Assignment no	11 (Group E)
Aim	**Queues are frequently used in computer programming, and a typical example is the creation of a job queue by an operating system. If the operating system does not use priorities, then the jobs are processed in the order they enter the system. Write C++ program for simulating job queue. Write functions to add job, display job and delete job from queue.**
Objective	To understand the concept of queue data structure To understand job queue of operating system & implement it using queue To use array in C++ to implement job queue scheduling (add job, display job & delete job)
Outcome	To understand ,design and implement queue data structure using array in C++ To write/implement user defined functions/modules for different operations of queue (create, insert, delete, display) To write menu driven, modular program in C++.
OS/Programming tools used	(64-Bit) 64-BIT Fedora 17 or latest 64-BIT Update of Equivalent Open source OS or latest 64-BIT Version and update of Microsoft Windows 7 Operating System onwards Programming Tools (64-Bit) Eclipse IDE with C++ (CDT Plugin) & linux gcc compiler

Theory related to assignment:

Queue is a collection whose elements are added at one end (the ***rear*** or ***tail*** of the queue) and removed from the other end (the ***front*** or ***head*** of the queue) i.e. queue is opened at both end. One end is always used to insert data (enqueue) and the other is used to remove data (dequeue).

A queue is a ***FIFO*** (first in, first out) data structure i.e., the data item stored first will be accessed first.

Example: Any waiting line is a queue:

- The check-out line at a grocery store
- The cars at a stop light
- An assembly line

Queue Operations:

enqueue : add an element to the tail of a queue

dequeue : remove an element from the head of a queue

first : examine the element at the head of the queue ("peek")

Other useful operations **(e.g. is the queue empty)**

<u>It is not legal to access the elements in the middle of the queue!</u>

ADT :

```
ADT representation using class for Set
Class Queue {
int item[MAX];          // array to hold queue elements of size MAX
int FRONT;
int REAR;
public: //Operator definition in ADT
void qInsert( q,val); // insert val in queue, pointed by q
void show(); //displays complete queue
int qDelete(Q); // delete an element from Q & returns it.
}
```

Pseudo Code:

Insert an item into queue

procedure qInsert(q, val)

Purpose: To insert val into queue pointed by q

Pre-condition: queue should not be full

Post condition: element by name val will be inserted into queue

```
        begin
    1.  if q->Rear = MAX-1   // check if queue is full
            a.  print "Overflow" and go to step 5
    end if
    2.  if q->FRONT = -1    // check if queue is empty
            a.  set q->FRONT = 0
    end if
    3.  set q->REAR = q->REAR +1    //increment rear by 1
    4.  set q->item[q->REAR]=val      // insert val
    5.  end
```

Delete an item from queue

```
procedure qDelete(q)
        Purpose : To delete an item from queue pointed by q
        Pre-condition: queue should not be empty
        Post condition: element deleted from queue will be returned
        Begin
    1. if q->FRONT = -1   // check if queue is EMPTY
          a. print "Underflow" and go to step 5
          b. return 0 and go to step 5
       end if
    2. set del_val=q->item[q->FRONT]   // item to be deleted saved in del_val
    3. if q->FRONT = q->REAR    // check if only 1 element in queue
          a. set q->FRONT = q->REAR = -1   // set queue to initial condition as empty
       else
             set  q->FRONT = q->FRONT+1    // increment FRONT by 1
       end if
    4. retun del_val
    5. end
```

In case of array implementation

Time Complexity: O(1)

Space Complexity: O(1)

Conclusion:

The queue data structure is used to simulate the job queue and performed primitive operations using C++

Review Questions:

1. What is set queue data structure and its applications?
2. What is priority queue?
3. How to use linked list to implement queue data structure?
4. How to use queue to implement stack data structure?
5. How to implement priority queue
6. How queue is used in computing problems?
7. What is circular queue & how it is different than normal linear queue?
8. What will be time complexity in case of linked list implementation of queue?