|| सिद्धिः भूषयते विद्याम् ||

avantika

UNIVERSITY

Discipline Core

Data Structures

CS4002

Assignment 1

Sheikh Muhammed Tadeeb (AU19B1014)

❖ Problem Statement:

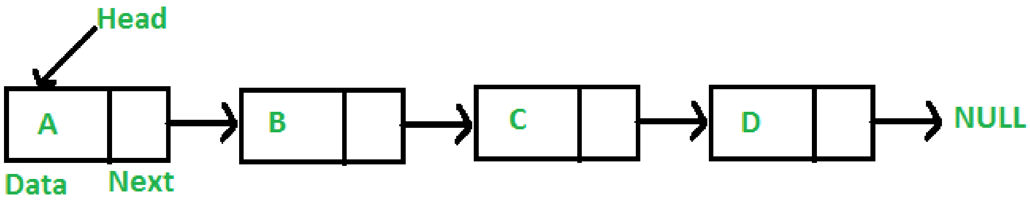
Creating a library management system for a university using appropriate datatype in C++, considering below points: -

1) Create the book category list.
2) Create a unique ID for each book list.
3) Add the books with their unique ID.
4) Searching the book in the category to be displayed.
5) Modify and traversing the shelf to be included in the code.

❖ Data-Type:

- **Linked-List:**

A linked list is a linear data structure, in which the elements are not stored at contiguous memory locations. The elements in a linked list are linked using pointers as shown in the below image:



In simple words, a linked list consists of nodes where each node contains a data field and a reference(link) to the next node in the list.

- **Why Linked-List?**

Time

Linked lists have most of their benefit when it comes to the insertion and deletion of nodes in the list. Unlike the dynamic array, insertion and deletion at any part of the list takes constant time.

However, unlike dynamic arrays, accessing the data in these nodes takes linear time because of the need to search through the entire list via pointers. It's also important to note that there is no way of optimizing search in linked lists. In the array, we could at least keep the array sorted. However, since we don't know how long the linked list is, there is no way of performing a binary search. Indexing - O(n), Insertion - O (1), Search - O(n). Deletion - O (1)

Space

Linked lists hold two main pieces of information (the value and pointer) per node. This means that the amount of data stored increases linearly with the number of nodes in the list. Therefore, the space complexity of the linked list is linear. Space - O(n).

❖ C++ Code:

```
#include <iostream>
#include <stdlib.h>
#include <bits/stdc++.h>
using namespace std;

class lib
{
        public:
                ~lib(){};
        public:
                string ctg;
                string ctgCode;
                string Bname;
                string Bcode;
                lib *next;
};

class linkedList : protected lib
{
        public:
                ~linkedList(){};

                typedef lib node;

                node* create_list();
                void display(node* head);
                void insert(node **head);
                void deletion(node** head);
                void modify(node **head);
                void BooksCount(node *head);
                void search(node *head);

                void call(){
                        node* head;
                        head = create_list();
                        display(head);

```

```
                    insert(&head);
                    display(head);
                    BooksCount(head);
                    search(head);

                    deletion(&head);
                    display(head);
                    BooksCount(head);

                    modify(&head);
                    char y;
                    cout<<endl<<endl<<endl<<endl<<endl<<"\tDo you want to see the data after all modifications(y/n) :- ";
                    cin>>y;
                    if (y == 'y' || y=='Y'){
                            cout<<endl<<endl<<endl<<endl<<endl<<"\tThe Final table is:- ";
                        display(head);
                    }else{
                            cout<<endl<<endl<<endl<<"\tThank You!";
                    }
             }

};

lib* linkedList :: create_list(){
                                        int n;
                                        node *head,*p;

        cout<<endl<<endl<<endl<<"\t\t\t\t##############  Library Management System  ##############";

        cout<<endl<<endl<<endl<<endl<<"\tEnter the nos of stock to be added = ";
                                        cin>>n;

                                        for (int i=0; i<n ; i++){
                                                if (i==0){
                                                        head = new node;
                                                        p = head;
                                                }
                                                else{
                                                        p->next = new node;
                                                        p = p->next;
                                                }

        cout<<endl<<endl<<"\tEnter Category Name:- ";
```

```
                                cin>>p->ctg;
                                cout<<endl<<"\tEnter Category Code:- ";
                                cin>>p->ctgCode;
                                cout<<endl<<"\tEnter Book Name:- ";
                                cin>>p->Bname;
                                cout<<endl<<"\tEnter Book Code:- ";
                                cin>>p->Bcode;
                            }
                            p->next = NULL;
                            return head;
                        }


struct arr {
    string ctgCode;
    int no_of_books;
};

arr total[3];

void linkedList :: BooksCount(node* head){
    char y;
        cout<<endl<<endl<<endl<<endl<<"\tDo you want to check status(y/n):- ";
        cin>>y;
        if(y == 'y' || y == 'Y'){
            for(int i=0; i<3;i++){
            string ca;
            cout<<endl<<"\tEnter The category codes:- ";
            cin>>ca;
            total[i].ctgCode = ca;
            }

            for(int i = 0; i<3; i++){
                node *p;
                p = head;
                string m = total[i].ctgCode;
                int count = 0;
                while(p != NULL){
                    if(p->ctgCode == m){
                        count++;
                    }
                    p = p->next;
                }
                total[i].no_of_books = count;
```

```
                    }

                    cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<"\t\t ______Status______";
                    // For printing the values in array.
                    for(int j = 0; j<3 ; j++ ){
                            cout<<endl<<endl<<"\t\t| "<<total[j].ctgCode<<"  "<<total[j].no_of_books<<"      |";

                    }
                    cout<<endl<<"\t\t ________________";
                    }else {
                            cout<<endl<<endl<<"You can check whenever we want";
                    }

      }

      void linkedList :: display(node* head){
                                                int count = 1;
                                                node *p;
                                                p = head;
      cout<<endl<<endl<<endl<<endl<<"\t\t\t\t\t\t\t Shelf Info.";
                                                cout<<endl<<"\t\t __________________________________________________________________________________________________";
                                                cout<<endl<<endl<<"\t\t Sno.\t\t CategoryName\t\t  CategoryCode\t\t  BookName\t\t  BookCode   ";
                                                while(p != NULL){
      cout<<endl<<"\t\t __________________________________________________________________________________________________";
      cout<<endl<<endl<<"\t\t  "<<count<<"\t\t  "<<p->ctg<<"\t\t  "<<p->ctgCode<<"\t\t   "<<p->Bname<<"\t\t  "<<p->Bcode;
                                                        p = p->next;
                                                        count++;
                                                }
                                                cout<<endl<<"\t\t __________________________________________________________________________________________________";
                                        }


      void linkedList :: insert(node **head){
                                        char y;
```

```
                int n;
    cout<<endl<<endl<<endl<<endl<<endl<<endl<<"___________________________________________________________________________________________________________________";
    cout<<endl<<endl<<"\t\t\t\t\t\t\tInsert";
                cout<<endl<<endl<<endl<<"\tDo you want to insert new data(y/n): ";
                cin>>y;
                if (y == 'y' || y == 'Y'){
    cout<<endl<<endl<<"\tHow many books data you want to enter:- ";
            cin>>n;
                for(int i = 1; i <= n ; i++) {
    cout<<endl<<endl<<endl<<"\tData "<<i<<")";
                    node *q,*p,*newone;
                    char y;
                        newone = new node;
                        string bcode,bname;
    cout<<endl<<endl<<"\tEnter Category Name:- ";
                        cin>>newone->ctg;
    cout<<endl<<"\tEnter Category Code:- ";
                        cin>>newone->ctgCode;
    cout<<endl<<"\tEnter Book Name:- ";
                        cin>>newone->Bname;
    cout<<endl<<"\tEnter Book Code:- ";
                        cin>>newone->Bcode;
                        p = *head;

    cout<<endl<<endl<<endl<<"\tDo you want to enter the book at last(y/n):- ";
                        cin>>y;
                        if (y=='y' || y == 'Y'){
```

```
                    while(p != 0){
        q = p;
        p = p->next;
                    }
        if(p == 0){         // Inserting in last
            q->next = newone;
            newone->next = NULL;
        }
                }else{
        cout<<endl<<endl<<endl<<"\tEnter The Book CODE before you need to place this new book:- ";
        cin>>bcode;
                    if (p->Bcode == bcode){     // Inserting at the beginning
    newone->next = p;
    *head = newone;
                    }
                    else{
        while(p != 0 && p->Bcode != bcode){
        q = p;
        p = p->next;
                    }
        if (p->Bcode == bcode){   // Inserting in middle
            q->next = newone;
            newone->next = p;
        }else if(p == 0){         // Inserting in last
            q->next = newone;
            newone->next = NULL;
```

```cpp
          }else{
                cout<<endl<<"\tNo such book code exist. Kindly enter a valid book code";
          }
                                                              }
                                                        }

                                                  }

                                            }
                                            else{
                                                  cout<<endl<<"\tThank You";
                                            }
                                      }


      void linkedList :: search(node* head){
            int count = 1;
            string bcode;
            node *p;
            p = head;
            cout<<endl<<endl<<endl<<endl<<endl<<endl<<"___________________________________________________________________________________________________________________";
            cout<<endl<<endl<<"\t\t\t\t\t\t\t\tSearch";
            cout<<endl<<endl<<endl<<endl<<"\tEnter the Book Code whose details you wanna see:- ";
            cin>>bcode;
            while(p != NULL){
                  if(p->Bcode == bcode){
                        cout<<endl<<endl<<endl<<"\t\t\t\t\t\t\t\t"<<p->Bcode<<" Info.";
                        cout<<endl<<"\t\t                               _______________________________________________________________________________________________";
                        cout<<endl<<endl<<"\t\t  Sno.\t\t CategoryName\t\t CategoryCode\t\t  BookName\t\t  BookCode  ";
                        cout<<endl<<"\t\t                               _______________________________________________________________________________________________";
                        cout<<endl<<endl<<"\t\t  "<<count<<"\t\t  "<<p->ctg<<"\t\t         "<<p->ctgCode<<"\t\t   "<<p->Bname<<"\t\t  "<<p->Bcode;
```

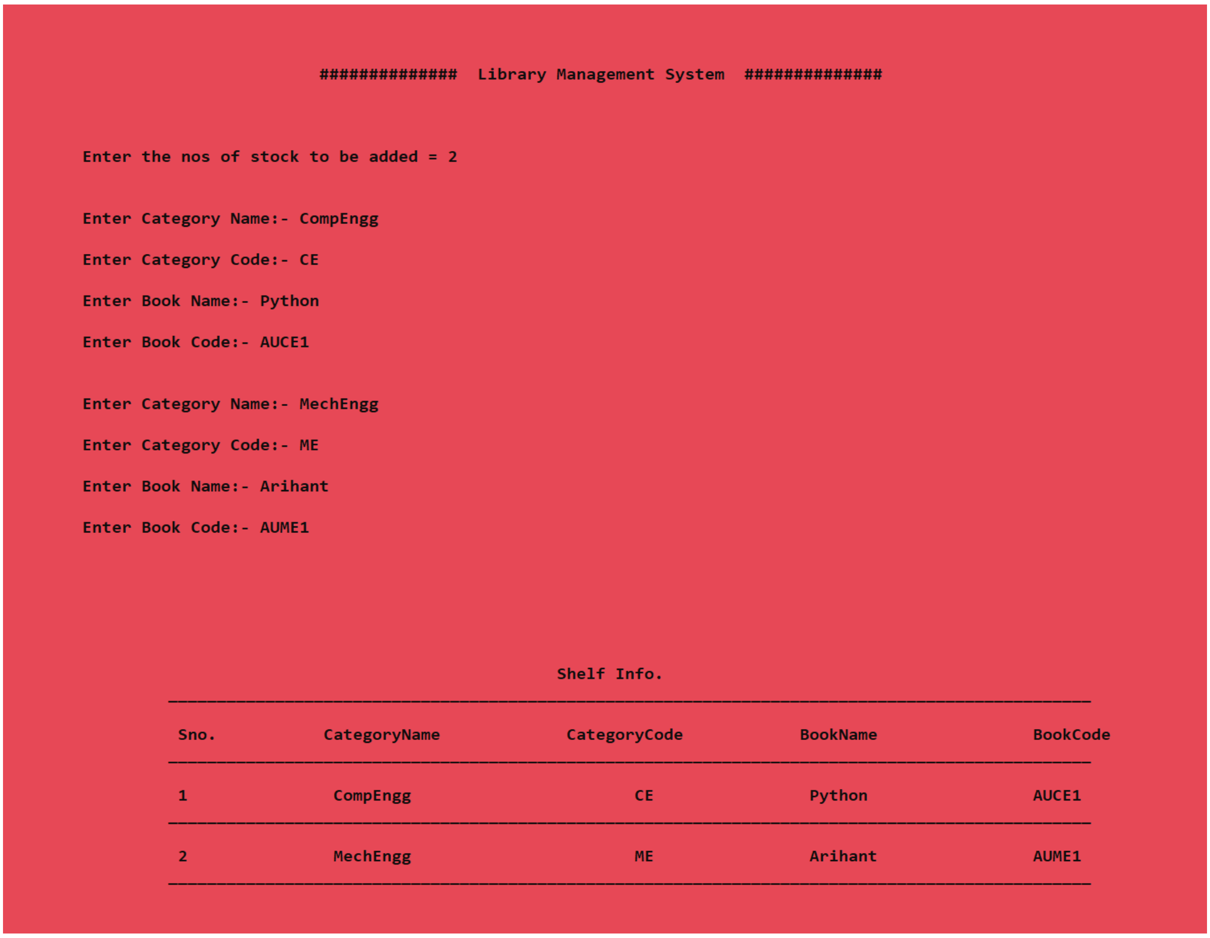
```
                    cout<<endl<<"\t\t_______________________________________________________________________________________________";
                    break;
             }else{
                    p = p->next;
             }
       }
}


void linkedList :: deletion(node** head){
                                        char y;
cout<<endl<<endl<<endl<<endl<<endl<<endl<<"___________________________________________________________________________________________________________________";
cout<<endl<<endl<<"\t\t\t\t\t\t\tDelete";
cout<<endl<<endl<<endl<<endl<<"\tDo you want to issue some book(y/n):- ";
                                        cin>>y;
                                        if (y == 'y' || y == 'Y'){
                                               string bcode;
                                               node *p,*q;
                                               cout<<endl<<"\tEnter the book code issued:- ";
                                               cin>>bcode;
                                               p = *head;
                                               if(p->Bcode == bcode){ //Deleting the 1st element
                                                      *head = p->next;
                                                      delete p;
                                               }else{
                                                      while(p->Bcode != bcode && p!=NULL){
                                                             q = p;
                                                             p = p->next;
                                                      }
                                                          if (p->Bcode == bcode){        // Deletion in middle
                                                             q->next = p->next;
                                                             delete p;
                                                             }else if(p == NULL){    // If no book exist
```

```
                        cout<<endl<<"\tNo such book is present in the shelf";
                    }else{ // Deleting the last one
                        q->next = NULL;
                        delete p;
                    }
                }

            }else{
                cout<<endl<<"\tThank You !";
            }
        }

    void modify_branch(lib *p){
            int count = 1;
            cout<<endl<<endl<<endl<<endl<<"\t\t\t\t\t\t\t Current Info.";
            cout<<endl<<"\t\t _______________________________________________________________________________________________";
            cout<<endl<<endl<<endl<<"\t\t  Sno.\t\t CategoryName\t\t CategoryCode\t\t  BookName\t\t  BookCode  ";
            cout<<endl<<"\t\t _______________________________________________________________________________________________";
            cout<<endl<<endl<<"\t\t  "<<count<<"\t\t  "<<p->ctg<<"\t\t          "<<p->ctgCode<<"\t\t   "<<p->Bname<<"\t\t  "<<p->Bcode;
            cout<<endl<<"\t\t _______________________________________________________________________________________________";

            cout<<endl<<endl<<endl<<endl<<"\tEnter the new CategoryName:- ";
            cin>>p->ctg;
            cout<<endl<<"\tEnter the new CategoryCode:- ";
            cin>>p->ctgCode;
            cout<<endl<<"\tEnter the new BookName:- ";
            cin>>p->Bname;
            cout<<endl<<"\tEnter the new BookCode:- ";
            cin>>p->Bcode;

            cout<<endl<<endl<<endl<<endl<<"\t\t\t\t\t\t\t Updated Info.";
```

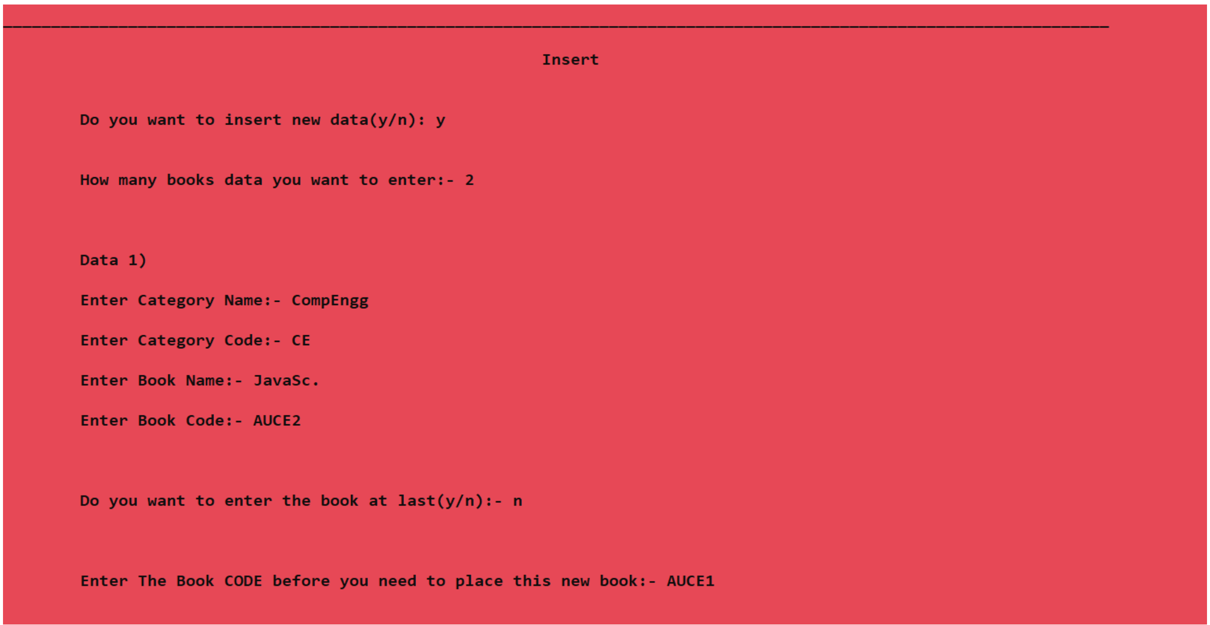
```
                cout<<endl<<"\t\t _______________________________________________________________________________________________";
                cout<<endl<<endl<<"\t\t  Sno.\t\t CategoryName\t\t CategoryCode\t\t  BookName\t\t  BookCode  ";
                cout<<endl<<"\t\t _______________________________________________________________________________________________";
                cout<<endl<<endl<<"\t\t  "<<count<<"\t\t  "<<p->ctg<<"\t\t         "<<p->ctgCode<<"\t\t   "<<p->Bname<<"\t\t  "<<p->Bcode;
                cout<<endl<<"\t\t _______________________________________________________________________________________________";
    }

    void linkedList :: modify(node **head){
        char y;
        int count =1;
        string bcode;
        node *p;
        cout<<endl<<endl<<endl<<endl<<endl<<endl<<"_____________________________________________________________________________________________________________________";
        cout<<endl<<endl<<"\tDo you want to modify existing data(y/n) :- ";
        cin>>y;
        if (y == 'y' || y == 'Y'){
            cout<<endl<<endl<<"\tEnter the Book code whose data you want to modify :- ";
            cin>>bcode;
            p = *head;

            if(p->Bcode == bcode){            // Modifying at the first node
                modify_branch(*head);
            }else{
                while(p->Bcode != bcode && p != NULL){
                p = p->next;
                }
                  if(p->Bcode == bcode){          // Modifying in the middle
                    modify_branch(p);
                    }else{                          // No element Found
                        cout<<endl<<endl<<"\tNo such Book Code exist. Kindly enter a valid book code";
                    }
            }
```

```
       }else{
              cout<<endl<<endl<<"\tThank You !";
       }
}

int main()
{
    system("Color C0");
    linkedList obj;
    obj.call();
       return 0;
}
```

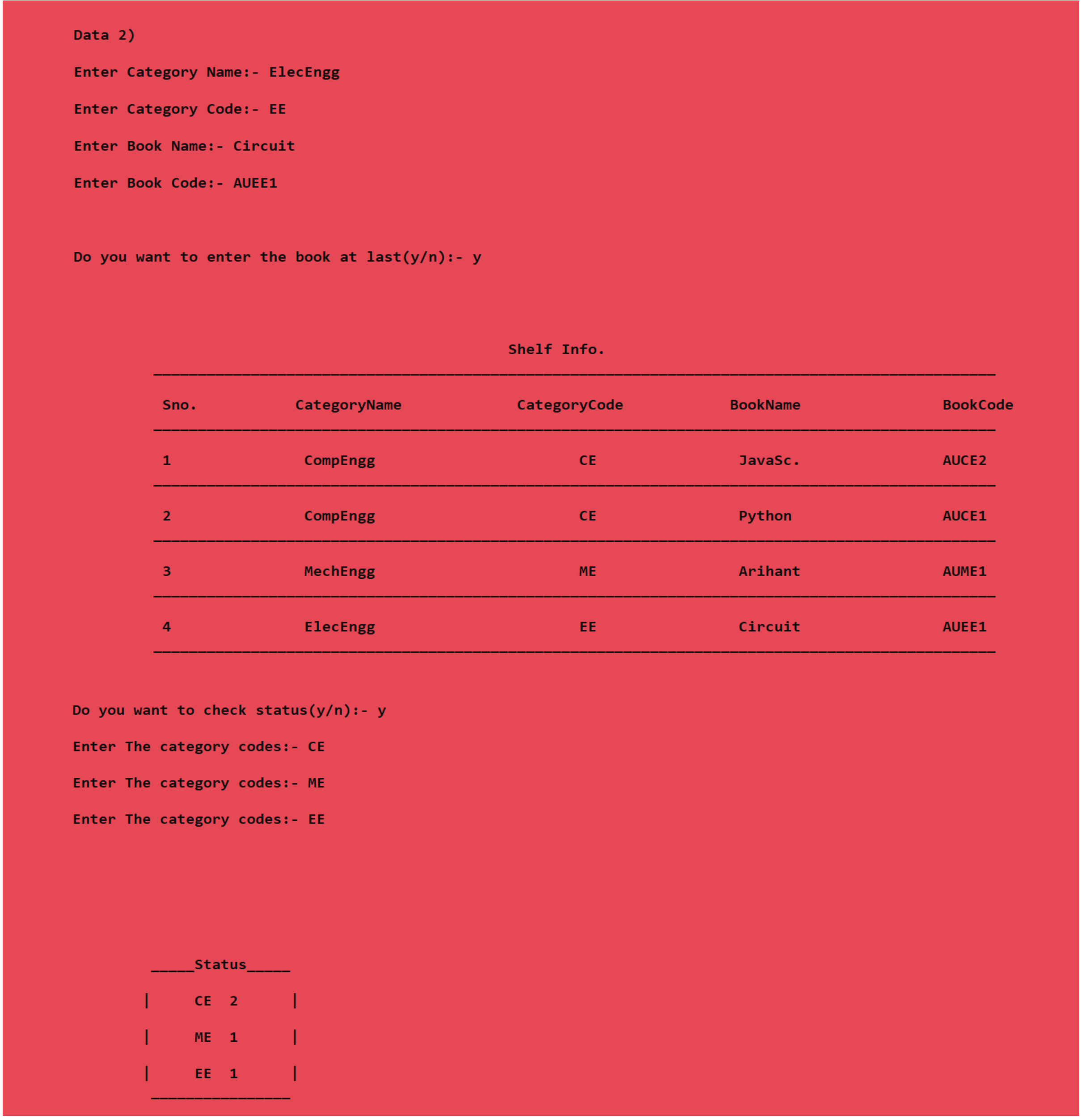
❖ C++ Output:



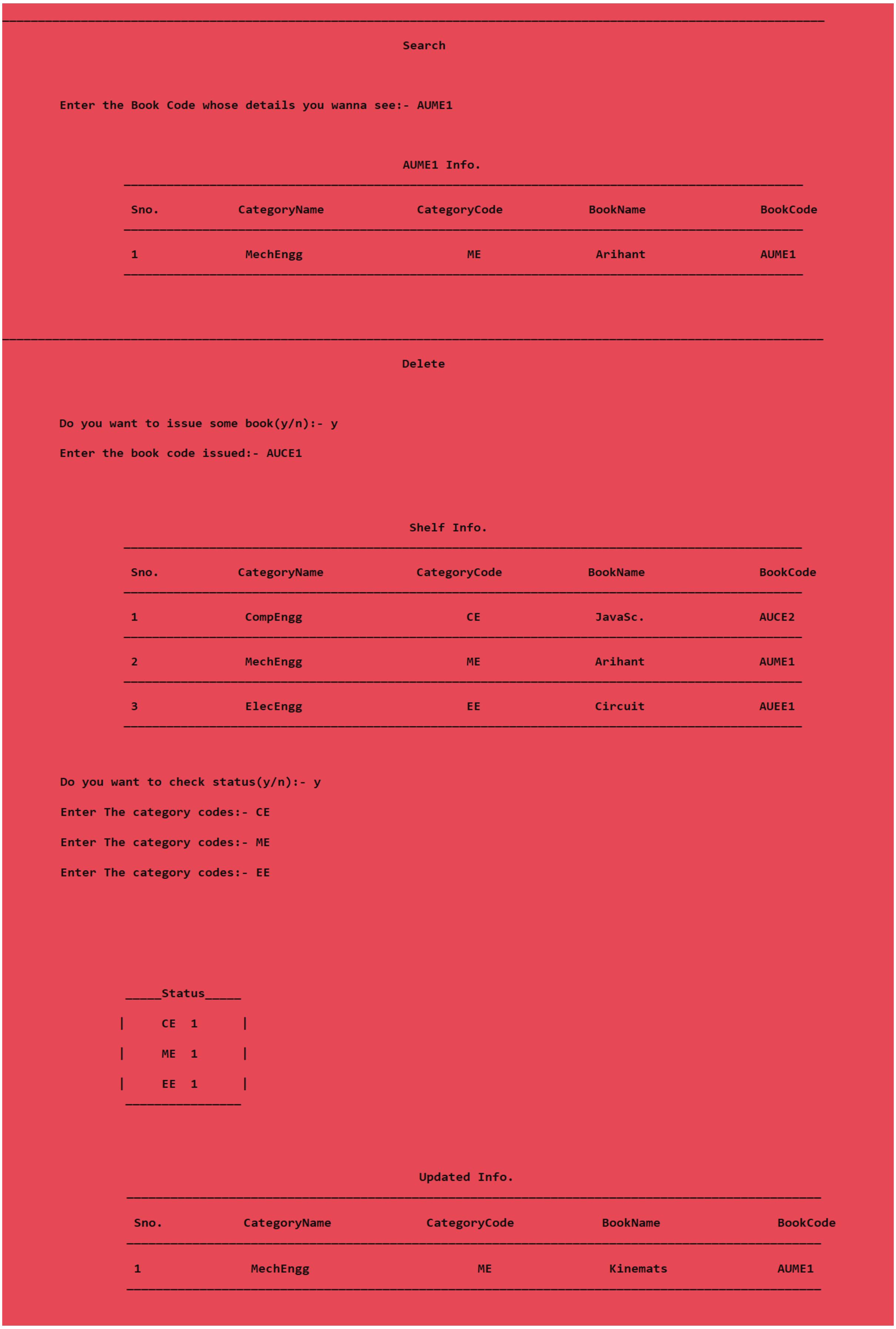
```
                  ##############  Library Management System  ##############


     Enter the nos of stock to be added = 2


     Enter Category Name:- CompEngg

     Enter Category Code:- CE

     Enter Book Name:- Python

     Enter Book Code:- AUCE1


     Enter Category Name:- MechEngg

     Enter Category Code:- ME

     Enter Book Name:- Arihant

     Enter Book Code:- AUME1




                                        Shelf Info.
                ______________________________________________________________________________

                 Sno.          CategoryName           CategoryCode          BookName            BookCode
                ______________________________________________________________________________

                 1              CompEngg                      CE             Python              AUCE1
                ______________________________________________________________________________

                 2              MechEngg                      ME             Arihant             AUME1
                ______________________________________________________________________________
```



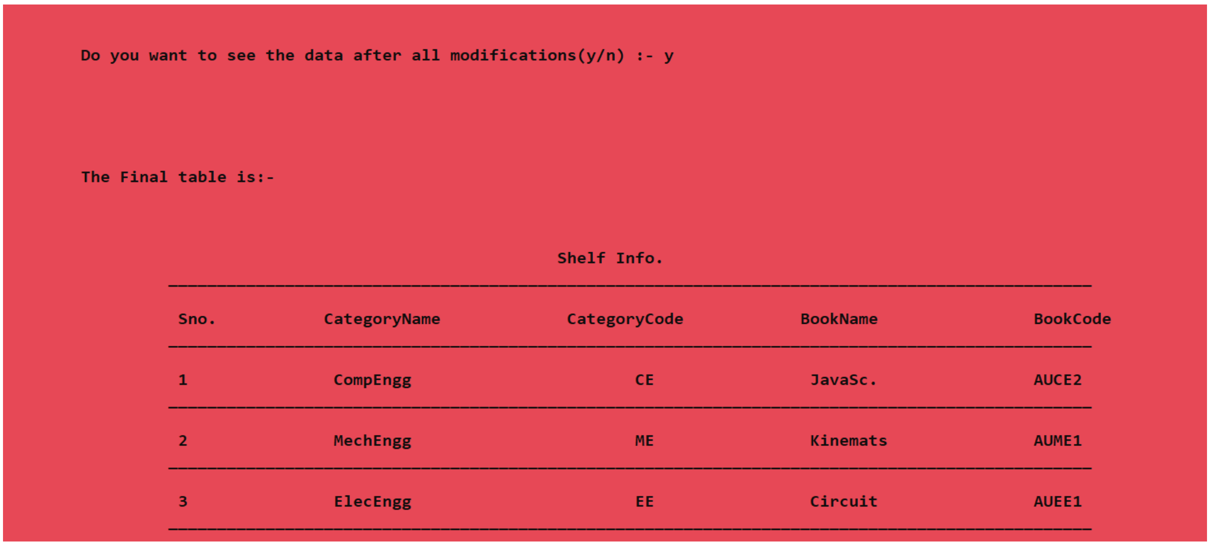
```
_____________________________________________________________________________________________________________________

                                                        Insert

        Do you want to insert new data(y/n): y

        How many books data you want to enter:- 2


        Data 1)

        Enter Category Name:- CompEngg

        Enter Category Code:- CE

        Enter Book Name:- JavaSc.

        Enter Book Code:- AUCE2


        Do you want to enter the book at last(y/n):- n


        Enter The Book CODE before you need to place this new book:- AUCE1
```



```
        Data 2)

        Enter Category Name:- ElecEngg

        Enter Category Code:- EE

        Enter Book Name:- Circuit

        Enter Book Code:- AUEE1


        Do you want to enter the book at last(y/n):- y


                                              Shelf Info.
                 _________________________________________________________________________________________________

                  Sno.            CategoryName            CategoryCode            BookName                BookCode
                 _________________________________________________________________________________________________

                  1                CompEngg                   CE                   JavaSc.                 AUCE2
                 _________________________________________________________________________________________________

                  2                CompEngg                   CE                   Python                  AUCE1
                 _________________________________________________________________________________________________

                  3                MechEngg                   ME                   Arihant                 AUME1
                 _________________________________________________________________________________________________

                  4                ElecEngg                   EE                   Circuit                 AUEE1
                 _________________________________________________________________________________________________

        Do you want to check status(y/n):- y

        Enter The category codes:- CE

        Enter The category codes:- ME

        Enter The category codes:- EE




                 _____Status_____
                |     CE  2      |
                |     ME  1      |
                |     EE  1      |
                 ________________
```



```
____________________________________________________________________________________________________________
                                                        Search


        Enter the Book Code whose details you wanna see:- AUME1


                                                        AUME1 Info.
                        ______________________________________________________________________________________
                        Sno.            CategoryName            CategoryCode            BookName                BookCode
                        ______________________________________________________________________________________
                        1                MechEngg                         ME             Arihant                AUME1
                        ______________________________________________________________________________________



____________________________________________________________________________________________________________
                                                        Delete


        Do you want to issue some book(y/n):- y

        Enter the book code issued:- AUCE1



                                                         Shelf Info.
                        ______________________________________________________________________________________
                        Sno.            CategoryName            CategoryCode            BookName                BookCode
                        ______________________________________________________________________________________
                        1                CompEngg                         CE             JavaSc.                AUCE2
                        ______________________________________________________________________________________
                        2                MechEngg                         ME             Arihant                AUME1
                        ______________________________________________________________________________________
                        3                ElecEngg                         EE             Circuit                AUEE1
                        ______________________________________________________________________________________


        Do you want to check status(y/n):- y

        Enter The category codes:- CE

        Enter The category codes:- ME

        Enter The category codes:- EE




                        _____Status_____
                        |     CE   1      |
                        |     ME   1      |
                        |     EE   1      |
                        ________________



                                                          Updated Info.
                        ______________________________________________________________________________________
                        Sno.            CategoryName            CategoryCode            BookName                BookCode
                        ______________________________________________________________________________________
                        1                MechEngg                         ME             Kinemats               AUME1
                        ______________________________________________________________________________________
```

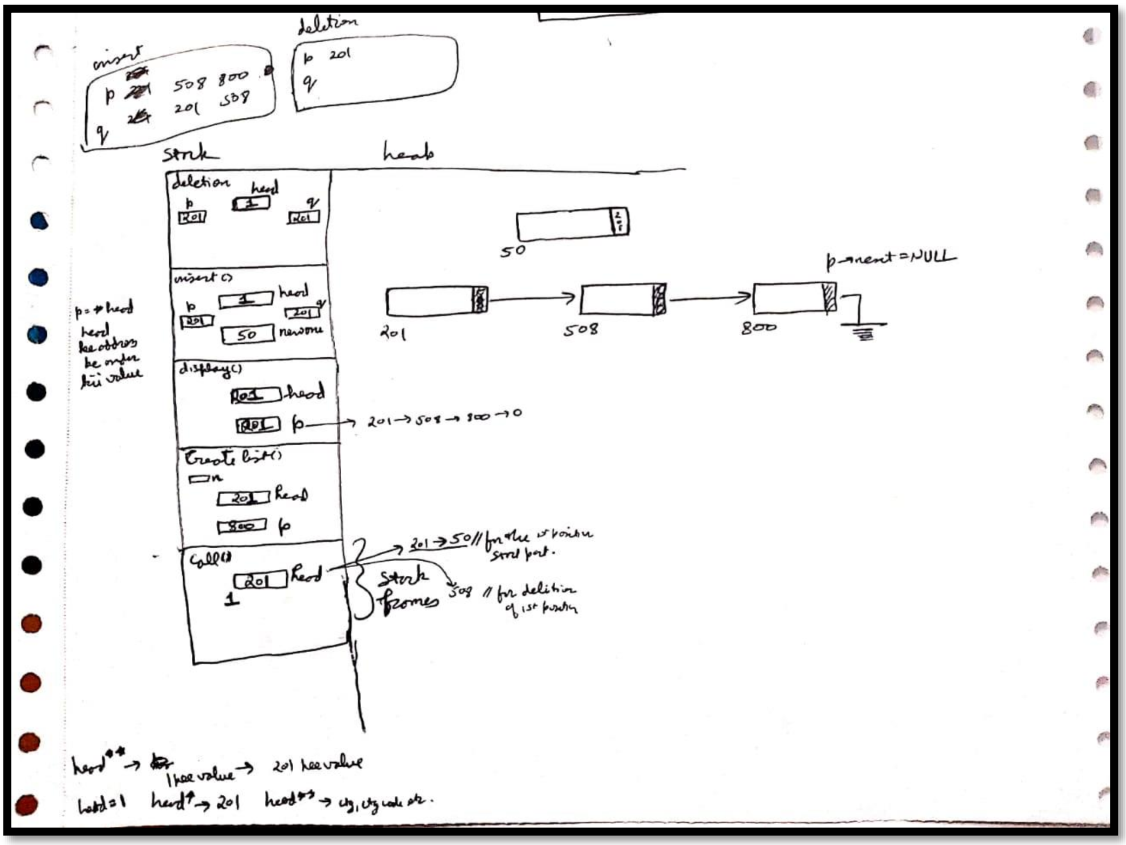


```
Do you want to see the data after all modifications(y/n) :- y



The Final table is:-


                                        Shelf Info.
         ______________________________________________________________________________________

         Sno.           CategoryName             CategoryCode            BookName                BookCode
         ______________________________________________________________________________________

         1               CompEngg                     CE                   JavaSc.               AUCE2
         ______________________________________________________________________________________

         2               MechEngg                     ME                   Kinemats              AUME1
         ______________________________________________________________________________________

         3               ElecEngg                     EE                   Circuit               AUEE1
         ______________________________________________________________________________________
```

❖ Conclusion:

Well, I was successfully able to implement and do the operations on linked-list. From the assignment I got to know the working of pointers and how different memory allocations takes place. Working with linked list is better than traditional arrays as it saves us lot of memory which in case of arrays is either exhausted or remain untouched which is again a memory wastage.

❖ Paper Work:



insert
p 508 800
q 201 508
deletion
p 201
q
stack
heap
deletion
head
p
201
1
q
201
insert()
head
p
201
1
201
q
50
newone
display()
201
head
201
p
201 → 508 → 800 → 0
Create list()
n
201
head
800
p
call()
201
head
1
201 → 50 // for the st position start part.
Stack frames
508 // for deletion of 1st position
50
201
508
800
p→next = NULL
p = *head
head kee address ke andar kii value
head** → 1kee value → 201 kee value
head = 1
head* → 201
head** → cg, cg code etc.