**Foundation University**

**School of Science and Technology**



**Data Structure Lab Report:3**

**Name: Wazzaif Ali**

**Roll No:095**

**Department of Engineering Technology**

## Exercises:

1. Write a menu-driven program to perform different operations with queue such a Enqueue ( ), Dequeue( ) and Display Queue( ).
2. In this Exercise, you have to take a single string as input. Using this input string, you have to create multiple queues in which each queue will comprise of separate word appearing

in the input string. Example: String = “Data Structure and Algo”

Q1 = D → a → t → a

Q2 = S → t → r → u → c → t → u → r → e

Q3 = a → n → d

Q4 = A → l → g → o

## Solution:

#include <iostream>

#include <queue>

#include <string>

#include <sstream>

using namespace std;

int main() {

// Standard library containers and IO are used

queue<int> q;

int choice, value;

do {

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

cout << "1. Enqueue (Add)\n";

cout << "2. Dequeue (Remove)\n";

cout << "3. Display Queue\n";

cout << "4. Create Queues from String\n";

cout << "5. Exit\n";

cout << "Enter your choice: ";

// Input validation and error handling

if (!(cin >> choice)) {

cout << "Invalid Input! Please enter a number.\n";

cin.clear();

cin.ignore(10000, '\n');

continue;

}

if (choice == 1) {

cout << "Enter value to add: ";

if (cin >> value) {

q.push(value);

cout << "Added " << value << " to queue.\n";

} else {

cout << "Invalid Input for value!\n";

cin.clear();

cin.ignore(10000, '\n');

}

}

else if (choice == 2) {

if (q.empty())

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

else {

cout << "Removed " << q.front() << " from queue.\n";

q.pop();

}

}

else if (choice == 3) {

if (q.empty())

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

else {

cout << "Queue elements (Front -> Back): ";

queue<int> temp = q;

while (!temp.empty()) {

cout << temp.front() << " ";

temp.pop();

}

cout << endl;

}

}

else if (choice == 4) {

cin.ignore();

string text;

cout << "Enter a string: ";

getline(cin, text);

stringstream ss(text);

string word;

int count = 1;

// Tokenizes the string into words

while (ss >> word) {

queue<char> q2;

// Modern C++ feature (C++11): Range-based for loop

for (char c : word) {

q2.push(c);

}

// Display logic

cout << "Q" << count++ << ": ";

while (!q2.empty()) {

cout << q2.front();

q2.pop();

if (!q2.empty()) cout << "-";

}

cout << endl;

}

}

else if (choice == 5) {

cout << "Program Ended.\n";

}

else {

cout << "Invalid Choice!\n";

}

} while (choice != 5);

return 0;

}

**Output:**





