Page No.
Date

Devanshu Surana
1032210755, PC-12
FDS Lab Assignment - 9

Problem statement :
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 a C program for simulating job queue. Write functions to add job and delete job from queue.

Objective :
1. To study Queue and its operations
2. To study the importance of queue as a data structure in computer science.

Theory :
- Write in brief about linear and circular queue. Write different applications of queue (job scheduling by CPU scheduler

→ (a) Linear queue.
A linear queue is linear data structure that serves request first, which has been arrived first i.e it follows FIFO policy. It consists of data elements which are connected in linear manner. It has two pointer i.e front and rear where the insertion takes place from rear and end while deletion takes place from front end.

(b) Circular queue :
A circular queue is also linear data structures like normal queue which follows the FIFO policy but it does not

end queue, it connects last position of the queue to first position of queue. It is also called as 'Ring Buffer'. If we want to insert from ~~from~~ front end, circular queue is used.

i) Enqueue - used to insert new element

ii) Dequeue - used to delete existing element

Platform:

- 64-bit open source linux or its derivatives
- Open source C programming tool like gcc/EclipseEditor

PSEUDO Code:

a) Add Q:

```
    void addq (q [], elem) {
      [initially front = rear = -1]
        if (front == size-1)
            print ('Queue full')
        else
            rear = rear +1;
          q [rear] = elem;
      }
```

b) Delete Q:

```
        int delq (q []) {
          if (front = = rear)
              return -1
          else
              front = front +1
              elem = q [front]
              return elem;
        }
```

c) is full ();

```
    int isfull(){
      if (rear == size -1)
        return 1;
      else
         return 0;
      }
```

d) is Empty ();

```
     int isEmpty (){
         if (rear == front)
            return 1;
         else
            return 0;
      }
```

e) Add ~~eq~~ CQ:

```
    void addcq (element){
      // initially front = rear = 0
      if (rear+1) % n == front
         print ('Queue Full')
      else
       rear = (rear+1) % n;
        Q [rear] = element;
      }
```

f) Detete CQ;

```
      int delcq (element){
   // initially front = rear = 0
         if (front == rear)
        print ('Empty Queue')
        else
```

```
front: (front + 1) % n
element - Q(front)
   return element
}
```

Time Complexity.

i) add q = O(1)
ii) del q = O(1)
iii) is full - O(1)
iv) Is empty = O(1)
v) add cq = O(1)
vi) delete cq = O(1)

Conclusion

Thus, implemented queue operations assignment using array concept.

FAQ's.

Ans i] Advantages:

- In linear queue, we can easily fetch out peak value.
- Insertion and deletion points are always fixed.
- Organizes data in simple linear order.

Diadvantages:

- Requires more memory space
- Inefficient way of utilization of memory.
- Can't insert data at any particular point.

Ans 2] Advantages
- Can insert data at any particular point
- Requires less memory space
- Efficient way of utillization of memory.

Disadvantages.
- Can't fetch peak value as it is circular
- No. of elements you store are limited to queue length.

Ans 3) The most common occurence of queue in computer application is scheduling of jobs.
Queues can be used as buffer on mp3 players and portable CD players.
Used in networking devices like routers, switches, etc.

2/11/22