Name : Farhan Ahmad

Sap id : 56193

Lab Task #09

```
#include <iostream>
using namespace std;
struct Node {
    int data;
    Node* next;
};
class Queue {
private:
    Node* front;
    Node* rear;
public:
    Queue() : front(NULL), rear(NULL) {}
    void enqueue(int item) {
        Node* newNode = new Node();
        newNode->data = item;
        newNode->next = NULL;
```

```
    if (rear == NULL) {
        front = rear = newNode;
    } else {
        rear->next = newNode;
        rear = newNode;
    }


}

int dequeue() {
    if (front == NULL) {
        cout << "Queue underflow! Cannot remove item." << endl;
        return -1;
    }

    int item = front->data;
    Node* temp = front;
    front = front->next;

    if (front == NULL) {
        rear = NULL;
```

```
    }
    delete temp;
    return item;
}
bool isEmpty() {
    return front == NULL;
}

int count() {
    int size = 0;



    Node* temp = front;

    while (temp != NULL) {
        size++;
        temp = temp->next;
    }
    return size;
}

void clear() {
```

```
    while (front != NULL) {
        Node* temp = front;
        front = front->next;
        delete temp;
    }
    rear = NULL;
}

void display() {
    if (front == NULL) {
        cout << "Queue is empty." << endl;
        return;
    }

    Node* temp = front;

    while (temp != NULL) {
        cout << temp->data << " ";
        temp = temp->next;
    }
    cout << endl;
}
```

```
    ~Queue() {
        clear();
    }
};

int main() {
    Queue q;
    q.enqueue(10);
    q.enqueue(20);
    q.enqueue(30);
    q.enqueue(40);
    q.enqueue(50);
    q.enqueue(70);

    cout << "Queue elements: ";

    q.display();
    cout << "Queue size: " << q.count() << endl;
    cout << "Dequeued item: " << q.dequeue() << endl;
    cout << "Queue elements after dequeue: ";
    q.display();
   cout << "Is queue empty? " << ((q.isEmpty()) ? "Yes" : "No") << endl;
```

```
    cout<<"After Clearing. \n";
    q.clear();
    cout << "Is queue empty? " << ((q.isEmpty()) ? "Yes" : "No") <<
endl;
    return 0;
}
```