Assignment7

1301058

Zhang Junming

exercise 1

Question

Define a structure type library book to represent each book in the library. Each book should be identified by the following IDs: book name (string), author name (string) and a library ID number (2 characters plus 7 digits). In this assignment, only "Sc" and "EN" are used (stand for science and engineering)) as the 2 characters in the library ID number.

Model Answer

Software Development Process

1. **Problem statement**

   Write a C program, ask user to choose functions, one is initialize a new book and the initialized new book could add into the file (librarybooks.txt) which contains

all the books the library has. Another is when user search book name, display the detail about this book, if not find this book, tell user that library does not have the book.

2. Analysis

Inputs:

1) The details about added book: (Book name, Author name and Library ID).

2) The book name which user wants to search.

Outputs:

1) Details about added book in Librarybooks.txt

2) Details about searched book.

Additional requirements or constraint

In library ID, only "Sc" and "EN" are used (stand for Science and Engineering)

3. Design

Algorithm

1. Adding 'stdio.h' ,'stdlib.h','string.h' and 'malloc.h' library.

2. Define menu(), addbook() and searchbook() three defined function.

3. Create struct: library_book

   (1) char book_name-represents the name of added book.
   (2) char author_name-represents the name of book author.
   (3) char library_ID-represents 7 digits of the book ID number.

4. Write main function.
   (1) int choose-represents options.

(2) Setting up a loop use 'do while', ask user to input a number choose functions, only input '3' could out this loop.

(3) Using defined function menu().

(4) Ask user enter 1~3 to choose function.

(5) Read and store entered number in **choose**.

(6) Receive EHTER.

(7) . Setting up 'switch' function of **choose**.

Case 1: using addbook defined function.

Case 2: using search defined function.

Case 3: display quit program.

Default when entered number is not in 1~3, display enter data error. Please enter again.

5. Write first defined function 'menu ()':

(1) Display the main menu.

(2) Ask user to input number to choose functions.

6. Write second defined function, 'addbook()':

(1) Struct library_book lib-lode library_book in lib.

(2) int choose - for choosing options.

(3) int count – for counting.

(4) Char ID1[10]="Sc" – represents Science library ID number.

(5) Char ID2[10]="EN" – represents Engineering library ID number.

(6) FILE *fp-represents librarybooks.txt

(7) Tell user how to use this function add book.

(8) Ask user input book name.

(9) Read and store book name in **book_name**.

(10) Ask user input author name.

(11) Read and store author name in **author_name**.

(12) Ask user choose book types, 1 represents science book and 2 represents engineering book.

(13) Read and store entered number in **choose**.

(14) Receive EHTER.

(15) Judge **choose**, if **choose** equal to 1.

(1) Setting up a loop use "do-while", when **count** equal to 7, out the loop.

(2) Add library ID to ID1 (add number behind Sc)

(3) Open librarybooks.txt

(4) Store and printf book name, book author and ID1 in librarybooks.txt.

(5) Close libraybooks.txt.

(6) Use system 'pause' and 'cls' clear screen after using this defined function.

(16) Judge **choose**, if **choose** equal to 1.

(1) Setting up a loop use "do-while", when **count** equal to 7, out the loop.

(2) Add library ID to ID2 (add number behind EN)

(3) Open librarybooks.txt

(4) Store and printf book name, book author and ID2 in librarybooks.txt.

(5) Close libraybooks.txt.

(6) Use system 'pause' and 'cls' clear screen after using this defined function.

(17) Judge **choose**, if **choose** did not equal to 1 or 2, tell user enter 1 or 2 and return menu.

(18) Use system 'pause' and 'cls' clear screen after using this defined function.

7. Write second defined function, 'searchbook()':
   (1) Struct library_book lib-lode library_book in lib.
   (2) int i,j.count - for counting.
   (3) char aim-represents the name of searching book.
   (4) FILE *fp-represents librarybooks.txt
   (5) char **bookname-for storing information in librarybooks.txt.
   (6) Open librarybooks.txt.

   (7) Setting up a loop use 'while', and counting how many items in librarybooks.txt.

   (8) Return to the outset of librarybooks.txt.

   (9) Request a block of memory of a give size for **bookname** and **bookname[i].**

   (10) Setting up a loop use 'for', put each item in **bookname[i]**.

   (11) Tell user how to follow this function.

   (12) Ask user enter the name of searching book.

   (13) Read and store data in **aim**.

(14) Setting up a loop use 'for', compare each item in **bookname[i]** and **aim**, if they are the same strings, display following information. At the same time, record the times which **bookname[i]** different with **item**.

(15) To release memory allocated with malloc().

(16) If the number of items and times of different items are same, display can't find this book.

(17) Use system 'pause' and 'cls' clear screen after using this defined function.

(18) Close librarybooks.txt.

(19) Use system 'pause' and 'cls' clear screen after using this defined function.

4. **Implementation**:

See the C code in file exercise1.c with comments.

```
/*
Name: Simple Program to add book and search book

File Name: exercise1.c

Copyright: Free

Author: Zhang Junming

Description: Ask user to choose functions using number (1~3):
             1- add book in librarybooks.txt
             2- search book name in librarybooks.txt and display the related information
             3- quit program;
*/
#include <stdio.h>
#include <string.h>
#include <stdlib.h>/* for malloc(), free() */
void menu();/* defined function of menu */
void addbook();/* declar a defined function addbook */
void searchbook();/* declar a defined function searchbook */
struct library_book /* setting up the structure declaration */
{
  char book_name[50];/* represents the name of added book */
  char author_name[20];/* represents the name of book author */
  char library_ID[8];/* represents 7 digits of the book ID number */
};

main()
{
int choose;/* represents options */
do/* setting up a loop for run following commands */
 {
  menu();
  printf("Please enter 1~3 to choose function\n");
  scanf("%d",&choose);
  getchar();/* receive EHTER */
  system("cls");/* system("cls") clears whatever is display on the screen */
  switch(choose)/* choose options */
  {
   case 1:addbook();break;/* 1- add book in librarybooks.txt  */
   case 2:searchbook();break;/* 2- search book name in librarybooks.txt and display the related information */
   case 3:printf("Quit program!!\n");break;
   default:printf("enter error! Please enter again!\n");/* when input wrong command, display enter data error */
  }
 }
while(choose!=3);/* if input 3, out of this loop */
system("pause");/* let result display on the screen before press any key to continue */
}

void menu()/* defined function of menu */
{
  printf("**********************\n");
  printf("******  menu  ********\n");
  printf("**  1- add book  *****\n");
  printf("**  2- search book  **\n");
  printf("**  3- quit  *********\n");
  printf("**********************\n\n");
}
```

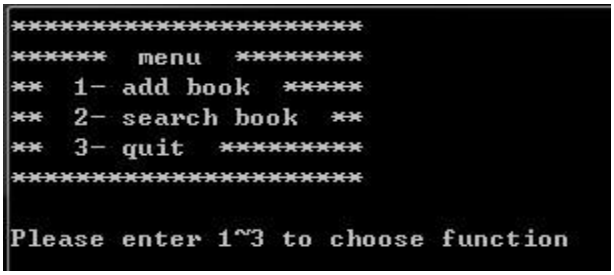
```c
void addbook()/* declar a defined function addbook */
{
  struct library_book lib;
  int choose;/* for choosing options */
  int count;/* for counting */
  char ID1[10] = "Sc";/* represents Science library ID number */
  char ID2[10] = "EN";/* represents Engineering library ID number */
  FILE*fp;/* represents librarybooks.txt */

  printf("Please follow 1~4 steps\n");

  printf("**********************\n");
  printf("(1 add book name \n");
  printf("(2 add author name \n");
  printf("(3 choose book types \n");
  printf("(4 add library ID number\n");
  printf("**********************\n\n");/* tell user how to use this function */

  printf("Please enter book name\n");
  gets(lib.book_name);
  printf("\n");

  printf("Please enter author name\n");
  gets(lib.author_name);
  printf("\n");

  printf("Please choose types: 1-science 2-engineering\n");
  scanf("%d",&choose);
  getchar();/* receive EHTER */
  if(choose==1)
   {
    do
     {
      printf("Please enter library ID (7 digits) \n");
      gets(lib.library_ID);
      count=strlen(lib.library_ID);/* counting how many digits in library_ID */
      }
     while(count!=7);/* limit library ID is a 7 digit number */
    strncat(ID1,lib.library_ID,10);/* add library ID to 'Sc' */
    fp=fopen("librarybooks.txt","a");/* open or create librarybooks.txt */
    if(fp==NULL){
    printf("Failed opening librarybooks.txt!\n");
    return;}
    fprintf(fp,"%s ",lib.book_name);/* input bookname in librarybooks.txt */
    fprintf(fp,"%s ",lib.author_name);/* input author in librarybooks.txt */
    fprintf(fp,"%s ",ID1);/* input library ID in librarybooks.txt */
    fprintf(fp,"\n");
    fclose(fp);/* when use file over, should close it */
    system("pause");/* let result display on the screen before press any key to continue */
    system("cls");/* system("cls") clears whatever is display on the screen */
   }
   if(choose==2)/* same as choose==1 */
   {
    do
    {
     printf("Please enter library ID\n");
     gets(lib.library_ID);
     count=strlen(lib.library_ID);
    }
    while(count!=7);
```

```
    strncat(ID2,lib.library_ID,10);/* add library ID to 'EN' */
    fp=fopen("librarybooks.txt","a");
    if(fp==NULL){
    printf("Failed opening librarybooks.txt!\n");
    return;}
    fprintf(fp,"%s ",lib.book_name);
    fprintf(fp,"%s ",lib.author_name);
    fprintf(fp,"%s ",ID2);
    fprintf(fp,"\n");
    fclose(fp);
    system("pause");/* let result display on the screen before press any key to continue */
    system("cls");/* system("cls") clears whatever is display on the screen */
   }
   if(choose!=1&&choose!=2)/* when enter error,back to menu */
    {
     printf("Please enter 1 or 2!\n");
     system("pause");/* let result display on the screen before press any key to continue */
     system("cls");/* system("cls") clears whatever is display on the screen */
     return;
    }
}

void searchbook()/* declar a defined function searchbook */
{
    struct library_book lib;
    int i=0,j=0,count=0;/* - for counting */
    char aim[50];/* represents the name of searching book */
    FILE *fp;/* represents librarybooks.txt */
    char **bookname; /* for storing information in librarybooks.txt */
    fp=fopen("librarybooks.txt","r");/* open librarybooks.txt */
    if(fp==NULL)
    {
        printf("Open File error!");
    }
    else
    {
        while(!feof(fp))/* before go to the end of librarybooks.txt */
        {
            fscanf(fp,"%s",lib.book_name);
            count++;/* counting how many items in librarybooks.txt */
        }
        rewind(fp);/* Return to the outset of librarybooks.txt */
        bookname = (char**)malloc(sizeof(char*)*count);/* Request a block of memory of a give size for bookname */
        for (i=0; i<count; i++)
        bookname[i] = (char*)malloc(sizeof(char)*20); /* Request a block of memory of a give size for bookname[i] */

        for (i=0;i<count;i++)
        {
            fscanf(fp,"%s",lib.book_name);
            strcpy(bookname[i],lib.book_name);/* put each item in bookname[i] */
        }
        printf("Please follow 1 step\n");

        printf("************************************\n");
        printf("(1 Please enter seachered information\n");
        printf("************************************\n\n");/* tell user how to use this function */
        printf("There are %d books in library\n\n",count);/* display how many books in librarybooks.txt */
        printf("Please enter name of searching book\n");
        gets(aim);

       for (i=0;i<count;i++)
       {
         if(strcmp(aim,bookname[i])==0){/* compare each item in bookname[i] and aim. if they are the same, return 0 to strcmp */
           printf("%s %s %s\n",bookname[i],bookname[i+1],bookname[i+2]);}
         else
           j=j++;/* record number of searching times that bookname[i] is different with aim */
       }
       for (i=0; i<count; i++)
           free(bookname[i]);/* To release memory allocated with malloc() */
           free(lib.book_name); /* To release memory allocated with malloc() */
    }
         if(i==j)/* it mean that didn't find aim in librarybooks.txt */
         {
           printf("can't find this book\n");
           system("pause");/* let result display on the screen before press any key to continue */
           system("cls");/* system("cls") clears whatever is display on the screen */
           return;
         }
    fclose(fp);
    system("pause");/* let result display on the screen before press any key to continue */
    system("cls");/* system("cls") clears whatever is display on the screen */
}
```

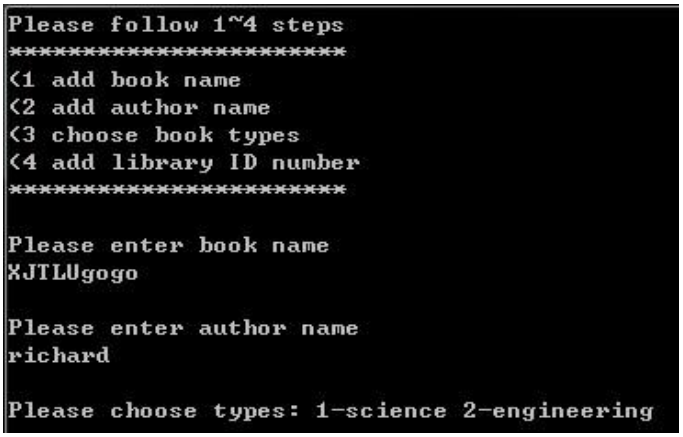
5. Testing:

The C program was tested by carrying out a set of experiments; and the C program output was verified successfully. For instance,

Menu



1- Add book



1 for science and 2 for engineering

When enter other number

```
Please follow 1~4 steps
************************
(1 add book name
(2 add author name
(3 choose book types
(4 add library ID number
************************

Please enter book name
XJTLUgogo

Please enter author name
richard

Please choose types: 1-science 2-engineering
3
Please enter 1 or 2!
Press any key to continue . . .
```

When enter 1 or 2

```
Please follow 1~4 steps
************************
(1 add book name
(2 add author name
(3 choose book types
(4 add library ID number
************************

Please enter book name
XJTLUgogo

Please enter author name
richard

Please choose types: 1-science 2-engineering
1
Please enter library ID (7 digits)
```

When enter error digits

```
Please choose types: 1-science 2-engineering
1
Please enter library ID (7 digits)
12345678
Please enter library ID (7 digits)
```

When enter right digits

```
Please choose types: 1-science 2-engineering
1
Please enter library ID (7 digits)
12345678
Please enter library ID (7 digits)
1234567
Press any key to continue . . .
```

succeed

Add to librarybooks.txt end

```
XJTLU aiai richard EN1234567
xixixixixi richard EN1234567
XJTLUgogo richard Sc1234567
```

No space between Sc || EN and ID number

2- Search book

Little program to add random number which represents book information

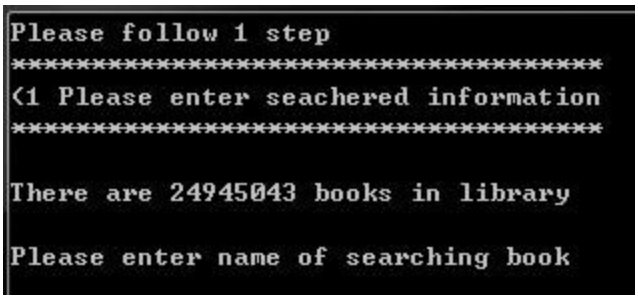
add information

```
#include <stdio.h>
#include <string.h>
#include <time.h>
#include <stdlib.h>
main()
{
int i;
char a[50];
char b[20];
char c[11];
FILE*fp;
srand((unsigned)time(NULL));
fp=fopen("librarybooks.txt","a");
for(i=0;i<9999;i++){
a[i]=(rand()%(1000000)+(0));
b[i]=(rand()%(10000)+(0));
c[i]=(rand()%(1000000)+(0));
fprintf(fp,"%d ",a[i]);
fprintf(fp,"%d ",b[i]);
fprintf(fp,"%d ",c[i]);
fprintf(fp,"\n");
}
fclose(fp);
}
```

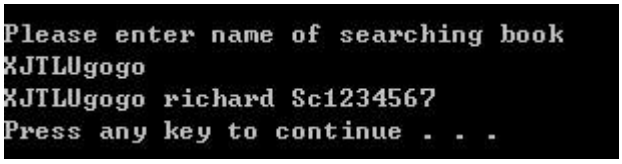
My librarybooks.txt

librarybooks	12/11/2014 7:50 PM	Text Document	105,147 KB

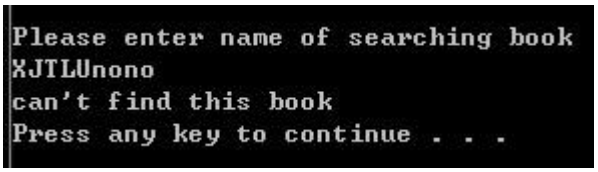
Need almost 10s to open this file when choose 2 to search book because librarybooks.txt is large



Enter existing book name



Enter not existing book name



3- Quit