Day 7

Write a program implementing insert, delete and display operation of Circular Queue.

|  |
| --- |
| #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; |
|  | } |