119) would still be required after combining. This combined database would now contain 1309 fields of information as opposed to a total of 934 in the two individual databases. Of course not all fields would have data entered into them. With this larger number of fields on each record, there is a greater chance of making an entry error by



```
Structure for database : A:ofmgmtco.dbf
Number of data records :       18
Date of last update    : 12/19/85
Field  Field name  Type        Width    Dec
    1  CATEGORY    Character      30
    2  TITLE       Character      45
    3  NUMBER      Character      10
    4  DATE        Date            8
    5  HARDCOPY    Character      20
    6  DISKCOPY    Character      20
    7  FILENAME    Character      12
** Total **                      146
```

Figure 10. Structure for Database A:OFMGMTCO.DBF.



```
Structure for database : A:ofreadng.dbf
Number of data records :      101
Date of last update    : 01/16/86
Field  Field name  Type        Width    Dec
    1  MONTH       Character      10
    2  DATE        Date            8
    3  SUBJECT     Character      45
    4  TO          Character      20
    5  FROM        Character      20
    6  HARDCOPY    Character      20
    7  DISKCOPY    Character      20
    8  FILENAME    Character      12
** Total **                      156
```

Figure 11. Structure for Database A:OFREADNG.DBF.

getting information in the wrong field. There is also a tendency to put data in every field which would cause retrieval problems. Suffice it to say that the number of fields in a database should be kept to a minimum.

B. Structured Programming

We have used a top down, structured approach in developing the program to search the office files. This approach can best be described by referring to Figure 12. The program which controls all the others is named OFMAIN.PRG, and it is shown at the top of the page. When this program is "run," the "sign on" program (SIGNON.PRG) is automatically executed (displays an introductory message) and waits on the user to "tell" it to proceed. On proceeding, the environmental parameters (e.g., SET BELL OFF, etc.) are automatically implemented, and a menu (generated by the format file named OFMAINMU.FMT) is displayed. At this point, the user sees the screen shown in Figure 13. The user now has the option of selecting one of ten file categories or exiting from the search program. Notice on Figure 12 that there is a first level program corresponding to each of these ten file categories. Also notice that there is a database file associated with each of these first level programs. Thus, for each file category, there is a "tailor-made" database and a set of "tailor-made" search programs to find the data for which the user is looking. By structuring the overall search routine in this manner, each program can be tested separately which makes trouble shooting much easier.

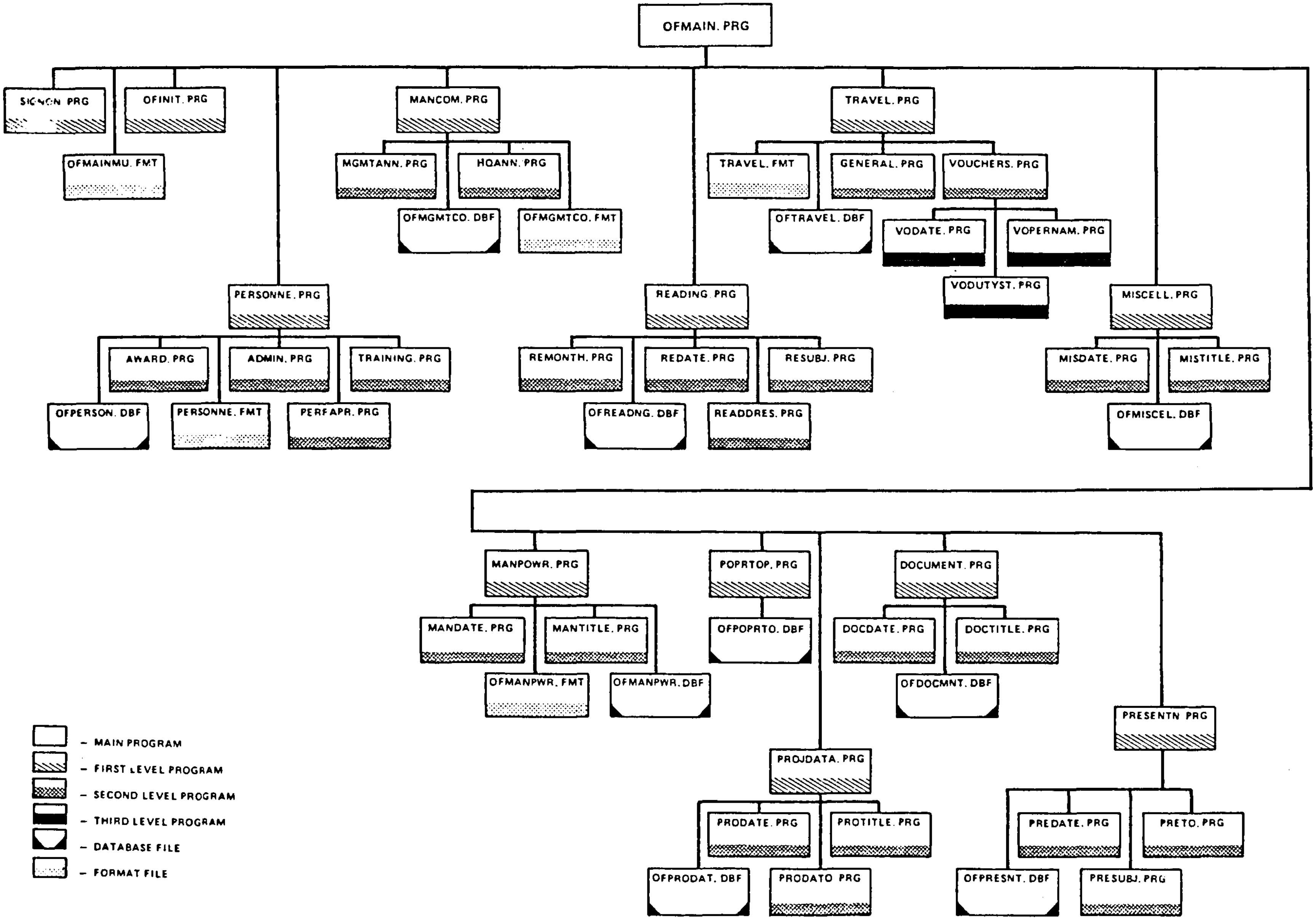
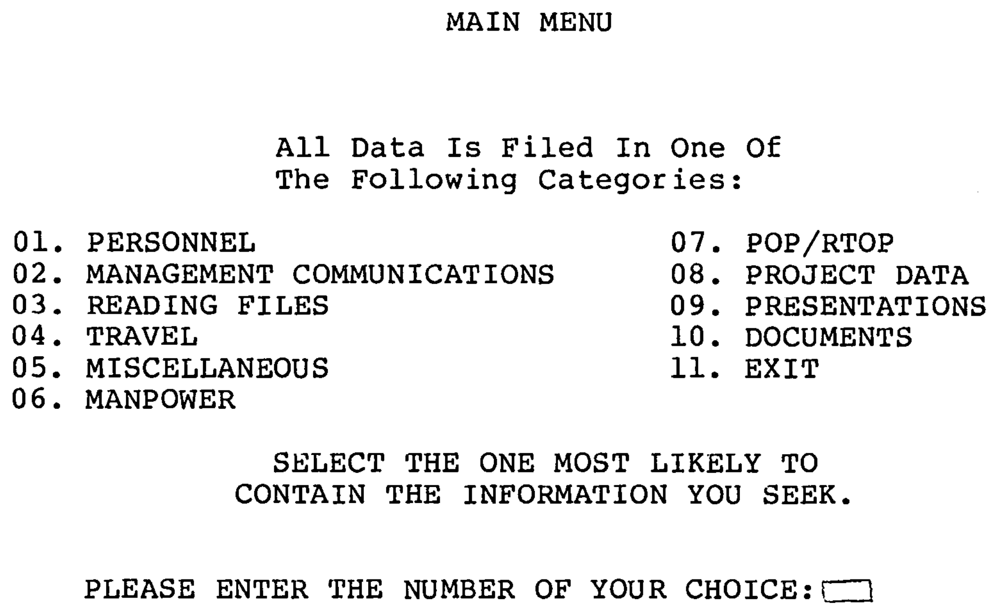


Figure 12. Program structure.



```
                    MAIN MENU


               All Data Is Filed In One Of
               The Following Categories:

01. PERSONNEL                             07. POP/RTOP
02. MANAGEMENT COMMUNICATIONS             08. PROJECT DATA
03. READING FILES                         09. PRESENTATIONS
04. TRAVEL                                10. DOCUMENTS
05. MISCELLANEOUS                         11. EXIT
06. MANPOWER

               SELECT THE ONE MOST LIKELY TO
             CONTAIN THE INFORMATION YOU SEEK.


      PLEASE ENTER THE NUMBER OF YOUR CHOICE:[  ]
```

Figure 13. Main Menu.

C. Logic/Hierarchy

The logic flow of the search routine is depicted in Figure 14. The travel file category is used as an example. Assuming that dBASE III is "loaded" into the computer, the user enters the command "DO OFMAIN" which initiates the search routine. The signon program is automatically executed and results in the display of screen 1 shown on Figure 14. When the user "presses any key to continue," the program executes the initialization program and sets the format according to the instructions in OFMAINMU.FMT which results in the display of the Main Menu (screen 2). The user selects the travel files to search and enters "04". By entering "04" the main program "calls" the travel program which displays screen 3. This screen contains another menu and asks the user to select either "VOUCHERS" or "GENERAL". A third choice is to return to the main menu. By entering "1", the vouchers program is activated, and screen 4 is displayed asking the user to pick "TRAVELER", "DUTY STATION", "DATE", or "PREVIOUS MENU". By entering "1', the user activates the program (VOPERNAM.PRG) that will search based on the traveler's name which is what screen 5 asks the user to enter. Once the name is entered, screen 6 shows all the places (duty station) the traveler has gone, when the traveler was there (date), and the location (D-2) of the hardcopy of his voucher(s). If the traveler had not traveled any, a message stating that fact would appear on the screen. By "pressing any key to continue," the program would go back to screen 4, where the user could then return to the previous menu or look for additional items in the voucher area.

3. SEARCH ROUTINES

Because of the power and versatility of dBASE III and its programming language, there are many ways to search, manipulate, and display the data in a database. This ability coupled with the specially designed database for each file category allows a "tailor-made" search and display routine to be created for each file category. These search routines tended to fall into one of two broad groups with slight variations in

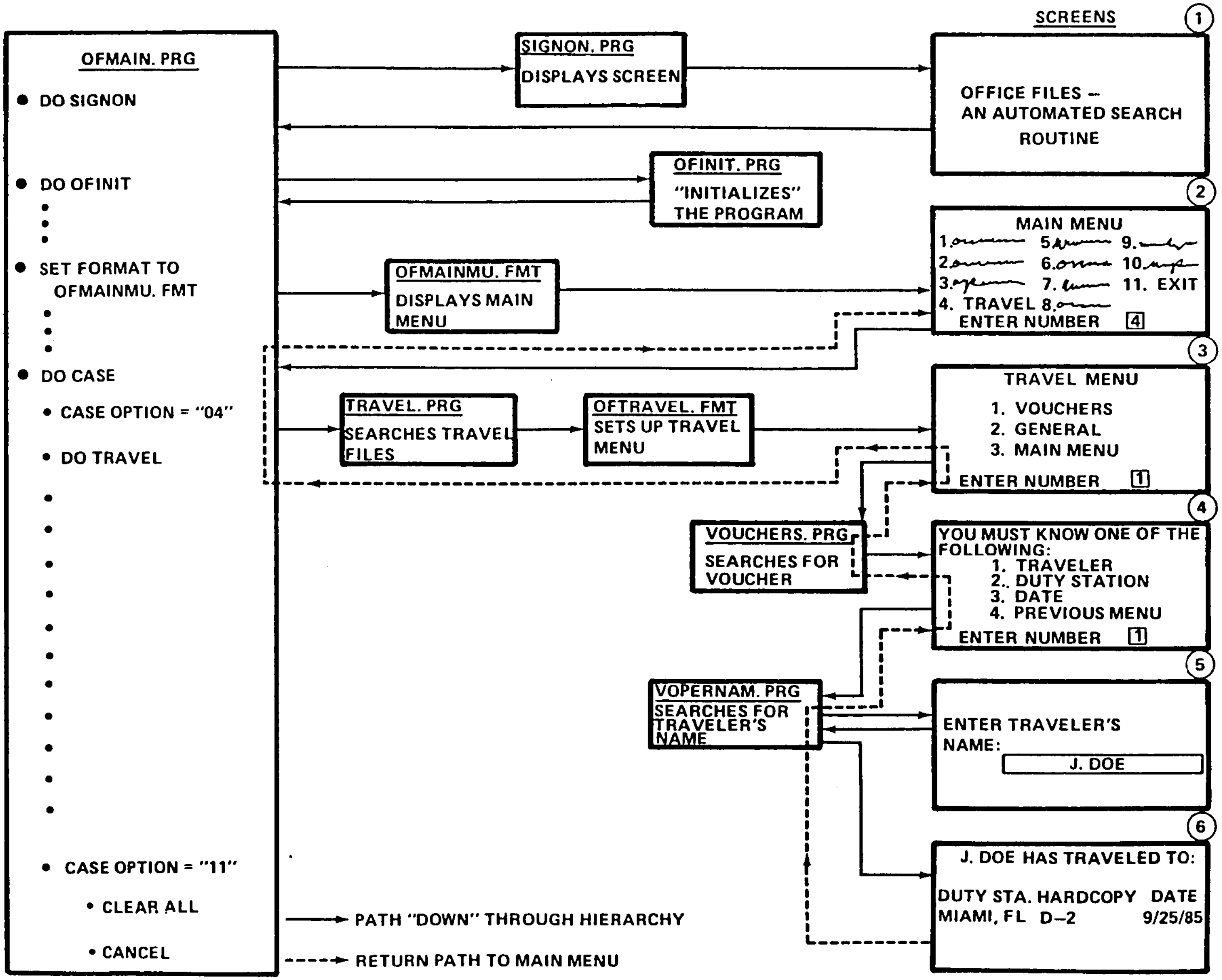
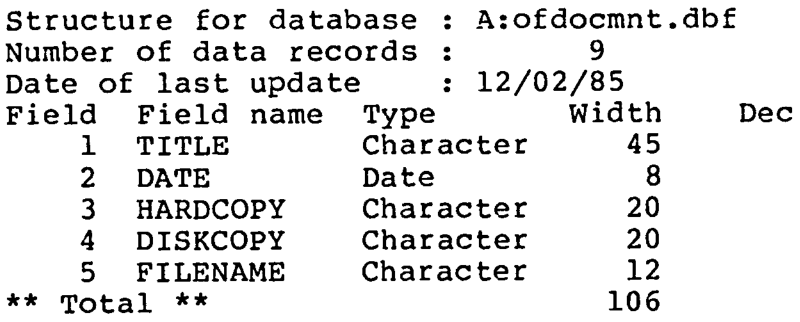


Figure 14. Program Logic.

the routines within a given group. The following paragraphs discuss these two groups of search routines.

A. Single-Entry Files

Before proceeding with a description of the search routine, an explanation of "single-entry files" seems appropriate. Consider the document database file (OFDOCMNT.DBF) whose structure is shown in Figure 15. It has two fields ("TITLE" and "DATE") on which to search. Of these two fields, one would expect the "TITLE" field to be unique, i.e., no two documents would have the same title. Therefore, a given title (if it is recorded in the documents database) would exist on one and only one record. Being recorded on a single record is what is meant by "single-entry file." Thus a search on the "TITLE" field would yield only one record meeting the title criteria. The next step is to explain how this fact is used in the search routine.



```
Structure for database : A:ofdocmnt.dbf
Number of data records :       9
Date of last update    : 12/02/85
Field  Field name  Type       Width    Dec
    1  TITLE       Character     45
    2  DATE        Date           8
    3  HARDCOPY    Character     20
    4  DISKCOPY    Character     20
    5  FILENAME    Character     12
** Total **                     106
```

Figure 15. Structure for Database A:OFDOCMNT.DBF.

In dBASE III, there are essentially four commands that can be used to search for specific data. These commands are "SEEK", "FIND", "DISPLAY", and "LOCATE". The first two commands will only work on a database indexed on the field which is being searched. For a large database the search will be accomplished quicker if the database is indexed. None of the databases in our office files are very large, so we have not indexed them. Therefore the "DISPLAY" and "LOCATE" commands are the only candidates for our use. To see how these commands are used, refer to Figure 16, which is a printout of the program that searches the "TITLE" field in the documents database (OFDOCUMNT.DBF). The numbers on the left were added to the printout to assist in describing the program and are not part of the program. The asterisk (*) in lines 1, 2, and 31 causes dBASE III to ignore the contents of that line. The asterisk allows comments about the program to be inserted in the program as an aid to understanding or identifying the program. Line 3 clears the screen. Line 4 establishes a memory variable that can contain as many characters as will fit between the quotation marks. Lines 5 and 6 instruct the user to enter the title. Lines 7 and 8 automatically capitalize the title entered from the keyboard. Line 9 instructs dBASE III to "read" the title that was entered and store it in the memory variable named "mtitle". Line 10 trims any blanks at the end of the memory variable. Blanks will occur if the title is entered starting at the left most position in the memory variable and is not long enough to take up all the blank spaces in the memory variable. Line 11 keeps unwanted information from appearing on the screen during the execution of the subsequent commands. Line 12 tells dBASE III to "locate" a record which meets the following condition: the record which has entered in its "TITLE" field the same title that was typed in from the keyboard. At this point suppose there is no record which contains the title for which we are searching. In that case dBASE III will have looked through the entire database and will encounter

```
1   *                             DOCTITLE.PRG          11/18/85

2   *Called from DOCUMENT.PRG

3   CLEAR
4   mtitle = "                                             "
5   @6,10 SAY "ENTER THE TITLE OF THE DOCUMENT FOR WHICH YOU ARE
6   SEARCHING:"
7   @8,17 GET mtitle PICTURE
8   "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
9   READ
10  mtitle = TRIM(mtitle)
11  SET TALK OFF
12  LOCATE FOR TITLE = "&mtitle"
13  IF EOF()
14     CLEAR
15     @6,10 SAY "NO DOCUMENT FOUND TITLED &mtitle"
16     @22,0
17     WAIT
18  ENDIF
19  IF .NOT. EOF()
20     CLEAR
21     @6,10 SAY "TITLE:" GET TITLE
22     @8,10 SAY "DATE:" GET DATE
23     @10,10 SAY "FILED IN FOLDER:" GET HARDCOPY
24     @12,10 SAY "DISKCOPY:" GET DISKCOPY
25     @14,10 SAY "FILENAME:" GET FILENAME
26     @22,0
27     WAIT
28  ENDIF
29  SET TALK ON
30  RETURN

31  * END OF DOCTITLE.PRG
```

Figure 16. DOCTITLE.PRG.

an "end of file" mark. Line 13 is an "if statement", and dBASE III will execute all the commands between this line and the "endif" line (line 18) when the specified condition of the "if statement" is met. Since we have assumed that dBASE III did not find a record that had the information for which we were looking, then the "end-of-file" mark has been encountered. That is what "IF EOF()" means. Therefore, dBASE III executes lines 14 through 17. Line 14 clears the screen, and line 15 displays a message stating the search was unsuccessful. The next two lines cause the program to interrupt its execution (which allows the operator to read the message on the screen) and cues the operator with the following message at the bottom of the screen: "PRESS ANY KEY TO CONTINUE...". When a key is pressed, dBASE III resumes at line 19 which (according to our assumption) is a false statement. Therefore dBASE III skips over all the subsequent command lines until it encounters an "endif" command. Line 29 sets the "talk" back on (remember that it was turned off in line 11). Line 30 "returns" dBASE III back to a higher program (in this case, "DOCUMENT.PRG") from which the DOCTITLE.PRG was "called".

Now we shall look at the sequence of commands, assuming that when line 12 is executed, a record meeting the criteria stated in that line is found. If a record is found, then the "end-of-file" condition is not true; and therefore, line 13 is a false statement (i.e., "IF EOF()" is not true). Then all the command lines between line 13 and line 18 ("endif") are skipped. Now line 19 is true. Therefore the commands in lines 20 through 27 are executed. Line 21 contains an "@...SAY, GET" command. What this command does is display (i.e., "SAY") on the screen at the specified position (row 6, column 10) the contents within the quotation marks (i.e., TITLE:); and immediately following the colon, it displays (i.e., "GET") in reverse video the contents of the TITLE field in the record which was just located. Lines 22 through 25 execute similar comments. Lines 26 and 27 make dBase III pause and cues the user to "PRESS ANY KEY TO CONTINUE...". When a key is pressed, line 29 turns the talk back on and line 30 is executed returning program control to the higher level program (DOCUMENT.PRG) from which the DOCTITLE.PRG was called.

Summarizing the search procedure for a single entry file, dBASE III looks for a record meeting the criteria specified by the user and if unsuccessful informs the user of that fact and if successful displays the information in that record.

B. Multiple-Entry Files

A multiple-entry file is one in which a specific value in a field to be searched might appear on more than one record in the database file. To illustrate this, look at the travel database shown in Figure 17. The "TRAVELER" field contains the name of the person on travel status, and of course no two travelers have the same name. However that does not mean that a given name only appears on one record in the database, because each trip a traveler makes is on a separate record. Therefore, if a traveler makes more than one trip, that name will appear more than once in a search of that field. A similar situation would exist on a search of the "DUTY_STA" field (more than one traveler can visit the same duty station or the same traveler could visit multiple times). Knowing that a search may yield more than one record, how do we extract and display information for multiple records?

```
Structure for database : A:oftravel.dbf
Number of data records :       9
Date of last update    : 01/16/86
Field  Field name  Type       Width    Dec
    1  CATEGORY    Character     10
    2  TITLE       Character     25
    3  TRAVELER    Character     25
    4  DUTY_STA    Character     25
    5  DATE        Date           8
    6  START_TDY   Date           8
    7  END_TDY     Date           8
    8  HARDCOPY    Character     20
** Total **                     130
```

Figure 17. Structure for Database A:OFTRAVEL.DBF.

Figure 18 illustrates one of the two routines that we use in this report. This figure is a printout of the search routine for a traveler's name. The line numbers on the left were added to assist in describing the routine and are not part of the routine. The information we would like to find is where did the person travel and on what date(s). Lines 3-5 cue the user to enter the name of the traveler for which

```
     *                          VOPERNAM.PRG              11/15/85

 1  * Called from vouchers.prg
 2  CLEAR
 3  MNAME = "                         "
 4  @10,20 SAY "PLEASE ENTER NAME OF TRAVELER:"
 5  @12,22 GET MNAME PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
 6  READ
 7  CLEAR
 8  MNAME = TRIM (MNAME)
 9  LOCATE FOR TRAVELER = "&MNAME"
10    IF EOF()
11      CLEAR
12      @4,20 SAY "&MNAME has NOT traveled this year."
13      @22,0
14      WAIT
15      LOOP
16    ENDIF
17    cntr = 8
18    @4,10 SAY "&mname has traveled to the following places this
19  year."
20    DO WHILE .NOT. EOF()
21      @6,10 SAY "DUTY STATION           HARDCOPY                DATE"
22      @cntr,10 SAY duty_sta
23      @cntr,35 SAY hardcopy
24      @cntr,55 SAY start_tdy
25      cntr = cntr + 2
26      CONTINUE
27    ENDDO
28  @22,0
29  WAIT
30  RETURN

31  * END OF VOPERNAM.PRG
```

Figure 18. VOPERNAM.PRG.

he is searching. Line 9 tells dBASE III to look for that name. If the name is not found (i.e., the end-of-file is encountered), dBASE III executes lines 10-16 which results in dBASE III displaying on the screen a message stating that the name that was entered has not traveled this year and returns control of the program to the higher program from which VOPERNAM.PRG was called. Line 17 defines a memory variable called "cntr" (counter) that will be used in a special way. Lines 18 and 19 are an "@...SAY" statement that causes the message in quotes to be displayed on the screen. Lines 20-27 establishes a "DO LOOP" which will continue to be executed until the end-of-file condition is met. If a record was not found, this part of the program is not executed. Line 21 is an "@...SAY" statement that creates "headers" for three columns. Line 22 is another "@...SAY" statement that displays the value in the "duty_sta" field at row 8 under the "DUTY STATION" header. Line 23 displays the value in the "hardcopy" field at row 8 under the "HARDCOPY" header, and line 24 displays the "start_tdy" field value at row 8 under the "DATE" header. Line 25 adds two to the "cntr" value which now makes cntr = 10. Line 26 is a "CONTINUE" statement which tells dBASE III to look for the next record which meets the "LOCATE" condition in line 9. If another record is found, then lines 21-25 are repeated. Notice that since cntr = 10, all the values for "duty_sta", "hardcopy", and "start_tdy" on the new record are displayed on line 10. Once again "cntr" is increased by two (cntr = 12) and dBASE III looks for another record that meets the "LOCATE" condition. The results are displayed on row 12 and the process is repeated until no new records are found (i.e., end-of-file is encountered). At this point dBASE III exits the "Do Loop," and executes lines 28 and 29 which cue the user to "press a key to continue." When a key is pressed, line 30 returns control to the program from which VOPERNAM.PRG was called. Summarizing the VOPERNAM.PRG routine, dBASE III looks for the name the user enters, displays a "no find message" if it cannot find a record meeting the specified condition, or displays a message stating that the traveler went to the following places this year and displays that information in a three column table below the message.

Another way of searching the database of a multiple-entry file is depicted in Figure 19. This is the printout of a routine (GENERAL.PRG) used to search the travel database for the category called "GENERAL". Lines 1-16 are essentially the same as in the routine described in the previous paragraphs of this section. There is a minor difference in that the "LOCATE" command (line 9) is for a predetermined condition (i.e., the user does not input a value). At line 17, major differences between this routine and the previous one begin to surface. Line 17 tells dBASE III to execute lines 18-27 if a record meeting the "LOCATE" command condition is found. Line 18 establishes headers for the information that is going to be subsequently displayed. Line 19 moves the cursor down to row 6. Line 20 keeps dBASE III from displaying the field title when a "DISPLAY" command is executed. Line 21 tells dBASE to display the "TITLE" and "HARDCOPY" fields for each record that meets the stated "FOR" criteria. Lines 22 and 23 turn the "talk" and "heading" back on. The remaining lines cause dBASE III to wait on the user to tell it to proceed before returning to the program from which GENERAL.PRG was called. Summarizing the GENERAL.PRG routine, dBASE III checks to see if a record meeting the predetermined "LOCATE" condition is found. If not, dBASE III displays a message to that effect and returns to the program from which GENERAL.PRG was called. If a record or records meeting the "LOCATE" condition is found, then dBASE III displays the appropriate information from these records in tabular form and waits on the user to tell it to proceed.

4. COMMENTS/SUGGESTIONS

dBASE III comes with an excellent, easy-to-use manual; but like most other programs or languages, repeated use and application uncover a lot of helpful hints and techniques that may or may not be included in the manual. Even though a hint or suggestion is included in the manual, the reason for or value of the suggestion is often not understood or appreciated until it is actually encountered in a practical application. A brief description of some of the "hints and suggestions" we found meaningful are included, and we are sure that the reader will find many others.

A. Input Errors

Once the search routines have been written, debugged, revised, and used successfully, are there are pitfalls or problems of which one should be aware? The answer is yes. There is always the chance, perhaps even the likelihood, of a data entry errors. These error can occur in two places. One of these is the entry of data when the search routine asks for it (e.g. "Please enter the name of traveler" in the travel routine). The name might be misspelled or entered incorrectly (e.g., first and middle initials might be used whereas in the database the first name might be spelled out). Also capitalization is very important. Use of the "PICTURE" command

```
*                                GENERAL.PRG      11/15/85

* called form travel.prg

CLEAR
@3,14 SAY "THE FOLLOWING ITEMS ARE FILED IN THE GENERAL
CATEGORY."
@23,0
WAIT
CLEAR
SET TALK OFF
LOCATE FOR CATEGORY = "GENERAL"
IF EOF()
  @10,20 SAY "NO DATA FILED IN GENERAL CATEGORY"
  @22,0
  WAIT
  SET TALK ON
  RETURN
ENDIF
IF .NOT. EOF()
  @5,10 SAY "TITLE                         HARDCOPY"
  @6,0
  SET HEADING OFF
  DISPLAY ALL TITLE, HARDCOPY FOR CATEGORY = "GENERAL"
  SET TALK ON
  SET HEADING ON
  @23,0
  WAIT
  RETURN
ENDIF

* END OF GENERAL.PRG
```

Figure 19. GENERAL.PRG.

helps somewhat in the data entry process by restricting the type of data accepted as input (for example all letters entered can be automatically capitalized). All data entered into our databases were capitalized, but this is not necessary and can be circumvented by use of the "UPPER" function in the search routines. However, we chose to enter all data in capitals and have all user entered data automatically converted to capitals.

The second area where data entry errors can occur are in the database themselves. Misspelling, punctuation, or entry into the wrong field can and does happen. Our databases were created, appended, and edited using the keyboard, but a better approach would be to write an input/edit program to guide the user in this process. This refinement will probably be added to our system later. Also entries should be kept as short and simple as possible. To accomplish this recommendation will require that memos, data, or other material generated within the office be given simple and short titles at the time of its generation. Otherwise, trying to rename an item or shorten its name at the time the information is entered into the database will just cause additional opportunities for errors to occur.

B. Data Display

There are essentially two commands for displaying data, and both are utilized in our routines. The "DISPLAY" command is used in some of those routines where there is more than one record's contents to be shown. This command has a major drawback in that it has a fixed format for displaying data. However, it is useful for scanning selected portions of an entire database, and this capability is used in some of the routines. The authors' preferred way of displaying data was to use the "@...SAY" command. There are two ways this command is used. The first is to use a "GET" option with it which results in the data being shown in reverse video. This technique is used when data from a single record is to be shown. The other way is to set up a counter and a "DO LOOP" and use the "@...SAY" command to display the desired information from multiple records in a tabular array. The advantage of this approach is being able to adjust the spacing in the tabular array for clarity. This technique will probably be used where possible when the authors update their search routines.

C. Indexing

None of the databases in the authors' system are indexed because none of the databases are very large (in this context, large refers to the number of records). For large databases the search is done faster if the database is indexed on the field being searched.

D. Revisions/Updates

Development of a system of search routines is done somewhat on a trial-and-error basis. The system presented in this report was done that way. In fact it was originally started as a means to become more proficient at using dBASE III. As each separate routine was written, debugged, and operated, ways to improve the routine would be discovered and incorporated into the next routine that was written. Some of the improvements have been mentioned in earlier paragraphs, but there was one major revision that was implemented after the system was "completed". This revision was the addition of a way to find data in some of the larger databases when the user might not have sufficient background information to find the item of interest. Figure 20 illustrates this concept. MISCELL.PRG is a search routine for the miscellaneous database (MISCELL.DBF), and as originally written required that either the "TITLE" or "DATE" of the item must be known. A third choice was added to the menu stating that if the "TITLE" or "DATE" were not known than the user should select item 3: "NEITHER OF THE ABOVE". If item 3 is selected than all the "TITLES" and corresponding "DATES" are displayed. If the desired item is seen, any key is pressed and now the "TITLE" or "DATE" option can be selected and entered, and all the information on that item will be displayed on the screen. If the desired information was not seen when item 3 was entered, the user returns to the Main Menu where another file can be selected. The point of this discussion is that the search routines are dynamic, and one should always be willing to incorporate suggestions or improvements.

```
*                                MISCELL.PRG           11/15/85

* Called from OFMAIN.PRG
USE ofmiscel.dbf
CLEAR
@8,16 SAY "MISCELLANEOUS FILES CONTAIN ONLY THOSE ITEMS THAT"
@10,16 SAY "DO NOT LOGICALLY BELONG IN ONE OF THE OTHER FILES."
@22,0
WAIT
DO WHILE .T.
  CLEAR
  OPTION = " "
  @4,16 SAY "YOU MUST KNOW ONE OF THE FOLLOWING ITEMS IN ORDER"
  @6,16 SAY "TO SEARCH THE MISCELLANEOUS FILES:"
  @9,25 SAY "1.TITLE"
  @11,25 SAY "2.DATE"
  @13,25 SAY "3.NEITHER OF THE ABOVE"
  @15,25 SAY "4.MAIN MENU"
  @18,16 SAY "PLEASE ENTER THE NUMBER OF YOUR CHOICE:"
  @18,56 GET OPTION PICTURE "9"
  READ
  DO CASE
    CASE OPTION = "1"
      DO MISTITLE
    CASE OPTION = "2"
      DO MISDATE
    CASE OPTION = "3"
      CLEAR
      DISPLAY ALL TITLE,DATE
      WAIT
      LOOP
    CASE OPTION = "4"
      CLEAR ALL
      RETURN
    OTHERWISE
      @23,15 SAY "INVALID CHOICE"
    WAIT
  ENDCASE
ENDDO

* END OF MISCELL.PRG
```

Figure 20. MISCELL.PROG.

PART III: OTHER FACTS

1. BIBLIOGRAPHY

Material for this report was based on the authors' personal experiences with dBASE III and on the following list of publications and short courses:

1. Allswang, John M., "dBASE III Is A Tough DBMS Act to Follow," Government Computer News, page 84, August 30, 1984.

2. dBASE III Manual, Ashton-Tate, 1984.

3. Prague, Cary N. and Hammitt, James C., Programming With dBASE, Tab Books, Inc., 1985.

4. Basic dBASE III, Training Course by BCSS, Sue Charles, February 1985.

5. Advanced dBASE III, Training Course by UAH, Jim Hudson, August 1985.

6. George T. Goley, IV., "dBASE III - Ashton-Tate's New Database Manager Addresses Many Of The Problems Associated With dBASE II," Popular Computing, March 1985.

7. Robert A. Byers, "Everyman's Database Primer Featuring dBASE III", Ashton-Tate, 1985.

8. Edward Jones, "Using dBASE III", Osborne-McGraw-Hill, 1985.

2. GLOSSARY

The following list of definitions has been included primarily as an aid to the non-technical user. We have assumed that the technical user will be familiar with most if not all the terminology used in this report.

database – Data that has been transformed into meaningful information by logically grouping and storing it.

database structure – The names, types (e.g. numerical, character, etc.), and sizes of the fields within the database.

field – An item of information within a record.

file – Generally used interchangeably with database.

program – A set of computer instructions that operates on input data and converts it to output.

prompt – A symbol, word, or descriptive phrase that is used to indicate the computer's readiness to accept data or commands.

record – A grouping of items of information that go together.

reverse video – An enhanced background differing from the normal background.

routine – A set of instructions used to perform a specific operation (used interchangeably with program).

software – Programs written to be used on a computer.

3. PROGRAM PRINTOUT

The following pages contain the complete system of programs that accomplish the search and retrieval of data within the authors' office files.

```
*                         OFMAIN.PRG                       11/12/85

* PURPOSE: serves as the main menu program for an automated search
*          routine of the GP-B Task Team files.

CLEAR ALL

* Sign on

DO ofsignon

* Initialize the program

DO ofinit

* Set up the main loop forever

DO WHILE .T.
   SET FORMAT TO ofmainmu
   option ="  "
* Erase any error message
   @23,0
* Activate ofmainmu and get option
   READ
   CLOSE FORMAT
   DO CASE
     case option = "01"
       DO personne
     case option = "02"
       DO mancom
     case option = "03"
       DO reading
     case option = "04"
       DO travel
     case option = "05"
       DO miscell
     case option = "06"
       DO manpower
     case option = "07"
       DO poprtop
     case option = "08"
       DO projdata
     case option = "09"
       DO presentn
     case option = "10"
       DO document
     case option = "11"
       CLEAR ALL
       CLEAR
       CANCEL
   OTHERWISE
     @23,15 SAY "INVALID CHOICE"
   WAIT
   ENDCASE
ENDDO
* End of OFMAIN.PRG
*                          OFSIGNON.PRG

* Called from Ofmain.prg to sign on program

CLEAR
@11,30 SAY "OFFICE FILES_____"
@13,25 SAY "AN AUTOMATED SEARCH ROUTINE"
@20,0
WAIT
RETURN
```

```
*                              OFINIT.PRG        11/25/85

* This program sets up the environment, default values for some * variables.

* "Set" values will go in here after some trouble shooting.
SET BELL OFF
SET EXACT ON
RETURN

* END OF OFINIT.PRG
*                             OFMAINMU.FMT          11/15/85

@ 1,34 SAY "MAIN MENU"
@ 5,25 SAY "All Data Is Filed In One Of"
@ 6,25 SAY "The Following Categories:"
@ 8,11 SAY "01. PERSONNEL                      07. POP/RTOP"
@ 9,11 SAY "02. MANAGEMENT COMMUNICATIONS      08. PROJECT DATA"
@ 10,11 SAY "03. READING FILES                  09. PRESENTATIONS"
@ 11,11 SAY "04. TRAVEL                         10. DOCUMENTS"
@ 12,11 SAY "05. MISCELLANEOUS                  11. EXIT"
@ 13,11 SAY "06. MANPOWER                               "
@ 15,25 SAY "SELECT THE ONE MOST LIKELY TO"
@ 16,23 SAY "CONTAIN THE INFORMATION YOU SEEK."
@ 19,15 SAY "PLEASE ENTER THE NUMBER OF YOUR CHOICE:"
@ 19,55 GET option PICTURE "99"
```

```
*                         PERSONNE.PRG          11/18/85

* CALLED FROM OFMAIN

USE OFPERSON
CLEAR
DO WHILE .T.
SET FORMAT TO personne.fmt
option =" "
@23,0
READ
CLOSE FORMAT
DO CASE
  case option = "1"
      DO award
  case option = "2"
      DO admin
  case option = "3"
      DO training
  case option = "4"
      DO perfapr
  case option = "5"
      RETURN
OTHERWISE
  @23,15 SAY "INVALID CHOICE"
ENDCASE
GO TOP
ENDDO
* END OF PERSONNE.PRG
*                          PERSONNE.FMT            11/15/85

@ 6,7 SAY "PERSONNEL FILES ARE DIVIDED INTO THE FOLLOWING CATEGORI"
@ 6,62 SAY "ES:"
@ 8,19 SAY "1. AWARDS"
@ 9,19 SAY "2. ADMINISTRATIVE"
@ 10,19 SAY "3. TRAINING"
@ 11,19 SAY "4. PERFORMANCE APPRAISALS"
@ 12,19 SAY "5. MAIN MENU"
@ 15,7 SAY "PLEASE PRESS THE NUMBER OF YOUR CHOICE:"
@ 15,47 GET OPTION PICTURE "9"
```

```
Structure for database : A:OFPERSON.dbf
Number of data records :       38
Date of last update    : 12/03/85
Field  Field name  Type       Width    Dec
    1  CATEGORY    Character     25
    2  TITLE       Character     20
    3  NAME        Character     25
    4  DATE        Date           8
    5  HARDCOPY    Character     20
    6  DISKCOPY    Character     20
    7  FILENAME    Character     12
** Total **                     131
```

```
*                          AWARD.PRG            11/13/85

* CALLED FROM PERSONNE.PRG

CLEAR
@3,14 SAY "If you see the data you are searching for, REMEMBER"
@5,14 SAY "its RECORD NUMBER for use on the next screen."
@23,0
WAIT
CLEAR
DISPLAY ALL TITLE, NAME, DATE FOR CATEGORY = "AWARDS"
@23,0
WAIT
recno ="  "
CLEAR
@5,10 SAY "ENTER RECORD NUMBER"
@5,30 GET recno PICTURE "99"
READ
GO VAL(recno)
CLEAR
@5,10 SAY "AWARD:" GET title
@7,10 SAY "RECIPIENT:" GET name
@9,10 SAY "DATE OF AWARD:" GET date
@11,10 SAY "HARDCOPY LOCATION:" GET hardcopy
@13,10 SAY "DISKCOPY LOCATION:" GET diskcopy
@15,10 SAY "FILENAME ON DISKCOPY:" GET filename
@23,0
WAIT
RETURN

* END OF AWARD.PRG
*                           ADMIN.PRG          11/25/85

* CALLED FROM PERSONNE.PRG

CLEAR
@3,14 SAY "If you see the data you are searching for, REMEMBER"
@5,14 SAY "its RECORD NUMBER for use on the next screen."
@23,0
WAIT
CLEAR
SET TALK OFF
LOCATE FOR CATEGORY = "AWARDS"
IF EOF()
   CLEAR
   @10,15 SAY "NO AWARDS HAVE BEEN GIVEN THIS YEAR."
   @22,0
   WAIT
   SET TALK ON
   RETURN
ENDIF
DISPLAY ALL TITLE,DATE FOR CATEGORY = "ADMINISTRATIVE"
@22,0
WAIT
CLEAR
MDATA = " "
@10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
@11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
@12,15 SAY "PRESS ANY KEY TO CONTINUE."
@12,41 GET MDATA PICTURE "!"
READ
   IF MDATA = "Y"
      RECNO = "  "
      CLEAR
      @5,10 SAY "ENTER RECORD NUMBER"
      @5,30 GET recno PICTURE "99"
      READ
      GO VAL(recno)
      CLEAR
      @5,10 SAY "TITLE:" GET title
      @7,10 SAY "DATE:" GET date
      @9,10 SAY "HARDCOPY LOCATION:" GET hardcopy
      @22,0
      WAIT
      SET TALK ON
      RETURN
ELSE
      SET TALK ON
      RETURN
ENDIF

* END OF ADMIN.PRG
```

```
DO WHILE .T.
SET TALK OFF
CLEAR
OPTION = " "
@3,10 SAY "You must know one of the following items in order"
@5,10 SAY "to search for training data:"
@8,19 SAY "1. PERSON'S NAME:"
@10,19 SAY "2. TITLE OF TRAINING COURSE:"
@12,19 SAY "3. PREVIOUS MENU:"
@16,7 SAY "PLEASE PRESS THE NUMBER OF YOUR CHOICE:"
@16,47 GET OPTION PICTURE "9"
READ
CLEAR
DO CASE
   CASE option = "1"
      CLEAR
      MNAME = "                         "
      @4,7 SAY "PLEASE ENTER NAME OF PERSON RECEIVING TRAINING:"
      @6,20 GET MNAME PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
      READ
      CLEAR
      MNAME = TRIM (MNAME)
      LOCATE FOR CATEGORY = "TRAINING" .AND. NAME = "&MNAME"
      IF EOF()
        CLEAR
        @4,20 SAY "&MNAME has NOT received any TRAINING this year."
        @22,0
        WAIT
        LOOP
      ENDIF
      cntr = 8
      @4,10 SAY "&mname has received the following TRAINING:"
      @6,10 SAY "COURSE TITLE             HARDCOPY           DATE"
      DO WHILE .NOT. EOF()
         @cntr,10 say title
         @cntr,37 SAY hardcopy
         @cntr,52 SAY date
         cntr = cntr + 2
         CONTINUE
      ENDDO
      @22,0
      WAIT
      CASE OPTION = "2"
         CLEAR
         MTITLE = "                    "
         @4,7 SAY "PLEASE ENTER TITLE OF TRAINING COURSE:"
         @6,20 GET MTITLE PICTURE "!!!!!!!!!!!!!!!!!!!!"
         READ
         CLEAR
         MTITLE = TRIM(MTITLE)
         LOCATE FOR CATEGORY = "TRAINING" .AND. TITLE = "&MTITLE"
         IF EOF()
            CLEAR
            @4,20 SAY "No one took &mtitle this year."
            @22,0
            WAIT
            LOOP
         ENDIF
         CNTR = 8
         @4,10 SAY "The following people took &mtitle:"
         @6,14 SAY "NAME                     HARDCOPY          DATE'
         DO WHILE .NOT. EOF()
            @CNTR,10 SAY NAME
            @CNTR,41 SAY HARDCOPY
            @CNTR,55 SAY DATE
            CNTR = CNTR + 2
            CONTINUE
         ENDDO
         @22,0
         WAIT

      CASE OPTION = "3"
         SET TALK ON
         RETURN
      OTHERWISE
         @23,15 SAY "INVALID CHOICE"
         WAIT
ENDCASE
ENDDO

* END OF TRAINING.PRG
*                            PERFAPR.PRG             11/25/85

*CALLED FROM PERSONNE.PRG

CLEAR
@3,14 SAY "If you see the data you are searching for, REMEMBER"
@5,14 SAY "its RECORD NUMBER for use on the next screen."
@23,0
WAIT
CLEAR
SET TALK OFF
LOCATE FOR CATEGORY = "PERFORMANCE APPRAISALS"
IF EOF()
   CLEAR
   @10,15 SAY "THERE ARE NO PERFORMANCE APPRAISALS LISTED UNDER THIS NAME
   @22,0
   WAIT
   SET TALK ON
   RETURN
ENDIF
DISPLAY ALL TITLE,NAME,DATE FOR CATEGORY = "PERFORMANCE APPRAISALS"
@22,0
WAIT
CLEAR
MDATA = "  "
@10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
@11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
@12,15 SAY "PRESS ANY KEY TO CONTINUE."
@12,41 GET MDATA PICTURE "!"
READ
   IF MDATA = "Y"
      RECNO = "  "
      CLEAR
      @10,25 SAY "ENTER RECORD NUMBER"
      @10,45 GET recno PICTURE "99"
      READ
      GO VAL(recno)
      CLEAR
      @5,10 SAY "NAME:" GET name
      @7,10 SAY "DATE:" GET date
      @9,10 SAY "HARDCOPY LOCATION:" GET hardcopy
      @23,0
      WAIT
      SET TALK ON
      RETURN
ELSE
   SET TALK ON
   RETURN
ENDIF

* END OF PERFAPR.PRG
*                         MANCOM.PRG                    11/15/85

* CALLED FROM OFMAIN.PRG

* Use management communications database.

USE OFMGMTCO
CLEAR
DO WHILE .T.
   SET FORMAT TO OFMGMTCO.FMT
   option = " "
* Clear any error message that might be present.
   @23,0
   READ
   CLOSE FORMAT
   DO CASE
     CASE OPTION = "1"
       DO mgmtann
     CASE OPTION = "2"
       DO hqann
     CASE OPTION ="3"
       RETURN
     OTHERWISE
       @23,15 SAY "INVALID CHOICE"
       WAIT
   ENDCASE
   GO TOP
ENDDO
* END OF MANCOM.PRG
```

```
*                        OFMGMTCO.FMT        11/15/85

@ 6,6 SAY "MANAGEMENT FILES are divided into the following categor"
@ 6,61 SAY "ies:"
@ 8,25 SAY "1. MSFC MANAGEMENT ANNOUNCEMENTS"
@ 10,25 SAY "2. HEADQUARTERS ANNOUNCEMENTS"
@ 12,25 SAY "3. MAIN MENU"
@ 16,6 SAY "PLEASE ENTER THE NUMBER OF YOUR CHOICE:"
@ 16,46 GET OPTION PICTURE "9"
```

```
Structure for database : A:OFMGMTCO.dbf
Number of data records :      15
Date of last update    : 11/13/85
Field  Field name  Type        Width    Dec
    1  CATEGORY    Character      30
    2  TITLE       Character      45
    3  NUMBER      Character      10
    4  DATE        Date            8
    5  HARDCOPY    Character      20
    6  DISKCOPY    Character      20
    7  FILENAME    Character      12
** Total **                      146
```

```
* CALLED FROM MANCOM.PRG
DO WHILE .T.
CLEAR
PROMPT1= " "
@3,10 SAY "You must know one of the following items in order"
@5,10 SAY "to search for a particular management announcement:"
@8,17 SAY "1. SUBJECT OF ANNOUNCEMENT(USUALLY A PERSON'S NAME)"
@10,17 SAY "2. ANNOUNCEMENT NUMBER"
@12,17 SAY "3. DATE OF ANNOUNCEMENT"
@14,17 SAY "4. NONE OF THE ABOVE"
@16,17 SAY "5. PREVIOUS MENU"
@19,10 SAY "PLEASE ENTER THE NUMBER OF YOUR CHOICE:"
@19,50 GET PROMPT1 PICTURE "9"
READ
CLEAR
SET TALK OFF
DO CASE
  CASE prompt1 = "1"
     mtitle = "                         "
     @3,10 SAY "ENTER TITLE:"
     @3,23 GET mtitle PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
     READ
     CLEAR
     mtitle = TRIM(mtitle)
     SET TALK OFF
     LOCATE FOR TITLE = "&MTITLE"
     IF EOF()
       @10,20 SAY "&mtitle NOT ON FILE."
       @20,0
       WAIT
       LOOP
     ELSE
       @8,15 SAY "NAME:" GET TITLE
       @10,15 SAY "ANNOUNCEMENT NO.:" GET NUMBER
       @12,15 SAY "DATE:" GET DATE
       @14,15 SAY "HARDCOPY:" GET HARDCOPY
       @20,0
       WAIT
     ENDIF
     LOOP
    CASE prompt1 = "2"
      mnumb = "       "
      @3,10 SAY "ENTER ANNOUNCEMENT NUMBER:"
      @3,37 GET mnumb PICTURE "!!#99"
      READ
      CLEAR
      mnumb = TRIM(mnumb)
      LOCATE FOR NUMBER = "&MNUMB"
      IF EOF()
        @10,20 SAY "&mnumb NOT ON FILE."
        @20,0
        WAIT
        LOOP
      ELSE
        @8,15 SAY "NAME:" GET title
       @10,15 SAY "ANNOUNCEMENT NO.:" GET number
       @12,15 SAY "DATE:" GET date
       @14,15 SAY "HARDCOPY:" GET hardcopy
       @20,0
       WAIT
    ENDIF
    LOOP
   CASE prompt1 = "3"
       mdate = "  /  /  "
       @3,10 SAY "ENTER DATE OF ANNOUNCEMENT:"
       @3,48 GET mdate PICTURE "99/99/99"
       READ
       CLEAR
       LOCATE FOR DATE = CTOD("&mdate")
       IF EOF()
         @10,20 SAY "NO ITEM FOUND DATED &MDATE"
         @20,0
         WAIT
         LOOP
       ELSE
         @8,15 SAY "NAME:" GET title
         @10,15 SAY "ANNOUNCEMENT NO.:" GET number
         @12,15 SAY "DATE:" GET date
         @14,15 SAY "HARDCOPY:" GET hardcopy
         @20,0
         WAIT
       ENDIF
       LOOP
   CASE prompt1 = "4"
       DISPLAY ALL TITLE,NUMBER,DATE FOR CATEGORY="MSFC MANAGEMENT ANNOUNCEM
       WAIT
       LOOP
   CASE prompt1 = "5"
     SET TALK ON
     RETURN
   OTHERWISE
     @23,15 SAY "INVALID CHOICE"
     WAIT
ENDCASE
ENDDO

* END OF MGMTANN.PRG
*                        HQANN.PRG           11/27/85

CLEAR
@3,14 SAY "If you see the data you are searching for, REMEMBER"
@5,14 SAY "its RECORD NUMBER for use on the next screen."
@23,0
WAIT
CLEAR
SET TALK OFF
LOCATE FOR CATEGORY = "HEADQUARTERS ANNOUNCEMENT"
IF EOF()
  CLEAR
  @10,15 SAY "NO HEADQUARTERS ANNOUNCEMENTS FOUND."
  @22,0
  WAIT
  SET TALK ON
  RETURN
ENDIF
DISPLAY ALL TITLE,DATE FOR CATEGORY = "HEADQUARTERS ANNOUNCEMENT"
@22,0
WAIT
CLEAR
MDATA = "  "
@10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
@11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
@12,15 SAY "PRESS ANY KEY TO CONTINUE."
@12,41 GET MDATA PICTURE "!"
READ
IF MDATA = 'Y'
   RECNO = "  "
   CLEAR
   @5,10 SAY "ENTER RECORD NUMBER"
   @5,30 GET recno PICTURE "99"
   READ
   GO VAL(recno)
   CLEAR
   @3,20 SAY "HEADQUARTERS ANNOUNCEMENT"
   @6,10 SAY "TITLE:" GET title
   @8,10 SAY "DATE OF ANNOUNCEMENT:" GET date
   @10,10 SAY "HARDCOPY LOCATION:" GET hardcopy
   @22,0
   WAIT
   SET TALK ON
   RETURN
ELSE
   SET TALK ON
   RETURN
ENDIF

* END OF HQANN.PRG
*                                 READING.PRG         11/15/85

USE ofreadng.dbf
CLEAR
@8,18 SAY "READING FILES ARE ORGANIZED BY MONTH AND CONTAIN"
@10,18 SAY "ONLY MATERIAL ORIGINATED BY THE TASK TEAM."
@22,0
WAIT
DO WHILE .T.
  CLEAR
  option = " "
  @4,18 SAY "YOU MUST KNOW ONE OF THE FOLLOWING ITEMS IN"
  @6,18 SAY "ORDER TO SEARCH THE READING FILES."
  @9,25 SAY "1. MONTH(displays ALL files for that month)"
  @11,25 SAY "2. DATE"
  @13,25 SAY "3. SUBJECT"
  @15,25 SAY "4. ADDRESSEE"
  @17,25 SAY "5. MAIN MENU"
  @20,16 SAY "PLEASE PRESS THE NUMBER OF YOUR CHOICE:"
  @20,56 GET option PICTURE "9"
  READ
  DO CASE
    CASE option = "1"
      DO remonth
    CASE option = "2"
      DO redate
    CASE option = "3"
      DO resubj
    CASE option = "4"
      DO readdres
    CASE option = "5"
      CLEAR ALL
      RETURN
    OTHERWISE
      @23,15 SAY "INVALID CHOICE"
    WAIT
  ENDCASE
ENDDO
* END OF READING.PRG
```

```
Structure for database : A:OFREADNG.dbf
Number of data records :       90
Date of last update    : 12/03/85
Field  Field name  Type       Width    Dec
    1  MONTH       Character     10
    2  DATE        Date           8
    3  SUBJECT     Character     45
    4  TO          Character     20
    5  FROM        Character     20
    6  HARDCOPY    Character     20
    7  DISKCOPY    Character     20
    8  FILENAME    Character     12
** Total **                     156
```

```
*                              REMONTH.PRG                    11/14/85

* called from READNG.PRG
SET TALK OFF
CLEAR
mmonth ="         "
@12,10 SAY "ENTER THE NAME OF THE MONTH:"
@12,40 GET mmonth PICTURE "!!!!!!!!!!"
READ
mmonth = TRIM(mmonth)
CLEAR
LOCATE FOR MONTH = "&mmonth"
IF .NOT. EOF()
  CLEAR
  @4,10 SAY "THE FOLLOWING DATA IS SHOWN FOR EACH ITEM IN THE &MMONTH FIL
  @9,15 SAY "DATE     ADDRESSEE            SUBJECT"
  @22,0
  WAIT
  CLEAR
  SET HEADING OFF
  @4,2 SAY "DATE     ADDRESSEE            SUBJECT"
  @6,0
  DISPLAY OFF ALL DATE,TO,SUBJECT FOR MONTH = "&mmonth"
  @23,0
  WAIT
  SET HEADING ON
  SET TALK ON
  RETURN
ENDIF
IF EOF()
  CLEAR
  @10,20 SAY "NO ITEMS FOUND THAT ARE DATED &MMONTH."
  @20,0
  WAIT
  SET TALK ON
  RETURN
ENDIF


* end of REMONTH.PRG
*                                   REDATE.PRG                   11/22/85

*   CALLED FROM READING.PRG

CLEAR
SET TALK OFF
mdate = "  /  /  "
@10,10 SAY "ENTER THE DATE FOR WHICH YOU ARE SEARCHING:"
@10,53 GET mdate PICTURE "99/99/99"
READ
CLEAR
SET HEADING OFF
LOCATE FOR DATE = CTOD("&mdate")
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING ITEMS WERE GENERATED ON &mdate"
  @6,10 SAY "IF YOU SEE THE ITEM YOU ARE SEEKING REMEMBER ITS RECORD NUMBER."
  @9,10 SAY "ADDRESSEE            SUBJECT"
  DISPLAY ALL TO, SUBJECT FOR DATE = CTOD("&mdate")
  SET HEADING ON
  @23,0
  WAIT
  CLEAR
  RECON = " "
  @10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
  @11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM. OTHERWISE,"
  @12,15 SAY "PRESS ANY OTHER KEY TO CONTINUE."
  @12,47 GET RECON PICTURE "!"
  READ
  IF RECON = "Y"
    CLEAR
    mrecno = "   "
    @10,20 SAY "PLEASE ENTER RECORD NUMBER:"
    @10,48 GET mrecno PICTURE "999"
    READ
    CLEAR
    GO VAL(mrecno)
    SET TALK ON
    @8,10 SAY "DATE:"
    @8,21 GET DATE
    @10,10 SAY "SUBJECT:"
    @10,21 GET SUBJECT
    @12,10 SAY "ADDRESSEE:"
    @12,21 GET TO
    @14,10 SAY "FROM:"
    @14,21 GET FROM
    @16,10 SAY "HARDCOPY:"
    @16,21 GET HARDCOPY
    @18,10 SAY "DISKCOPY:"
    @18,21 GET DISKCOPY
    @20,10 SAY "FILENAME:"
    @20,21 GET FILENAME
    @22,0
    WAIT
    LOOP
  ELSE
    LOOP
    ENDIF
```

```
ENDIF
IF EOF()
  CLEAR
  SET TALK ON
  SET HEADING ON
  @10,20 SAY "NO ITEM FOUND DATED &mdate."
  @20,0
  WAIT
  LOOP
ENDIF
*  END OF REDATE.PRG
*                              RESUBJ.PRG                    11/26/85

*   CALLED FROM READING.PRG
CLEAR
SET TALK OFF
msubj = "                                             "
@10,15 SAY "ENTER THE SUBJECT FOR WHICH YOU ARE SEARCHING:"
@11,15 GET msubj PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
msubj = TRIM(msubj)
CLEAR
SET HEADING OFF
LOCATE FOR SUBJECT = "&msubj"
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING ITEMS ARE TITLED &msubj."
  @6,10 SAY "IF YOU SEE THE ITEM YOU ARE SEEKING REMEMBER ITS RECORD NUMBER."
  @9,10 SAY "ADDRESSEE             DATE"
  DISPLAY ALL TO,DATE FOR SUBJECT = "&msubj"
  SET HEADING ON
  @23,0
  WAIT
  CLEAR
  recon = "  "
  @10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
  @11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
  @12,15 SAY "PRESS ANY OTHER KEY TO CONTINUE."
  @12,47 GET RECON PICTURE "!"
  READ
  IF RECON = "Y"
     CLEAR
     mrecno = "   "
     @10,20 SAY "PLEASE ENTER RECORD NUMBER:"
     @10,48 GET mrecno PICTURE "999"
     READ
     CLEAR
     GO VAL(mrecno)
     SET TALK ON
     @8,10 SAY "DATE:"
     @8,21 GET DATE
     @10,10 SAY "SUBJECT:"
     @10,21 GET SUBJECT
     @12,10 SAY "ADDRESSEE:"
     @12,21 GET TO
     @14,10 SAY "FROM:"
     @14,21 GET FROM
     @16,10 SAY "HARDCOPY:"
     @16,21 GET HARDCOPY
     @18,10 SAY "DISKCOPY:"
     @18,21 GET DISKCOPY
     @20,10 SAY "FILENAME:"
     @20,21 GET FILENAME
     @22,0
     WAIT
     LOOP
 ELSE
   SET TALK ON
      LOOP
  ENDIF
ENDIF
IF EOF()
   CLEAR
   SET TALK ON
   SET HEADING ON
   @10,14 SAY "NO ITEM FOUND TITLED &msubj."
   @20,0
   WAIT
   LOOP
ENDIF


*  END OF RESUBJ.PRG
*                              READDRES.PRG            11/27/85

*   CALLED FROM READING.PRG
CLEAR
SET TALK OFF
maddr = "                    "
@10,10 SAY "ENTER THE ADDRESSEE FOR WHICH YOU ARE SEARCHING:"
@10,58 GET maddr PICTURE "!!!!!!!!!!!!!!!!!!!!"
READ
maddr = TRIM(maddr)
CLEAR
SET HEADING OFF
LOCATE FOR TO = "&maddr"
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING ITEMS WERE ADDRESSED TO &maddr."
  @6,10 SAY "IF YOU SEE THE ITEM YOU ARE SEEKING REMEMBER ITS RECORD NUMBER."
  @9,10 SAY "SUBJECT                                       DATE"
  DISPLAY ALL SUBJECT,DATE FOR TO = "&maddr"
  SET HEADING ON
  @23,0
  WAIT
  CLEAR
  recon = " "
  @10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
  @11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
  @12,15 SAY "PRESS ANY OTHER KEY TO CONTINUE."
  @12,47 GET RECON PICTURE "!"
  READ
  IF RECON = "Y"
     CLEAR
     mrecno = "   "
     @10,20 SAY "PLEASE ENTER RECORD NUMBER:"
     @10,48 GET mrecno PICTURE "999"
     READ
     CLEAR
     GO VAL(mrecno)
     SET TALK ON
     @8,10 SAY "DATE:"
     @8,21 GET DATE
     @10,10 SAY "SUBJECT:"
     @10,21 GET SUBJECT
     @12,10 SAY "ADDRESSEE:"
     @12,21 GET TO
     @14,10 SAY "FROM:"
     @14,21 GET FROM
     @16,10 SAY "HARDCOPY:"
     @16,21 GET HARDCOPY
     @18,10 SAY "DISKCOPY:"
     @18,21 GET DISKCOPY
     @20,10 SAY "FILENAME:"
     @20,21 GET FILENAME
     @22,0
     WAIT
     LOOP
   ELSE
     SET TALK ON
     LOOP
  ENDIF
     ENDIF
IF EOF()
  CLEAR
  SET TALK ON
  SET HEADING ON
  @10,20 SAY "NO ITEM FOUND ADDRESSED TO &maddr."
  @20,0
  WAIT
  LOOP
ENDIF
*  END OF READDRES.PRG
```

```
*                            TRAVEL.PRG         11/18/85

* Called From ofmain.PRG

USE oftravel
CLEAR
DO WHILE .T.
SET FORMAT TO travel.fmt
option = " "
@23,0
READ
CLOSE FORMAT
DO CASE
   case option = "1"
        DO vouchers
   case option = "2"
        DO general
   case option = "3"
        CLEAR ALL
        RETURN
OTHERWISE
   @23,10 SAY "INVALID CHOICE"
   WAIT
ENDCASE
ENDDO

* END OF TRAVEL.PRG
*                      TRAVEL.FMT                  11/15/85
*
@3,31 SAY "TRAVEL MENU"
*
@8,13 SAY "Travel Files Consist of the Following Two Items:"
@10,31 SAY "1. VOUCHERS"
@11,31 SAY "2. GENERAL"
@12,31 SAY "3. MAIN MENU"
@14,13 SAY "PLEASE PRESS THE NUMBER OF YOUR CHOICE:"
@14,53 GET OPTION PICTURE "9"
```

```
Structure for database : A:OFTRAVEL.dbf
Number of data records :       7
Date of last update    : 12/03/85
Field  Field name  Type       Width    Dec
    1  CATEGORY    Character     10
    2  TITLE       Character     25
    3  TRAVELER    Character     25
    4  DUTY_STA    Character     25
    5  DATE        Date           8
    6  START_TDY   Date           8
    7  END_TDY     Date           8
    8  HARDCOPY    Character     20
** Total **                     130
```

```
*                                   VOUCHERS.PRG              02/13/86

* Called from travel.prg

SET TALK OFF
DO WHILE .T.
CLEAR
PROMPT1= " "
@2,28 SAY "VOUCHER MENU"
@5,16 SAY "You must know one of the following items in order"
@7,16 SAY "to search for voucher data:"
@10,25 SAY "1. PERSON'S NAME:"
@12,25 SAY "2. DUTY STATION:"
@14,25 SAY "3. DATE:"
@16,25 SAY "4. PREVIOUS MENU:"
@20,16 SAY "PLEASE PRESS THE NUMBER OF YOUR CHOICE:"
@20,56 GET PROMPT1 PICTURE "9"
READ
CLEAR
DO CASE
   CASE prompt1= "1"
     DO VOPERNAM
   CASE prompt1 = "2"
     DO VODUTYST
   CASE prompt1 = "3"
     DO VODATE
   CASE PROMPT1 = "4"
     SET TALK ON
     RETURN
   OTHERWISE
      @23,15 SAY "INVALID CHOICE"
      WAIT
ENDCASE
ENDDO

* END OF VOUCHERS.PRG
*                          VOPERNAM.PRG                 11/15/85

* Called from vouchers.prg
CLEAR
MNAME = "                         "
@10,20 SAY "PLEASE ENTER NAME OF TRAVELER:"
@12,22 GET MNAME PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
CLEAR
MNAME = TRIM (MNAME)
LOCATE FOR TRAVELER = "&MNAME"
  IF EOF()
    CLEAR
    @4,20 SAY "&MNAME has NOT traveled this year."
    @22,0
    WAIT
    LOOP
  ENDIF
  cntr = 8
  @4,10 SAY "&mname has traveled to the following places this year."
  DO WHILE .NOT. EOF()
    @6,10 SAY "DUTY STATION           HARDCOPY                DATE"
    @cntr,10 SAY duty_sta
    @cntr,35 SAY hardcopy
    @cntr,55 SAY start_tdy
    cntr = cntr + 2
    CONTINUE
  ENDDO
@22,0
WAIT
RETURN

* END OF VOPERNAM.PRG
*                                         VODUTYST.PRG           11/15/85

* CALLED FROM VOUCHERS.PRG
CLEAR
MSTATION = "                         "
@10,20 SAY "PLEASE ENTER THE NAME OF THE DUTY STATION:"
@12,29 GET MSTATION PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
MSTATION = TRIM(MSTATION)
LOCATE FOR DUTY_STA = "&MSTATION"
CLEAR
IF EOF()
  @6,15 SAY "NO ONE FROM THE TASK TEAM HAS TRAVELED TO &MSTATION THIS YEAR
  @22,0
  WAIT
  RETURN
ENDIF
CNTR = 9
DO WHILE .NOT. EOF()
  @4,10 SAY "THE FOLLOWING PERSONNEL TRAVELED TO &MSTATION THIS YEAR"
  @6,10 SAY "TRAVELER                  DATE       HARDCOPY"
  @CNTR,8 SAY TRAVELER
  @CNTR,34 SAY START_TDY
  @CNTR,49 SAY HARDCOPY
  CNTR = CNTR + 2
  CONTINUE
ENDDO
@22,0
WAIT
RETURN


* END OF VODUTYST.PRG
*                           VODATE.PRG                 11/15/85

* CALLED FROM VOUCHERS.PRG

CLEAR
MDATE = "  /  /  "
@10,15 SAY "PLEASE ENTER THE BEGINNING DATE OF TRAVEL:"
@10,60 GET MDATE PICTURE "99/99/99"
READ
LOCATE FOR START_TDY = CTOD("&MDATE")
CLEAR
IF EOF()
  @10,15 SAY "NO ONE FROM THE TASK TEAM BEGAN TRAVEL ON &MDATE."
  @22,0
  WAIT
ENDIF
CNTR = 9
DO WHILE .NOT. EOF()
  @4,10 SAY "THE FOLLOWING PERSONNEL BEGAN TRAVEL ON &MDATE:"
  @7,8 SAY "TRAVELER                    DUTY_STA           HARDCOPY"
  @CNTR,6 SAY TRAVELER
  @CNTR,33 SAY DUTY_STA
  @CNTR,56 SAY HARDCOPY
  CNTR = CNTR + 2
  CONTINUE
ENDDO
@22,0
WAIT
RETURN

* END OF VODATE.PRG
*                                GENERAL.PRG       11/15/85

* called form travel.prg

CLEAR
@3,14 SAY "THE FOLLOWING ITEMS ARE FILED IN THE GENERAL CATEGORY."
@23,0
WAIT
CLEAR
SET TALK OFF
LOCATE FOR CATEGORY = "GENERAL"
IF EOF()
  @10,20 SAY "NO DATA FILED IN GENERAL CATEGORY"
  @22,0
  WAIT
  SET TALK ON
  RETURN
ENDIF
IF .NOT. EOF()
  @5,10 SAY "TITLE                   HARDCOPY"
  @6,0
  SET HEADING OFF
  DISPLAY ALL TITLE, HARDCOPY FOR CATEGORY = "GENERAL"
  SET TALK ON
  SET HEADING ON
  @23,0
  WAIT
  RETURN
ENDIF

* END OF GENERAL.PRG
*                              MISCELL.PRG            11/15/85

* Called from OFMAIN.PRG
USE ofmiscel.dbf
CLEAR
@8,16 SAY "MISCELLANEOUS FILES CONTAIN ONLY THOSE ITEMS THAT"
@10,16 SAY "DO NOT LOGICALLY BELONG IN ONE OF THE OTHER FILES."
@22,0
WAIT
DO WHILE .T.
  CLEAR
  OPTION = " "
  @4,16 SAY "YOU MUST KNOW ONE OF THE FOLLOWING ITEMS IN ORDER"
  @6,16 SAY "TO SEARCH THE MISCELLANEOUS FILES:"
  @9,25 SAY "1.TITLE"
  @11,25 SAY "2.DATE"
  @13,25 SAY "3.NEITHER OF THE ABOVE"
  @15,25 SAY "4.MAIN MENU"
  @18,16 SAY "PLEASE ENTER THE NUMBER OF YOUR CHOICE:"
  @18,56 GET OPTION PICTURE "9"
  READ
  DO CASE
    CASE OPTION = "1"
      DO MISTITLE
    CASE OPTION = "2"
      DO MISDATE
    CASE OPTION = "3"
      CLEAR
      DISPLAY ALL TITLE,DATE
      WAIT
      LOOP
    CASE OPTION = "4"
      CLEAR ALL
      RETURN
    OTHERWISE
      @23,15 SAY "INVALID CHOICE"
    WAIT
  ENDCASE
ENDDO

* END OF MISCELL.PRG
```

```
Structure for database : A:OFMISCEL.dbf
Number of data records :       31
Date of last update    : 12/03/85
Field  Field name  Type       Width    Dec
    1  TITLE       Character     45
    2  DATE        Date           8
    3  KEYWORD     Character     12
    4  HARDCOPY    Character     20
    5  DISKCOPY    Character     20
    6  FILENAME    Character     12
** Total **                     118
```

```
*                            MISTITLE.PRG                 11/15/85

* called from MISCELL.PRG

CLEAR
mtitle ="                                             "
@12,15 SAY "ENTER THE TITLE FOR WHICH YOU ARE SEARCHING:"
@14,15 GET mtitle PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
mtitle = TRIM(mtitle)
CLEAR
SET TALK OFF
LOCATE FOR TITLE = "&mtitle"
IF .NOT. EOF()
  CLEAR
  @4,10 SAY "THE FOLLOWING ITEMS WERE FOUND TITLED &mtitle:"
  @6,0
  DISPLAY OFF ALL DATE,HARDCOPY,KEYWORD FOR TITLE = "&mtitle"
  @23,0
  WAIT
  SET TALK ON
  RETURN
ENDIF
IF EOF()
  CLEAR
  @10,20 SAY "NO ITEMS FOUND THAT WERE TITLED &MTITLE."
  @20,0
  WAIT
  SET TALK ON
  RETURN
ENDIF


* end of MISTITLE.PRG
*                                 MISDATE.PRG

*   CALLED FROM MISCELL.PRG
CLEAR
SET TALK OFF
mdate = "  /  /  "
@10,10 SAY "ENTER THE DATE FOR WHICH YOU ARE SEARCHING:"
@10,53 GET mdate PICTURE "99/99/99"
READ
CLEAR
LOCATE FOR DATE = CTOD("&mdate")
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING ITEMS ARE DATED &mdate:"
  @6,0
  DISPLAY OFF ALL TITLE, HARDCOPY FOR DATE = CTOD("&mdate")
  @23,0
  WAIT
  CLEAR
  SET TALK ON
  LOOP
ENDIF
IF EOF()
  CLEAR
  SET TALK ON
  @10,20 SAY "NO ITEM FOUND DATED &mdate."
  @20,0
  WAIT
  LOOP
ENDIF
*  END OF MISDATE.PRG
*                             MANPOWER.PRG                11/15/85

*CALLED FROM OFMAIN.PRG
USE OFMANPWR.DBF
CLEAR
DO WHILE .T.
SET FORMAT TO OFMANPWR.FMT
option = " "
@23,0
READ
CLOSE FORMAT
DO CASE
   case option = "1"
     DO mantitle
   case option = "2"
     DO mandate
   case option = "3"
     CLEAR ALL
     RETURN
OTHERWISE
   @23,15 SAY "INVALID CHOICE"
ENDCASE
GO TOP
ENDDO
```

```
*                                OFMANPWR.FMT                  11/15/85

@6,10 SAY "YOU MUST KNOW ONE OF THE FOLLOWING IN ORDER TO SEARCH"
@7,10 SAY "THE MANPOWER FILES:"
@10,25 SAY "1. TITLE"
@12,25 SAY "2. DATE"
@14,25 SAY "3. MAIN-MENU"
@18,10 SAY "PLEASE PRESS THE NUMBER OF YOUR CHOICE:"
@18,50 GET OPTION PICTURE "9"
```

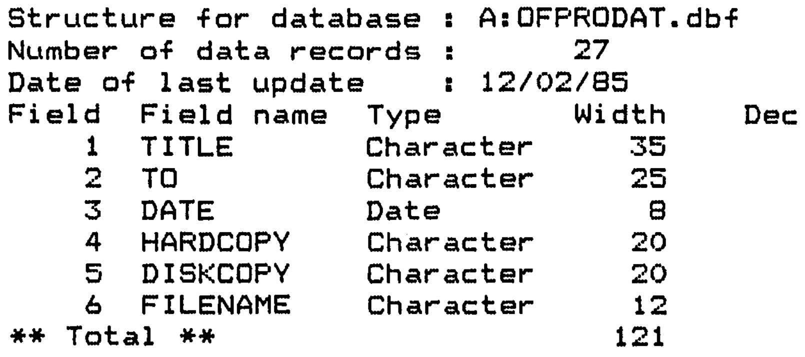
```
Structure for database : A:OFMANPWR.dbf
Number of data records :       16
Date of last update    : 11/14/85
Field  Field name  Type        Width    Dec
    1  TITLE       Character     45
    2  DATE        Date           8
    3  HARDCOPY    Character     20
    4  DISKCOPY    Character     20
    5  FILENAME    Character     12
** Total **                     106
```

```
*                               MANTITLE.PRG             11/15/85

*CALLED FROM MANPOWER.PRG
CLEAR
SET TALK OFF
mtitle = "                                             "
@12,15 SAY "ENTER THE TITLE FOR WHICH YOU ARE SEARCHING:"
@14,15 GET mtitle PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
mtitle = TRIM(mtitle)
LOCATE FOR TITLE = "&MTITLE"
IF .NOT. EOF()
   CLEAR
   @4,10 SAY "THE FOLLOWING ITEMS WERE FOUND TITLED &mtitle:"
   @6,0
   DISPLAY OFF ALL DATE, HARDCOPY FOR TITLE = "&mtitle"
   @23,0
   WAIT
   SET TALK ON
   RETURN
ENDIF
IF EOF()
   CLEAR
   SET TALK ON
   @10,20 SAY "NO ITEM FOUND TITLED &mtitle."
   @22,0
   WAIT
   RETURN
ENDIF

*END OF MANTITLE.PRG
```

```
*                        MANDATE.PRG                11/15/85

* CALLED FORM MANPOWER.PRG
CLEAR
SET TALK OFF
mdate = "  /  /  "
@12,10 SAY "ENTER THE DATE FOR WHICH YOU ARE SEARCHING:"
@12,53 GET mdate PICTURE "99/99/99"
READ
CLEAR
LOCATE FOR DATE = CTOD("&mdate")
IF .NOT. EOF()
   @4,10 SAY "THE FOLLOWING ITEMS ARE DATED &mdate:"
   @6,0
   DISPLAY OFF ALL TITLE, HARDCOPY FOR DATE = CTOD("&mdate")
   @23,0
   WAIT
   CLEAR
   SET TALK ON
   LOOP
ENDIF
IF EOF()
   CLEAR
   SET TALK ON
   @10,20 SAY "NO ITEM FOUND DATED &mdate."
   @22,0
   WAIT
   LOOP
ENDIF

* END OF MANDATE.PRG
*                              POPRTOP.PRG           11/18/85

*   CALLED FROM OFMAIN.PRG

USE ofpoprto
SET TALK OFF
DO WHILE .T.
CLEAR
OPTION = " "
@7,12 SAY "YOU MUST KNOW ONE OF THE FOLLOWING ITEMS IN ORDER"
@8,12 SAY "TO SEARCH THE POP/RTOP FILES:"
@11,25 SAY "1.TITLE"
@13,25 SAY "2.DATE"
@15,25 SAY "3.MAIN MENU"
@18,12 SAY "PLEASE ENTER THE NUMBER OF YOUR CHOICE:"
@18,51 GET OPTION PICTURE "9"
READ
DO CASE
  CASE OPTION = "1"
    CLEAR
    mtitle = "                         "
    @10,20 SAY "ENTER THE TITLE FOR WHICH YOU ARE LOOKING:"
    @12,27 GET mtitle PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
    READ
    mtitle = TRIM(MTITLE)
    LOCATE FOR TITLE = "&MTITLE"
    IF EOF()
      CLEAR
      @10,20 SAY "NO ITEM FOUND TITLED &MTITLE."
      @22,0
      WAIT
      LOOP
    ENDIF
    IF .NOT. EOF()
      CLEAR
      @8,15 SAY "TITLE:" GET TITLE
      @10,15 SAY "DATE:" GET DATE
      @12,15 SAY "FILED IN FOLDER:" GET HARDCOPY
      @14,15 SAY "DISKCOPY;" GET DISKCOPY
      @16,15 SAY "FILENAME:" GET FILENAME
      @22,0
      WAIT
      LOOP
    ENDIF
  CASE OPTION = "2"
    CLEAR
    mdate = "  /  /  "
    @10,20 SAY "ENTER THE DATE FOR WHICH YOU ARE LOOKING:"
    @12,36 GET mdate PICTURE "99/99/99"
    READ
    LOCATE FOR DATE = CTOD("&mdate")
    IF EOF()
      CLEAR
      @10,20 SAY "NO ITEM FOUND DATED &mdate."
      @22,0
      WAIT
      LOOP
    ENDIF
    IF .NOT. EOF()
      CLEAR
      @8,15 SAY "TITLE:" GET TITLE
      @10,15 SAY "DATE:" GET DATE
      @12,15 SAY "FILED IN FOLDER:" GET HARDCOPY
      @14,15 SAY "DISKCOPY:" GET DISKCOPY
      @16,15 SAY "FILENAME:" GET FILENAME
      @22,0
      WAIT
      LOOP
  CASE OPTION = "3"
    CLEAR
    SET TALK ON
    CLEAR ALL
    RETURN
  OTHERWISE
    @23,15 SAY "INVALID CHOICE"
    WAIT
ENDCASE
ENDDO

*  END OF POPRTOP.PRG
```

```
Structure for database : A:OFPOPRTO.dbf
Number of data records :       4
Date of last update    : 11/14/85
Field  Field name  Type        Width    Dec
    1  TITLE       Character      25
    2  DATE        Date            8
    3  HARDCOPY    Character      20
    4  DISKCOPY    Character      20
    5  FILENAME    Character      12
** Total **                       86
```

```
*                           PROJDATA.PRG           11/18/85

* CALLED FROM OFMAIN.PRG
DO WHILE .T.
  CLEAR
  USE OFPRODAT
  SET TALK OFF
  OPTION = " "
  @6,15 SAY "YOU MUST KNOW ONE OF THE FOLLOWING ITEMS IN ORDER"
  @8,15 SAY "TO SEARCH THE PROJECT DATA FILES:"
  @10,30 SAY "1.TITLE"
  @12,30 SAY "2.ADDRESSEE"
  @14,30 SAY "3.DATE"
  @16,30 SAY "4.NONE OF THE ABOVE"
  @18,30 SAY "5.MAIN MENU"
  @20,15 SAY "PLEASE PRESS THE NUMBER OF YOUR CHOICE:"
  @20,55 GET OPTION PICTURE "9"
  READ
  DO CASE
    CASE OPTION = "1"
      DO PROTITLE
    CASE OPTION = "2"
      DO PRODATO
    CASE OPTION = "3"
      DO PRODATE
    CASE OPTION = "4"
      CLEAR
      DISPLAY OFF ALL TITLE,TO,DATE
      WAIT
    CASE OPTION = "5"
      CLEAR ALL
      RETURN
    OTHERWISE
      @23,10 SAY "INVALID CHOICE"
      WAIT
  ENDCASE
ENDDO

* END OF PROJDATA.PRG
```



```
Structure for database : A:OFFRODAT.dbf
Number of data records :       27
Date of last update    : 12/02/85
Field  Field name  Type        Width    Dec
    1  TITLE       Character      35
    2  TO          Character      25
    3  DATE        Date            8
    4  HARDCOPY    Character      20
    5  DISKCOPY    Character      20
    6  FILENAME    Character      12
** Total **                      121
```

```
*                             PROTITLE.PRG           11/26/85

*   CALLED FROM PROJDATA.PRG

CLEAR
SET TALK OFF
mtitle = "                                   "
@10,15 SAY "ENTER THE TITLE FOR WHICH YOU ARE SEARCHING:"
@11,20 GET mtitle PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
CLEAR
SET HEADING OFF
CLEAR
SET TALK OFF
LOCATE FOR TITLE = "&mtitle"
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING ITEMS ARE TITLED &mtitle."
  @6,10 SAY "IF YOU SEE THE ITEM YOU ARE SEEKING REMEMBER ITS RECORD NUMBEF
  SET HEADING OFF
  @9,10 SAY "ADDRESSEE                  DATE"
  @10,0
  DISPLAY ALL TO,DATE FOR TITLE = "&mtitle"
  SET HEADING ON
  @23,0
  WAIT
  CLEAR
  RECON = " "
  @10,20 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
  @11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
  @12,15 SAY "PRESS ANY OTHER KEY TO CONTINUE."
  @12,47 GET RECON PICTURE "!"
  READ
  IF RECON = "Y"
     CLEAR
     mrecno = "   "
     @10,20 SAY "PLEASE ENTER RECORD NUMBER:"
     @10,48 GET mrecno PICTURE "999"
     READ
     CLEAR
     GO VAL(mrecno)
     SET TALK ON
     @8,10 SAY "DATE:"
     @8,21 GET DATE
     @10,10 SAY "TITLE:"
     @10,21 GET TITLE
     @12,10 SAY "ADDRESSEE:"
     @12,21 GET TO
     @14,10 SAY "HARDCOPY:"
     @14,21 GET HARDCOPY
     @16,10 SAY "DISKCOPY:"
     @16,21 GET DISKCOPY
     @18,10 SAY "FILENAME:"
     @18,21 GET FILENAME
     @22,0
     WAIT
     LOOP
   ELSE
```

```
     LOOP
   ENDIF
ENDIF
IF EOF()
   CLEAR
   SET TALK ON
   SET HEADING ON
   @10,20 SAY "NO ITEM FOUND TITLED &mtitle."
   @20,0
   WAIT
   LOOP
ENDIF


*  END OF PROTITLE.PRG
*                              PRODATO.PRG         11/26/85

*   CALLED FROM PROJDATA.PRG

CLEAR
maddr = "                         "
@10,15 SAY "ENTER THE ADDRESSEE FOR WHICH YOU ARE SEARCHING:"
@12,27 GET maddr PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
maddr = TRIM(maddr)
CLEAR
SET HEADING OFF
CLEAR
SET TALK OFF
LOCATE FOR TO = "&maddr"
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING ITEMS WERE ADDRESSED TO &maddr."
  @6,10 SAY "IF YOU SEE THE ITEM YOU ARE SEEKING REMEMBER ITS RECORD NUMBE
  SET HEADING OFF
  @9,15 SAY "TITLE                          DATE"
  @10,0
  DISPLAY ALL TITLE,DATE FOR TO = "&maddr"
  SET HEADING ON
  @23,0
  WAIT
  CLEAR
  RECON = "   "
  @10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
  @11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
  @12,15 SAY "PRESS ANY OTHER KEY TO CONTINUE."
  @12,47 GET RECON PICTURE "!"
  READ
  IF RECON = "Y"
     CLEAR
     mrecno = "   "
     @10,20 SAY "PLEASE ENTER RECORD NUMBER:"
     @10,48 GET mrecno PICTURE "999"
  READ
  CLEAR
  GO VAL(mrecno)
  SET TALK ON
  @8,10 SAY "DATE:"
  @8,21 GET DATE
  @10,10 SAY "TITLE:"
  @10,21 GET TITLE
  @12,10 SAY "ADDRESSEE:"
  @12,21 GET TO
  @14,10 SAY "HARDCOPY:"
  @14,21 GET HARDCOPY
  @16,10 SAY "DISKCOPY:"
  @16,21 GET DISKCOPY
  @18,10 SAY "FILENAME:"
  @18,21 GET FILENAME
  @22,0
  WAIT
  LOOP
ELSE
```

```
  LOOP
  ENDIF
ENDIF
IF EOF()
  CLEAR
  SET TALK ON
  SET HEADING ON
  @10,20 SAY "NO ITEM FOUND ADDRESSED TO &maddr."
  @20,0
  WAIT
  LOOP
ENDIF



*  END OF PRODATO.PRG
*                               PRODATE.PRG             11/27/85

*   CALLED FROM PROJDATA.PRG

CLEAR
SET TALK OFF
mdate = "  /  /  "
@10,10 SAY "ENTER THE DATE FOR WHICH YOU ARE SEARCHING:"
@10,53 GET mdate PICTURE "99/99/99"
READ
CLEAR
SET HEADING OFF
LOCATE FOR DATE = CTOD("&mdate")
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING ITEMS WERE GENERATED ON &mdate"
  @6,10 SAY "IF YOU SEE THE ITEM YOU ARE SEEKING REMEMBER ITS RECORD NUMBE
  @9,10 SAY "ADDRESSEE                    TITLE"
  DISPLAY ALL TO, TITLE FOR DATE = CTOD("&mdate")
  SET HEADING ON
  @23,0
  WAIT
  CLEAR
  recon = " "
  @10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
  @11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
  @12,15 SAY "PRESS ANY OTHER KEY TO CONTINUE."
  @12,47 GET RECON PICTURE "!"
  READ
  IF RECON = "Y"
     CLEAR
     mrecno = "  "
     @10,20 SAY "PLEASE ENTER RECORD NUMBER:"
     @10,48 GET mrecno PICTURE "99"
     READ
     CLEAR
     GO VAL(mrecno)
     SET TALK ON
     @8,10 SAY "DATE:"
     @8,21 GET DATE
     @10,10 SAY "TITLE:"
     @10,21 GET TITLE
     @12,10 SAY "ADDRESSEE:"
     @12,21 GET TO
     @14,10 SAY "HARDCOPY:"
     @14,21 GET HARDCOPY
     @16,10 SAY "DISKCOPY:"
     @16,21 GET DISKCOPY
     @18,10 SAY "FILENAME:"
     @18,21 GET FILENAME
     @22,0
     WAIT
     LOOP
  ELSE
     SET TALK ON
     LOOP
ENDIF
     ENDIF
```

```
IF EOF()
     CLEAR
     SET TALK ON
     @10,20 SAY "NO ITEM FOUND DATED &mdate."
     @20,0
     WAIT
     LOOP
ENDIF

*   END OF PRODATE.PRG
*                         PRESENTN.PRG          11/18/85

* Called from OFMAIN.PRG

USE ofpresnt
CLEAR
@6,30 SAY "PRESENTATION FILES"
@9,10 SAY "IN ADDITION TO THOSE PRESENTATIONS IN WHICH A TASK"
@10,10 SAY "TEAM MEMBER PARTICIPATED, THESE FILES ALSO CONTAIN PRESENTATIOI
@11,10 SAY "GIVEN BY STANFORD OR HQ AT WHICH MSFC WAS NOT PRESENT."
@21,0
WAIT
DO WHILE .T.
CLEAR
option = " "
@6,15 SAY "YOU MUST KNOW ONE OF THE FOLLOWING ITEMS IN ORDER"
@7,15 SAY "TO SEARCH THE PRESENTATION FILES:"
@9,25 SAY "1.SUBJECT"
@11,25 SAY "2.PRESENTED TO"
@13,25 SAY "3.DATE"
@15,25 SAY "4.NONE OF THE ABOVE"
@17,25 SAY "5.MAIN MENU"
@19,15 SAY "PLEASE ENTER THE NUMBER OF YOUR CHOICE:"
@19,55 GET option PICTURE "9"
READ
DO CASE
  CASE OPTION = "1"
    DO PRESUBJ
  CASE OPTION = "2"
    DO PRETO
  CASE OPTION = "3"
    DO PREDATE
  CASE OPTION = "4"
    CLEAR
    DISPLAY OFF ALL SUBJECT,TO,DATE
    WAIT
    LOOP
  CASE OPTION = "5"
    CLEAR ALL
    RETURN
  OTHERWISE
    @23,15 SAY "INVALID CHOICE
  WAIT
ENDCASE
ENDDO

* END OF PRESENTN.PRG
```

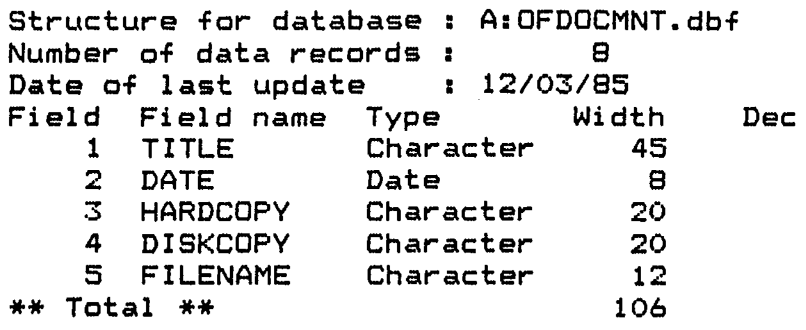
```
Structure for database : A:OFPRESNT.dbf
Number of data records :       13
Date of last update    : 12/03/85
Field  Field name  Type       Width    Dec
    1  INTERNAL    Logical        1
    2  SUBJECT     Character     25
    3  TO          Character     25
    4  GIVEN       Logical        1
    5  DATE        Date           8
    6  HARDCOPY    Character     20
    7  DISKCOPY    Character     20
    8  FILENAME    Character     12
** Total **                     113
```

```
*                            PRESUBJ.PRG                 11/27/85

*   CALLED FROM PRESENTN.PRG

CLEAR
SET TALK OFF
msubj = "                        "
@10,8 SAY "ENTER THE SUBJECT FOR WHICH YOU ARE SEARCHING:"
@10,54 GET msubj PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
msubj = TRIM(msubj)
CLEAR
SET HEADING OFF
LOCATE FOR SUBJECT = "&msubj"
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING PRESENTATIONS ARE TITLED &msubj."
  @6,10 SAY "IF YOU SEE THE ITEM YOU ARE SEEKING REMEMBER ITS RECORD NUMBEF
  @9,10 SAY "GIVEN TO                  DATE"
  @10,0
  DISPLAY ALL TO,DATE FOR SUBJECT = "&msubj"
  SET HEADING ON
  @23,0
  WAIT
  CLEAR
  recon = " "
  @10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
  @11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
  @12,15 SAY "PRESS ANY OTHER KEY TO CONTINUE."
  @12,47 GET RECON PICTURE "!"
  READ
  IF RECON = "Y"
     CLEAR
     mrecno = "  "
     @10,20 SAY "PLEASE ENTER RECORD NUMBER:"
     @10,48 GET mrecno PICTURE "99"
     READ
     CLEAR
     GO VAL(mrecno)
     SET TALK ON
     @6,10 SAY "DATE:"
     @6,21 GET DATE
     @8,10 SAY "SUBJECT:"
     @8,21 GET SUBJECT
     @10,10 SAY "GIVEN TO:"
     @10,21 GET TO
     @12,10 SAY "GIVEN?:"
     @12,21 GET GIVEN
     @14,10 SAY "HARDCOPY:"
     @14,21 GET HARDCOPY
     @16,10 SAY "DISKCOPY:"
     @16,21 GET DISKCOPY
     @18,10 SAY "FILENAME:"
     @18,21 GET FILENAME
     @22,0
     WAIT
     LOOP
ELSE
```

```
  SET TALK ON
  LOOP
ENDIF
  ENDIF
IF EOF()
  CLEAR
  SET TALK ON
  SET HEADING ON
  @10,20 SAY "NO PRESENTATIONS FOUND TITLED &msubj."
  @20,0
  WAIT
  LOOP
ENDIF


*  END OF PRESUBJ.PRG
*                              PRETO.PRG            11/26/85

*   CALLED FROM PRESENTN.PRG

CLEAR
SET TALK OFF
mto = "                         "
@10,15 SAY "ENTER THE NAME OF THE PERSON OR ORGANIZATION TO"
@11,15 SAY "WHICH THE PRESENTATION WAS GIVEN:"
@14,27 GET mto PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
CLEAR
SET HEADING OFF
SET TALK OFF
LOCATE FOR TO = "&mto"
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING PRESENTATIIONS WERE GIVEN TO &mto."
  @6,10 SAY "IF YOU SEE THE ITEM YOU ARE SEEKING REMEMBER ITS RECORD NUMBE
  SET HEADING OFF
  @9,10 SAY "SUBJECT                    DATE"
  @10,0
  DISPLAY ALL SUBJECT,DATE FOR TO = "&mto"
  SET HEADING ON
  @23,0
  WAIT
  CLEAR
  RECON = " "
  @10,20 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
  @11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
  @12,15 SAY "PRESS ANY OTHER KEY TO CONTINUE."
  @12,47 GET RECON PICTURE "!"
  READ
  IF RECON = "Y"
     CLEAR
     mrecno = "   "
     @10,20 SAY "PLEASE ENTER RECORD NUMBER:"
     @10,48 GET mrecno PICTURE "999"
     READ
     CLEAR
     GO VAL(mrecno)
     SET TALK ON
     @6,10 SAY "DATE:"
     @6,21 GET DATE
     @8,10 SAY "SUBJECT:"
     @8,21 GET SUBJECT
     @10,10 SAY "GIVEN TO:"
     @10,21 GET TO
     @12,10 SAY "GIVEN?"
     @12,21 GET GIVEN
     @14,10 SAY "HARDCOPY:"
     @14,21 GET HARDCOPY
     @16,10 SAY "DISKCOPY:"
     @16,21 GET DISKCOPY
     @18,10 SAY "FILENAME:"
     @18,21 GET FILENAME
     @22,0
     WAIT
     LOOP
ELSE
     LOOP
     ENDIF
ENDIF
IF EOF()
   CLEAR
   SET TALK ON
   SET HEADING ON
   @10,20 SAY "NO PRESENTATIONS GIVEN TO &mto THIS YEAR."
   @20,0
   WAIT
   LOOP
ENDIF


*  END OF PRETO.PRG
*                               PREDATE.PRG          11/26/85

*   CALLED FROM PRESNTN.PRG

CLEAR
SET TALK OFF
mdate = "  /  /  "
@10,10 SAY "ENTER THE DATE FOR WHICH YOU ARE SEARCHING:"
@10,53 GET mdate PICTURE "99/99/99"
READ
CLEAR
SET HEADING OFF
LOCATE FOR DATE = CTOD("&mdate")
IF .NOT. EOF()
  @4,10 SAY "THE FOLLOWING ITEMS WERE GENERATED ON &mdate"
  @6,10 SAY "IF YOU SEE THE ITEM YOU ARE SEEKING REMEMBER ITS RECORD NUMI
  @9,10 SAY "SUBJECT                    PRESENTED TO:"
  @23,0
  DISPLAY ALL SUBJECT,TO FOR DATE = CTOD("&mdate")
  SET HEADING ON
  @23,0
  WAIT
  CLEAR
  RECON = " "
  @10,15 SAY "IF YOU SAW THE DATA YOU WERE LOOKING FOR,"
  @11,15 SAY "PLEASE PRESS 'Y' TO CONFIRM.  OTHERWISE,"
  @12,15 SAY "PRESS ANY OTHER KEY TO CONTINUE."
  @12,47 GET RECON PICTURE "!"
  READ
  IF RECON = "Y"
     CLEAR
     mrecno = "   "
     @10,20 SAY "PLEASE ENTER RECORD NUMBER:"
     @10,48 GET mrecno PICTURE "99"
     READ
     CLEAR
     GO VAL(mrecno)
     SET TALK ON
     @6,10 SAY "DATE:"
     @6,21 GET DATE
     @8,10 SAY "SUBJECT:"
     @8,21 GET SUBJECT
     @10,10 SAY "PRESENTED TO:"
     @10,25 GET TO
     @12,10 SAY "GIVEN?:"
     @12,21 GET GIVEN
     @14,10 SAY "HARDCOPY:"
     @14,21 GET HARDCOPY
     @16,10 SAY "DISKCOPY:"
     @16,21 GET DISKCOPY
     @18,10 SAY "FILENAME:"
     @18,21 GET FILENAME
     @22,0
     WAIT
     LOOP
  ELSE
     LOOP
```

```
            ENDIF
       ENDIF
       IF EOF()
          CLEAR
          SET TALK ON
          SET HEADING ON
          @10,20 SAY "NO PRESENTATION FOUND DATED &mdate."
          @20,0
          WAIT
          LOOP
       ENDIF

       *  END OF PREDATE.PRG
```

```
*                                       DOCUMENT.PRG             11/18/85

* CALLED FROM OFMAIN.PRG

USE ofdocmnt.dbf
CLEAR
DO WHILE .T.
CLEAR
PROMPT1 = " "
@5,15 SAY "You must know one of the following items in order"
@7,15 SAY "to search for document information:"
@10,25 SAY "1. TITLE:"
@12,25 SAY "2. DATE:"
@14,25 SAY "3. MAIN MENU:"
@18,15 SAY "PLEASE PRESS THE NUMBER OF YOUR CHOICE:"
@18,55 GET PROMPT1 PICTURE "9"
READ
CLEAR
DO CASE
   case prompt1 = "1"
        DO DOCTITLE
   case prompt1 = "2"
        DO DOCDATE
   case prompt1 = "3"
        CLEAR ALL
        RETURN
 OTHERWISE
     @23,10 SAY "INVALID CHOICE"
 ENDCASE
 ENDDO

 * END OF DOCUMENT.PRG
```



```
Structure for database : A:OFDOCMNT.dbf
Number of data records :       8
Date of last update    : 12/03/85
Field  Field name  Type       Width    Dec
    1  TITLE       Character     45
    2  DATE        Date           8
    3  HARDCOPY    Character     20
    4  DISKCOPY    Character     20
    5  FILENAME    Character     12
** Total **                     106
```

```
*                              DOCTITLE.PRG          11/18/85

*Called from DOCUMENT.PRG

CLEAR
mtitle = "                                             "
@6,10 SAY "ENTER THE TITLE OF THE DOCUMENT FOR WHICH YOU ARE SEARCHING:"
@8,17 GET mtitle PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
READ
mtitle = TRIM(mtitle)
SET TALK OFF
LOCATE FOR TITLE = "&mtitle"
IF EOF()
   CLEAR
   @6,10 SAY "NO DOCUMENT FOUND TITLED &mtitle"
   @22,0
   SET TALK ON
   WAIT
ENDIF
IF .NOT. EOF()
   CLEAR
   @6,10 SAY "TITLE:" GET TITLE
   @8,10 SAY "DATE:" GET DATE
   @10,10 SAY "FILED IN FOLDER:" GET HARDCOPY
   @12,10 SAY "DISKCOPY:" GET DISKCOPY
   @14,10 SAY "FILENAME:" GET FILENAME
   @22,0
   SET TALK ON
   WAIT
ENDIF
RETURN

* END OF DOCTITLE.PRG
*                           DOCDATE.PRG                  11/18/85

*Called from DOCUMENT.PRG

CLEAR
SET TALK OFF
mdate = "  /  /  "
@8,10 SAY "ENTER THE DATE (OF THE DOCUMENT) FOR WHICH YOU ARE SEARCHING:
@10,36 GET mdate PICTURE "99/99/99"
READ
LOCATE FOR DATE = CTOD("&mdate")
IF EOF()
   CLEAR
   @6,10 SAY "NO DOCUMENT FOUND DATED &mdate"
   @22,0
   SET TALK ON
   WAIT
ENDIF
IF .NOT. EOF()
   CLEAR
   @6,10 SAY "TITLE:" GET TITLE
   @8,10 SAY "DATE:" GET DATE
   @10,10 SAY "FILED IN FOLDER:" GET HARDCOPY
   @12,10 SAY "DISKCOPY:" GET DISKCOPY
   @14,10 SAY "FILENAME:" GET FILENAME
   @22,0
   SET TALK ON
   WAIT
ENDIF
RETURN

* END OF DOCDATE.PRG
```

1. REPORT NO.
NASA TM-86550

2. GOVERNMENT ACCESSION NO.

3. RECIPIENT'S CATALOG NO.

4. TITLE AND SUBTITLE

Search and Retrieval of Office Files Using dBASE III

5. REPORT DATE
August 1986

6. PERFORMING ORGANIZATION CODE
PF16

7. AUTHOR(S)
William L. Breazeale and Charlotte R. Talley

8. PERFORMING ORGANIZATION REPORT #

9. PERFORMING ORGANIZATION NAME AND ADDRESS

George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama 35812

10. WORK UNIT NO.
M-534

11. CONTRACT OR GRANT NO.

12. SPONSORING AGENCY NAME AND ADDRESS

National Aeronautics and Space Administration
Washington, D.C. 20546

13. TYPE OF REPORT & PERIOD COVERED

Technical Memorandum

14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

Prepared by Advanced Projects Office.

16. ABSTRACT

This document describes a method of automating the office files retrieval process using a commercially available software package (dBASE III). The resulting product is a menu-driven computer program which requires no computer skills to operate. One part of the document is written for the potential user who has minimal computer experience and uses sample menu "screens" to explain the program; while a second part is oriented towards the computer literate individual and includes rather detailed descriptions of the methodology and search routines. Although much of the programming techniques are explained, this document is not intended to be a tutorial on dBASE III. Hopefully, the document will serve as a stimulus for other applications of dBASE III.

17. KEY WORDS
dBASE III, File Search, Automation,
Menu-Driven, Computer Program,
Database

18. DISTRIBUTION STATEMENT
Unclassified - Unlimited

Subject Category 61

19. SECURITY CLASSIF. (of this report)
Unclassified

20. SECURITY CLASSIF. (of this page)
Unclassified

21. NO. OF PAGES
85

22. PRICE
A05

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

NASA-Langley, 1986

End of Document