1. Implementation Code (Main function)

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
int count=0;
struct{
   int id;
   int quantity;
   char name[30];
   }book[15],sort[15];
   // function prototype
void Menu ();
void insert_book();
void delete_a_book_by_id();
void search_a_book_by_id(int c,int id);
void search_a_book_by_name_binarysearch(char nob[],int high,int low);
void Display_all_books_sorted_by_name();
void Display_all_books_unsorted();

int main()
{
// read information from the file
   int i=0,j;
    FILE *fptr;
    fptr=fopen("E:\\library.txt","r");
    if(fptr==NULL)
      printf("File could not be open");
    else
    {
       while(!feof(fptr))
       {
         fscanf(fptr,"%d",&book[i].id);
         fgets(book[i].name,30,fptr);
         fscanf(fptr,"%d",&book[i].quantity);
         i++;
```

```
      }
    }
     count=i-1;
//repeat function gets
     for(i=0;i<count;i++)
     {
       for(j=0;book[i].name[j]!='\n';)
          j++;
       book[i].name[j]='\0'; }
     fclose(fptr);
    Menu();
//put information in file
   FILE *pt;
     pt=fopen("E:\\library.txt","w");
     if(pt==NULL)
        printf("file could not be open");
     else
     {
      for(j=0;j<count;j++)

fprintf(pt,"%d%s\n%d\n",book[j].id,book[j].name,book[j].quantity);
     }
     fclose(pt);
// End
    return 0;
}

//  menu
void Menu (){
   int  x=1,ID,i;
   char choice[3],NAME[30];
   int  num;
   while(x)
     {
        printf("Welcome to the library\n\n1-Insert a book.\n2-Delete a book
by id.\n3-Search a book by id.\n4-Search a book by name.\n5-Display all
books sorted by name.\n6-Display all books unsorted.\n");
        printf("====================================\n");
        printf("Enter your choice[1..6]: \n");
```

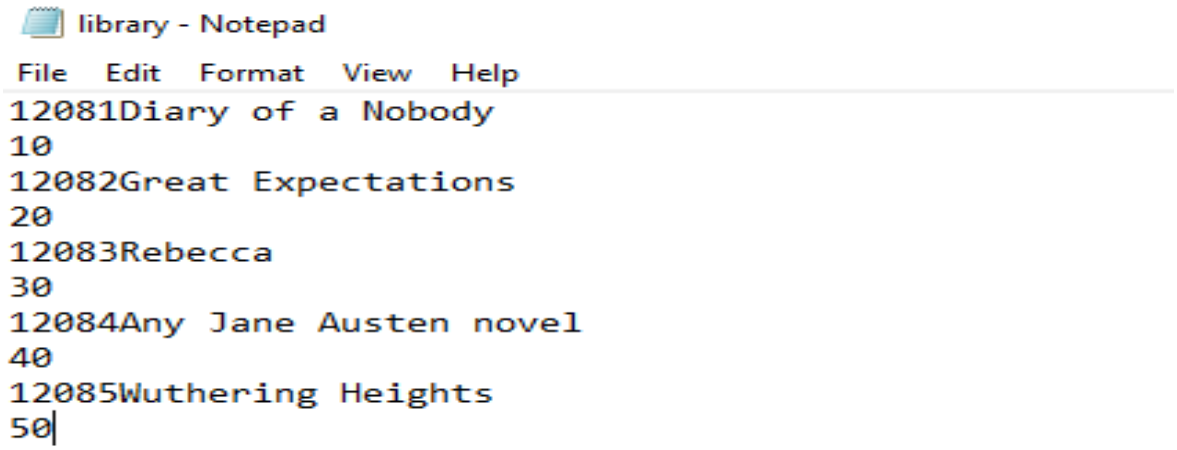
```
        scanf("%d",&num);
        if(num>6||num<=0) break;
       switch(num)
        {
        case 1:
          insert_book();
          break;
        case 2:
           delete_a_book_by_id();
          break;
        case 3:
           printf("Enter the id of the book : \n");
           scanf("%d",&ID);
           search_a_book_by_id(count,ID);
          break;
        case 4:
          Display_all_books_sorted_by_name();
          printf("Enter the name of the book :\n");
          fflush(stdin);
          gets(NAME);
          search_a_book_by_name_binarysearch(NAME,count-1,0);
          break;
        case 5:
           Display_all_books_sorted_by_name();
           for(i=0;i<count;i++)
              printf("%-10d%-
25s%d\n",sort[i].id,sort[i].name,sort[i].quantity);
          break;
        case 6:
           Display_all_books_unsorted();
          break;
        }
        printf("You want to choice again ?! \n");
        scanf("%s",choice);

if((choice[0]=='N'||choice[0]=='n')&&(choice[1]=='O'||choice[1]=='o'))
           x--;
        else
          system("cls");   } }
```

2. Function Codes and their corresponding Screenshots of Output Screen

a. Create a text file

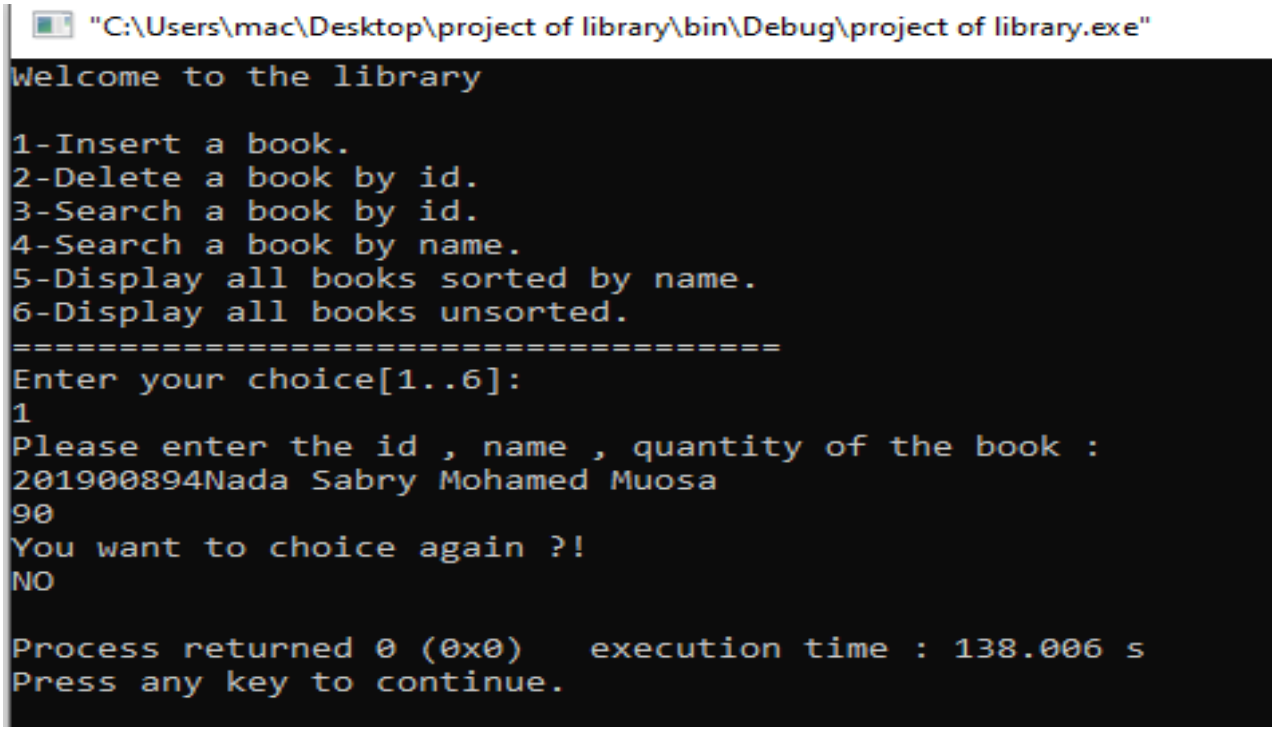
Screenshot of the text file including some books with their ids, name, and quantity)

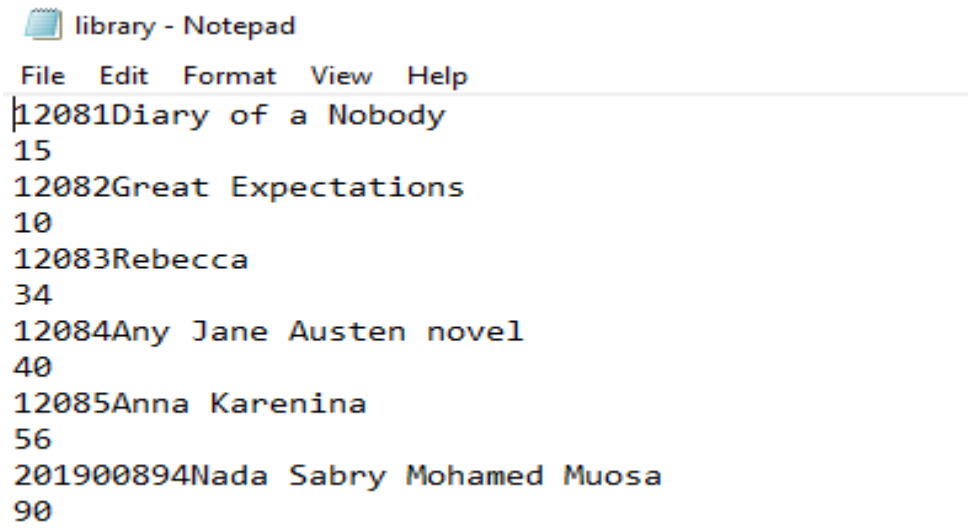


b. Insert a book

The function and the screenshot of output screen (Insert your id, complete name, any grade as a book), text file after insertion.

```
void insert_book()
{
   printf("Please enter the id , name , quantity of the book :\n");
   scanf("%d",&book[count].id);
   gets(book[count].name);
   scanf("%d",&book[count].quantity);
   count++;
}
```



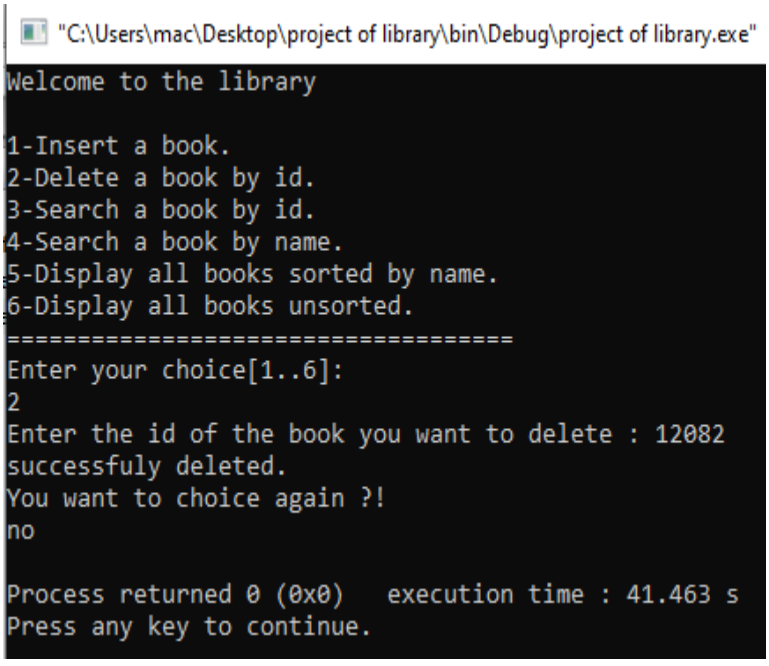


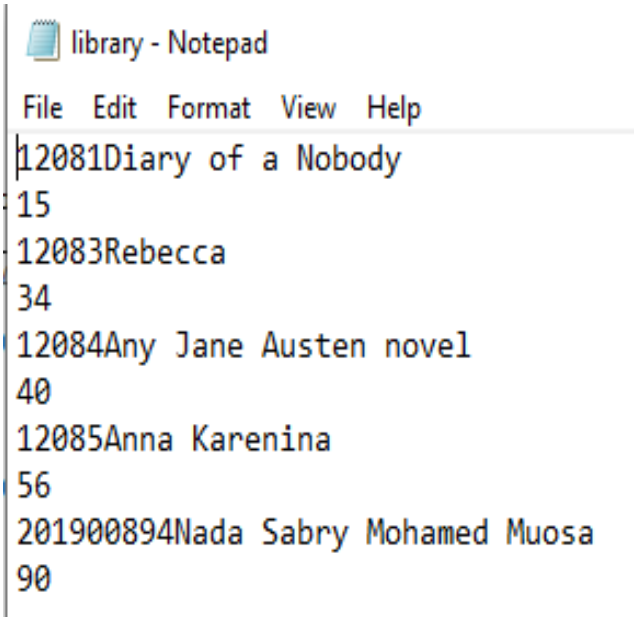
c. Delete a book by id

The function and the screenshot of output screen (choose an id to delete, but not your id), text file after deletion.

```
void delete_a_book_by_id()
{
  int id,x,i,j,z;
  printf("Enter the id of the book you want to delete : ");
  scanf("%d",&id);
  z=count;
  for(i=0;i<count;i++)
  {
```

```
        if(id==book[i].id){
          for(j=i;j<count-1;j++)
          {
            book[j].id=book[j+1].id;
            book[j].quantity=book[j+1].quantity;
            for(x=0;book[j].name[x]!='\0';x++)
                  book[j].name[x]='\0';
            strcpy(book[j].name,book[j+1].name);
          }
        count--;
        break;
    }
}
if(count==z)
   printf("Not Found\n");
else
    printf("successfuly deleted.\n");
}
```



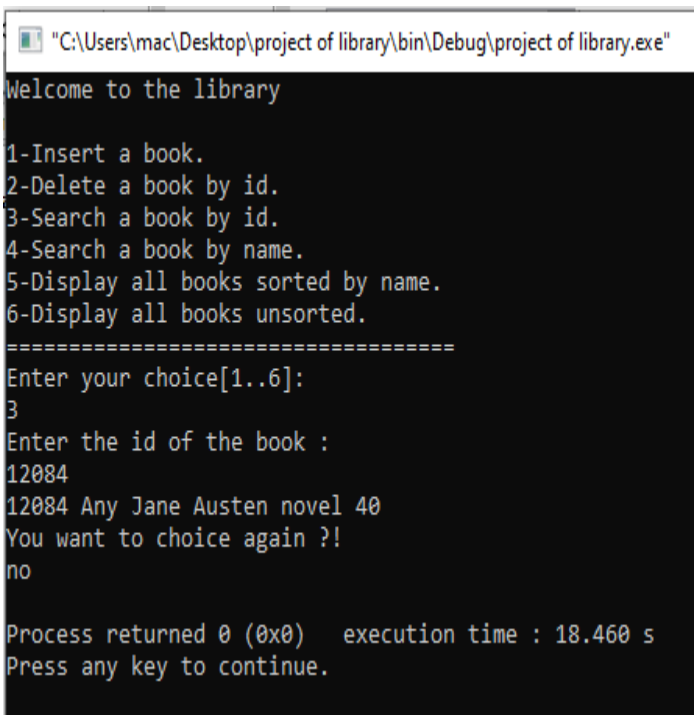


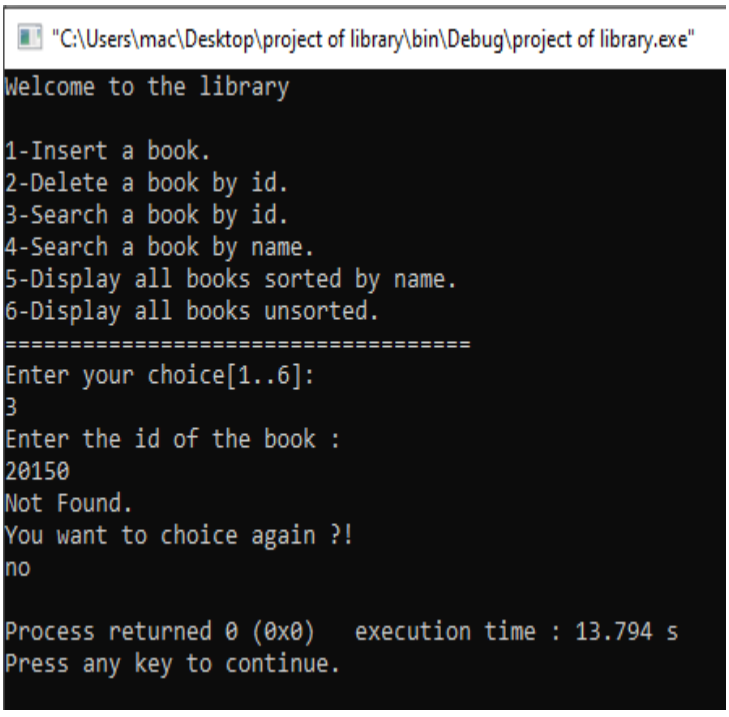
d. Search a book by id and display its name and quantity using linear search recursively. If not exist, display "Not found".

The function and the screenshot of output screen

- Choose ***your id*** to display.
- Choose an ***id that doesn't exist*** in your file.

```
void search_a_book_by_id(int c,int id)
{ static y=0;
   if(y==c){
      printf("Not Found.\n");
      return;}
   else if(book[y].id==id)
   {
      printf("%d %s
%d\n",book[y].id,book[y].name,book[y].quantity);
      return;
   }
   else{
      y++;
      search_a_book_by_id(c,id);
   }
}
```



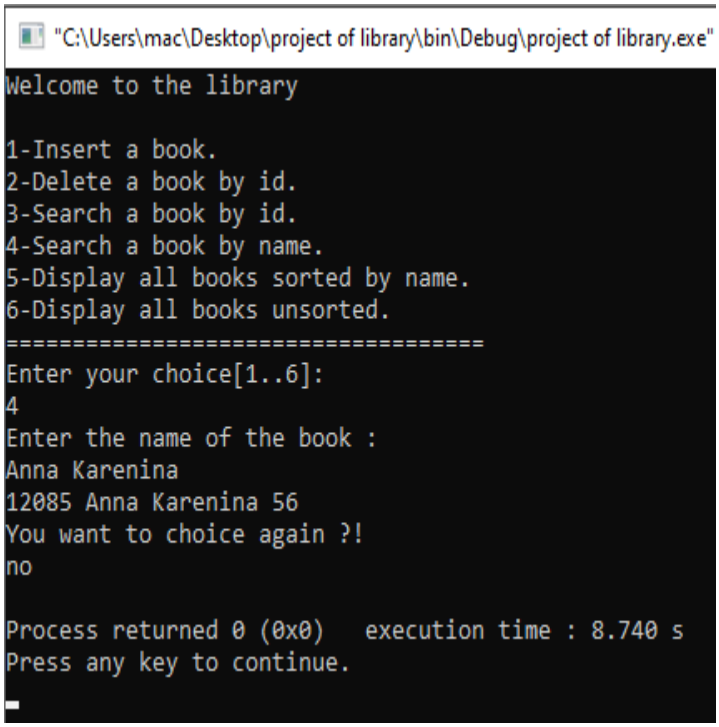


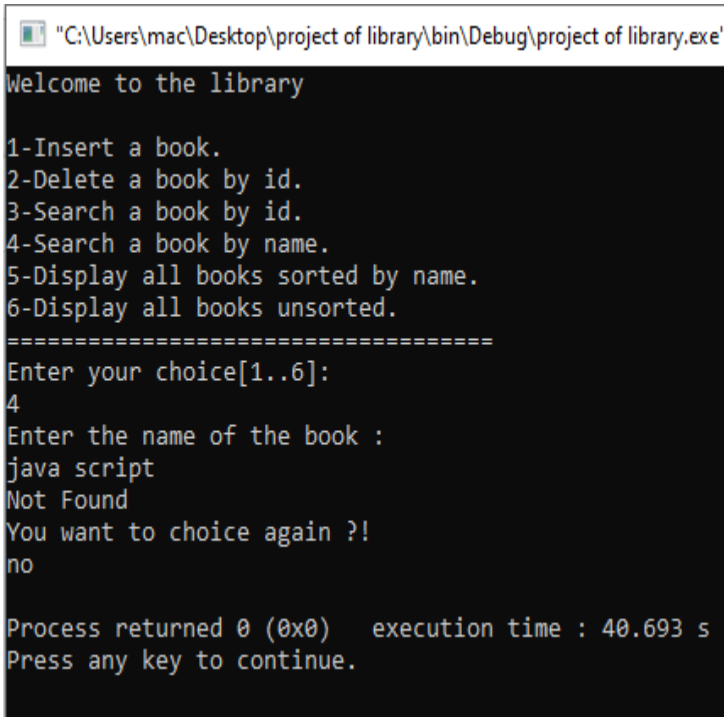
e. Search a book by name and display its id and quantity using binary search. If not exist, display "Not found".

The function and the screenshot of output screen

- Choose an ***id that exist*** to display.
- Choose an ***id that doesn't exist*** in your file.

```
void search_a_book_by_name_binarysearch(char nob[],int high,int
low)
{
   int result;
  int middle=(low+high)/2;
  result=strcmp(nob,sort[middle].name);
  if(low>high)
   {printf("Not Found\n");return;}
  if(result==0)
   {printf("%d %s
%d\n",sort[middle].id,sort[middle].name,sort[middle].quantity);retur
n;}
  else if(strcmp(nob,sort[middle].name)>0)
   return search_a_book_by_name_binarysearch(nob,high,middle+1);
  else
   return search_a_book_by_name_binarysearch(nob,middle-1,low);
}
```

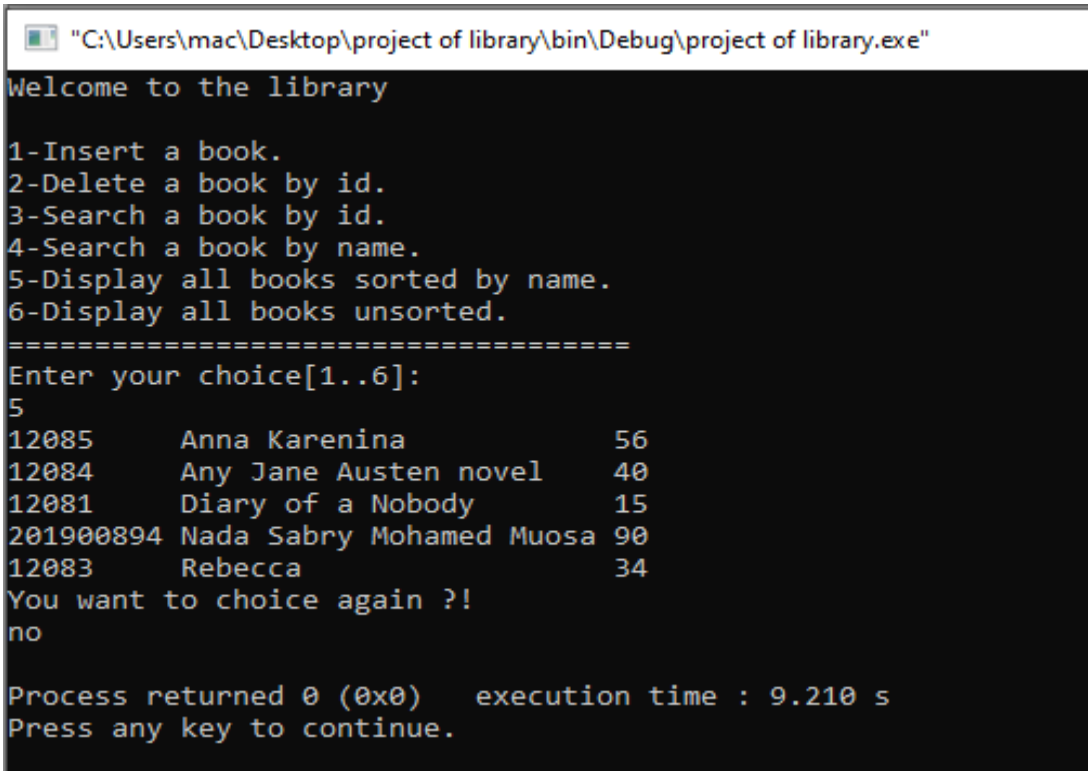




f. Display all books sorted by name, and their corresponding ids and quantity.

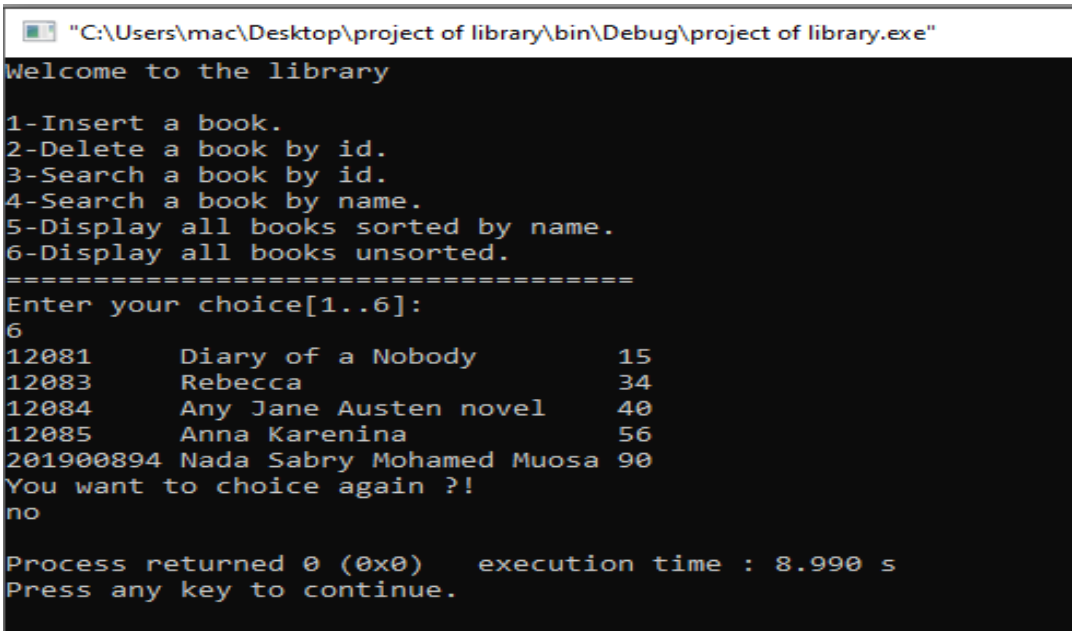
The function and the screenshot of output screen of all sorted books including your name.

```
void Display_all_books_sorted_by_name()
{ int i,j,t;
  char tmp[30];
  for(i=0;i<count;i++)
  {
    sort[i].id=book[i].id;
    sort[i].quantity= book[i].quantity;
    strcpy( sort[i].name,book[i].name); }
 for(i=0;i<count;i++)
  {
    for(j=i+1;j<count;j++)
    {
       if(strcmp(sort[i].name,sort[j].name)>0)
       {
          strcpy(tmp,sort[i].name);
          strcpy(sort[i].name,sort[j].name);
          strcpy(sort[j].name,tmp);
          t=sort[i].id;
          sort[i].id=sort[j].id;
          sort[j].id=t;
          t=sort[i].quantity;
          sort[i].quantity=sort[j].quantity;
          sort[j].quantity=t;
       }
    }
  }
}
```



g. Display all books unsorted, their ids, names and quantity (as entered)
The function and the screenshot of output screen of all unsorted books including your name.

```
void Display_all_books_unsorted()
{
   int i;
   for(i=0;i<count;i++)
      printf("%-10d%-
25s%d\n",book[i].id,book[i].name,book[i].quantity);
}
```



h. Ask if you want another operation

The screenshot of output screen when you ask the user if he wants another operation, reply one time by ‘yes’ and another time by ‘no’.