Programming Exercises 03 CMSC 28

QUEUE IMPLEMENTATION

OBJECTIVE

To be able to write a program for the queue implementation and apply it in a program application.

INSTRUCTIONS

1. You are to work on this activity individually.
2. You are to write a code for a program that follows the details in the PROGRAM SPECIFICATION section. There should only be one program (one source code file) for this PE.
3. You are to develop your program following the structured programming approach. No global variable declarations are allowed.
4. You only need to submit the source code file you have created. Name your source code file using your surname followed by the PE # similar to this example: Rizal_03.cpp

PROGRAM SPECIFICATION

This program has two parts: ***queue implementation*** and ***queue application***. You are to implement the queue first and then use that in implementing the program application.

Queue Implementation

Each cell of the queue should be able to hold two integers. The following are the expected operations for the queue:

- INIT(): creates and returns an empty queue
- EMPTY(Q): returns true if queue Q is empty, otherwise, returns false
- FRONT(Q): returns a copy of the front item of queue Q
- ENQUEUE(x, Q): inserts item x in queue Q
- DEQUEUE(Q): removes and returns the front item of queue Q

Queue Application

The following is the specification of the application you will implement using the queue you have created. The application is about a checkout counter system in any large-scale store in the city (NCCC, SM Department store, or even Lots for Less, and other similar stores).

The checkout system has the following main menu:

```
[Name of your chain/shop/store] Counter 0:

[1] Fall in line
[2] Serve customer
[3] Next customer
[4] Closing time
[0] Exit
Enter choice:
```

Note: replace [Name of your chain/shop/store] with the name of your shop or something.

Details of the menu items are as follows…

1. **Fall in line**: Once selected, the program should ask the user for the customer's name (string, you can use lastnames) of the person falling in line and that customer's total amount of the items to be paid (float type, output should be upto 4 decimal places).

2. **Serve customer**: Once selected, the program should proceed with the process of checking out the order of the customer who is first in line. For this process, the customer info (name and total amount of the items to be paid) will be removed from the queue and the program displays a message with the following format:
   "Now serving [name of customer] with total amount payable of [total amount of the items to be paid]."

   If the queue is empty, the program should not proceed with performing the process for the menu and instead display the following message:
   "The QUEUE is EMPTY. No orders to serve."

3. **Next customer**: Once selected, the program should display the name of the customer to be served next. The program should display a message with the following format:
   "Next order: For [customer name] with total amount payable of [total amount of the items to be paid]."

   If the queue is empty, the program should not proceed with performing the process for the menu and instead display the following message:
   "The QUEUE is EMPTY. No order to serve."

4. **Closing time**: Once selected, the program should proceed with the process of serving all the customers left in the queue. For each customer being served, display the same output message format from menu item [2].

   After serving the last customer, the program should inform the user that all the customers have been served.

   If the queue is empty, the program should not proceed with performing the process for the menu and instead display the following message:
   "The QUEUE is EMPTY. No customer/s to serve."

0. **Exit**: Once selected, the program should terminate with a message informing the user that the checkout system has been terminated.

Note: After processing the selected main menu item, the program should loop back to the main menu. The program should end only when the user selects the main menu item Exit.