Assignment 6:

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 and delete job from queue.

|  |
| --- |
| /\*-----------------------------------------------------  Assignment 6:  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 and delete job from queue.  -----------------------------------------------------  \*/  #include <iostream>  using namespace std;  class queue  {  int \*arr;  int front, rear ;  int MAX ;  public :  queue(int maxsize = 10);  void insert ( int item );  int deleteQue( );  void displayQueue();  };  queue :: queue( int maxsize )  {  MAX = maxsize ;  arr = new int [ MAX ];  front = -1 ;  rear = -1 ;  }  void queue :: insert ( int item )  {  if ( rear == MAX - 1 )  {  cout << "\nQueue is full" ;  return ;  }  rear++ ;  arr[rear] = item ;  if ( front == -1 )  front = 0 ;  }  int queue :: deleteQue( )  {  int data ;  if ( front == -1 || front > rear)  {  cout << "\nQueue is Empty\n" ;  // return NULL ;  }  data = arr[front] ;  arr[front] = 0;  front++ ;  return data ;  }  void queue::displayQueue()  {  if ( front == -1 || front > rear)  {  cout << "\nQueue is Empty";  return;  }  cout << "\nElements in Queue are:" ;  for (int i = front; i <= rear; i++)  cout << arr[i] << " ";  cout << endl;  }  int main()  {  int choice;  int size;  cout << "Enter the size of queue: ";  cin >> size;  queue q(size);  do {  cout << "1. Insert Elements" << endl;  cout << "2. Delete Element" << endl;  cout << "3. Display Elements" << endl;  cout << "4. Exit" << endl;  cin >> choice;  switch (choice) {  case 1:  // Inserting elements in Circular Queue  for(int i = 0; i < 5; i++){  int element;  cout << "Insert an element - ";  cin >> element;  q.insert(element);  }  break;  case 2:  q.deleteQue();  break;  case 3:  q.displayQueue();  break;  case 4:  break;  default:  cout << "Wrong input"<< endl;  break;  }  }while(choice != 4);  } |