19BIT0292

Bhaumik Tandan

DIGITAL ASSIGNMENT-2

DATA STRUCTURES AND ALGORITHMS LABORATORY

CSE2011

L57+L58

Q1) Create a linked list Swap two adjacent elements by

(a) Interchanging the elements itself.
(b) Adjusting only the pointers (and not data)

In both cases, assume that elements are stored in single linked list and also with double linked list.

(a) swap_item.h

CODE

```
#pragma once

void swap_item(struct node* p,struct node* q)
{
   int t=p->d;
   p->d=q->d;
   q->d=t;
}
```

(b) swap_pointer.h

CODE

```
#pragma once

void swap_next(struct node* p,struct node* q)
{
   struct node* t =q->n;
   q->n=p->n;
   p->n=t;
}

void swap_pointer(struct node* p,struct node* q)
```

```
{
   struct node* t=p->n;
   p->n=q->n;
   q->n=t;

   swap_next(p->n,q->n);
}

void swap_pointer_change_head(struct node* q)
{
   struct node* t=h;
   h=q->n;
   q->n=t;
   swap_next(h,q->n);
}
```

swap_pointer_dll.h

```
#pragma once

void swap_prev(struct node *p,struct node *q)
{
   struct node* t =q->p;
   q->p=p->p;
   p->p=t;
}

void swap_pointer_dll(struct node *p,struct node *q)
{
   q->p->n=p;

   if(p==h)
      h=q;

   else
   p->p->n=q;

   if(q->n)
   q->n->p=p;

   p->n->p=q;

   swap_next(p,q);
```

```
    swap_prev(p,q);
}
```

single_ll.c

CODE

```
#include "ll.h"//Click here for the souce code of ll.h
#include "swap_item.h"//Click here for the souce code of swap_item.h
#include "swap_pointer.h"//Click here for the source code of swap_pointer.h
#include "get_node.h"//Click here for the souce code of get_node.h
#include "swap.h"//Click here for the souce code of swap.h
#include "menu.h"//Click here for the souce code of menu.h
main()
{
    takeinput();
    menu();
}
```

OUTPUT

Swapping items

```
Enter the length of the number of elements: 5
Enter the elements: 1 2 3 4 5


1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 4

1->2->3->4->5->(null)

1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 5

240880->240912->165040->165072->470496->(null)

1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 2

Enter the node positions to be exchanged(space seperated): 1 4

The new linked list elements are:-
4->2->3->1->5->(null)
The new linked list address are:-
240880->240912->165040->165072->470496->(null)
```

Swapping pointers

```
1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 1

Enter the length of the number of elements: 6
Enter the elements: 10 20 30 40 50 60


1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 5

470720->470752->470784->470816->470848->470880->(null)

1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 3

Enter the node positions to be exchanged(space seperated): 6 1

The new linked list elements are:-
60->20->30->40->50->10->(null)
The new linked list address are:-
470880->470752->470784->470816->470848->470720->(null)

1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 6
PS C:\Users\bhaum\OneDrive\Desktop\DSA CODES\linked list\linked
```

double_ll.c

CODE

#include "../doubly_linked_list.h"//Click here for the souce code of doubly_linked_list.h
#include "swap_item.h"//Click here for the souce code of swap_item.h
#include "swap_pointer.h"//Click here for the source code of swap_pointer.h
#include "get_node.h"//Click here for the souce code of get_node.h
#include "swap_pointer_dll.h"//Click here for the souce code of swap_pointer_dll.h
#include "swap.h"//Click here for the souce code of swap.h
#include "menu.h"//Click here for the souce code of menu.h

```
main()
{
   takeinput();
   menu();
}
```

OUTPUT

Swapping items

```
Enter the length of the number of elements: 5
Enter the elements: 1 2 3 4 5


1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 2

Enter the node positions to be exchanged(space seperated): 3 5

The new linked list elements are:-
(null)<-1<->2<->5<->4<->3->(null)
The new linked list address are:-
(null)<-240880<->240912<->165040<->165072<->470496->(null)
```

Swapping pointers

```
1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 5 1

(null)<-240880<->240912<->165040<->165072<->470496->(null)
```

Display linked list

```
Enter the length of the number of elements: 5
Enter the elements: 1 2 3 4 5


1)Set linked list
2)Swap items
3)Swap pointers
4)Display linked list
5)Display address
6)Exit
 Enter your choice: 4

(null)<-1<->2<->3<->4<->5->(null)
```

CLICK HERE FOR GITHUB LINK OF WHOLE SOURCE CODE

Q2) Given single circular linked list containing a set of data. Obtain the following from this data structure:

(a) Reverse the direction of links

(b) Forgiven two elements in the list, find the distance (number of nodes) between them.

cll.h

CODE

```
#include<stdio.h>
#include<stdlib.h>

#pragma once
struct node
{
   int d;
   struct node *n;
} * l;

typedef struct node node;

void ins(int n)
{
   node *t = (node *)malloc(sizeof(node));
   t->d = n;

   if (!l)
   {
      l = t;
      t->n = l;
      return;
   }

  t->n=l->n;
  l->n=t;
  l=t;
}

void disp()
{
   printf("\n-");
```

```
    node *t = l;
    int i=0;

    do
    {
       t = t->n;
       printf("->%d",t->d);
       i++;
    }
    while(t!=l);

    printf("--\n|");
    for(int j=0;j<i*3+2;j++)
    printf("_");
    printf("|");

}

void takeinput()
{
    int n;
    printf("\nEnter the number of elements: ");
    scanf("%d",&n);
    printf("Enter the elements: ");

    for (int i = 0; i < n; i++)
    {
       int t;
       scanf("%d",&t);
       ins(t);
    }
}
```

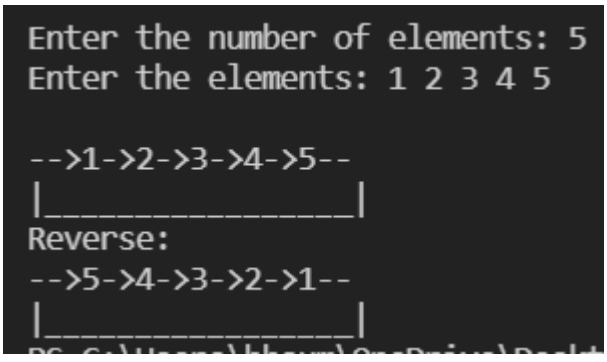
(a) reverse.h

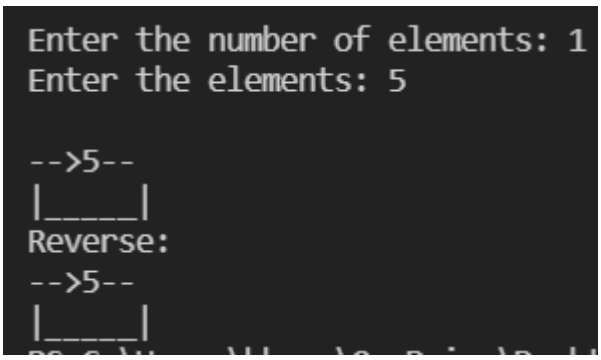
CODE

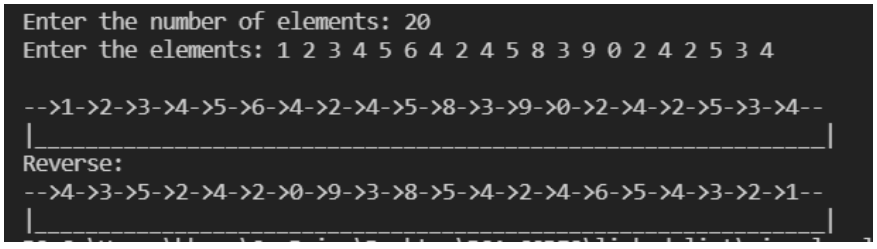
```
#include "cll.h"

void reverse()
{
   if(!l)
   {
      printf("\nList empty");
      return;
```

```
    }

    node *prev=l,*c=l->n;
    l=c;

    do
    {
       node* t=c->n;
       c->n=prev;
       prev=c;
       c=t;
    }
    while(c!=l);

}

main()
{
    takeinput();
    disp();
    reverse();
    printf("\nReverse: ");
    disp();
}
```

OUTPUT





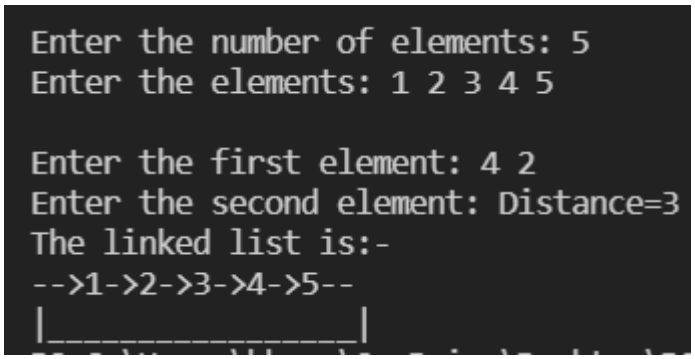


(b) distance.h

CODE

```
#include "cll.h"

void distance()
{
   if(!l)
   {
      printf("\nList empty");
      return;
   }

   printf("\nEnter the first element: ");
   int f,s;
   scanf("%d",&f);
   printf("Enter the second element: ");
   scanf("%d",&s);

   node* p=l;

   while(p->d!=f)
   p=p->n;

   f=0;

   while(p->d!=s && ++f)//used to increment as well && does not goes
to 2nd if first found false
      p=p->n;

   printf("Distance=%d",f);
}

main()
{
   takeinput();
   distance();
   printf("\nThe linked list is:-");
   disp();
}
```

OUTPUT



CLICK HERE FOR GITHUB LINK OF WHOLE SOURCE CODE

Q3) It is required to maintain a library database using a number of lists as mentioned: The list BOOKS contains the information like title, accession number, and tag field (to indicate whether a book is issued or not) for all the books in a library. Note that each book can be there in multiple copies, but there accession numbers are different. Another list SUBSCRIBERS will contain the name, borrower number and the list of books (with date of issues) he has issued. Assume that a subscriber can issue up to five books at the most and no two copies of the same book.

(a) Design a suitable data structure using single linked list.

header_function.h

CODE

```
#pragma once

#include<stdbool.h>
#include<string.h>
#include <termios.h>
#include <unistd.h>


struct  book_node
{
   char title[30];
   int accession_number;
   bool tag;
   struct  book_node *n;
}*head_book;

typedef struct book_node book_node;


struct  subscriber_node
{
   char name[30];
   int borrower_number;
```

```
    char password[30];
    book_node *book[5];
    char date[5][20];
    int number;
    struct  subscriber_node *n;
}*head_subscriber;

typedef struct subscriber_node subscriber_node;

//global functions
void admin();
void borrower();
void print_book_direct(subscriber_node* p);
void print_book(int borrower_number);
void disp_avail_book();
void print_subscriber(char book[30]);
void issue_directly(subscriber_node* p,char book[30],char date[30]);
void issue_book(char book[30],int borrower_number,char date[30],char
pas[30]);
void return_directly(subscriber_node* p,char book[30]);
void return_book(char book[30],int borrower_number,char pas[30]);
void add_book(char title[30],int accession_number);
book_node* set_tag(char book[30],bool tag);
void home();
subscriber_node* subscriber_exists(int borrower_number,char
password[30]);
void add_subscriber(char name[30],int borrower_number,char
password[30]);


#include "book.h"
#include "suscriber.h"
#include "./operation/return.h"
#include "./operation/issue.h"
#include "./operation/display/print_books.h"
#include "./operation/display/subscriber_of_book.h"


void __attribute__((constructor))  setup();
#include "./default_opearations/setup.h"//contains default operation
```

book.h

CODE

```
#pragma once

void add_book(char title[30],int accession_number)
{
    book_node *t=(book_node*)malloc(sizeof(book_node));
    strcpy(t->title,title);
    t->accession_number=accession_number;
```

```
    t->tag=0;
    t->n=0;

    if(!head_book)
    {
        head_book=t;
        printf("\n Book \"%s\" added with accession number
%d",title,accession_number);
        return;
    }

    book_node *p=head_book;

    while(p->n)
    if(p->accession_number==accession_number)//check if the accession
number is unique or not
    {
        printf("\n Accession number %d already assigned with
book \"%s\"",accession_number,p->title);
        return;
    }
    else
    p=p->n;

    if(p->accession_number==accession_number)
    {
        printf("\n Accession number %d already assigned with
book \"%s\"",accession_number,p->title);
        return;
    }

    printf("\n Book \"%s\" added with accession number
%d",title,accession_number);
    p->n=t;
}

book_node* set_tag(char book[30],bool tag)//return 1 on successful
change
{

    book_node *p=head_book;
    while(p && !(strcmp(p->title,book)==0 && p->tag!=tag))//find the
book wit book number and not isssued
    p=p->n;

    if(!p)
        return 0;

    p->tag=tag;

    return p;
}
```

subscriber.h

CODE

```
#pragma once

subscriber_node* subscriber_exists(int borrower_number,char
password[30])
{
   subscriber_node *p=head_subscriber;

   while(p && !(p->borrower_number==borrower_number &&
strcmp(p->password,password)==0))
   p=p->n;

   return p;
}

void add_subscriber(char name[30],int borrower_number,char
password[30])
{
   subscriber_node
*t=(subscriber_node*)malloc(sizeof(subscriber_node));
   strcpy(t->name,name);
   t->borrower_number=borrower_number;
   strcpy(t->password,password);
   t->number=0;
   t->n=0;

   if(!head_subscriber)
   {
      head_subscriber=t;
      printf("\n Borrower %s with borrower number %d
added",name,borrower_number);
      return;
   }

    subscriber_node *p=head_subscriber;

   while(p->n)
   if(p->borrower_number==borrower_number)//check if the borrower
number is unique or not
   {
      printf("\n Borrower number %d already exists with
%s",borrower_number,p->name);
      return;
   }
   else
   p=p->n;

   if(p->borrower_number==borrower_number)
   {
      printf("\n Borrower number %d already exists with
```

```
%s",borrower_number,p->name);
        return;
    }

    printf("\n Borrower %s with borrower number %d
added",name,borrower_number);

    p->n=t;
}
```

(b) Write a menu driven program using C for the following:

(i) To issue a book

/operation/issue.h

CODE

```
#pragma once

void issue_directly(subscriber_node* p,char book[30],char date[20])
{
    if(p->number==5)
    {
        printf("\n %s has exceeded the limit of issuing
books",p->name);//indicate limit exceeds
        return;
    }

    for(int i=0;i<p->number;i++)
    if(strcmp(p->book[i]->title,book)==0)// indicate book already issued
    {
        printf("\n %s has already issued %s with accession number %d",p-
>name,book,p->book[i]->accession_number);
        return;//indicate book already issued
    }

    book_node *issue=set_tag(book,1);

    if(!issue)
    {
        printf("\n %s is not available in the library",book);
        return;
    }

    int n=p->number;
    p->number=n+1;

    p->book[n]=issue;
```

```
    strcpy(p->date[n],date);

    printf("\n %s with accession number %d has been issued to %s with
borrower number %d on %s",book,p->book[n]->accession_number,p-
>name,p->borrower_number,date);
}

void issue_book(char book[30],int borrower_number,char date[10],char
pas[30])
{
    subscriber_node *p=subscriber_exists(borrower_number,pas);

    if(!p)
    printf("\nWrong credentials");
    else
    issue_directly(p,book,date);

}
```

(ii) Return a book

/operation/return.h

CODE

```
#pragma once

void return_directly(subscriber_node* p,char book[30])
{
    if(p->number==0)
    {
        printf("\n %d) %s has no pending returns",p->borrower_number,p-
>name);//indicate no book
        return;
    }

    int f=-1;
    for(int i=0;i<p->number;i++)
    if(strcmp(p->book[i]->title,book)==0)// indicate book issued
    {
        f=i;
        break;
    }

    if(f==-1)
    {
        printf("\n %d) %s has not issued book %s",p->borrower_number,p-
>name,book);//indicate no book
        return;
    }

    p->book[f]->tag=0;//set to not issued
```

```
    printf("\n %s with accession number %d has been returned by %s with
borrower number %d which was issued on %s",
    book,p->book[f]->accession_number,p->name,p-
>borrower_number,p->date[f]);

    p->book[f]=0;
    for(int i=f;i<p->number-1;i++)
    {
        p->book[i]=p->book[i+1];
        strcpy(p->date[i],p->date[i+1]);
    }

    p->number-=1;
}

void return_book(char book[30],int borrower_number,char pas[30])
{
    subscriber_node
*p=subscriber_exists(borrower_number,pas);//imported in login

    if(!p)
    printf("\nWrong credentials");
    else
    return_directly(p,book);
}
```

(iii) Show the list of books issued by a subscriber

/operation/display/print_book.h

CODE

```
#pragma once

void print_book_direct(subscriber_node* p)
{
    if(p->number==0)
    {
        printf("\n %d) %s has not issued any book",p->borrower_number,p-
>name);
        return;//indicate book not found
    }

    printf("\n Books issued by %s are: ",p->name);

    for(int i=0;i<p->number;i++)
    printf("\n  %d) %s on %s",p->book[i]->accession_number,p->book[i]-
>title,p->date[i]);
}

void print_book(int borrower_number)
```

```
{
   subscriber_node *p=head_subscriber;

   printf("\n");

   while(p && p->borrower_number!=borrower_number)
   p=p->n;

   if(!p)
   printf("\n %d does not exists in the record",borrower_number);

   else
   print_book_direct(p);

}

void disp_avail_book()
{
   bool f=0;
   printf("\n\n Available books in the library are:-");

   book_node *p=head_book;

   while(p)
   {
      if(p->tag==0)
      {
         printf("\n  %d)%s",p->accession_number,p->title);
         f=1;
      }

      p=p->n;
   }

   if(!f)
   printf("\n  No books available");
}
```

(iv) Given a title, find out to whom it has issued.

/operation/display/subscriber_of_book.h

CODE

```
#pragma once


void print_subscriber(char book[30])
{
```

```
    subscriber_node *p=head_subscriber;

    printf("\n\n %s has been issued by: ",book);

    int f=-1;

    while(p)
    {
       for(int i=0;i<p->number;i++)
       if(!strcmp(p->book[i]->title,book))
       {
          printf("\n  %d) %s on %s: %d",p->borrower_number,p->name,p-
>date[i],p->book[i]->accession_number);
          f=1;
          break;
       }
       p=p->n;
    }

    if(f==-1)
    printf("\n  No borrowers found");
}
```

For the purpose of test the bellow code has been added

default_operations/setup.h

```
  #pragma once
void setup() {//it runs automatically

  #include "header.h"
  add_default_subscribers();
    add_default_books();
    add_default_issue();

    // bellow all are just for testing
    test_return_operations();
    test_display_book_operation();
    test_subscriber_disp_operation();

    printf("\n----------Default operations ended----------");
}
```

default_operations/subscriber.h

```
#pragma once

void add_default_subscribers()
{
   printf("\n");
   add_subscriber("Rohan Gupta",221,"1234");
   add_subscriber("Bhaumik Tandan",292,"1234");
   add_subscriber("Rohan Gupta",321,"1234");
   add_subscriber("Rahul Misra",221,"1234");//will not get added
   add_subscriber("Naman Sethi",121,"1234");
}
```

Duplicate borrower number will not get added

```
Borrower Rohan Gupta with borrower number 221 added
Borrower Bhaumik Tandan with borrower number 292 added
Borrower Rohan Gupta with borrower number 321 added
Borrower number 221 already exists with Rohan Gupta
Borrower Naman Sethi with borrower number 121 added
```

default_operations/books.h

```
#pragma once

void add_default_books()
{
   printf("\n");
   add_book("Three men in a boat",1564);
   add_book("7 ages",9564);
   add_book("Macbeth",3253);
   add_book("End game",1014);
   add_book("Dance of dragons",1024);
   add_book("Invisible Man",1114);
   add_book("Dead end",1814);
   add_book("Clash of Kings",7814);
   add_book("Three men in a boat",1564);//will not get added
    add_book("7 ages",1264);//will get added as accession number changed
   add_book("Three men in a boat",1253);
   add_book("7 ages",1164);
   add_book("End game",1019);
   add_book("Macbeth",9564);//will not get added
   add_book("Invisible Man",1214);
   add_book("End game",1004);
}
```

Duplicate accession number will not get added

```
Book "Three men in a boat" added with accession number 1564
Book "7 ages" added with accession number 9564
Book "Macbeth" added with accession number 3253
Book "End game" added with accession number 1014
Book "Dance of dragons" added with accession number 1024
Book "Invisible Man" added with accession number 1114
Book "Dead end" added with accession number 1814
Book "Clash of Kings" added with accession number 7814
Accession number 1564 already assigned with book "Three men in a boat"
Book "7 ages" added with accession number 1264
Book "Three men in a boat" added with accession number 1253
Book "7 ages" added with accession number 1164
Book "End game" added with accession number 1019
Accession number 9564 already assigned with book "7 ages"
Book "Invisible Man" added with accession number 1214
Book "End game" added with accession number 1004
```

default_operations/issue.h

```
#pragma once

void add_default_issue()
{
   printf("\n");
   issue_book("Macbeth",221,"12/2/2013","1234");
   issue_book("Three men in a boat",221,"13/3/2013","1234");
   issue_book("7 ages",221,"13/3/2013","1234");
   issue_book("Invisible Man",221,"13/9/2013","1234");

   issue_book("Three men in a boat",221,"13/3/2013","1234");//will not get issued as already issued
   issue_book("Three men in a boat",292,"13/3/2013","1234");//will get issued by different person
   issue_book("Dead end",292,"23/3/2013","1234");

   issue_book("End game",221,"13/3/2014","1234");
   issue_book("Clash of Kings",221,"23/3/2013","1234");//will not get issued as limit exeded

   issue_book("Last supper",292,"23/3/2013","1234");//will not get issued as book not available

   issue_book("Dead end",321,"23/3/2013","1234");//will not get issued as book not available as we only had one copy already issued by 292
}
```

Out of stock book will not get issued, than 5 books will not get issued and the book which is not in library record will show not available

```
Macbeth with accession number 3253 has been issued to Rohan Gupta with borrower number 221 on 12/2/2013
Three men in a boat with accession number 1564 has been issued to Rohan Gupta with borrower number 221 on 13/3/2013
7 ages with accession number 9564 has been issued to Rohan Gupta with borrower number 221 on 13/3/2013
Invisible Man with accession number 1114 has been issued to Rohan Gupta with borrower number 221 on 13/9/2013
Rohan Gupta has already issued Three men in a boat with accession number 1564
Three men in a boat with accession number 1253 has been issued to Bhaumik Tandan with borrower number 292 on 13/3/2013
Dead end with accession number 1814 has been issued to Bhaumik Tandan with borrower number 292 on 23/3/2013
End game with accession number 1014 has been issued to Rohan Gupta with borrower number 221 on 13/3/2014
Rohan Gupta has exceeded the limit of issuing books
Last supper is not available in the library
Dead end is not available in the library
```

default_operations/return.h

```
#pragma once

void test_return_operations()
{
   printf("\n");
    return_book("Three men in a boat",121,"1234");//will not return as he has not issued any book
    return_book("Clash of Kings",221,"1234");//will not return as he has not issued the given book
   return_book("Dead end",292,"1234");
}
```

Borrower which does not have pending return and borrowers which do not issue that particular book will display message

```
121) Naman Sethi has no pending returns
221) Rohan Gupta has not issued book Clash of Kings
Dead end with accession number 1814 has been returned by Bhaumik Tandan with borrower number 292 which was issued on 23/3/2013
```

default_operations/disp_book.h

```
#pragma once

void test_display_book_operation()
{
   printf("\n");
   print_book(221);
   print_book(121);//will print not issued message
}
```

User who has not issued any book message will be displayed

```
Books issued by Rohan Gupta are:
 3253) Macbeth on 12/2/2013
 1564) Three men in a boat on 13/3/2013
 9564) 7 ages on 13/3/2013
 1114) Invisible Man on 13/9/2013
 1014) End game on 13/3/2014

121) Naman Sethi has not issued any book
```

default_operations/disp_subscriber.h

```
#pragma once

void test_subscriber_disp_operation()
{
```

```
        printf("\n");
        print_subscriber("Three men in a boat");
        print_subscriber("Dance of dragons");
    }
```

Books that does not have subscriber then menu gets displayed

```
 Three men in a boat has been issued by:
  221) Rohan Gupta on 13/3/2013: 1564
  292) Bhaumik Tandan on 13/3/2013: 1253

 Dance of dragons has been issued by:
  No borrowers found
--------------------Default operations ended-----------------------

1)Login as borrower
2)Login as Admin
3)Exit
Enter your choice:
```

Menu demo

```
--------------------Default operations ended-----------------------

1)Login as borrower
2)Login as Admin
3)Exit
Enter your choice: 2

 Enter the login username: bhaumik.tandan
  Enter the Password: *********


1)Add book
2)Add borrower
3)Display all the available books
4)Display the list of all the subscribers
5)Get the list of books issued by a borrower
6)Get the list of borrowers of a book
7)Go to Home Page
8)Exit
Enter your choice: 3


 Available books in the library are:-
  1024)Dance of dragons
  1814)Dead end
  7814)Clash of Kings
  1264)7 ages
  1164)7 ages
  1019)End game
  1214)Invisible Man
  1004)End game
```

```
1)Add book
2)Add borrower
3)Display all the available books
4)Display the list of all the subscribers
5)Get the list of books issued by a borrower
6)Get the list of borrowers of a book
7)Go to Home Page
8)Exit
Enter your choice: 4


 Borrowers in the library are:-
  221)Rohan Gupta
  292)Bhaumik Tandan
  321)Rohan Gupta
  121)Naman Sethi

1)Add book
2)Add borrower
3)Display all the available books
4)Display the list of all the subscribers
5)Get the list of books issued by a borrower
6)Get the list of borrowers of a book
7)Go to Home Page
8)Exit
Enter your choice: 5

Enter the borrower number: 292


 Books issued by Bhaumik Tandan are:
  1253) Three men in a boat on 13/3/2013
```

```
1)Add book
2)Add borrower
3)Display all the available books
4)Display the list of all the subscribers
5)Get the list of books issued by a borrower
6)Get the list of borrowers of a book
7)Go to Home Page
8)Exit
Enter your choice: 1


 Enter the book details:-
  Title: Dark riders
  Accession number: 9087

 Book "Dark riders" added with accession number 9087
```

```
1)Add book
2)Add borrower
3)Display all the available books
4)Display the list of all the subscribers
5)Get the list of books issued by a borrower
6)Get the list of borrowers of a book
7)Go to Home Page
8)Exit
Enter your choice: 2


 Enter the details:-
  Name: Bhaumik
  Borrower number: 280
  Password: ****

 Borrower Bhaumik with borrower number 280 added
```

```
1)Add book
2)Add borrower
3)Display all the available books
4)Display the list of all the subscribers
5)Get the list of books issued by a borrower
6)Get the list of borrowers of a book
7)Go to Home Page
8)Exit
Enter your choice: 6

Enter the book name: 7 ages


 7 ages has been issued by:
  221) Rohan Gupta on 13/3/2013: 9564

1)Add book
2)Add borrower
3)Display all the available books
4)Display the list of all the subscribers
5)Get the list of books issued by a borrower
6)Get the list of borrowers of a book
7)Go to Home Page
8)Exit
Enter your choice: 7


1)Login as borrower
2)Login as Admin
3)Exit
Enter your choice: 3
```

```
1)Login as borrower
2)Login as Admin
3)Exit
Enter your choice: 1

Enter the borrower number: 292
  Password: ****


1)Issue a book
2)Return a book
3)Show issued books
4)Show available books
5)Log out
6)Exit
 Enter your choice: 3

 Books issued by Bhaumik Tandan are:
  1253) Three men in a boat on 13/3/2013
```

```
1)Issue a book
2)Return a book
3)Show issued books
4)Show available books
5)Log out
6)Exit
 Enter your choice: 4


 Available books in the library are:-
  1024)Dance of dragons
  1814)Dead end
  7814)Clash of Kings
  1264)7 ages
  1164)7 ages
  1019)End game
  1214)Invisible Man
  1004)End game
```

```
 1)Issue a book
 2)Return a book
 3)Show issued books
 4)Show available books
 5)Log out
 6)Exit
  Enter your choice: 1

  Enter the name of the book to be borrowed: Dance of dragons
  Enter the date: 12/1/16

  Dance of dragons with accession number 1024 has been issued to Bhaumik Tandan with borrower number 292 on 12/1/16

 1)Issue a book
 2)Return a book
 3)Show issued books
 4)Show available books
 5)Log out
 6)Exit
  Enter your choice: 2

  Enter the name of the book to be returned: Dance of dragons

  Dance of dragons with accession number 1024 has been returned by Bhaumik Tandan with borrower number 292 which was issued on 12/1/16

 1)Issue a book
 2)Return a book
 3)Show issued books
 4)Show available books
 5)Log out
 6)Exit
  Enter your choice: 5

  1)Login
 2)Go to home page
 3)Exit
   Enter your choice: 3
PS C:\Users\bhaum\OneDrive\Desktop\da>
```

CLICK HERE FOR GITHUB LINK OF WHOLE SOURCE CODE