CL2001 – Data Structure Lab

Lab Task # 07

Note:

- Copied task will be awarded **zero** marks.
- Use comments wherever applicable.
- Submit a pdf file containing all your C++ code with all possible screenshots of every task output on Google Classroom. The name of file should be your roll no followed by your name (roll-no-name.pdf) i.e., (24P-1234-Ali.pdf).
- Variables and functions names should be meaningful.

Problem: 1

You are tasked with developing an application for a cafeteria ordering using a queue data structure implemented with a linked list in C++. The cafeteria serves a variety of food items and beverages, and the system must efficiently manage incoming orders while ensuring that they are processed in the order they are received. Menu item contain id, name and price.

Requirements:

1. **Menu Display:** The application should display a menu of available food items and beverages, along with their corresponding prices.
2. **Order Placement:** Customers should be able to place orders by selecting items from the menu. Each order should be added to a queue for processing.
3. **Order Processing:** Orders should be processed in the order they are received. As orders are processed, they should be removed from the queue, and the next order should be displayed.
4. **Queue Management:** The application should provide options to add new orders, remove processed orders, and display the current orders in the queue.

Problem: 2

You are tasked with creating a menu-driven queue management system for a doctor's office. The system should allow the doctor's attendant to add patients to a queue, process them one by one, see who's next, and display the patient queue. Here's a detailed breakdown of the task:

Implement a simple queue data structure to manage the flow of patients waiting to see the doctor.

Your program should first as how many seats are available in the waiting area, it means the max no of patients that can wait in the queue.

Create a menu-driven system with the following options:

1. Add Patient to Queue: The attendant can add a patient's name to the end of the queue.
2. Send next patient to the doctor: The attendant can process the next patient in the queue and send them to the doctor for examination. Remove the patient from the queue at this stage.
3. See who's next: The program should only display the name of the next patient.