NAME: SYED AWAIS HAIDER

REG:090

**Assignment 3**

#include <iostream> using namespace std;

class CircularQueue { private:

int \*arr; int front; int rear; int size; int count;

public:

CircularQueue(int s) { if (s < 3) s = 3;

else if (s > 7) s = 7; size = s;

arr = new int[size]; front = -1;

rear = -1;

count = 0;

cout << "Queue initialized with size: " << size << endl;

}

bool isFull() {

return count == size;

}

bool isEmpty() { return count == 0;

}

void enqueue(int value) { if (isFull()) {

resize();

}

if (front == -1) front = 0; rear = (rear + 1) % size;

arr[rear] = value; count++;

cout << "Enqueued: " << value << endl;

}

void dequeue() { if (isEmpty()) {

cout << "Queue is empty! Nothing to dequeue.\n"; return;

}

cout << "Dequeued: " << arr[front] << endl; front = (front + 1) % size;

count--;

if (count == 0) { front = rear = -1;

cout << "Queue is now empty. Resetting front and rear.\n";

}

}

void resize() {

int newSize = size \* 2;

int \*newArr = new int[newSize];

cout << "Queue is full. Resizing to " << newSize << "...\n"; for (int i = 0; i < count; i++) {

newArr[i] = arr[(front + i) % size];

}

delete[] arr; arr = newArr; size = newSize; front = 0;

rear = count - 1;

}

void display() { if (isEmpty()) {

cout << "Queue is empty.\n"; return;

}

cout << "Queue elements: "; for (int i = 0; i < count; i++) {

cout << arr[(front + i) % size] << " ";

}

cout << endl;

}

~CircularQueue() { delete[] arr;

}

};

int main() { CircularQueue q(4);

q.enqueue(10); q.enqueue(20); q.enqueue(30); q.enqueue(40); q.display();

q.enqueue(50); q.display();

q.dequeue();

q.dequeue();

q.display();

q.enqueue(60); q.enqueue(70); q.display();

q.dequeue();

q.dequeue();

q.dequeue();

q.dequeue();

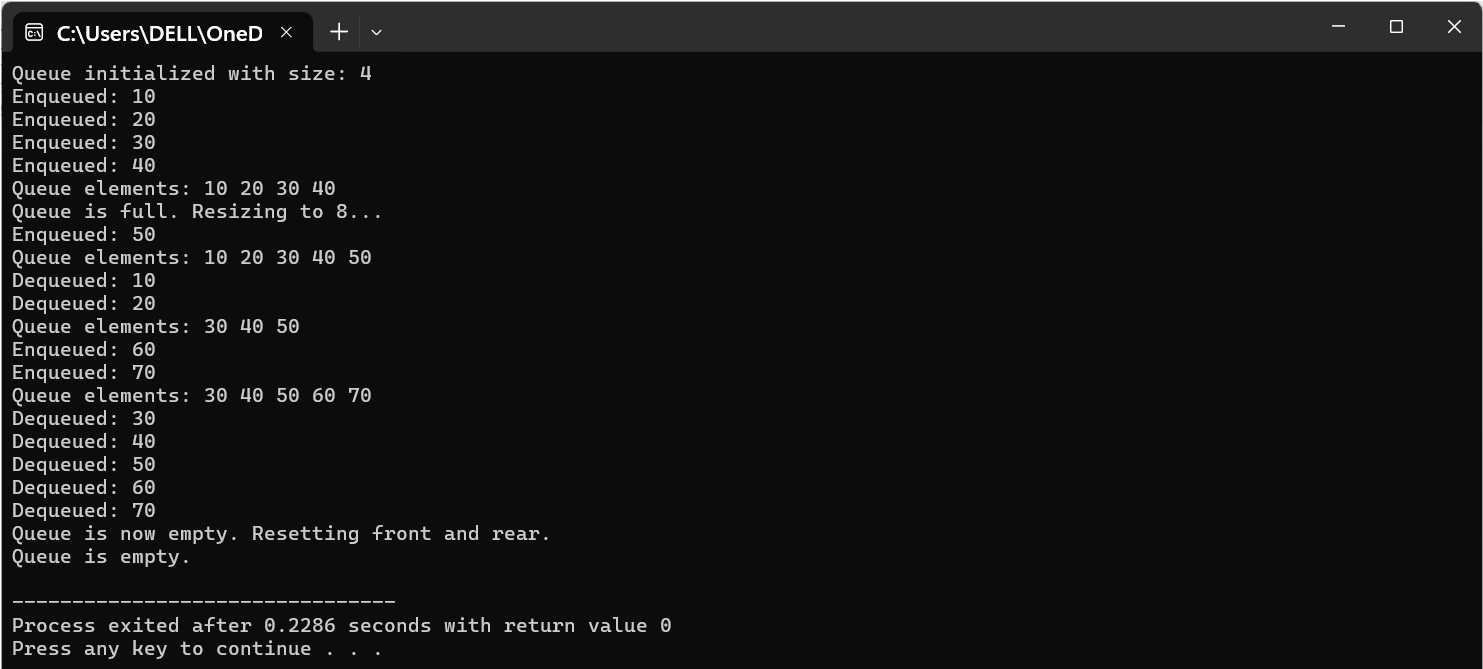
q.dequeue();

q.display();

return 0;

}

**OUTPUT;**

****