CC4 Laboratory Activity #6
Prepared by: Rey Benjamin M. Baquirin, MSCS

Topics Covered: Queues

Estimated Completion Time: 1 meeting (2 hours)

Objectives:

1. To appreciate and understand how the queue data structure behaves when inserting and deleting elements.

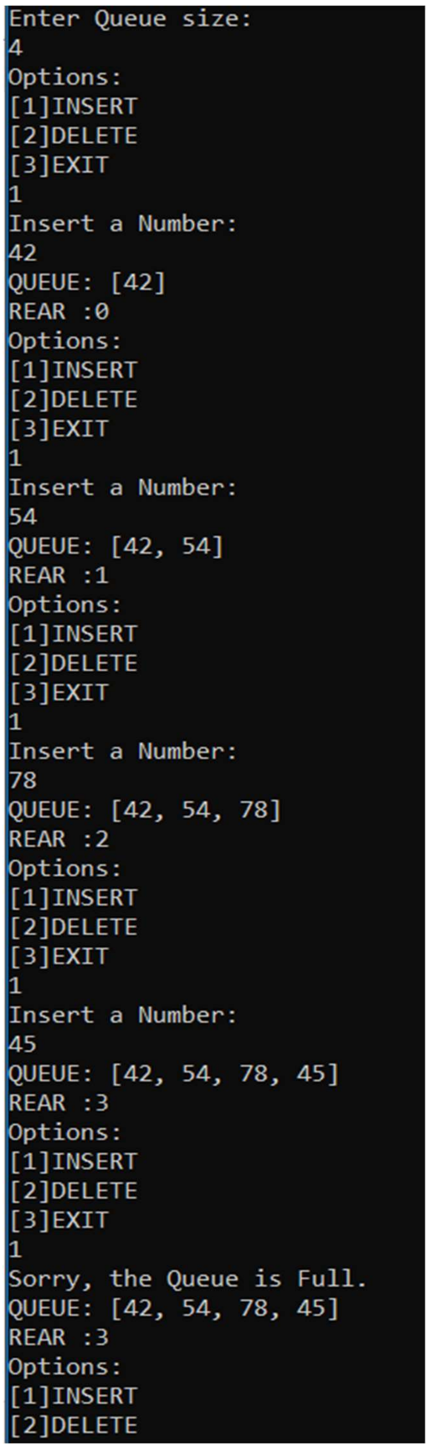
2. To be able to simulate how queues insert and delete elements.

Problem: Create a running program that simulates how the queue data structure works by:

a. Asking the user to choose whether to INSERT, DELETE or EXIT.
b. Writing the appropriate program logic to check whether it is possible to Insert or Delete based on the Rear Index and thereby displaying error messages otherwise.
c. Deleting elements from the Front and Inserting Elements from the Rear.

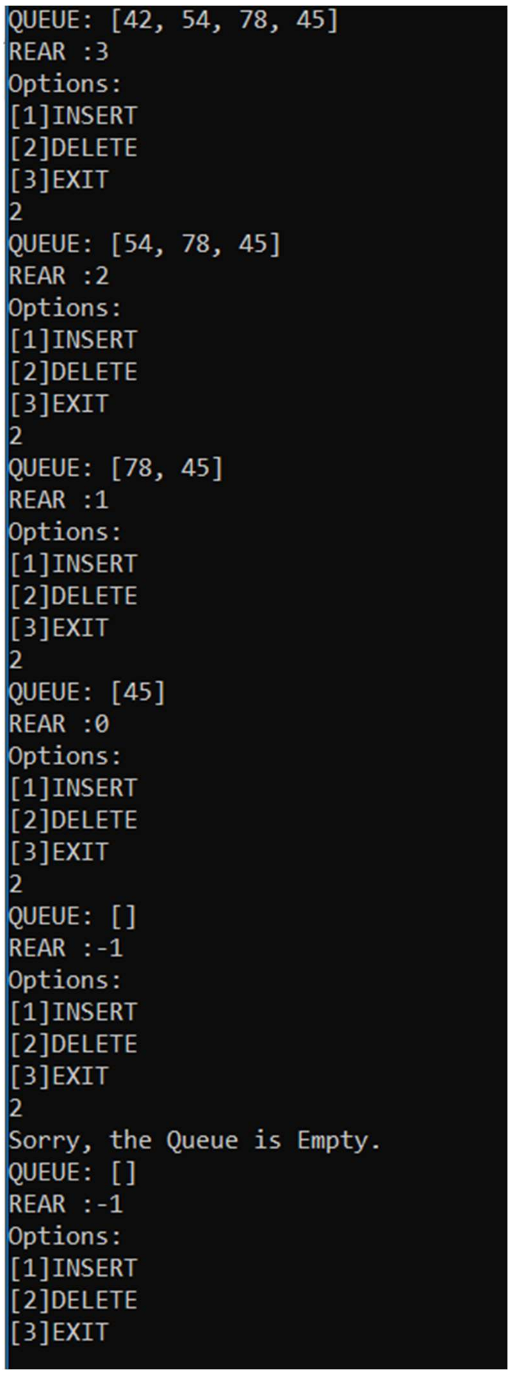
Sample Output:

INSERT



```
Enter Queue size:
4
Options:
[1]INSERT
[2]DELETE
[3]EXIT
1
Insert a Number:
42
QUEUE: [42]
REAR :0
Options:
[1]INSERT
[2]DELETE
[3]EXIT
1
Insert a Number:
54
QUEUE: [42, 54]
REAR :1
Options:
[1]INSERT
[2]DELETE
[3]EXIT
1
Insert a Number:
78
QUEUE: [42, 54, 78]
REAR :2
Options:
[1]INSERT
[2]DELETE
[3]EXIT
1
Insert a Number:
45
QUEUE: [42, 54, 78, 45]
REAR :3
Options:
[1]INSERT
[2]DELETE
[3]EXIT
1
Sorry, the Queue is Full.
QUEUE: [42, 54, 78, 45]
REAR :3
Options:
[1]INSERT
[2]DELETE
```

DELETE



```
QUEUE: [42, 54, 78, 45]
REAR :3
Options:
[1]INSERT
[2]DELETE
[3]EXIT
2
QUEUE: [54, 78, 45]
REAR :2
Options:
[1]INSERT
[2]DELETE
[3]EXIT
2
QUEUE: [78, 45]
REAR :1
Options:
[1]INSERT
[2]DELETE
[3]EXIT
2
QUEUE: [45]
REAR :0
Options:
[1]INSERT
[2]DELETE
[3]EXIT
2
QUEUE: []
REAR :-1
Options:
[1]INSERT
[2]DELETE
[3]EXIT
2
Sorry, the Queue is Empty.
QUEUE: []
REAR :-1
Options:
[1]INSERT
[2]DELETE
[3]EXIT
```