Project – Hospital Management System

1 Introduction

This is a GROUP project that contributes to 20% of the total grade of this course. Each group shall consist of 4 or 5 members.

2 Objectives

- To evaluate students' skills at writing a program using C++ programming language.
- To apply **linked list** data structures along with **stack** or **queue** data structures to store and manage data.
- To utilize **searching** and **sorting** algorithms to support data manipulation.

3 Hospital Management System

This project involves designing and implementing a comprehensive Hospital Management System using the C++ programming language. The goal is to create an efficient system capable of managing patient records, scheduling appointments, and facilitating medical treatments. This system aims to improve the operational efficiency of hospital staff, allowing them to access and update patient information, manage scheduling, and track treatment processes easily.

4 Project Requirements

Write a program in C++ to demonstrate your ability to apply stack or queue data structures, searching and sorting techniques in designing your solutions.

Data Management:

- Use a linked-list to manage an inventory of patient records, consisting of patient ID, name, and treatment type.
- Implement stacks or queues to handle patient admissions and discharges.

Transaction Handling:

- Track the medical treatment history of each patient using a separate list.
- Implement functions to add new treatments, delete old or canceled appointments, and edit existing patient records.

Search and Sort Functionality:

- Ability to search for patients based on name, patient ID, or treatment type.
- Implement sorting algorithms to organize patients based on treatment dates, priority of medical attention, and length of stay.

Reports and Summaries:

- Generate a summary report displaying the status of admitted patients, patients scheduled for discharge, and out-patients treatments sorted by department.
- Choose from bubble, selection, insertion, shell, quick, and merge sorting techniques for organizing summary reports.

User Interface Functions:

- Patient Display: Show all patients listed by admitted and scheduled for Discharge patients.
- Transaction Management: Manage admissions, discharges, and medical treatments.
- Detailed Views: Ability to view detailed information and status of each patient.

Administrative Functions:

- Manage medical records, generate reports, and update patient statuses.

System Operation:

- Your program should operate continuously until the user chooses to terminate the program.
- Include at least ten (10) preset patients in your program as hardcoded data.

5 Project Presentation

Presentation: Each group is required to do a maximum of 15 minutes (including Q&A) presentation of their project in class. All group members **MUST** take part in the presentation. Students must wear appropriate attire for the presentation.

6 Submission

- All documents must be submitted via the link provided in eLEAP latest by **24th June 2024 (Monday) by 5.00pm**. Any late submission will get 20% of Project mark penalty deduction per day;
- Submit your answer in a zipped file containing only the working C++ source codes (*.h and *.cpp). Use the following zipped file naming format for submission:

  LecGroupNo_ProjectGroupNo#.zip *E.g: LG1_ProjGroup01.zip*
- Each group is required to submit *only* one copy of the group assignment.
- Files submitted with *incorrect format* will *not be entertained*.
- Any plagiarism (similarity between two files more than 50%) will be graded as a zero for the two files.
- Attach snapshots or files showing evidence of your team's online collaborations in your report (e.g., social media chats, emails, collaboration tool screenshots, etc.).