(1.) Implement a circular queue.

Code :

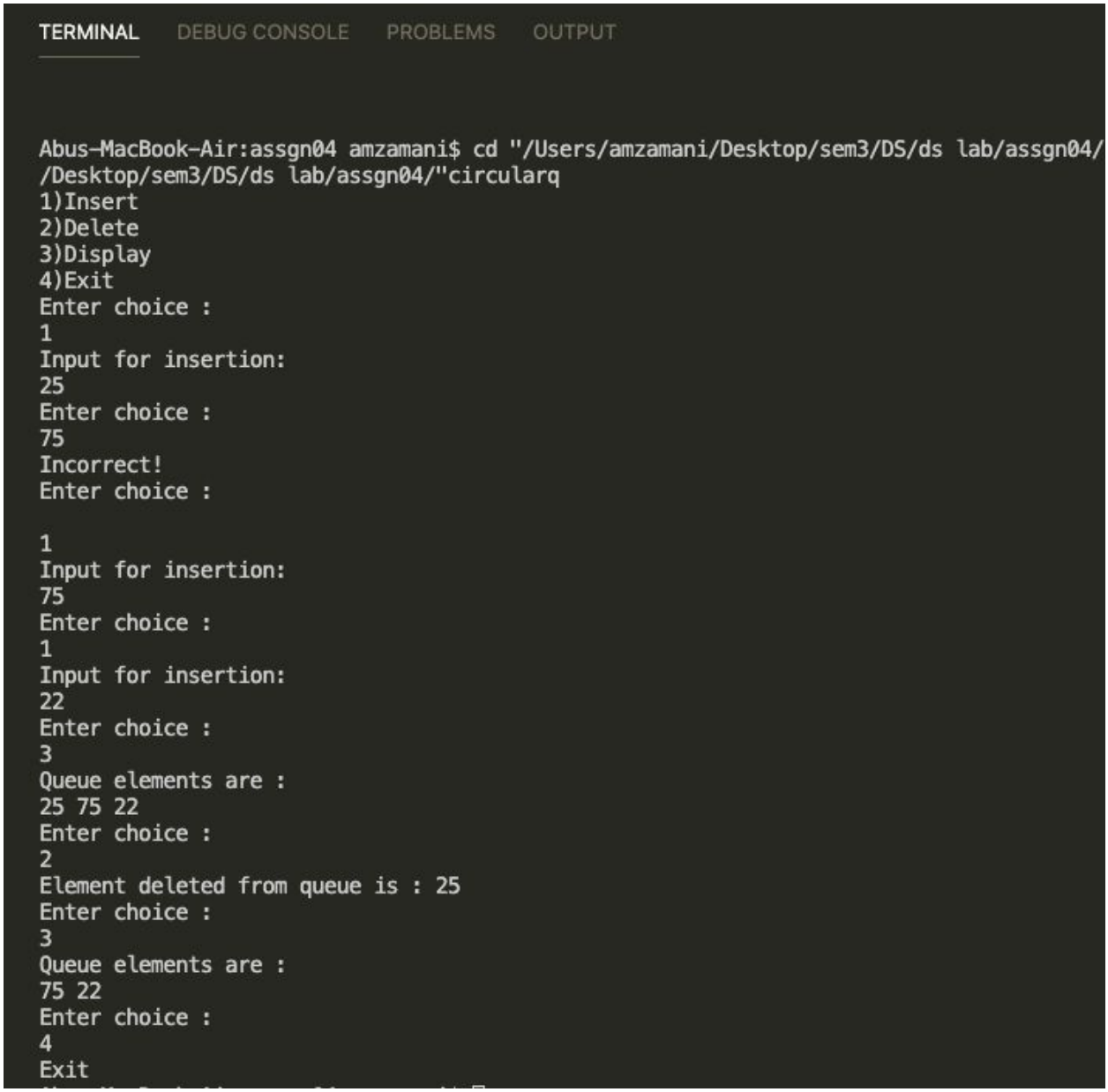
```
#include <iostream>
using namespace std;
int cqueue[5];
int front = -1, rear = -1, n = 5;
void insertCQ(int val) {
if ((front == 0 && rear == n - 1) || (front == rear + 1)) {
cout<< "Queue Overflow \n";
return; }
if (front == -1) {
front = 0;
rear = 0; }
else {
if (rear == n - 1) rear = 0;
else
rear = rear + 1;
}
cqueue[rear] = val;
}
void deleteCQ() {
if (front == -1) {
cout << "Queue Underflow\n";
return; }
cout << "Element deleted from queue is : " << cqueue[front] << endl;
if (front == rear) {
front = -1;
rear = -1; }
else {
if (front == n - 1) front = 0;
else
front = front + 1;
} }
void displayCQ() {
int f = front, r = rear; if (front == -1)
{
```

```
cout << "Queue is empty" << endl;
return; }
cout << "Queue elements are :\n"; if (f <= r)

{
while (f <= r) {
cout << cqueue[f] << " ";
f++; }
} else {
while (f <= n - 1) {
cout << cqueue[f] << " ";
f++; }
f = 0;
while (f <= r) {
cout << cqueue[f] << " ";
f++; }
}
cout << endl; }
int main() {
int ch, val;
cout << "1)Insert\n"; cout << "2)Delete\n"; cout << "3)Display\n"; cout << "4)Exit\n";
do
{

cout << "Enter choice : " << endl; cin >> ch;
switch (ch)
{
case 1:
cout << "Input for insertion: " << endl; cin >> val;
insertCQ(val);
break;
case 2: deleteCQ(); break;
case 3: displayCQ(); break;
case 4:
cout << "Exit\n"; break;
default:
cout << "Incorrect!\n";
}
} while (ch != 4); return 0;
}
```

Output :


TERMINAL
DEBUG CONSOLE
PROBLEMS
OUTPUT
Abus-MacBook-Air:assgn04 amzamani$ cd "/Users/amzamani/Desktop/sem3/DS/ds lab/assgn04/
/Desktop/sem3/DS/ds lab/assgn04/"circularq
1)Insert
2)Delete
3)Display
4)Exit
Enter choice :
1
Input for insertion:
25
Enter choice :
75
Incorrect!
Enter choice :
1
Input for insertion:
75
Enter choice :
1
Input for insertion:
22
Enter choice :
3
Queue elements are :
25 75 22
Enter choice :
2
Element deleted from queue is : 25
Enter choice :
3
Queue elements are :
75 22
Enter choice :
4
Exit

(2) Implement a priority queue.

Code :

```
#include <iostream>
#include <cstdio>
#include <cstring>
#include <cstdlib>
using namespace std;
struct node
{
   int priority;
   int info;
   struct node *link;
};
class Priority_Queue
{
private:
   node *front;

public:
   Priority_Queue()
   {
       front = NULL;
   }
   /*
* Insert into Priority Queue */
   void insert(int item, int priority)
   {
       node *tmp, *q;
       tmp = new node;
       tmp->info = item;
       tmp->priority = priority;
       if (front == NULL || priority < front->priority)
       {
           tmp->link = front;
           front = tmp;
       }
       else
```

```
        {
            q = front;
            while (q->link != NULL && q->link->priority <= priority)
                q = q->link;
            tmp->link = q->link;
            q->link = tmp;
        }
    }
    /*
* Delete from Priority Queue */
    void del()
    {
        node *tmp;
        if (front == NULL)
            cout << "Queue Underflow\n";
        else
        {
            tmp = front;
            cout << "Deleted item is: " << tmp->info << endl;
            front = front->link;
            free(tmp);
        }
    }
    void display()
    {
        node *ptr;
        ptr = front;
        if (front == NULL)
            cout << "Queue is empty\n";
        else
        {
            cout << "Queue is :\n";
            cout << "Priority Item\n";
            while (ptr != NULL)
            {
                cout << ptr->priority << " " << ptr->info << endl;
                    ptr = ptr->link;
            }
        }
    }
};
int main()
```

```
{
   int choice, item, priority;
   Priority_Queue pq;
   do
   {
       cout << "1.Insert\n";
       cout << "2.Delete\n";
       cout << "3.Display\n";
       cout << "4.Quit\n";
       cout << "Enter your choice : ";
       cin >> choice;
       switch (choice)
       {


               case 1 : cout<< "Input the item value to be added in the queue : ";
           cin >> item;
           cout << "Enter its priority : ";
           cin >> priority;
           pq.insert(item, priority);
           break;
       case 2:
           pq.del();
           break;
       case 3:
           pq.display();
           break;
       case 4:
           break;
       default:
           cout << "Wrong choice\n";
       }
   } while (choice != 4);
   return 0;
}
```

Output :

```
TERMINAL  DEBUG CONSOLE  PROBLEMS  OUTPUT

/Desktop/sem3/DS/ds lab/assgn04/"priorityq
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice : 1
Input the item value to be added in the queue : 22
Enter its priority : 1
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice : 1
Input the item value to be added in the queue : 75
Enter its priority : 2
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice : 1
Input the item value to be added in the queue : 66
Enter its priority : 3
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice : 3
Queue is :
Priority Item
1 22
2 75
3 66
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice : 2
Deleted item is: 22
1.Insert
2.Delete
3.Display
4.Quit
Enter your choice : 3
Queue is :
Priority Item
2 75
3 66
```