

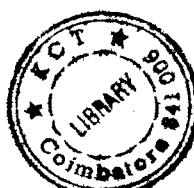


LIB 05 (01)

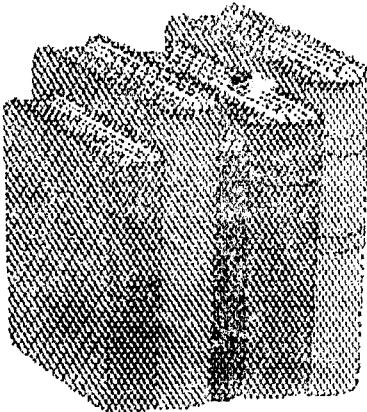
P-1102

Technology.

P-1102



LIBRARY MANAGEMENT SYSTEM



11

Case Study By

Hariharan, S, Jitesh Menon, P & Kannan, N
October 1994

Parashuram

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641 006



Department of

LABORATORY RECORD

Name Roll No.

Class Branch

Certified Bonafide Record of Work Done by

Mr.

Place Staff in Charge

Date

University Register No.

Submitted for the University Practical Examination

Held on

Internal Examiner

External Examiner

CONTENTS

TOPIC	PAGE NUMBER
SYNOPSIS	1
1.0 INTRODUCTION	2
1.1 Introduction about the problem	2
1.2 Need for computing the problem	3
1.3 Problem definition	3
1.4 Solution details	7
2.0 DESIGN OVERVIEW	7
2.1 Introduction	8
2.2 Issues involved in design	9
2.3 Design details	10
3.0 IMPLEMENTATION DETAILS	10
3.1 Introduction	10
3.2 Main modules used	10
3.3 Description of modules	15
3.4 Flow chart	9
CONCLUSION	27
FUTURE SCOPE	28
BIBLIOGRAPHY & SOFTWARE USED	31
APPENDIX	33
a) Source Code	34
b) Example Outputs	72

SYNOPSIS:

Library management is a separate branch of science dealing with the classification of book, journals, papers and their classification. Even today, researchers and librarians continue to work on newer and efficient methods of management. The enormous size in the number of books and the tedious task of maintenance and classification has always been a great challenge.

The task of classification, maintaining a set of databases of the member names, names of books, updating the database....etc. always results in a large number of records and files. Searching through the records and classifying them is not an easy task. Moreover, as the number of books increases the volume of paper material needed to maintain them increases proportionately. Now with the advent of computers, this task has been made all the more easier. Computers work at an extremely faster rate when compared to humans. The amount of paper work and paper material is almost nil.

In this case study, we have basically simplified the process of library management by computerizing the entire system. Simultaneously we have provided new facilities, basic enough to be understood by any librarian who has absolutely no knowledge of computers. To achieve this we adopted Microsoft FoxPro -- a highly advanced database management system. This way, the users have the advantage of saving time, money and space. The result -- an overall increase in the efficiency of library management.

facilities of instantaneous access to databases and answers to simple queries in no more than few seconds.

1.3 Problem definition

The problem could be basically defined as an intricate classification of books based on various fields and maintenance of a database based on various records. The definition could be further attenuated as the simultaneous updating, classification and maintenance of books with an optimal saving in time and physical space.

1.4 Solution details

The solution to the above mentioned problem primarily involves the use of a database management software that contains features necessary for basic programming. Using various databases essentially eases the entire understanding of the problem as a whole. This way updating or increasing the size of the databases will become easier and simpler.

Simple basic algorithms facilitate quicker searching techniques and classification methods. Minimal opening of files should be ensured to save loading and response time. Relational key hierarchy should be used to as far as possible to

link databases, since relational databases are not supported by this version of FoxPro.

This method involves efficient access and improvised enhancement techniques. A more interactive method of user input is intended. As far as possible commands should be avoided and key driven interactive user interfaces created. We assume the real time user has absolutely no knowledge of a computer.

2.0 DESIGN OVERVIEW

2.1 Introduction

An overview of the design involves primarily the creation of databases and modules to maintain these databases. Our basic design involves using extensive pop-up menus supported by FoxPro that provide better user interaction. The rest of the modules contain data entry or data display by the computer. Since, this system was being carried out for our college library, we had to be all the more careful with interface design and dialogs. The librarian who had to use it obviously had no idea of a computer. We have tried to provide on line help for every part of the system.

The usage of a single relational key has been maintained throughout. The use of a relational key facilitates access to other database, to which we need a link. the relational hierarchy of databases is not fully available in FoxPro. Thus updating a single database does not simultaneously update the other.

The system has been designed in such a way that it supports further modification or addition. Extensive usage's of simple codes and a large number of smaller modules facilitate easier understanding of the code. Thus, any further improvement in the code may be carried out with ease.

2.2 Issues involved in design

The design process was not a walk through. We had to go through many hurdles before we could arrive at any particular optimal solution.

For example, in the borrowing module -- the system had to check whether there existed such a member borrowing that book. For this it had to use the member database. Secondly, it had to check whether the book with the given accession number was present. This it did by checking the name database. Thirdly, it checks to see if the book is available for borrowing. Fourthly, it checks whether the member has got a free card and then all the databases are used once again to update after borrowing. This process involves a lot of searching. So, one of the main issue was speed that went sluggish when large databases were used. The similar problem was faced while handling the lending section.

Other problems could involve hardware constraints. The system generally requires a large secondary memory for optimal storage and a sufficiently fast processor for quicker access.

2.3 Design Details

We restricted the number of databases, as far as possible, to the minimum. This way we could reduce access time and understanding was more lucid. Although,

we could not compromise for the large amount of searching done while browsing through the fields of these large databases.

Databases used:

- Member database.
- Book's database.
- Circulation database

Book database – NAME.dbf

NAME	TYPE	WIDTH	DECIMAL	DESCRIPTION
BOOK_NAME	Character	75	0	Name of the book
AUTHOR	Character	30	0	Author Name
VOLUME	Numeric	3	0	Volume of book
EDITION	Character	10	0	Edition of book
SUBJECT	Character	15	0	Subject of book
COPIES	Numeric	2	0	Copies available
T_OR_REF	Character	1	0	Text/Reference Book
BNK_OR_GNR	Character	1	0	Book bank/General
CALL_NO	Character	10	0	Call Number
ACCN_NO	Numeric	10	0	Accession number
STATUS	Character	1	0	Book available
PUBLISHER	Character	30	0	Publisher name
PRICE	Numeric	7	2	Price of book
DATE	Date	8	0	Date Purchased
AGENTS	Character	30	0	Agents for books
BILL_NO	Numeric	6	0	Bill Number

Member database – MEMBER dbf

NAME	TYPE	WIDTH	DECIMAL	DESCRIPTION
NAME	Character	25	0	Name of the member
CARD_NO	Numeric	5	0	Card Number of member
ROLL_NO	Character	7	0	Roll Number of member
F_OR_S	Character	7	0	Staff/Student
MAX_BKS	Numeric	1	0	Maximum Books
AC_BNK	Character	1	0	Access to Book bank

Circulation database – CIRC.dbf

NAME	TYPE	WIDTH	DECIMAL	DESCRIPTION
ROLL_NO	Character	7	0	Roll No. of the member
TOT_BOOK	Numeric	1	0	Total Books allowed
ACCN_NO_1	Numeric	1	0	Accession no. of book 1
ACCN_NO_2	Numeric	1	0	Accession no. of book 2
ACCN_NO_3	Numeric	1	0	Accession no. of book 3
ACCN_NO_4	Numeric	1	0	Accession no. of book 4
ACCN_NO_5	Numeric	1	0	Accession no. of book 5
DUE_DATE_1	Date	8	0	Due date of Book 1
DUE_DATE_2	Date	8	0	Due date of Book 2
DUE_DATE_3	Date	8	0	Due date of Book 3
DUE_DATE_4	Date	8	0	Due date of Book 4
DUE_DATE_5	Date	8	0	Due date of Book 5
FINE	Numeric	5	2	Fine imposed

• *Book Database:*

The book database primarily deals with storage of books and their related fields. The fields are as shown above. Basic fields which are of importance are included.

• *Member Database:*

The member database deals with the storage of member names and their allotment details. The fields are as shown above. Specific details maximum book allowed are of prime importance.

• *Circulation Database:*

The circulation database contains details regarding circulation. It's nothing but a form of book card available in the library.

3.0 IMPLEMENTATION DETAILS

3.1 Introduction

The entire case study has been implemented using the world famous database management software -- Microsoft FoxPro. The FoxPro software was developed by the Microsoft corporation and is basically intended to abet database program and develop custom built database software.

The FoxPro program is a sophisticated piece of a software and has very easy to use interface. It only takes a matter of few minutes to have a basic understanding of the software. However, its the programming capability of FoxPro that makes it an outstanding database management system. A few simple commands form the basic tools for good programming. A primary reason for our choosing FoxPro as the software for our case study was its flexibility and ease of use with which we could produce the desired result.

The minimal hardware requirement is an XT computer with at least 640 K B of memory and a 20-40 Mb hard disk with FoxPro loaded. This configuration is purely a minimal one and is not very effective. Drawbacks could be extremely slow access and less storage. A better and preferable configuration would be an AT 386DX computer, VGA monitor, greater than 120 Mb hard disk, 640 KB base memory, above 2 Mb extended random access memory and a Fox Pro

compiler. A printer is purely optional, though highly recommended for on-line library operations.

3.2 Main Modules used

- Addition Module
- Deletion Module
- Sorting Module
- Help Module
- Book-Return Module
- Lending Module

3.3 Description of Modules

1. Addition Module:

This module is used to enter new books or new members. The user should enter the information of the book for addition of a book. For entering new members, member details should be furnished. This module mainly accesses the member database for addition of members and book database for addition of books.

2. Deletion Module:

The deletion module uses a single particular key for deletion of member or a book. In case of a book the key used is the accession number and for member deletion the Roll Number is taken as the key. Using a single key facilitates both access and maintenance. This module primarily accesses the member database for deletion of members and book database for deletion of books.

3. Sorting Module:

The sorting module involves the sorting of books based on any particular field. For members, it could be on the name or Roll number. For sorting books it could be on the name, author, subject, accession number.. etc. A temporary file is used, to which the sorted new sorted file is written on to. The sorting module accesses member databases for sorting based on member fields. For sorting based on book fields it accesses the book databases.

5. Lending Module:

The lending module maintains the circulation database of the systems. People or members returning books get their names entered into the lending-circulation database. The name, roll number, the account of books, the fine due..etc. get updated automatically. Fine receipts are issued once a fine has been paid. A fine

can be paid immediately or could be summed up and paid towards the end. The lending module accesses all the three databases for cross checking of information. This module and the borrowing module are the longest and most complicated of all the modules. Cross checking of information for verification of validity of both the user details and the book details makes it more difficult to assimilate.

6. Book-Return Module:

The borrowing module also helps in the maintenance of the circulation database. Once books are borrowed by members of the library, their names are entered into their roll cards and an account is kept of the book issued, roll no, number of books and date issued. This module is similar to the lending module. The only difference is that this aids in the return of the book by a member. The process of access of databases is almost the same for both. The book return-module controls the entire database and is very important for full and perfect ordering of the books. Updation and deletion are periodically carried out to ensure perfect ordering of the books.

7. Help Module.

The help module provides on-line help for each of the topics. Detailed help explaining the steps for each process have been illustrated with the aid of pop-up

menus and windows. On-line help facilitates even the most tyro user who has no idea of the software.

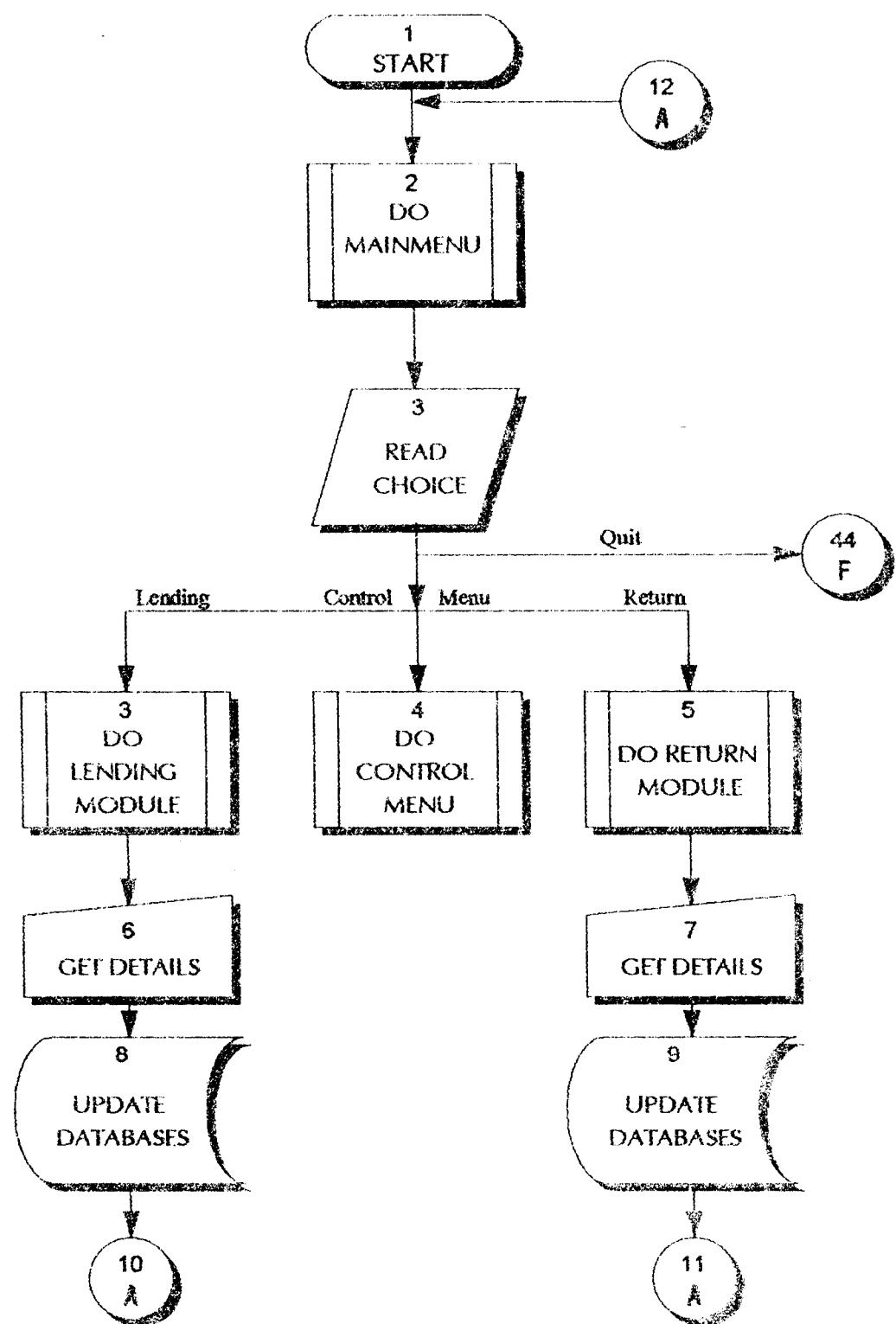


Chart 1

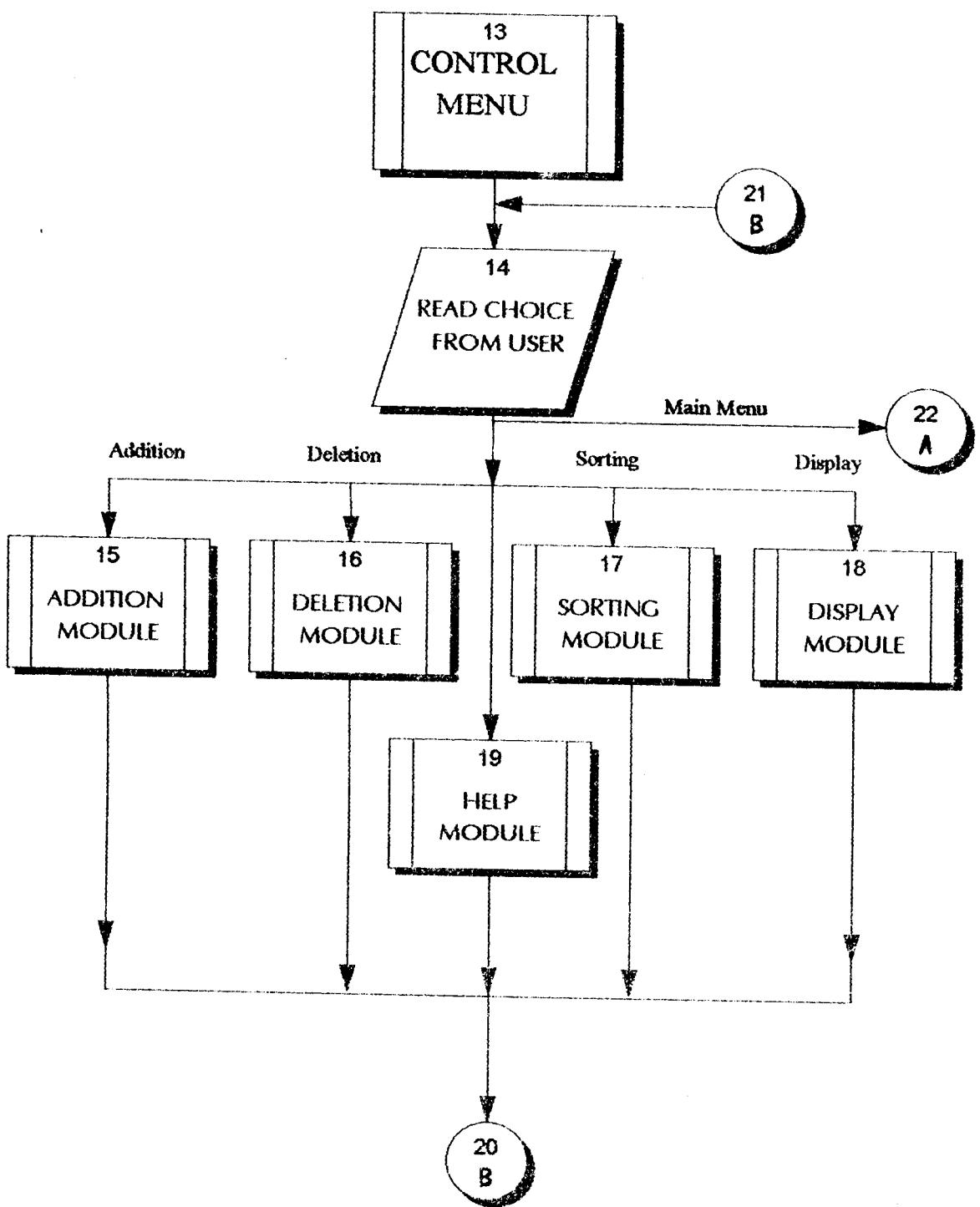


Chart 2

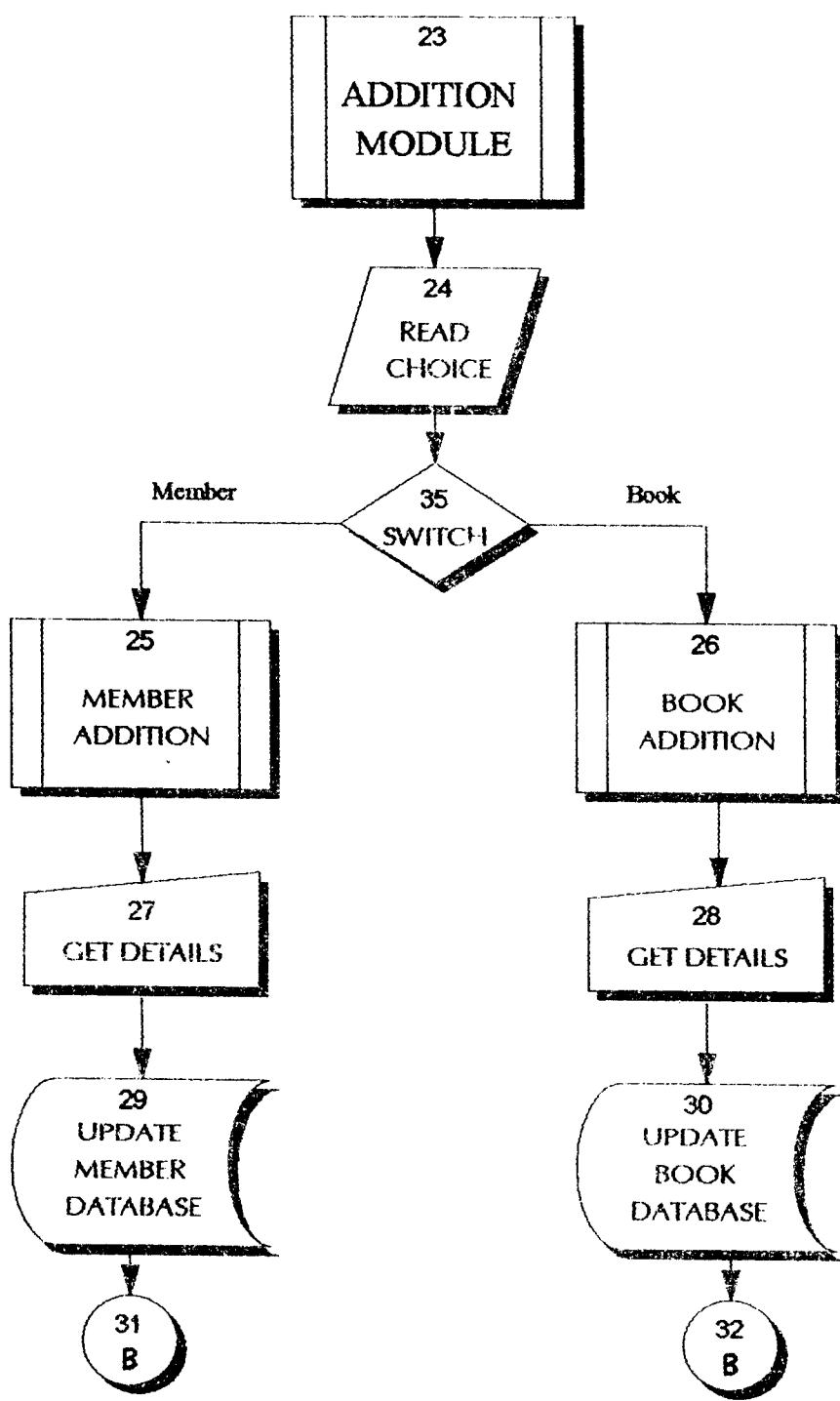


Chart 3

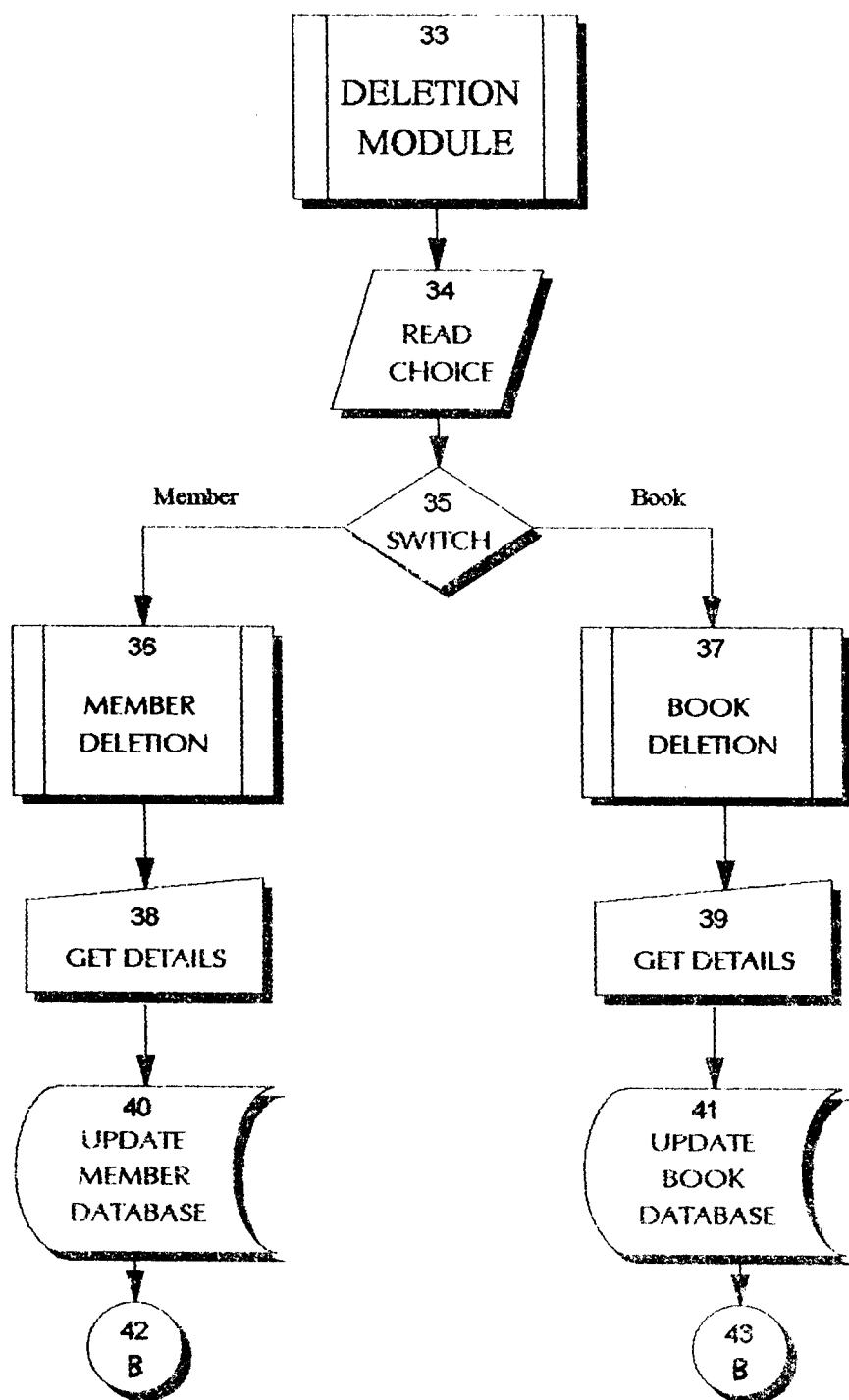


Chart 4

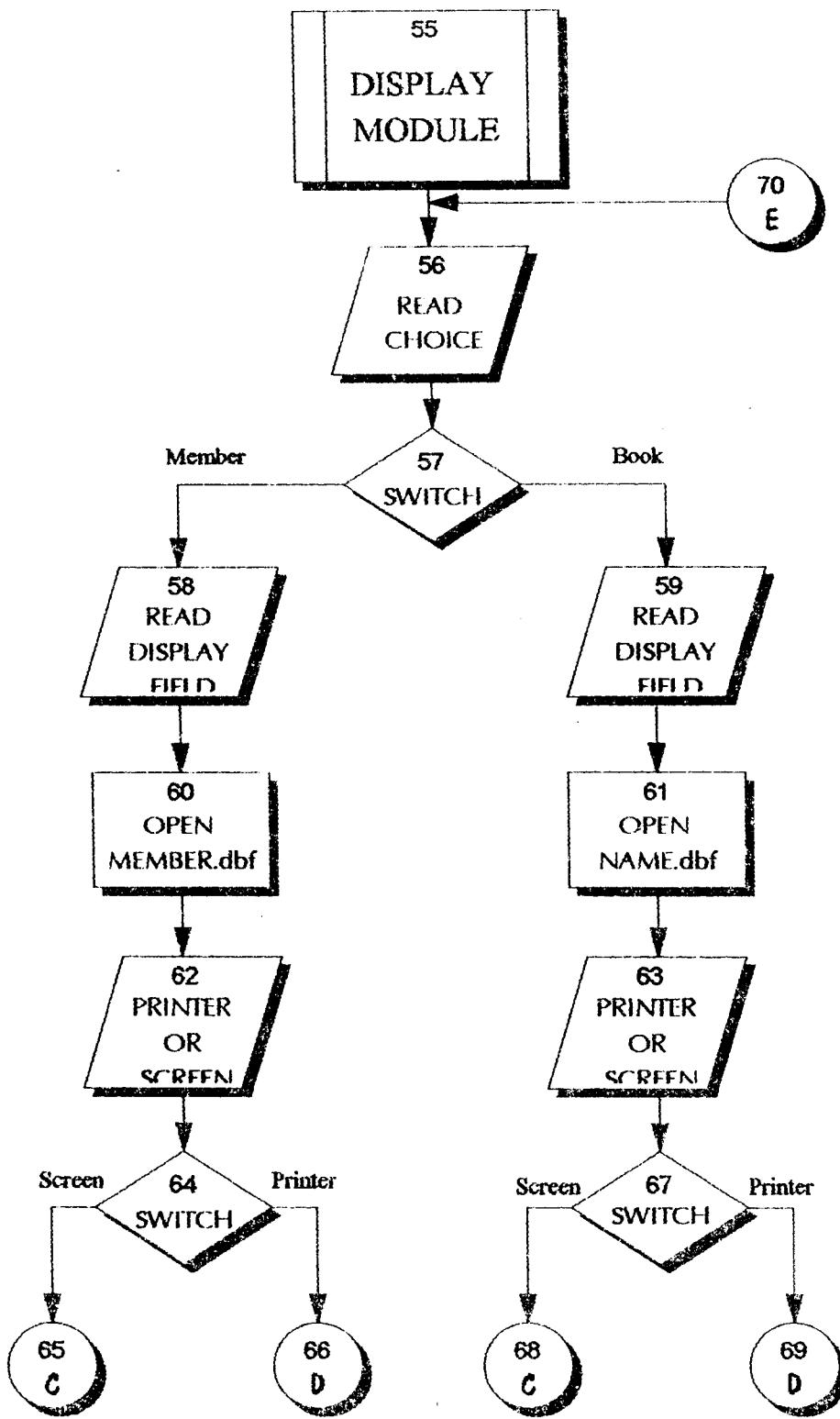


Chart 6

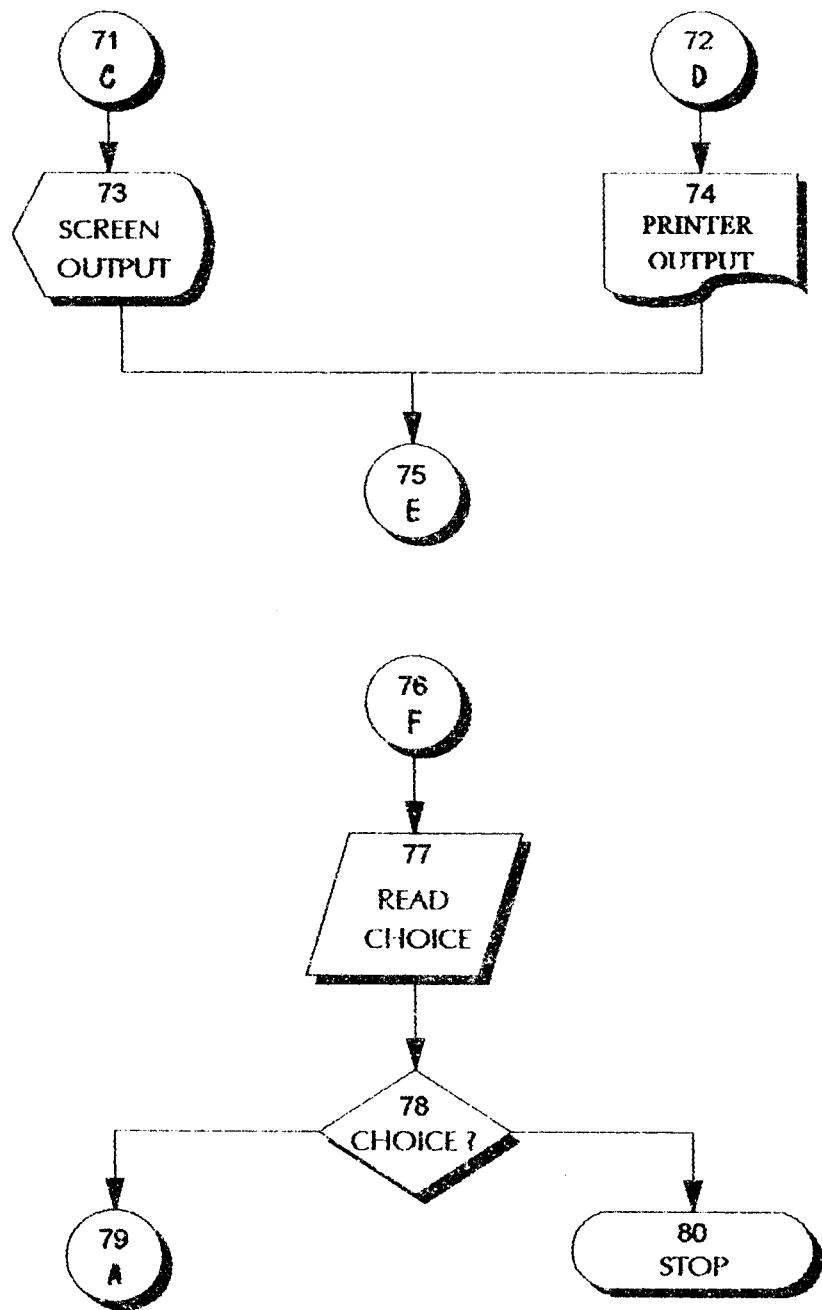


Chart 7

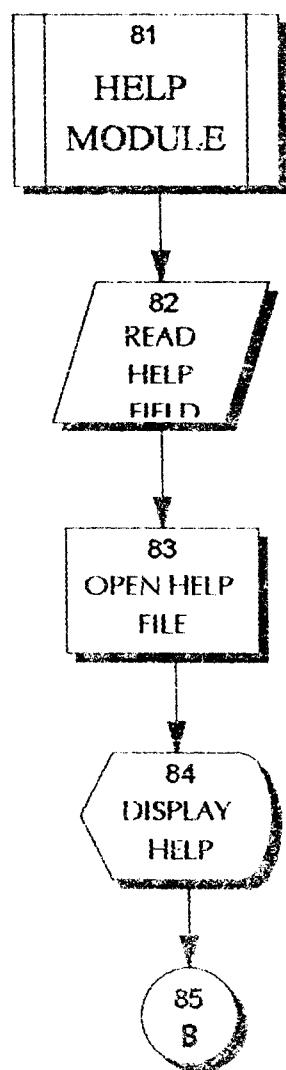


Chart 8

CONCLUSION

We were able to achieve the aim of this case study, which was to create a library management system with all major facilities. The case study was mainly custom designed for our college library. This very reason instigated into us a feeling of inspiration and motivation that was very important for our success. We fully feel that this system would be highly efficient and extremely practical for our library.

Another very simple theory that we could prove was that utilization of computers in library management produced results that had no match with the human work. Jobs like sorting and classification was done in a matter of seconds. These same jobs took hours when done by humans.

The only drawback we faced was speed. As the database grew in size, the access time and secondary storage posed a great problem. The only way out is to have a higher configuration system.

FUTURE SCOPE

SCOPE FOR FURTHER EXPANSION

We feel that the case study could have been even better if we had built it under a graphics environment like windows.

Under a Graphic User Interface a higher level of interactivity was possible with icons which picturized the entire process of understanding. The entire process of appending the database could be beautified. For example, we could store the member photos and their details into the member database. This ensures proper identification. Book covers could be scanned using a scanner and stored in the book database. One need not go into the library to search for a book.

Such a system would be a librarians paradise. Going even further we could have the job of the librarian being done by the computer. This would involve speech and character recognition, an area still under research. But, all this requires a lot of money and a higher configuration system and moreover, it is not practical in a country like ours.

BIBLIOGRAPHY

1. Illustrated Fox Pro by Robert Granillo, BPB publications.
2. Fox Pro in a day by Rayman.
3. The basics of Library Science by Mary Stivens, McGraw Hill.

SOFTWARE USED

1. Microsoft Fox Pro -- Ver 2.0 for LAN.
2. Microsoft Word -- Ver 6.0 for Windows.
3. Micrografx ABC Flow Charter -- Ver 2.0 for Windows.
4. Instant Artist

System: LIBRARY MANAGEMENT SYSTEM
Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N
10/20/94 15:27:10
Database Structure Summary

4 databases in the system

MEMBER.DBF
NAME.DBF
CIRC.DBF
TEMP.DBF (Temporary swap database)

Structure for database : MEMBER.DBF

Number of data records : 13

Last updated : 10/07/94 at 13:01

Field	Field name	Type	Width	Dec	Start	End
1	NAME	Character	25		1	25
2	CARD_NO	Numeric	5		26	30
3	ROLL_NO	Character	7		31	37
4	F_OR_S	Character	1		38	38
5	MAX_BKS	Numeric	1		39	39
6	AC_BNK	Character	1		40	40
* Total **			41			

Used by: BORBOOK.PRG

: RETBOOK.PRG
: CASE_SM2
: ADDMEM.PRG
: LISTMEMN.PRG
: LISTMEMR.PRG
: LISTMEMC.PRG
: DELMEMB.PRG
: DELMEMR.PRG

(procedure in DEFMENU.PRG)

Structure for database : NAME.DBF

Number of data records : 25

Last updated : 10/07/94 at 15:28

Field	Field name	Type	Width	Dec	Start	End
1	BOOK_NAME	Character	75		1	75
2	AUTHOR	Character	30		76	105
3	VOLUME	Numeric	3		106	108
4	EDITION	Character	10		109	118
5	SUBJECT	Character	15		119	133
6	COPIES	Numeric	2		134	135
7	T_OR_REF	Character	1		136	136
8	BNK_OR_GNR	Character	1		137	137
9	CALL_NO	Character	10		138	147
10	ACCN_NO	Numeric	10		148	157
11	STATUS	Character	1		158	158
12	PUBLISHER	Character	30		159	188
13	PRICE	Numeric	7	2	189	195
14	DATE	Date	8		196	203
15	AGENTS	Character	30		204	233
16	BILL_NO	Numeric	6		234	239
** Total **			240			

Used by: BORBOOK.PRG

: CASE_SM1
: ADDBOOK.PRG
: DELBOOK.PRG
: SORTBOOK.PRG
: SORTMEM.PRG
: SERBOOKN.PRG
: SERBOOKA.PRG

(procedure in DEFMENU.PRG)

Structure for database : CIRC.DBF

Number of data records : 5

Last updated : 10/07/94 at 12:57

Field	Field name	Type	Width	Dec	Start	End
1	ROLL_NO	Character	7		1	7
2	TOT_BOOKS	Numeric	1		8	8
3	ACCN_NO_1	Numeric	10		9	18
4	ACCN_NO_2	Numeric	10		19	28
5	ACCN_NO_3	Numeric	10		29	38
6	ACCN_NO_4	Numeric	10		39	48
7	ACCN_NO_5	Numeric	10		49	58
8	DUE_DATE_1	Character	8		59	66
9	DUE_DATE_2	Character	8		67	74
10	DUE_DATE_3	Character	8		75	82
11	DUE_DATE_4	Character	8		83	90
12	DUE_DATE_5	Character	8		91	98
13	FINE	Numeric	5	2	99	103
** Total **			104			

Used by: BORBOOK.PRG

: RETBOOK.PRG

Structure for database : TEMP.DBF

Number of data records : 17

Last updated : 10/07/94 at 13:16

Field	Field name	Type	Width	Dec	Start	End
1	BOOK_NAME	Character	75		1	75
2	AUTHOR	Character	30		76	105
3	VOLUME	Numeric	3		106	108
4	EDITION	Character	10		109	118
5	SUBJECT	Character	15		119	133
6	COPIES	Numeric	2		134	135
7	T_OR_REF	Character	1		136	136
8	BNK_OR_GNR	Character	1		137	137
9	CALL_NO	Character	10		138	147
10	ACCN_NO	Numeric	10		148	157
11	STATUS	Character	1		158	158
12	PUBLISHER	Character	30		159	188
13	PRICE	Numeric	7	2	189	195
14	DATE	Date	8		196	203
15	AGENTS	Character	30		204	233
16	BILL_NO	Numeric	6		234	239
** Total **			240			

System: LIBRARY MANAGEMENT SYSTEM
Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N
10/20/94 15:27:10
Tree Diagram

MAINMENU.PRG

- BORBOOK.PRG
 - MEMBER.DBF (database)
 - NAME.DBF (database)
 - CIRC.DBF (database)
 - CAPSLOCK() (function in FoxPro)
 - UPPER() (function in FoxPro)
 - EOF() (function in FoxPro)
 - DTOC() (function in FoxPro)
 - DATE() (function in FoxPro)
- RETBOOK.PRG
 - MEMBER.DBF (database)
 - NAME.DBF (database)
 - CIRC.DBF (database)
 - CAPSLOCK() ... (function in FoxPro)
 - UPPER() ... (function in FoxPro)
 - EOF() ... (function in FoxPro)
 - DATE() ... (function in FoxPro)
 - CTOD() (function in FoxPro)
 - CHR() (function in FoxPro)
- DEFMENU.PRG
 - MENU() (function in FoxPro)
 - PAD() (function in FoxPro)
 - PROMPT() (function in FoxPro)
 - BOOKPROC (procedure in DEFMENU.PRG)
 - ADDBOOK.PRG
 - NAME.DBF (database)
 - CAPSLOCK() ... (function in FoxPro)
 - REPLICATE() (function in FoxPro)
 - CHR() ... (function in FoxPro)
 - DELBOOK.PRG
 - NAME.DBF (database)
 - FOUND() (function in FoxPro)
 - SUB_FOR_SER (procedure in DEFMENU.PRG)
 - PROMPT() ... (function in FoxPro)
 - CASE_SM1 (procedure in DEFMENU.PRG)
 - NAME.DBF (database)
 - TEMP.DBF (database)
 - SERBOOKN.PRG
 - NAME.DBF (database)
 - UPPER() ... (function in FoxPro)
 - EOF() ... (function in FoxPro)
 - CHR() ... (function in FoxPro)
 - SERBOOKA.PRG
 - NAME.DBF (database)
 - UPPER() ... (function in FoxPro)
 - EOF() ... (function in FoxPro)
 - CHR() ... (function in FoxPro)
 - SORTBOOK.PRG
 - NAME.DBF (database)
 - PROMPT() ... (function in FoxPro)
 - CASE_SM1... (procedure in DEFMENU.PRG)
 - MEMPROC (procedure in DEFMENU.PRG)
 - ADDMEM.PRG

```
    MEMBER.DBF (database)
    CAPSLOCK() ... (function in FoxPro)
    CHR() ... (function in FoxPro)
  SUB_FOR_DEL (procedure in DEFMENU.PRG)
    PROMPT() ... (function in FoxPro)
    CASE_SM3 (procedure in DEFMENU.PRG)
      DELMEMB.PRG
        MEMBER.DBF (database)
        UPPER() ... (function in FoxPro)
        FOUND() ... (function in FoxPro)
      DELMEMR.PRG
        MEMBER.DBF (database)
        FOUND() ... (function in FoxPro)
  SUB_FOR_LIST (procedure in DEFMENU.PRG)
    PROMPT() ... (function in FoxPro)
    CASE_SM2 (procedure in DEFMENU.PRG)
      MEMBER.DBF (database)
      TEMP1.DBF (database)
      TEM1P1.DBF (database)
      LISTMEMN.PRG
        MEMBER.DBF (database)
        UPPER() ... (function in FoxPro)
        EOF() ... (function in FoxPro)
        CHR() ... (function in FoxPro)
      LISTMEMR.PRG
        MEMBER.DBF (database)
        UPPER() ... (function in FoxPro)
        EOF() ... (function in FoxPro)
        CHR() ... (function in FoxPro)
      LISTMEMC.PRG
        MEMBER.DBF (database)
        EOF() ... (function in FoxPro)
        CHR() ... (function in FoxPro)
  SORTMEM.PRG
    NAME.DBF (database)
    PROMPT() ... (function in FoxPro)
    CASE_SM1... (procedure in DEFMENU.PRG)
  HELPPROC (procedure in DEFMENU.PRG)
  STOPROC (procedure in DEFMENU.PRG)
```

10/20/94
15:51

BORBOOK.PRG
Copyright, ANONYMOUS, 1994
LIBRARY MANAGEMENT SYSTEM



Page 1

*: Program: BORBOOK.PRG
*: System: LIBRARY MANAGEMENT SYSTEM
*: Author: HARIHARAN. S, JITESH MENON. P, KANNAN.
N
*: Copyright (c) 1994, ANONYMOUS
*: Last modified: 10/20/94 14:30
*: Called by: MAINMENU.PRG
*: Calls: CAPSLOCK() (function in ?)
*: : UPPER() (function in ?)
*: : EOF() (function in ?)
*: : DTOC() (function in ?)
*: : DATE() (function in ?)
*: Uses: MEMBER.DBF
*: : NAME.DBF
*: : CIRC.DBF
*: Documented 10/20/94 at 14:45 FoxDoc
version 2.10e

*BORBOOK.PRG

*Module for borrowing a book

SET TALK OFF

CAPSTATUS = CAPSLOCK(.T.)

DEFINE WINDOW SERWIN1 FROM 03,05 TO 15, 60 SHADOW TITLE "B
orrow a book"

ACTIVATE WINDOW SERWIN1

*DO WHILE .T.

*

*Searches for the Roll Number in the member database

CLEAR

@ 0,0 SAY 'Enter the Roll No. of the student or press Q to
quit'

ACCEPT '>' TO M.ROL

CLEAR

```

*Searches for the Roll Number in the member database
CLEAR
@ 0,0 SAY 'Enter the Roll No. of the student or press Q to quit'
ACCEPT '>' TO M_ROL
CLEAR
- IF UPPER(M_ROL) = 'Q'
    CLEAR
    DEACTIVATE WINDOW SERWIN1
    RETURN
- ENDIF

    USE MEMBER

    LOCATE FOR UPPER(ROLL_NO) = UPPER(M_ROL)
    IF EOF()
        ? "Cannot find student roll number ", M_ROL
        WAIT
        DEACTIVATE WINDOW SERWIN1
    RETURN
- ENDIF
    STORE MAX_BKS TO MAX
    *
    *
*Searches for Acctn. No in the book database

    @ 1,0 SAY 'Enter the Acctn. No. of the book or press 0 to quit'
    INPUT '>' TO MNO
    CLEAR
    - IF MNO = 0
        CLEAR
        DEACTIVATE WINDOW SERWIN1
    RETURN
- ENDIF

    USE NAME

    LOCATE FOR ACCTN_NO = MNO
    IF EOF()
        ? "Cannot find given Acctn. No. of the book", MNO
        WAIT
        DEACTIVATE WINDOW SERWIN1
    RETURN
- ENDIF

    - IF UPPER(STATUS) = 'N'
        ? "Book has already been borrowed."
        WAIT
        DEACTIVATE WINDOW SERWIN1
    RETURN
- ENDIF
    REPLACE STATUS WITH 'N'
    *
    *

*Searches the circulation database to check whether there is an
*existing record for this roll number

    USE CIRC

    LOCATE FOR UPPER(ROLL_NO) = UPPER(M_ROL)

```

```
--IF EOF()
  GO BOTTOM
  APPEND BLANK
ENDIF
*-
*-
REPLACE ROLL_NO WITH M.ROL.
IF TOT_BOOKS < MAX
  DO CASE
    --CASE ACCN_NO_1 = 0
      REPLACE DUE_DATE_1 WITH DTOC(DATE() + 15)
      REPLACE ACCN_NO_1 WITH MNO
    --CASE ACCN_NO_2 = 0
      REPLACE DUE_DATE_2 WITH DTOC(DATE() + 15)
      REPLACE ACCN_NO_2 WITH MNO
    --CASE ACCN_NO_3 = 0
      REPLACE DUE_DATE_3 WITH DTOC(DATE() + 15)
      REPLACE ACCN_NO_3 WITH MNO
    --CASE ACCN_NO_4 = 0
      REPLACE DUE_DATE_4 WITH DTOC(DATE() + 15)
      REPLACE ACCN_NO_4 WITH MNO
    --CASE ACCN_NO_5 = 0
      REPLACE DUE_DATE_5 WITH DTOC(DATE() + 15)
      REPLACE ACCN_NO_5 WITH MNO
  ENDCASE
  TOT = TOT_BOOKS + 1
  REPLACE TOT_BOOKS WITH TOT
--ELSE
  CLEAR
  @ 0,1 SAY 'Roll number '+M.ROL+' has reached the maximum limit'
  @ 1,1 SAY 'His/Her maximum limit is '
  @ 1,26 SAY MAX
  WAIT
ENDIF
DEACTIVATE WINDOW SERWIN1
CAPSTATUS = CAPSLOCK(.T.)
SET TALK ON
SET HEADING ON
CLOSE DATABASES
CLEAR
RETURN
```

*: EOF: BORBOOK.PRG

```

1  ****
2  *:
3  *:          Program:           DELMEMB.PRG
4  *:
5  *:          System: LIBRARY MANAGEMENT SYSTEM
6  *:          Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N
7  *:          Copyright (c) 1994, ANONYMOUS
8  *:          Last modified: 10/20/94    14:58
9  *:
10 *:         Called by: CASE_SM3          (procedure in DEFMENU.PRG)
11 *:
12 *:         Calls: UPPER()          (function in ?)
13 *:                  : FOUND()          (function in ?)
14 *:
15 *:         Uses: MEMBER.DBF,
16 *:
17 *:         Documented 10/20/94 at 15:27          FoxDoc version 2.10e
18 *: ****
19
20      SET TALK OFF
21      USE MEMBER
22      DEFINE WINDOW WIN1 FROM 03,05 TO 22,70 SHADOW TITLE "Deletion of a me
mber"
23      ACTIVATE WINDOW WIN1
24      CLEAR
25      INPUT 'How many members do you want to delete : ' TO MEMNO
26      CLEAR
27      FOR I=1 TO MEMNO
28          CLEAR
29          ? 'Member', I
30          ? '_____'
31          ?
32          INPUT "Enter the member name (exactly as entered) : " TO M.MNAME
33          ACCEPT "Are you sure (Y/N) ? " TO M.CHoice
34          ?
35          IF LOWER (M.CHoice) = 'y'
36              LOCATE FOR UPPER(NAME) = UPPER(M.MNAME)
37              IF FOUND() = .F.
38                  ? "Cannot find member with given name", M.ROLL
39                  ? "Record Not deleted, Try again !!""
40                  WAIT
41              ?
42          EXIT
43          ELSE
44              DELETE FOR UPPER(NAME) = UPPER(M.MNAME)
45              ? "Specified member record deleted."
46              ?
47              WAIT
48          ENDIF
49      ENDIF
50      IF LOWER (M.CHoice) = 'n'
51          ? "Record has not been deleted"
52          ?
53          WAIT
54      ENDIF
55  ENDFOR
56  PACK
57  DEACTIVATE WINDOW WIN1

```

```
58     CLOSE DATABASES
59     RETURN
60
61     *: EOF: DELMEMB.ACT
```

```
1  ****
2  *:
3  *:
4  *:      Program: DELMEMR.PRG
5  *:
6  *:      System: LIBRARY MANAGEMENT SYSTEM
7  *:      Author: HARIHARAN, S., JITESH MENON, P., KANNAN, N
8  *:      Copyright (c) 1994, ANONYMOUS
9  *:      Last modified: 10/20/94    14:58
10 *:      Called by: CASE_SM3          (procedure in DEFMEMO.PRG)
11 *:
12 *:      Calls: FOUND()           (function in ?)
13 *:
14 *:      Uses: MEMBER.DBF
15 *:
16 *:      Documented 10/20/94 at 15:27
17 *:***** FoxDoc version 2.10
*****  
****  
18
19     SET TALK OFF
20     USE MEMBER
21     DEFINE WINDOW WIN1 FROM 03,05 TO 22,70 SHADOW TITLE "Deletion of a
mber"
22     ACTIVATE WINDOW WIN1
23     CLEAR
24     INPUT 'How many members do you want to delete : ' TO MEMNO
25     CLEAR
26     FOR I=1 TO MEMNO
27         CLEAR
28         ? 'Member', I
29         ? '-----'
30         ?
31         INPUT "Enter the Roll Number for the book : " TO M.ROLL
32         ACCEPT "Are you sure (Y/N) ? " TO M.CHoice
33         ?
34         IF LOWER (M.CHoice) = 'y'
35             LOCATE FOR ROLL_NO = M.ROLL
36             IF FOUND() = .F.
37                 ? "Cannot find member with given Roll number ", M.ROLL
38                 ? "Record Not deleted, Try again !!"
39                 WAIT
40                 ?
41             V--EXIT
42             ELSE
43                 DELETE FOR ROLL_NO = M.ROLL
44                 ? "Specified member record deleted."
45             ?
```

```

98      ON SELECTION POPUP POPQUIT DO STOPROC
99
100     *Display and activate the menu bar
101     ACTIVATE MENU MAINMENU
102     SET TALK ON
103
104
105     *
118   ****
119   *!
120   *!      Procedure: BOOKPROC
121   *!
122   *!      Called by: DEFMENU.PRG
123   *!
124   *!          Calls: ADDBOOK.PRG
125   *!                  : DELBOOK.PRG
126   *!                  : SUB_FOR_SER      (procedure in DEFMENU.PRG)
127   *!                  : SORTBOOK.PRG
128   *!
129   ****
*****  

118   PROCEDURE BOOKPROC
119   *Display the names of the pad and the menu
120   PARAMETER MMENU, MPAD, MPROMPT
121
122   DO CASE
123   CASE MPROMPT = 'Add a book'
124       DO ADDBOOK
125   CASE MPROMPT = 'Delete a book'
126       DO DELBOOK
127   CASE MPROMPT = 'Search for a book'
128       DO SUB_FOR_SER
129   CASE MPROMPT = 'Sort books'
130       DO SORTBOOK
131   ENDCASE
132   RETURN
133   *
134
135   *
148   ****
*****  

149   *!
150   *!      Procedure: MEMPROC
151   *!
152   *!      Called by: DEFMENU.PRG
153   *!
154   *!          Calls: ADDMEM.PRG
155   *!                  : SUB_FOR_DEL      (procedure in DEFMENU.PRG)
156   *!                  : SUB_FOR_LIST      (procedure in DEFMENU.PRG)
157   *!                  : SORTMEM.PRG
158   *!
159   ****
*****  

148   PROCEDURE MEMPROC
149   *Display the names of the pad and the menu
150   PARAMETER MMENU, MPAD, MPROMPT
151
152   DO CASE
153   CASE MPROMPT = 'Add a member'
154       DO ADDMEM
155   CASE MPROMPT = 'Delete a member'
156       DO SUB_FOR_DEL

```

```

157      CASE MPROMPT = 'List members'
158          DO SUB_FOR_LIST
159      CASE MPROMPT = 'Sort members'
160          DO SORTMEM
161      ENDCASE
162      RETURN
163      *
164      *
165      *
166      ****
167      ****
168      ****
169      ****
170      ****
171      ****
172      ****
173      ****
174      *!
175      *! Procedure: HELPPROC
176      *!
177      *! Called by: DEFMENU.PRG
178      *!
179      ****
180      ****
181      ****
182      ****
183      ****
184      ****
185      ****
186      ****
187      ****
188      ****
189      ****
190      ****
191      ****
192      ****
193      ****
194      ****
195      ****
196      ****
197      ****
198      ****
199      ****
200      ****
201      ****
202      ****
203      ****
204      ****
205      ****
206      ****
207      ****
208      ****
209      ****
210      ****
211      ****

```

173 PROCEDURE HELPPROC
 *Display the names of the pad and the menu
 PARAMETER MMENU, MPAD, MPROMPT

177 DO CASE

178 CASE MPROMPT = 'Lending'
 DEFINE WINDOW HELP FROM 10,00 TO 15, 60 CLOSE
 SHADOW COLOR SCHEME 1 TITLE "Help - Lending" DOUBLE
 ACTIVATE WINDOW HELP
 @ 0,0 SAY 'Use this command for lending a book to a member.'
 @ 1,0 SAY 'The inputs it requires are the the Roll No. and Accn.'

184 WAIT
 DEACTIVATE WINDOW HELP

185 CASE MPROMPT = 'Receiving'
 DEFINE WINDOW HELP FROM 10,00 TO 15, 60 CLOSE
 SHADOW COLOR SCHEME 1 TITLE "Help - Receiving" DOUBLE
 ACTIVATE WINDOW HELP
 @ 0,0 SAY 'Use this command for receiving a book from a member.'
 @ 1,0 SAY 'The inputs it requires are the the Roll No. and Accn.'

192 WAIT
 DEACTIVATE WINDOW HELP

193 CASE MPROMPT = 'Members'
 DEFINE WINDOW HELP FROM 10,00 TO 15, 60 CLOSE
 SHADOW COLOR SCHEME 1 TITLE "Help - Members" DOUBLE
 ACTIVATE WINDOW HELP
 @ 0,0 SAY 'Members are the people who have rights to borrow'
 @ 1,0 SAY 'or use the library and its facilities'
 WAIT

201 DEACTIVATE WINDOW HELP

202 CASE MPROMPT = 'Books'
 DEFINE WINDOW HELP FROM 10,00 TO 15, 60 CLOSE
 SHADOW COLOR SCHEME 1 TITLE "Help - Books" DOUBLE
 ACTIVATE WINDOW HELP
 @ 0,0 SAY 'Books are the entities of any Book Library'
 @ 1,0 SAY 'These entities can either be borrowed or returned.'
 WAIT

209 DEACTIVATE WINDOW HELP

210 CASE MPROMPT = 'Reservation'
 DEFINE WINDOW HELP FROM 10,00 TO 15, 60 CLOSE

212 SHADOW COLOR SCHEME 1 TITLE "Help - Reservation" DOUBLE
213 ACTIVATE WINDOW HELP
214 @ 0,0 SAY 'Use this command for reserving a book.'
215 @ 1,0 SAY 'A member may reserve a book for future borrowing.'
216 WAIT
217 DEACTIVATE WINDOW HELP
218 --CASE MPROMPT = 'Sorting'
219 DEFINE WINDOW HELP FROM 10,00 TO 15, 60 CLOSE
220 SHADOW COLOR SCHEME 1 TITLE "Help - Sorting" DOUBLE
221 ACTIVATE WINDOW HELP
222 @ 0,0 SAY 'Use this command for sorting based on the given keys.'
223 @ 1,0 SAY 'Keys could be Accn. No., Roll No., Member Name .. etc'
224 WAIT
225 DEACTIVATE WINDOW HELP
226 --CASE MPROMPT = 'Addition'
227 DEFINE WINDOW HELP FROM 10,00 TO 15, 60 CLOSE SHADOW
228 COLOR SCHEME 1 TITLE "Help - Addition" DOUBLE
229 ACTIVATE WINDOW HELP
230 @ 0,0 SAY 'Use this command for Adding a member or book'
231 @ 1,0 SAY 'The inputs it requires are simple to understand.'
232 WAIT
233 DEACTIVATE WINDOW HELP
234 --CASE MPROMPT = 'Deletion'
235 DEFINE WINDOW HELP FROM 10,00 TO 15, 60 CLOSE SHADOW
236 COLOR SCHEME 1 TITLE "Help - Deletion" DOUBLE
237 ACTIVATE WINDOW HELP
238 @ 0,0 SAY 'Use this command for Deleting a book or member.'
239 @ 1,0 SAY 'The inputs it requires are self understandable'
240 WAIT
241 DEACTIVATE WINDOW HELP
242 --CASE MPROMPT = 'Searching'
243 DEFINE WINDOW HELP FROM 10,00 TO 15, 60 CLOSE
244 SHADOW COLOR SCHEME 1 TITLE "Help - Searching" DOUBLE
245 ACTIVATE WINDOW HELP
246 @ 0,0 SAY 'Use this command for Searching a book or member.'
247 @ 1,0 SAY 'The inputs it requires are variou keys.'
248 WAIT
249 DEACTIVATE WINDOW HELP
250 --ENDCASE
251 RETURN
252 *-----
253 *-----
254 *-----
255 ****
256 ****

263 *!
264 *! Procedure: STOPROC
265 *!
266 *! Called by: DEFMENU.PRG
267 *!
268 ****

262 PROCEDURE STOPROC
263 *Return to the FOX pro command window
264 SET CLOCK OFF
265 CLEAR
266 DEACTIVATE WINDOW MAIN
267 DEACTIVATE POPUP POPBOOK
268 DEACTIVATE POPUP POPCIR
269 DEACTIVATE POPUP POPHELP
270 DEACTIVATE POPUP POPQUIT

```

271     RETURN
272     *
273
274     *
285 ***!*****!
286 *!
287 *!      Procedure: SUB_FOR_SER
288 *!
289 *!      Called by: BOOKPROC          (procedure in DEFMENU.PRG)
290 *!
291 *!          Calls: PROMPT()        (function in ?)
292 *!                  : CASE_SM1      (procedure in DEFMENU.PRG)
293 *!
294 ***!*****!
285 PROCEDURE SUB_FOR_SER
286 *Procedure for sub menu creation for searching a book
287 DEFINE POPUP SUB_MENU_1 FROM 6,15 COLOR SCHEME 2 TITLE "Search" SHAD
W
288
289     DEFINE BAR 1 OF SUB_MENU_1 PROMPT "Search using \<Book name"
290     DEFINE BAR 2 OF SUB_MENU_1 PROMPT "Search using \<Author"
291     ON SELECTION POPUP SUB_MENU_1 DO CASE_SM1 WITH PROMPT()
292     ACTIVATE POPUP SUB_MENU_1
293     RETURN
294
295
311 ***!*****!
312 *!
313 *!      Procedure: CASE_SM1
314 *!
315 *!      Called by: SUB_FOR_SER      (procedure in DEFMENU.PRG)
316 *!                  : SORTBOOK.PRG
317 *!                  : SORTMEM.PRG
318 *!
319 *!          Calls: SERBOOKN.PRG
320 *!                  : SERBOOKA.PRG
321 *!
322 *!          Uses: NAME.DBF
323 *!                  : TEMP.DBF
324 *!
325 ***!*****!
311 PROCEDURE CASE_SM1
312 *Procedure for case structure for submenu
313 PARAMETER MPROMPT
314
315     DO CASE
316         CASE MPROMPT = 'Search using Book name'
317             DO SERBOOKN
318         CASE MPROMPT = 'Search using Author'
319             DO SERBOOKA
320             *CASE mprompt = 'Search for a book'
321         ENDCASE
322     RETURN
323     *

```

```

271      RETURN
272      *
273
274      *
275      ****!
276      *!
277      *!      Procedure: SUB_FOR_SER
278      *!
279      *!      Called by: BOOKPROC          (procedure in DEFMENU.PRG)
280      *!
281      *!      Calls: PROMPT()           (function in ?)
282      *!                  : CASE_SM1        (procedure in DEFMENU.PRG)
283      *!
284      *!*****
285      PROCEDURE SUB_FOR_SER
286      *Procedure for sub menu creation for searching a book
287      DEFINE POPUP SUB_MENU_1 FROM 6,15 COLOR SCHEME 2 TITLE "Search" SHAD
W
288
289      DEFINE BAR 1 OF SUB_MENU_1 PROMPT "Search using \<Book name"
290      DEFINE BAR 2 OF SUB_MENU_1 PROMPT "Search using \<Author"
291      ON SELECTION POPUP SUB_MENU_1 DO CASE_SM1 WITH PROMPT()
292      ACTIVATE POPUP SUB_MENU_1
293      RETURN
294
295
296      ****!
297      *!*****
298      *!
299      *!      Procedure: CASE_SM1
300      *!
301      *!      Called by: SUB_FOR_SER        (procedure in DEFMENU.PRG)
302      *!                  : SORTBOOK.PRG
303      *!                  : SORTMEM.PRG
304      *!
305      *!      Calls: SERBOOKN.PRG
306      *!                  : SERBOOKA.PRG
307      *!
308      *!      Uses: NAME.DBF
309      *!                  : TEMP.DBF
310      *!
311      *!*****
312      PROCEDURE CASE_SM1
313      *Procedure for case structure for submenu
314      PARAMETER MPROMPT
315
316      DO CASE
317      CASE MPROMPT = 'Search using Book name'
318          DO SERBOOKN
319      CASE MPROMPT = 'Search using Author'
320          DO SERBOOKA
321          *CASE mprompt = 'Search for a book'
322      ENDCASE
323      RETURN
324      *

```

```

324
325      *
326      *!*****  

327      *!
328      *!      Procedure: SUB_FOR_LIST
329      *!
330      *!      Called by: MEMPROC          (procedure in DEFMENU.PRG)
331      *!
332      *!      Calls: PROMPT()           (function in ?)
333      *!              : CASE_SM2        (procedure in DEFMENU.PRG)
334      *!
335      *!*****  

336      PROCEDURE SUB_FOR_LIST
337      *Procedure for sub menu creation for searching a book
338      DEFINE POPUP SUB_MENU_2 FROM 6,15 COLOR SCHEME 2 TITLE "List" SHADOW
339
340      DEFINE BAR 1 OF SUB_MENU_2 PROMPT "List by \<Name"
341      DEFINE BAR 2 OF SUB_MENU_2 PROMPT "List by \<Roll-No"
342      DEFINE BAR 3 OF SUB_MENU_2 PROMPT "List by \<Card-No"
343      DEFINE BAR 4 OF SUB_MENU_2 PROMPT "\<Display database"
344      ON SELECTION POPUP SUB_MENU_2 DO CASE_SM2 WITH PROMPT()
345      ACTIVATE POPUP SUB_MENU_2
346      RETURN
347
348
349      *!*****  

350      *!
351      *!      Procedure: CASE_SM2
352      *!
353      *!      Called by: SUB_FOR_LIST      (procedure in DEFMENU.PRG)
354      *!
355      *!      Calls: LISTMEMN.PRG
356      *!              : LISTMEMR.PRG
357      *!              : LISTMEMC.PRG
358      *!
359      *!      Uses: MEMBER.DBF
360      *!              : TEMP1.DBF
361      *!              : TEMP1.DBF
362      *!
363      *!*****  

364      PROCEDURE CASE_SM2
365      *Procedure for case structure for submenu
366      PARAMETER MPROMPT
367
368      DO CASE
369      CASE MPROMPT = 'List by Name'
370          DO LISTMEMN
371      CASE MPROMPT = 'List by Roll-No'
372          DO LISTMEMR
373      CASE MPROMPT = 'List by Card-No'
374          DO LISTMEMC
375      CASE MPROMPT = 'Display database'
376          USE MEMBER
377          BROWSE
378          CLOSE DATABASES
379      ENDCASE
380      RETURN
381      * - - - -
382

```

```

383      *-----
384
395 *!*****  

*****  

396 *!  

397 *!      Procedure: SUB_FOR_DEL  

398 *!  

399 *!      Called by: MEMPROC          (procedure in DEFMENU.PRG)  

400 *!  

401 *!          Calls: PROMPT()        (function in ?)  

402 *!                  : CASE_SM3    (procedure in DEFMENU.PRG)  

403 *!  

404 *!*****  

*****  

395 PROCEDURE SUB_FOR_DEL  

396 *Procedure for sub menu creation for searching a book  

397 DEFINE POPUP SUB_MENU_3 FROM 6,20 COLOR SCHEME 2 TITLE "Deletion" S  

DOW  

398     DEFINE BAR 1 OF SUB_MENU_3 PROMPT "Using \<Name"  

399     DEFINE BAR 2 OF SUB_MENU_3 PROMPT "Using \<Roll No."  

400     ON SELECTION POPUP SUB_MENU_3 DO CASE_SM3 WITH PROMPT()  

401     ACTIVATE POPUP SUB_MENU_3  

402     RETURN  

403  

404  

405  

416 *!*****  

*****  

417 *!  

418 *!      Procedure: CASE_SM3  

419 *!  

420 *!      Called by: SUB_FOR_DEL      (procedure in DEFMENU.PRG)  

421 *!  

422 *!          Calls: DELMEMB.PRG  

423 *!                  : DELMEMR.PRG  

424 *!  

425 *!*****  

*****  

416 PROCEDURE CASE_SM3  

417 *Procedure for case structure for submenu  

418 PARAMETER MPROMPT  

419  

420     DO CASE  

421     CASE MPROMPT = 'Using Name'  

422         DO DELMEMB  

423     CASE MPROMPT = 'Using Roll No.'  

424         DO DELMEMR  

425     ENDCASE  

426     RETURN  

427     *-----  

428  

429  

430  

431     *: EOF: DEFMENU.PRG

```


10/20/94

15:33

ADDBOOK.PRG
Copyright, ANONYMOUS, 1994
LIBRARY MANAGEMENT SYSTEM

```
@ 0,23 SAY MTEMP
@ 1,0 SAY REPLICATE(CHR(205),64)
?
@ 18,0 SAY REPLICATE(CHR(205),64)
@ 2,2 SAY 'Book Name : ' GET BOOK_NAME
@ 3,2 SAY 'Author : ' GET AUTHOR
@ 4,2 SAY 'Volume : ' GET VOLUME
@ 5,2 SAY 'Edition : ' GET EDITION
@ 6,2 SAY 'Subject : ' GET SUBJECT
@ 7,2 SAY 'Copies : ' GET COPIES
@ 8,2 SAY 'Text/Ref. : ' GET T_OR_REF
@ 9,2 SAY 'Bank/Gen. : ' GET BNK_OR_GNR
@ 10,2 SAY 'Call No. : ' GET CALL_NO
@ 11,2 SAY 'Acctn. No. : ' GET ACCN_NO
@ 12,2 SAY 'Available : ' GET STATUS
@ 13,2 SAY 'Publisher : ' GET PUBLISHER
@ 14,2 SAY 'Price @ 15,2 SAY 'Date : ' GET
@ 16,2 SAY 'Agents : ' GET AGENTS
@ 17,2 SAY 'Bill No. : ' GET BILL_NO
READ
ENDFOR
? "Information has been appended to Book Database"
WAIT
SET CONFIRM OFF
DEACTIVATE WINDOW WIN1
CLOSE DATABASES
RETURN
```

*: EOF: ADDBOOK.PRG

10/20/94
15:35

ADDBOOK.ACT
Copyright, ANONYMOUS, 1994
LIBRARY MANAGEMENT SYSTEM

```
1 *:*****  
*****  
2 *:  
3 *: Program: D:\FP2\CASE\ADDBOOK.PRG  
4 *:  
5 *: System: LIBRARY MANAGEMENT SYSTEM  
6 *: Author: HARIHARAN. S, JITESH MENON. P, K  
ANNAN. N  
7 *: Copyright (c) 1994, ANONYMOUS  
8 *: Last modified: 10/20/94 14:52  
9 *:  
10 *: Called by: BOOKPROC (procedure in  
DEFMENU.PRG)  
11 *:  
12 *: Calls: CAPSLOCK() (function in  
?)  
13 *: : REPLICATE() (function in  
?)  
14 *:  
15 *:  
16 *: Uses: NAME.DBF  
17 *:  
18 *: Documented 10/20/94 at 15:27  
FoxDoc version 2.10e  
19 *:*****  
*****  
20 *procedure for adding books  
21  
22 SET TALK OFF  
23 SET CONFIRM ON  
24  
25 USE NAME  
26  
27 DEFINE WINDOW WIN1 FROM 03,05 TO 24,70 SHADOW T  
ITLE "Addition of a book"  
28 ACTIVATE WINDOW WIN1  
29 CLEAR  
30 INPUT 'How many books do you want to add : ' TO  
BOOKNO  
31 CAPSTATUS = CAPSLOCK(.T.)  
32 CLEAR  
33 FOR I=1 TO BOOKNO  
34 CLEAR  
35 SET CONFIRM ON
```

10/20/94
15:38

ADDBOOK.ACT
Copyright, ANONYMOUS, 1994
LIBRARY MANAGEMENT SYSTEM

```
36      APPEND BLANK
37      GO BOTTOM
38      STORE I TO MTEMP
39      @ 0,2 SAY 'Information for book '
40      @ 0,23 SAY MTEMP
41      @ 1,0 SAY REPLICATE(CHR(205),64)
42      ?
43      @ 18,0 SAY REPLICATE(CHR(205),64)
44      @ 2,2 SAY 'Book Name : ' GET BOOK_NAME
45      @ 3,2 SAY 'Author : ' GET AUTHOR
46      @ 4,2 SAY 'Volume : ' GET VOLUME
47      @ 5,2 SAY 'Edition : ' GET EDITION
48      @ 6,2 SAY 'Subject : ' GET SUBJECT
49      @ 7,2 SAY 'Copies : ' GET COPIES
50      @ 8,2 SAY 'Text/Ref. : ' GET T_OR_REF
52      |   @ 10,2 SAY 'Call No. : ' GET CALL_NO
53      @ 11,2 SAY 'Accn. No. : ' GET ACCN_NO
54      @ 12,2 SAY 'Available : ' GET STATUS
55      @ 13,2 SAY 'Publisher : ' GET PUBLISHER
56      @ 14,2 SAY 'Price : ' GET PRICE
57      @ 15,2 SAY 'Date : ' GET DATE
58      @ 16,2 SAY 'Agents : ' GET AGENTS
59      @ 17,2 SAY 'Bill No. : ' GET BILL_NO
60      READ
61      ENDFOR
62      ? "Information has been appended to Book Database
se"
63      WAIT
64      SET CONFIRM OFF
65      DEACTIVATE WINDOW WIN1
66      CLOSE DATABASES
67      RETURN
68
69      *: EOF: ADDBOOK.ACT
```

10/20/94

ADDMEM.PRG

15:41

Copyright, ANONYMOUS, 1994
LIBRARY MANAGEMENT SYSTEM

*:*****

*: Program: ADDMEM.PRG
*: System: LIBRARY MANAGEMENT SYSTEM
*: Author: HARIHARAN. S, JITESH MENON. P, KANNAN.
N
*: Copyright (c) 1994, ANONYMOUS
*: Last modified: 09/17/94 12:11
*: Called by: MEMPROC (procedure in DEFMEM
U.PRG)
*: Calls: CAPSLOCK() (function in ?)
*: : CHR() (function in ?)
*: Uses: MEMBER.DBF
*: Documented 10/20/94 at 14:45

version 2

*:*****

*procedure for adding members

```
SET TALK OFF
CAPSTATUS = CAPSLOCK(.T.)
USE MEMBER
GO BOTTOM
DEFINE WINDOW WIN1 FROM 03,05 TO 15,67 SHADOW TITLE "Addition of a member"
ACTIVATE WINDOW WIN1
CLEAR
INPUT 'How many members do you want to add : ' TO MEMNO
CLEAR
FOR I=1 TO MEMNO
  CLEAR
  SET CONFIRM ON
  APPEND BLANK
  GO BOTTOM
  STORE I TO MTEMP
  @ 0,2 SAY "Information for Member"
  @ 0,24 SAY MTEMP
  @ 1,0 SAY REPLICATE(CHR(205), 61)
  @ 8,0 SAY REPLICATE(CHR(205), 61)
```

10/20/94
15:43

ADDMEM.PRG
Copyright, ANONYMOUS, 1994
LIBRARY MANAGEMENT SYSTEM

```
@ 2,2 SAY 'Member Name : ' GET NAME
@ 3,2 SAY 'Card Number : ' GET CARD_NO
@ 4,2 SAY 'Roll Number : ' GET ROLL_NO
@ 5,2 SAY 'Student/Staff (S/F) : ' GET F_OR_S
@ 6,2 SAY 'Max. Number of books allowed : ' GET MAX_BKS
@ 7,2 SAY 'Access to book bank (Y/N) : ' GET AC_BNK
READ
ENDFOR
? "Information has been appended to Members Database"
WAIT
DEACTIVATE WINDOW WIN1
CLOSE DATABASES
RETURN
```

*: EOF: ADDMEM.PRG

10/20/94
15:55

DELBOOK.PRG
Copyright, ANONYMOUS, 1994
LIBRARY MANAGEMENT SYSTEM

```
*:*****  
*****  
*: Program: DELBOOK.PRG  
*: System: LIBRARY MANAGEMENT SYSTEM  
*: Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N  
*: Copyright (c) 1994, ANONYMOUS  
*: Last modified: 09/17/94 12:48  
*: Called by: BOOKPROC (procedure in DEFMENU  
.PRG)  
*: Calls: FOUND() (function in ?)  
*: Uses: NAME.DBF  
*: Documented 10/20/94 at 14:45 FoxDoc  
version 2.10e  
*:*****  
*****  
*procedure for deleting books  
  
SET TALK OFF  
USE NAME  
DEFINE WINDOW WIN1 FROM 03,05 TO 10,70 SHADOW TITLE "Deletion of a book"  
ACTIVATE WINDOW WIN1  
CLEAR  
INPUT 'How many books do you want to delete (or 0 to quit)  
' TO BOOKNO  
IF BOOKNO = 0  
    CLEAR  
    RETURN  
ENDIF  
CLEAR  
FOR I=1 TO BOOKNO  
    CLEAR  
    ? 'Book', I  
    ? '----'  
    ?  
    INPUT "Enter the Acn. Number for the book : " TO M.ACC  
ACCEPT "Are you sure (Y/N) ? " TO M.CHOICE  
?  
IF LOWER (M.CHOICE) = 'y'
```

10/20/94
15:56

DELBOOK.PRG
Copyright, ANONYMOUS, 1994
LIBRARY MANAGEMENT SYSTEM

```
LOCATE FOR ACCN_NO = M. ACC
IF FOUND() = .F.
    ? "Cannot find record with given accession number
", M. ACC
    ? "Record Not deleted, Try again !!""
    WAIT
    ?
    EXIT
ELSE
    DELETE FOR ACCN_NO = M. ACC
    ? "Specified record deleted."
    ?
    WAIT
ENDIF
ENDIF
IF LOWER (M.CHOICE) = 'n'
    ? "Record has not been deleted"
    ?
    WAIT
ENDIF
ENDFOR
PACK
DEACTIVATE WINDOW WIN1
CLOSE DATABASES
RETURN
```



1 *: ****
2 *:
3 *: Program: LISTMEMN.PRG
4 *:
5 *: System: LIBRARY MANAGEMENT SYSTEM
6 *: Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N
7 *: Copyright (c) 1994, ANONYMOUS
8 *: Last modified: 10/20/94 14:56
9 *:
10 *: Called by: CASE_SM2 (procedure in DEFMENU.PRG)
11 *:
12 *: Calls: UPPER() (function in ?)
13 *: : EOF() (function in ?)
14 *: : CHR() (function in ?)
15 *:
16 *: Uses: MEMBER.DBF
17 *:
18 *: Documented 10/20/94 at 15:27 FoxDoc version 2.10
19 *: ****

20 SET TALK OFF
21 DEFINE WINDOW SERWIN1 FROM 03,05 TO 15, 60 SHADOW TITLE "SEARCH BY
ME"
22 ACTIVATE WINDOW SERWIN1
23 DO WHILE .T.
24 CLEAR
25 @ 0,0 SAY 'Enter the name of the student or press Q to quit'
26 ACCEPT '>' TO MNAME
27 CLEAR
28 IF UPPER(MNAME) = 'Q'
29 CLEAR
30 EXIT
31 V ENDIF
32 USE MEMBER
33
34 LOCATE FOR UPPER(NAME) = UPPER(MNAME)
35 IF EOF()
36 ? "Cannot find student name ", MNAME
37 WAIT
38 LOOP
39 V ENDIF
40 DO WHILE .NOT. EOF()
41 CLEAR
42 @ 1,0 SAY REPLICATE(CHR(205), 53) : , GET NAME PICTURE '@'
43 @ 8,0 SAY REPLICATE(CHR(205), 53) : , GET CARD_NO PICTURE '@'
44 @ 2,3 SAY 'Student Name' : , GET ROLL_NO PICTURE '@'
45 @ 3,3 SAY 'Card Number' : , GET F_OR_S PICTURE '@'
46 @ 4,3 SAY 'Roll Number' : , GET MAX_BKS PICTURE '@'
47 @ 5,3 SAY 'Student/Staff' : , GET AC_BNK PICTURE '@'
48 @ 6,3 SAY 'Max. Books allowed' : , GET
49 @ 7,3 SAY 'Access to book bank': , GET
50 CONTINUE
51
52 ENDDO
53 WAIT
54 ENDDO
55 DEACTIVATE WINDOW SERWIN1
56 SET TALK ON
57 SET HEADING ON
58 CLOSE DATABASES
59 CLEAR
60
61 RETURN

```

1 *:*****
2 *:
3 *: Program: LISTMEMR.PRG
4 *:
5 *: System: LIBRARY MANAGEMENT SYSTEM
6 *: Author: HARIHARAN. S. JITESH MENON. P, KANNAN. N
7 *: Copyright (c) 1994, ANONYMOUS
8 *: Last modified: 10/20/94 14:57
9 *:
10 *: Called by: CASE_SM2 (procedure in DEFMENU.PRG)
11 *:
12 *: Calls: UPPER() (function in ?)
13 *: : EOF() (function in ?)
14 *: : CHR() (function in ?)
15 *:
16 *: Uses: MEMBER.DBF
17 *:
18 *: Documented 10/20/94 at 15:27 FoxDoc version 2.10e
19 *:*****
20
21 SET TALK OFF
22 DEFINE WINDOW SERWIN1 FROM 03,05 TO 15, 60 SHADOW TITLE "SEARCH BY NAME"
23 ACTIVATE WINDOW SERWIN1
24 DO WHILE .T.
25   CLEAR
26   @ 0,0 SAY 'Enter the Roll No. of the student or press Q to quit'
27   ACCEPT '>' TO MNAME
28   CLEAR
29   IF UPPER(MNAME) = 'Q'
30     CLEAR
31     EXIT
32   ENDIF
33
34   USE MEMBER
35
36   LOCATE FOR UPPER(ROLL_NO) = UPPER(MNAME)
37   IF EOF()
38     ? "Cannot find student roll number ", MNAME
39     WAIT
40   LOOP
41 ENDIF
42 DO WHILE .NOT. EOF()
43   CLEAR
44   @ 1,0 SAY REPLICATE(CHR(205), 53)
45   @ 8,0 SAY REPLICATE(CHR(205), 53)
46   @ 2,3 SAY 'Student Name' : 'GET NAME PICTURE '@T'
47   @ 3,3 SAY 'Card Number' : 'GET CARD_NO PICTURE '@T'
48   @ 4,3 SAY 'Roll Number' : 'GET ROLL_NO PICTURE '@T'
49   @ 5,3 SAY 'Student/Staff' : 'GET F_OR_S PICTURE '@T'
50   @ 6,3 SAY 'Max. Books allowed' : 'GET MAX_BKS PICTURE '@T'
51   @ 7,3 SAY 'Access to book bank:' : 'GET AC_BNK PICTURE '@T'
52   CONTINUE
53 ENDDO
54 WAIT
55 ENDDO
56 DEACTIVATE WINDOW SERWIN1
57 SET TALK ON
58 SET HEADING ON
59 CLOSE DATABASES
60 CLEAR
61 RETURN

```

```
1 *:*****  
****:  
2 *:  
3 *:      Program: LISTMEMC.PRG  
4 *:  
5 *:      System: LIBRARY MANAGEMENT SYSTEM  
6 *:      Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N  
7 *:      Copyright (c) 1994, ANONYMOUS  
8 *:      Last modified: 10/20/94    14:57  
9 *:  
10 *:      Called by: CASE_SM2          (procedure in DEFMENU.PRG)  
11 *:  
12 *:      Calls: EOF()           (function in ?)  
13 *:              : CHR()           (function in ?)  
14 *:  
15 *:      Uses: MEMBER.DBF  
16 *:  
17 *:      Documented 10/20/94 at 15:27          FoxDoc version 2.10e  
18 *:*****
```

```
*****  
19  
20      SET TALK OFF  
21      DEFINE WINDOW SERWIN1 FROM 03,05 TO 15, 60 SHADOW TITLE "SEARCH BY N  
ME"  
22      ACTIVATE WINDOW SERWIN1  
23      DO WHILE .T.  
24          CLEAR  
25          @ 0,2 SAY 'Enter the card number of the student or 0 to quit.'  
26          INPUT ">" TO M.CARD  
27          CLEAR  
28          IF M.CARD = 0  
29              CLEAR  
30              EXIT  
31          ENDIF  
32  
33          USE MEMBER  
34  
35          LOCATE FOR CARD_NO = M.CARD  
36          IF EOF()  
37              ? "Cannot find card number", M.CARD  
38              WAIT  
39          LOOP  
40          ENDIF  
41          DO WHILE .NOT. EOF()  
42              CLEAR  
43              @ 1,0 SAY REPLICATE(CHR(205), 53)  
44              @ 8,0 SAY REPLICATE(CHR(205), 53)  
45              @ 2,3 SAY 'Student Name' : 'GET NAME PICTURE '@T'  
46              @ 3,3 SAY 'Card Number' : 'GET CARD_NO PICTURE '@T'  
47              @ 4,3 SAY 'Roll Number' : 'GET ROLL_NO PICTURE '@T'  
48              @ 5,3 SAY 'Student/Staff' : 'GET F_OR_S PICTURE '@T'  
49              @ 6,3 SAY 'Max. Books allowed' : 'GET MAX_BKS PICTURE '@T'  
50              @ 7,3 SAY 'Access to book bank' : 'GET AC_BNK PICTURE '@T'  
51              CONTINUE  
52          ENDDO  
53          WAIT  
54      ENDDO  
55      DEACTIVATE WINDOW WIN1  
56      SET TALK ON  
57      SET HEADING ON  
58      CLOSE DATABASES  
59      CLEAR  
60      DFTUDM
```

```

1 ****
2 *: Program: MAINMENU.PRG
3 *:
4 *: System: LIBRARY MANAGEMENT SYSTEM
5 *: Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N
6 *:
7 *: Copyright (c) 1994, ANONYMOUS
8 *: Last modified: 10/20/94 14:59
9 *:
10 *: Calls: BORBOOK.PRG
11 *: : RETBOOK.PRG
12 *: : DEFMENU.PRG
13 *:
14 *: Documented 10/20/94 at 15:27
15 *:*****
16 *Main menu
17 CLEAR
18 CLEAR ALL
19 SET TALK OFF
20 DO WHILE .T.
21   DEFINE WINDOW MAIN1 FROM 05,25 TO 16, 56 TITLE
22     " MAIN MENU " PANEL SHADOW
23   DEFINE WINDOW MAIN2 FROM 00,01 TO 02, 78 DOUBLE SHADOW
24   DEFINE WINDOW MAIN3 FROM 20,01 TO 22, 78 SHADOW
25   ACTIVATE WINDOW MAIN2
26     @ 0,27 SAY 'LIBRARY MANAGEMENT SYSTEM'
27   ACTIVATE WINDOW MAIN3
28     @ 0,8 SAY 'Case study lab by Hariharan. S,
29       JITESH MENON. P AND KANNAN. N'
30   ACTIVATE WINDOW MAIN1
31     @ 2, 8 SAY '1. Lend'
32     @ 3, 8 SAY '2. Receive'
33     @ 4, 8 SAY '3. Control menu'
34     @ 5, 8 SAY '4. Help'
35     @ 6, 8 SAY '5. Exit'
36     ?
37     WAIT ' Enter your choice : ' TO CHOICE
38
39
40   DO CASE
41     CASE CHOICE = '1'
42       DO BORBOOK
43     CASE CHOICE = '2'
44       DO RETBOOK
45     CASE CHOICE = '3'
46       DO DEFMENU
47     CASE CHOICE = '4'
48       DO DEFMENU
49     CASE CHOICE = '5'
50       DEACTIVATE WINDOW MAIN1
51       DEACTIVATE WINDOW MAIN2
52       DEACTIVATE WINDOW MAIN3
53   EXIT
54   OTHERWISE
55     DEFINE WINDOW POP1 FROM 3, 35 TO 8, 70 SHADOW
56     ACTIVATE WINDOW POP1
57     @ 1, 5 SAY 'Wrong Choice entered'

```

FoxDoc version 2.10e

```
58     WAIT
59     DEACTIVATE WINDOW POP1
60   ENDCASE
61     DEACTIVATE WINDOW MAIN1
62     DEACTIVATE WINDOW MAIN2
63     DEACTIVATE WINDOW MAIN3
64   ENDDO
65     SET TALK ON
66   *: EOF: MAINMENU.PRG
```

```
1  *: ****
2  *:
3  *:           Program: RETBOOK.PRG
4  *:
5  *:           System: LIBRARY MANAGEMENT SYSTEM
6  *:           Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N
7  *:           Copyright (c) 1994, ANONYMOUS
8  *:           Last modified: 10/20/94      15:02
9  *:
10 *:          Called by: MAINMENU.PRG
11 *:
12 *:          Calls: CAPSLOCK()          (function in ?)
13 *:                  : UPPER()          (function in ?)
14 *:                  : EOF()           (function in ?)
15 *:                  : DATE()          (function in ?)
16 *:                  : CTOD()          (function in ?)
17 *:                  : CHR()           (function in ?)
18 *:
19 *:          Uses: MEMBER.DBF
20 *:                  : NAME.DBF
21 *:                  : CIRC.DBF
22 *:
23 *:          Documented 10/20/94 at 15:27          FoxDoc version 2.10
24 *: ****
*****  
25
26     SET TALK OFF
27     CAPSTATUS = CAPSLOCK(.T.)
28     DEFINE WINDOW SERWIN1 FROM 03,05 TO 15, 60 SHADOW TITLE "Return a b  
k"
29     ACTIVATE WINDOW SERWIN1
30     *DO WHILE .T.
31     *-----
32     *Searches for the Roll Number in the member database
33     CLEAR
34     @ 0,0 SAY 'Enter the Roll No. of the student or press Q to quit'
35     ACCEPT '>' TO M.ROL
36     CLEAR
37     IF UPPER(M.ROL) = 'Q'
38     CLEAR
39     DEACTIVATE WINDOW SERWIN1
40     RETURN
41   ENDIF
42
43     USE MEMBER
44
45     LOCATE FOR UPPER(ROLL_NO) = UPPER(M.ROL)
```

```

46      IF EOF()
47          ? "Cannot find student roll number ", M.ROL
48          WAIT
49          DEACTIVATE WINDOW SERWIN1
50      ← RETURN
51  └─ENDIF
52      * STORE MAX_BKS TO max
53      *-
54      *-
55      *Searches for Accn. No in the book database
56
57      @ 1,0 SAY 'Enter the Accn. No. of the book or press 0 to quit'
58      INPUT ' > ' TO MNO
59      CLEAR
60      IF MNO = 0
61          CLEAR
62          DEACTIVATE WINDOW SERWIN1
63      ← RETURN
64  └─ENDIF
65
66      USE NAME
67
68      LOCATE FOR ACCN_NO = MNO
69      IF EOF()
70          ? "Cannot find given Accn. No. of the book", MNO
71          WAIT
72          DEACTIVATE WINDOW SERWIN1
73      ← RETURN
74  └─ENDIF
75      *-
76      *Searches the circulation database to check whether there is an
77      *existing record for this roll number
78
79      USE CIRC
80      LOCATE FOR ROLL_NO = M.ROL
81      IF EOF()
82          CLEAR
83          @ 1, 1 SAY 'This person has not borrowed any book - ' +M.ROL
84          WAIT
85          DEACTIVATE WINDOW SERWIN1
86      ← RETURN
87  └─ENDIF
88
89      DO CASE
90      CASE ACCN_NO_1 = MNO
91          REPLACE ACCN_NO_1 WITH 0
92          DAYS = (DATE() - CTOD(DUE_DATE_1))
93          IF DAYS > 0
94              T_FINE = FINE + DAYS * 0.50
95              REPLACE FINE WITH T_FINE
96          ENDIF
97          REPLACE DUE_DATE_1 WITH ''
98      CASE ACCN_NO_2 = MNO
99          REPLACE ACCN_NO_2 WITH 0
100         DAYS = (DATE() - CTOD(DUE_DATE_2))
101        IF DAYS > 0
102            T_FINE = FINE + DAYS * 0.50
103            REPLACE FINE WITH T_FINE
104        ENDIF
105        REPLACE DUE_DATE_2 WITH ''
106    CASE ACCN_NO_3 = MNO

```

```

107      REPLACE ACCN_NO_3 WITH 0
108      DAYS = (DATE() - CTOD(DUE_DATE_3))
109      IF DAYS > 0
110          T_FINE = FINE + DAYS * 0.50
111          REPLACE FINE WITH T_FINE
112      ENDIF
113      REPLACE DUE_DATE_3 WITH ''
114      CASE ACCN_NO_4 = MNO
115          REPLACE ACCN_NO_4 WITH 0
116          DAYS = (DATE() - CTOD(DUE_DATE_4))
117          IF DAYS > 0
118              T_FINE = FINE + DAYS * 0.50
119              REPLACE FINE WITH T_FINE
120          ENDIF
121          REPLACE DUE_DATE_4 WITH ''
122      CASE ACCN_NO_5 = MNO
123          REPLACE ACCN_NO_1 WITH 0
124          DAYS = (DATE() - CTOD(DUE_DATE_5))
125          IF DAYS > 0
126              T_FINE = FINE + DAYS * 0.50
127              REPLACE FINE WITH T_FINE
128          ENDIF
129          REPLACE DUE_DATE_5 WITH ''
130      OTHERWISE
131          CLEAR
132          @ 1,1 SAY 'The book with the Acn. No'
133          @ 1,28 SAY MNO
134          @ 2,1 SAY 'has not been borrowed by Roll No. '+M.ROL
135          ?
136          WAIT
137          DEACTIVATE WINDOW SERWIN1
138      RETURN
139  ENDCASE
140      TOT = TOT_BOOKS - 1
141      REPLACE TOT_BOOKS WITH TOT
142
143      IF FINE > 0
144          CLEAR
145          @ 0,0 SAY REPLICATE(CHR(205), 52)
146          @ 6,0 SAY REPLICATE(CHR(205), 52)
147          @ 1,7 SAY 'Roll Number : '+M.ROL
148          @ 2,7 SAY 'Acn. No : '
149          @ 2,30 SAY MNO PICT '99999'
150          @ 3,7 SAY 'Lag period : '
151          @ 3,30 SAY DAYS PICT '999.99'
152          @ 3,34 SAY 'days'
153          @ 4,7 SAY 'Fine (0.5/day) : '
154          @ 4,30 SAY (DAYS*0.50) PICT '999.99'
155          @ 5,7 SAY 'Total Fine : '
156          @ 5,30 SAY FINE PICT '999.99'
157          ?
158          ACCEPT 'Collect fine now (Y/N) ? ' TO CHOICE
159  ENDIF
160
161  DO CASE
162      CASE UPPER(CHOICE) = 'Y'
163          REPLACE FINE WITH 0
164          CLEAR
165          @ 1,1 SAY 'Collect cash and issue receipt'
166      CASE UPPER(CHOICE) = 'N'
167          CLEAR

```

```
168     @ 1,1 SAY 'Inform member to pay cash as soon as possible.'
169     ENDCASE
170     ?
171     WAIT
172
173     USE NAME
174
175     LOCATE FOR ACCN_NO = MNO
176     REPLACE STATUS WITH 'Y'
177
178     DEACTIVATE WINDOW SERWIN1
179     CLEAR
180     CLOSE DATABASES
181     SET TALK ON
182
183 .    *: EOF: RETBOOK.PRG
```

```
1  *:*****
2  *:
3  *: Procedure file: SORTMEM.PRG
4  *:
5  *:           System: LIBRARY MANAGEMENT SYSTEM
6  *:           Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N
7  *:           Copyright (c) 1994, ANONYMOUS
8  *:           Last modified: 10/20/94      15:04
9  *:
10 *:           Set by: MEMPROC          (procedure in DEFMENU.PRG)
11 *:
12 *:           Calls: PROMPT()        (function in ?)
13 *:                     : CASE_SM1   (procedure in DEFMENU.PRG)
14 *:
15 *:           Uses: NAME.DBF
16 *:
17 *:           Documented 10/20/94 at 15:27          FoxDoc version 2.10e
18 *:*****
```

```
19 *Procedure for sorting members
20 CLEAR
21 SET TALK OFF
22 USE NAME
23
24 DEFINE POPUP SM_2 FROM 6,15 COLOR SCHEME 2 TITLE "Sort" SHADOW
25 DEFINE BAR 1 OF SM_2 PROMPT "Sort by \<Name"
26 DEFINE BAR 2 OF SM_2 PROMPT "Sort by \<Roll No."
27 DEFINE BAR 3 OF SM_2 PROMPT "Sort by \<Student/Staff"
28 DEFINE BAR 4 OF SM_2 PROMPT "Sort by \<Card No."
29 DEFINE BAR 5 OF SM_2 PROMPT "Sort by \<Total Books"
30 ON SELECTION POPUP SM_2 DO CASE_SM1 WITH PROMPT()
31
32 ACTIVATE POPUP SM_2
33
34 -----
35
36 *!*****
```

```
37 *!
38 *!           Procedure: CASE_SM2
39 *!
40 *!           Called by: SUB_FOR_LIST          (procedure in DEFMENU.PRG)
41 *!
42 *!
```

43 *! Calls: LISTMEMN.PRG

```
58 *! : LISTMEMR.PRG
59 *! : LISTMEMC.PRG
60 *!
61 *!     Uses: MEMBER.DBF
62 *!     : TEMP1.DBF
63 *!     : TEM1P1.DBF
64 *!
65 *!*****PROCEDURE CASE_SM2
*****  
51 PROCEDURE CASE_SM2
52 *Procedure for case structure for submenu
53 PARAMETER MPROMPT
54
55 DO CASE
56   CASE MPROMPT = 'Sort by Book name'
57     USE MEMBER
58     SORT TO TEMP1 ON BOOK_NAME
59     CLOSE DATABASES
60     COPY FILE TEMP1.DBF TO MEMBER.DBF
61   CASE MPROMPT = 'Sort by Author'
62     USE MEMBER
63     SORT TO TEMP1 ON BOOK_NAME
64     CLOSE DATABASES
65     COPY FILE TEMP1.DBF TO MEMBER.DBF
66   CASE MPROMPT = 'Sort by Accn. No.'
67     USE MEMBER
68     SORT TO TEMP1 ON BOOK_NAME
69     CLOSE DATABASES
70     COPY FILE TEMP1.DBF TO MEMBER.DBF
71   CASE MPROMPT = 'Sort by Subject'
72     USE MEMBER
73     SORT TO TEMP1 ON BOOK_NAME
74     CLOSE DATABASES
75     COPY FILE TEM1P1.DBF TO MEMBER.DBF
76   CASE MPROMPT = 'Sort by Total books'
77 ENDCASE
78 RETURN
79
80
81 *: EOF: SORTMEM.PRG
```

```

1 *:*****  

*****  

2 *:  

3 *: Procedure file: SORTBOOK.PRG  

4 *:  

5 *: System: LIBRARY MANAGEMENT SYSTEM  

6 *: Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N  

7 *: Copyright (c) 1994, ANONYMOUS  

8 *: Last modified: 10/20/94 14:53  

9 *:  

10 *: Set by: BOOKPROC (procedure in DEFMENU.PRG)  

11 *:  

12 *: Calls: PROMPT() (function in ?)  

13 *: : CASE_SM1 (procedure in DEFMENU.PRG)  

14 *:  

15 *: Uses: NAME.DBF  

16 *:  

17 *: Documented 10/20/94 at 15:27 FoxDoc version 2.10e  

18 *:*****  

*****  

19 *Procedure for sorting a book  

20 CLEAR  

21 SET TALK OFF  

22 USE NAME  

23  

24 DEFINE POPUP SM_1 FROM 6,15 COLOR SCHEME 2 TITLE "Sort" SHADOW  

25 DEFINE BAR 1 OF SM_1 PROMPT "Sort by \<Book name"  

26 DEFINE BAR 2 OF SM_1 PROMPT "Sort by \<Author"  

27 DEFINE BAR 3 OF SM_1 PROMPT "Sort by Acc\<n. No."  

28 DEFINE BAR 4 OF SM_1 PROMPT "Sort by \<Subject"  

29 ON SELECTION POPUP SM_1 DO CASE_SM1 WITH PROMPT()  

30 ACTIVATE POPUP SM_1  

31  

32 *-----  

33  

49 *!*****  

*****  

50 *!  

51 *! Procedure: CASE_SM1  

52 *!  

53 *! Called by: SUB_FOR_SER (procedure in DEFMENU.PRG)  

54 *! : SORTBOOK.PRG  

55 *! : SORTMEM.PRG  

56 *!  

57 *! Calls: SERBOOKN.PRG  

58 *! : SERBOOKA.PRG  

59 *!  

60 *! Uses: NAME.DBF  

61 *! : TEMP.DBF  

62 *!  

63 *!*****  

*****  

49 PROCEDURE CASE_SM1  

50 *Procedure for case structure for submenu  

51 PARAMETER MPROMPT  

52  

53 DO CASE  

54 CASE MPROMPT = 'Sort by Book name'  

55 USE NAME  

56 SORT TO TEMP ON BOOK_NAME  

57 CLOSE DATABASES

```

```

58   COPY FILE TEMP.DBF TO NAME.DBF
59   CASE MPROMPT = 'Sort by Author'
60     USE NAME
61     SORT TO TEMP ON AUTHOR
62     CLOSE DATABASES
63     COPY FILE TEMP.DBF TO NAME.DBF
64   CASE MPROMPT = 'Sort by Accn. No.'
65     USE NAME
66     SORT TO TEMP ON ACCN_NO
67     CLOSE DATABASES
68     COPY FILE TEMP.DBF TO NAME.DBF
69   CASE MPROMPT = 'Sort by Subject'
70     USE NAME
71     SORT TO TEMP ON SUBJECT
72     CLOSE DATABASES
73     COPY FILE TEMP.DBF TO NAME.DBF
74 END CASE
75 RETURN
76
77
78 *: EOF: SORTBOOK.PRG

```

```

1 *: ****
2 *:
3 *:      Program: SERBOOKN.PRG
4 *:
5 *:      System: LIBRARY MANAGEMENT SYSTEM
6 *:      Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N
7 *:      Copyright (c) 1994, ANONYMOUS
8 *:      Last modified: 10/20/94    14:56
9 *:
10 *:      Called by: CASE_SM1          (procedure in DEFMENU.PRG)
11 *:
12 *:      Calls: UPPER()           (function in ?)
13 *:              : EOF()            (function in ?)
14 *:              : CHR()            (function in ?)
15 *:
16 *:      Uses: NAME.DBF
17 *:
18 *:      Documented 10/20/94 at 15:27
19 *: **** FoxDoc version 2.10e
*****  

20
21   SET TALK OFF
22   SET HEADING OFF
23   PRIVATE MNAME
24   DEFINE WINDOW SERWIN1 FROM 03,05 TO 20, 65 FLOAT
25   SHADOW TITLE "Search by Author"
26   ACTIVATE WINDOW SERWIN1
27   DO WHILE .T.
28     CLEAR
29     @ 0,0 SAY 'Enter the name of the book or press Q to quit. '
30     ACCEPT '>' TO MNAME
31     CLEAR
32     IF UPPER(MNAME) = 'Q'
33       CLEAR
34     EXIT
35     DEACTIVATE WINDOW SERWIN1
36   ENDIF
37
38   USE NAME

```

```

39      LOCATE FOR UPPER(BOOK_NAME) = UPPER(MNAME)
40
41      IF EOF()
42          ? " Cannot find the book ", MNAME
43          WAIT
44      LOOP
45
46      ┌─ ENDIF
47      DO WHILE .NOT. EOF()
48          CLEAR
49          @ 0,0 SAY REPLICATE (CHR(205), 59)
50          @ 14,0 SAY REPLICATE (CHR(205), 59)
51          @ 1,4 SAY 'Book Name : ' GET BOOK_NAME PICTURE '@T'
52          @ 2,4 SAY 'Author : ' GET AUTHOR PICTURE '@T'
53          @ 3,4 SAY 'Volume : ' GET VOLUME PICTURE '@T'
54          @ 4,4 SAY 'Edition : ' GET EDITION PICTURE '@T'
55          @ 5,4 SAY 'Copies : ' GET COPIES PICTURE '@T'
56          @ 6,4 SAY 'Text/Ref. : ' GET T_OR_REF PICTURE '@T'
57          @ 7,4 SAY 'Bank/Gen. : ' GET BNK_OR_GNR PICTURE '@T'
58          @ 8,4 SAY 'Call No. : ' GET CALL_NO PICTURE '@T'
59          @ 9,4 SAY 'Acctn. No. : ' GET ACCN_NO PICTURE '@T'
60          @ 10,4 SAY 'Publisher : ' GET PUBLISHER PICTURE '@T'
61          @ 11,4 SAY 'Price : ' GET PRICE PICTURE '@T'
62          @ 12,4 SAY 'Agents : ' GET AGENTS PICTURE '@T'
63          @ 13,4 SAY 'Bill No. : ' GET BILL_NO PICTURE '@T'
64          CONTINUE
65      ENDDO
66      ?
67      WAIT
68  ENDDO
69  DEACTIVATE WINDOW SERWIN1
70  SET TALK ON
71  SET HEADING ON
72  CLOSE DATABASES
73  CLEAR
74  RETURN
75  *: EOF: SERBOOKN.PRG
1  *:***** ****
2  *:
3  *: Program: SERBOOKA.PRG
4  *:
5  *: System: LIBRARY MANAGEMENT SYSTEM
6  *: Author: HARIHARAN. S, JITESH MENON. P, KANNAN. N
7  *: Copyright (c) 1994, ANONYMOUS
8  *: Last modified: 10/20/94 14:55
9  *:
10 *: Called by: CASE_SM1 (procedure in DEFMENU.PRG)
11 *:
12 *: Calls: UPPER() (function in ?)
13 *: : EOF() (function in ?)
14 *: : CHR() (function in ?)
15 *:
16 *: Uses: NAME.DBF
17 *:
18 *: Documented 10/20/94 at 15:27
19 *:***** ****
20
21      SET TALK OFF

```

```

22      SET HEADING OFF
23      PRIVATE MNAME
24      DEFINE WINDOW SERWIN1 FROM 03,05 TO 20, 65 FLOAT SHADOW TITLE "Search
by Author"
25      ACTIVATE WINDOW SERWIN1
26      DO WHILE .T.
27          CLEAR
28          @ 0,0 SAY 'Enter the name of the Author or press Q to quit. '
29          ACCEPT '>' TO MNAME
30          CLEAR
31          IF UPPER(MNAME) = 'Q'
32              CLEAR
33              EXIT
34          DEACTIVATE WINDOW SERWIN1
35      ENDIF
36
37      USE NAME
38
39      LOCATE FOR UPPER(AUTHOR) = UPPER(MNAME)
40      IF EOF()
41          ? " Cannot find the author ", MNAME
42          WAIT
43      LOOP
44  ENDIF
45  DO WHILE .NOT. EOF()
46      CLEAR
47      @ 0,0 SAY REPLICATE (CHR(205), 59)
48      @ 14,0 SAY REPLICATE (CHR(205), 59)
49      @ 1,4 SAY 'Book Name : ' GET BOOK_NAME PICTURE '@T'
50      @ 2,4 SAY 'Author : ' GET AUTHOR PICTURE '@T'
51      @ 3,4 SAY 'Volume : ' GET VOLUME PICTURE '@T'
52      @ 4,4 SAY 'Edition : ' GET EDITION PICTURE '@T'
53      @ 5,4 SAY 'Copies : ' GET COPIES PICTURE '@T'
54      @ 6,4 SAY 'Text/Ref. : ' GET T_OR_REF PICTURE '@T'
55      @ 7,4 SAY 'Bank/Gen. : ' GET BNK_OR_GNR PICTURE '@T'
56      @ 8,4 SAY 'Call No. : ' GET CALL_NO PICTURE '@T'
57      @ 9,4 SAY 'Accn. No. : ' GET ACCN_NO PICTURE '@T'
58      @ 10,4 SAY 'Publisher : ' GET PUBLISHER PICTURE '@T'
59      @ 11,4 SAY 'Price : ' GET PRICE PICTURE '@T'
60      @ 12,4 SAY 'Agents : ' GET AGENTS PICTURE '@T'
61      @ 13,4 SAY 'Bill No. : ' GET BILL_NO PICTURE '@T'
62      CONTINUE
63  ENDDO
64  ?
65  WAIT
66  ENDDO
67  DEACTIVATE WINDOW SERWIN1
68  SET TALK ON
69  SET HEADING ON
70  CLOSE DATABASES
71  CLEAR
72  RETURN
73
74  *: EOF: SERBOOKA.PRG

```

MAIN MENU

Enter your choice :

- 1. ADD
- 2. REMOVE
- 3. DISPLAY
- 4. SEARCH
- 5. EXIT

Case study lab by Hariharan. S, Jitesh Menon. P and Karthik. N

Lend a book to the member

LIBRARY MANAGEMENT SYSTEM

Borrow a book

Roll Number : 91CSE09
Accn. No. : 4

Book borrowed !!

(Press any key to continue)

Case study lab by Hariharan. S, Jitesh Menon. P and Karthik. N

LIBRARY MANAGEMENT SYSTEM

Return a book

Roll Number : 91CSE09
Accn. No. :

No fine. The book has been returned

<Press any key to continue>

Case study lab by Hariharan. S, Jitesh Menon. P and Kannan. N

LIBRARY MANAGEMENT SYSTEM

Control Menu for LMS

BOOK DETAILS

MEMBER DETAILS

HELP

QUIT

- Add a book
- Delete a book
- Search for a book
- Sort books
- Display books

Case study lab by Hariharan. S, Jitesh Menon. P and Kannan. N

LIBRARY MANAGEMENT SYSTEM

MAIN MENU

- 1. Lend
- 2. Receive
- 3. Control menu
- 4. Help
- 5. Exit

Enter your choice :

Case study lab by Hariharan. S, Jitesh Menon. P and Kannan. N

LIBRARY MANAGEMENT SYSTEM

Control Menu for LMS

BOOK DETAILS

MEMBER DETAILS

HELP

QUIT

- Add a book
- Delete a book
- Search for a book
- Sort books

Case study lab by Hariharan. S, Jitesh Menon. P and Kannan. N

LIBRARY MANAGEMENT SYSTEM

Control Menu for LMS

BOOK DETAILS MEMBER DETAILS HELP QUIT

Add a book	
Delete a book	
Search for a book	Search
Sort book	Search using Book name
Display b	Search using Author

Case study lab by Hariharan. S, Jitesh Menon. P and Kannan. N

LIBRARY MANAGEMENT SYSTEM

Control Menu for LMS

BOOK DETAILS MEMBER DETAILS HELP QUIT

Add a book	
Delete a book	
Search for a book	Search
Sort book	Sort
Display b	Sort by Book name
	Sort by Author

Case study lab by Hariharan. S, Jitesh Menon. P and Kannan. N

LIBRARY MANAGEMENT SYSTEM

Addition of a book
information for book 1

Book Name	:	BORLAND C++
Author	:	PETERSON
Volume	:	3
Edition	:	1995
Subject	:	COMPUTER
Copies	:	2
Text/Ref.	:	?
Bank/Gen.	:	6
Call No.	:	54
Access. No.	:	9885
Available	:	Y
Publisher	:	WATTE GROUP PRESS
Price	:	150.00
Date	:	11/11/95
Agents	:	HIGGIN BOTHAMS
Bill No.	:	23

LIBRARY MANAGEMENT ITEM

Deletion of a book

BOOK 1

Accession Number : 5
Are you sure (Y/N) ? Y

Record deleted
<Space to CONTINUE/ESC to quit>

Case study lab by Hariharan, S., Jitleen Morris, P. and Kannan, M.

6:38:50 am

LIBRARY MANAGEMENT SYSTEM

Control Menu for LMS

BOOK DETAILS

MEMBER DETAILS

HELP

QUIT

Sort
Sort by Name
Sort by Roll No.

Add a member
Delete a member
Search members

Case study lab by Hariharan, S, Jitresh Menon, P and Kannan, R

6:38:41 am

LIBRARY MANAGEMENT SYSTEM

Control Menu for LMS

BOOK DETAILS

MEMBER DETAILS

HELP

QUIT

Lending
Receiving
Members
Books
Reservation
Sorting

Case study lab by Hariharan, S, Jitresh Menon, P and Kannan, R

8:40:03 am

LIBRARY MANAGEMENT SYSTEM

Control Menu for LMS

BOOK DETAILS MEMBER DETAILS HELP

QUIT

Return to Fox Pro
 Quit to DOS prompt

Case study lab by Hariharan. S, Jitesh Menon. P and Kannan. N

LIBRARY MANAGEMENT SYSTEM

Borrow a book

Enter the Roll No. of the student or press 0 to quit.

3 SICSE09

Case study lab by Hariharan. S, Jitesh Menon. P and Kannan. N