RIPHAH INTERNATIONAL UNIVERSITY

Data Structures & Alogorithms

Batch: Fall-2024

Manahil Waseem

Sap: 54035

Submitted to: Ms. Sanam Rehana

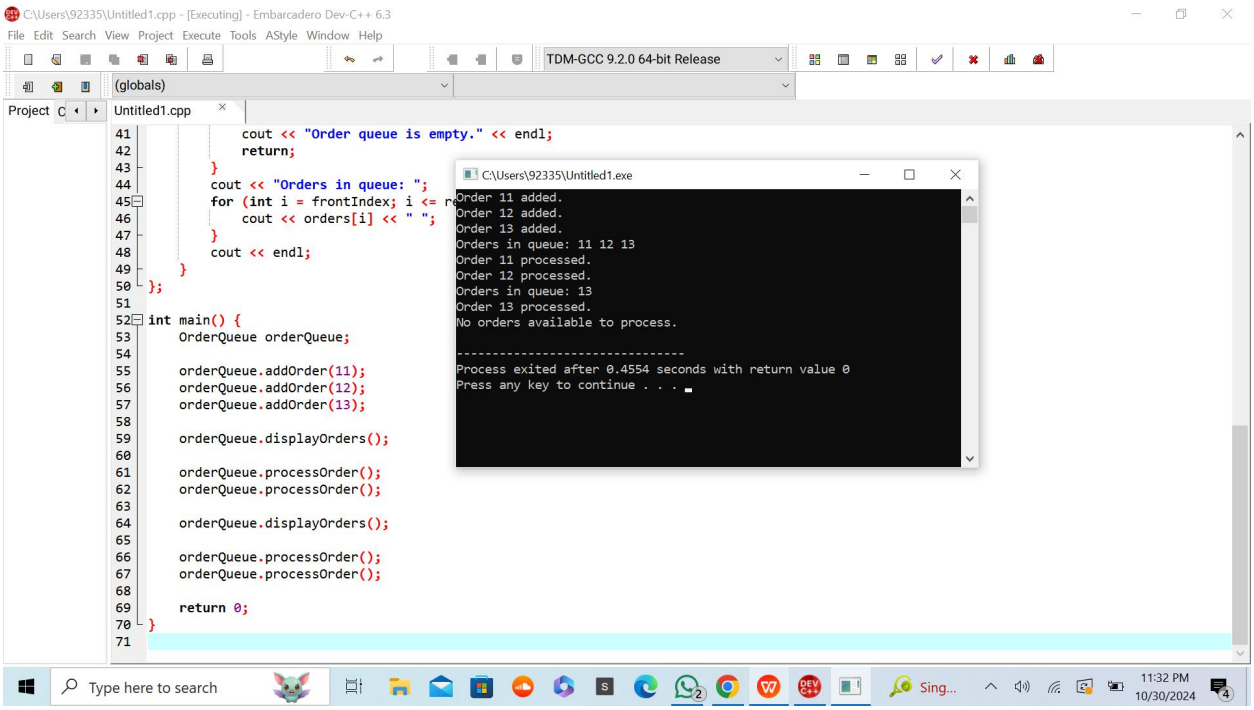
Date : 02/10/2024

TASK:1

```
#include <iostream>
using namespace std;

class OrderQueue {
private:
    static const int maxSize = 10;
    int orders[maxSize];
    int frontIndex;
    int rearIndex;

public:
    OrderQueue()
        : frontIndex(-1), rearIndex(-1) {}

    void addOrder(int orderID) {
        if (rearIndex == maxSize - 1) {
            cout << "Queue is full. Cannot add order " << orderID << "." << endl;
            return;
        }
        if (frontIndex == -1) {
            frontIndex = 0;
        }
        orders[++rearIndex] = orderID;
        cout << "Order " << orderID << " added." << endl;
    }
```

```
    void processOrder() {
        if (frontIndex == -1 || frontIndex > rearIndex) {
            cout << "No orders available to process." << endl;
            return;
        }
        int orderID = orders[frontIndex++];
        cout << "Order " << orderID << " processed." << endl;
        if (frontIndex > rearIndex) {
            frontIndex = rearIndex = -1;
        }
    }

    void displayOrders() {
        if (frontIndex == -1) {
            cout << "Order queue is empty." << endl;
            return;
        }
        cout << "Orders in queue: ";
        for (int i = frontIndex; i <= rearIndex; i++) {
            cout << orders[i] << " ";
        }
        cout << endl;
    }
};

int main() {
    OrderQueue orderQueue;

    orderQueue.addOrder(11);
```

```
    orderQueue.addOrder(12);
    orderQueue.addOrder(13);

    orderQueue.displayOrders();

    orderQueue.processOrder();
    orderQueue.processOrder();

    orderQueue.displayOrders();

    orderQueue.processOrder();
    orderQueue.processOrder();

    return 0;
}
```

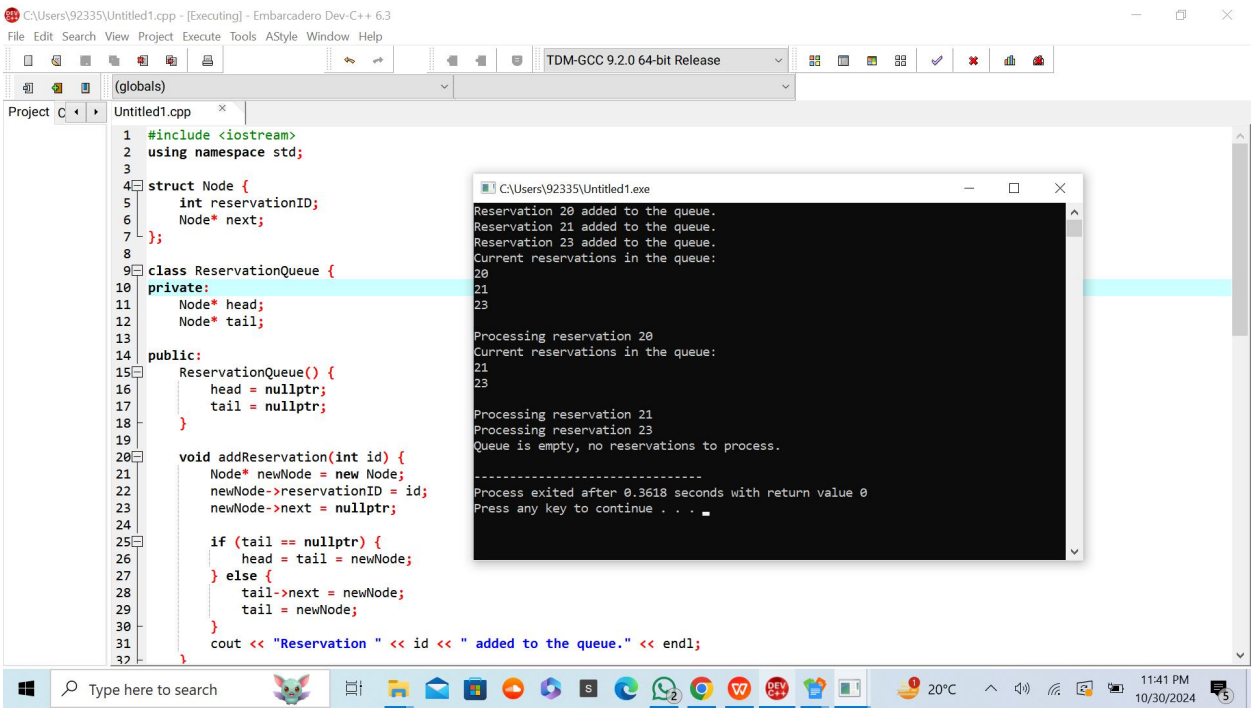


TASK:2

```
#include <iostream>
using namespace std;

struct Node {
   int reservationID;
   Node* next;
};

class ReservationQueue {
private:
   Node* head;
   Node* tail;

public:
   ReservationQueue() {
      head = nullptr;
      tail = nullptr;
   }

   void addReservation(int id) {
      Node* newNode = new Node;
      newNode->reservationID = id;
      newNode->next = nullptr;

      if (tail == nullptr) {
```

```
        head = tail = newNode;
    } else {
        tail->next = newNode;
        tail = newNode;
    }
    cout << "Reservation " << id << " added to the queue." << endl;
}

void processReservation() {
    if (head == nullptr) {
        cout << "Queue is empty, no reservations to process." << endl;
        return;
    }

    Node* temp = head;
    head = head->next;

    if (head == nullptr) {
        tail = nullptr;
    }
    cout << "Processing reservation " << temp->reservationID << endl;
    delete temp;
}

void showReservations() {
    if (head == nullptr) {
        cout << "No current reservations in the queue." << endl;
        return;
    }
```

```
        Node* temp = head;
        cout << "Current reservations in the queue:" << endl;
        while (temp != nullptr) {
            cout << temp->reservationID << endl;
            temp = temp->next;
        }
        cout << endl;
    }
};

int main() {
    ReservationQueue queue;
    queue.addReservation(20);
    queue.addReservation(21);
    queue.addReservation(23);
    queue.showReservations();
    queue.processReservation();
    queue.showReservations();
    queue.processReservation();
    queue.processReservation();
    queue.processReservation();

    return 0;
}
```


C:\Users\92335\Untitled1.cpp - [Executing] - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project
Untitled1.cpp
#include <iostream>
using namespace std;
struct Node {
int reservationID;
Node* next;
};
class ReservationQueue {
private:
Node* head;
Node* tail;
public:
ReservationQueue() {
head = nullptr;
tail = nullptr;
}
void addReservation(int id) {
Node* newNode = new Node;
newNode->reservationID = id;
newNode->next = nullptr;
if (tail == nullptr) {
head = tail = newNode;
} else {
tail->next = newNode;
tail = newNode;
}
cout << "Reservation " << id << " added to the queue." << endl;
}
C:\Users\92335\Untitled1.exe
Reservation 20 added to the queue.
Reservation 21 added to the queue.
Reservation 23 added to the queue.
Current reservations in the queue:
20
21
23
Processing reservation 20
Current reservations in the queue:
21
23
Processing reservation 21
Processing reservation 23
Queue is empty, no reservations to process.
Process exited after 0.3618 seconds with return value 0
Press any key to continue . . .
Type here to search
20°C
11:41 PM
10/30/2024