**FOUNDATION UNIVERSITY SCHOOL OF SCIENCE AND TECHNOLOGY**



DSA LAB REPORT 03

LAB TASKS

Name: Warda Javed

094

**(1)**

#include <iostream>

#define SIZE 5 // Maximum size of queue

using namespace std;

class Queue {

int items[SIZE];

int front, rear;

public:

Queue() {

front = -1;

rear = -1;

}

// Function to insert an element (Enqueue)

void enqueue(int value) {

if (rear == SIZE - 1)

cout << "Queue is full!\n";

else {

if (front == -1)

front = 0;

rear++;

items[rear] = value;

cout << value << " inserted into queue.\n";

}

}

// Function to remove an element (Dequeue)

void dequeue() {

if (front == -1 || front > rear)

cout << "Queue is empty!\n";

else {

cout << "Deleted: " << items[front] << endl;

front++;

}

}

// Function to display queue elements

void display() {

if (front == -1 || front > rear)

cout << "Queue is empty!\n";

else {

cout << "Queue elements are: ";

for (int i = front; i <= rear; i++)

cout << items[i] << " ";

cout << endl;

}

}

};

int main() {

cout << "Queue Operations Program \n\n";

Queue q;

int choice, value;

do {

cout << "\n--- Queue Menu ---\n";

cout << "1. Enqueue\n2. Dequeue\n3. Display\n4. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

cout << "Enter value to insert: ";

cin >> value;

q.enqueue(value);

break;

case 2:

q.dequeue();

break;

case 3:

q.display();

break;

case 4:

cout << "Exiting program...\n";

break;

default:

cout << "Invalid choice!\n";

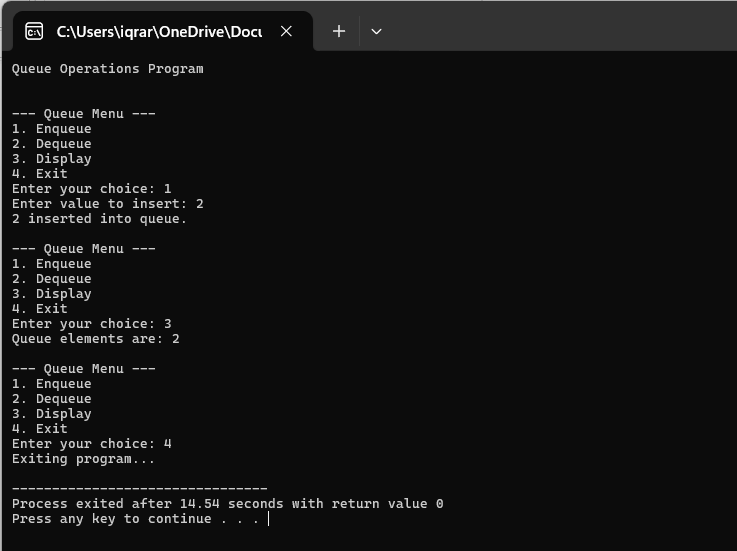
}

} while (choice != 4);

return 0;

}

OUTPUT:



**(2)**

#include <iostream>

#include <queue>

#include <sstream>

using namespace std;

int main() {

cout << "Queue Program Demonstration\n";

cout << "----------------------------\n\n";

string input;

cout << "Enter a string: ";

getline(cin, input);

// Convert the input string into words

stringstream ss(input);

string word;

int queueNum = 1;

// Each word is treated as a separate queue

while (ss >> word) {

queue<char> q; // Queue for each word

for (char c : word)

q.push(c);

// Display the queue content

cout << "Q" << queueNum << " = ";

while (!q.empty()) {

cout << q.front();

q.pop();

if (!q.empty()) cout << " -> ";

}

cout << endl;

queueNum++;

}

return 0;

}

OUTPUT:

