Green University of Bangladesh
Department of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering
Semester: (Summer, Year:2022), B.Sc. in CSE (Day)

LAB REPORT NO 03

Course Title: Data Structure Lab
Course Code: CSE 106 Section: DA

Lab Experiment Name:
Quick Sort and Merge Sort in C Programming Language

Student Details

Name	ID
Obaydur Rahman	213902018

Submission Date : 07-08-2022
Course Teacher's Name : Farhana Akter Sunny

Lab Report Status

Marks: ………………………………… Signature: ....
Comments: .... Date:..............................

1. TITLE OF THE LAB EXPERIMENT

- Implement a program for converting an infix expression to postfix expression using stack.
- Implement a circular queue.

2. AIM

Implementing infix expression to postfix expression conversion program and circular queue.

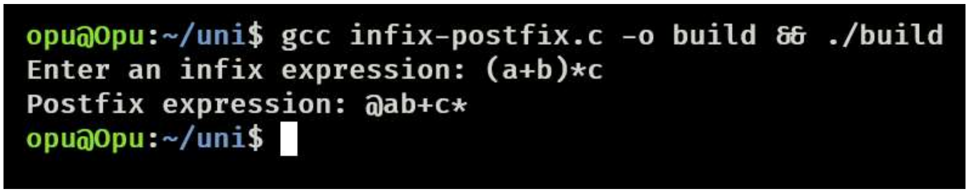
3. DESIGN

Stack: Stack is a linear data structure in which the insertion and deletion operations are performed at only one end. In a stack, adding and removing of elements are performed at a single position which is known as "top". That means, a new element is added at top of the stack and an element is removed from the top of the stack. In stack, the insertion and deletion operations are performed based on LIFO (Last In First Out) principle. In a stack, the insertion operation is performed using a function called "push" and deletion operation is performed using a function called "pop".

Queue: Queue is also an abstract data type or a linear data structure, just like stack data structure, in which the first element is inserted from one end called the REAR (also called tail), and the removal of existing element takes place from the other end called as FRONT (also called head). This makes queue as FIFO (First In First Out) data structure, which means the element inserted first will be removed first.

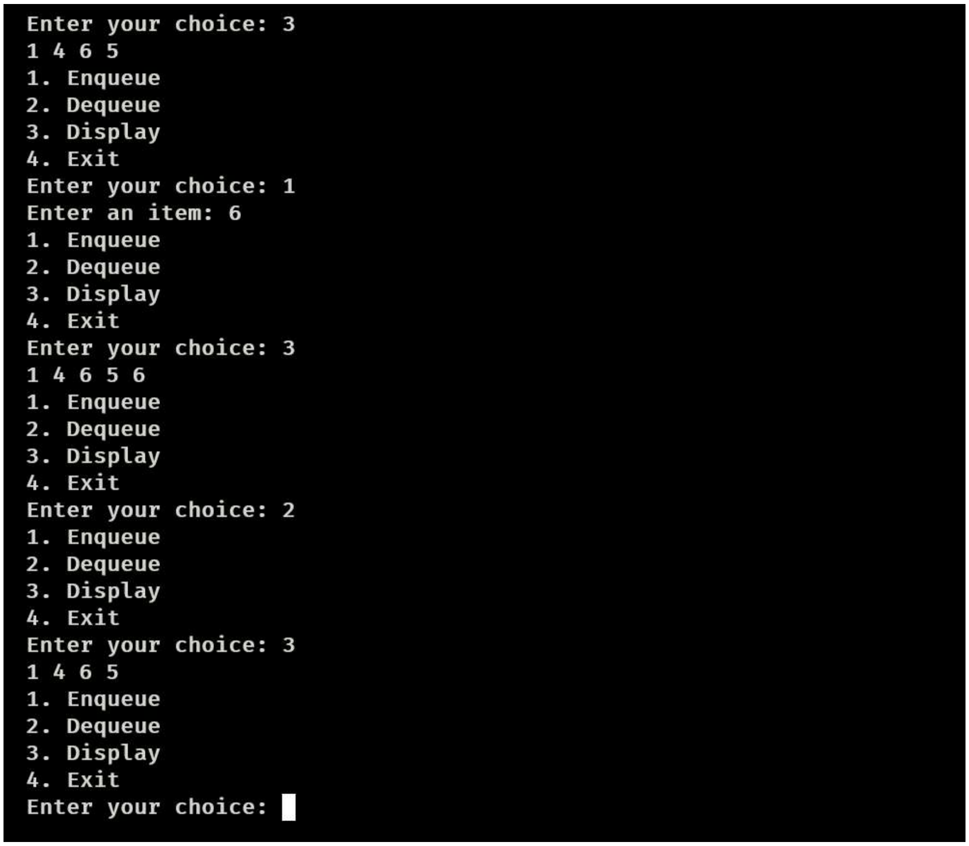
4. TEST RESULT / OUTPUT

Infix to postfix:



```
opu@Opu:~/uni$ gcc infix-postfix.c -o build && ./build
Enter an infix expression: (a+b)*c
Postfix expression: @ab+c*
opu@Opu:~/uni$ 
```

Circular Queue:



```
Enter your choice: 3
1 4 6 5
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 1
Enter an item: 6
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 3
1 4 6 5 6
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 2
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 3
1 4 6 5
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 
```

5. ANALYSIS AND DISCUSSION

By implementing the infix to post fix conversion application and circular queue using stack I learned and understood how stack and queue works.