CS2106: Data Structures and Algorithms – Final Project

Due Date 08th February 2023

Requirements

·   Use 3 to 4 data structures that we covered in class(queue, stack, binary search tree, linked list,) in a hypothetical real-life management system.

·   Justify the use of the chosen data structures for the problem.

·   Creative ideas are encouraged and welcome!

Code

- Appropriate use of variables
- Appropriate use of methods
- Clear and comprehensive commenting
- Easy to modify (e.g. via instance variables)
- Correct algorithms
- Made good use of appropriate features of the python language
- Good code layout
- Reliability of code (error handling)
- Source code(to be submitted as a soft copy)

CLI

- All CLI elements working
- Appropriate use of CLI elements
- How meaningful are any messages and screen prompts?
- Intuitive design
- Data validation
- Output formatted
- Graphic display of output

APPLICATION ENHANCEMENTS

- Add enhancements to the implementation

REPORT

- Minimum 1200 words.
- Discussion justifying the design of your application.
- Discussion of any bugs and/or weaknesses in your change algorithm, if you don't think there are any then say so. You should consider uncaught exceptions to be a weakness.

- ⬚ Discussion of strong points or novel features of your application.
- ⬚ A critical discussion and evaluation of your learning experiences, reflection.
- ⬚ Layout, formatting and structure of report.

Example of Project Title

A hospital management system involves scheduling incoming patients' appointments with doctors based on the reason of admission to the hospital.

Patients appointments are made on a first come first served basis. A doctor has a weekly hourly schedule where s/he is either: available for an appointment, on appointment, or outside office. If the patient already visited the hospital for the same doctor specialty, it is desirable that the patient be scheduled to the same doctor s/he previously visited.

- · Doctor schedule can be represented using a 2-D array of dimension 7(days)*24(hours). The values inside the array (0, 1, or 2) can represent the 3 availabilities status of the doctor.
- · Patients are scheduled an appointment with a doctor based the order of their arrivals, so a queue data structure of patients can be used.
- · Patients record need to be stored for fast lookup. Therefore, a BST might be used to store the patient record with the SSN as the unique search key.

Example of Hospital Management Menu:

1. Lookup Patient Information
2. Add a patient to the queue
3. Schedule Appointment for enqueued patients
4. Add New doctor
5. Lookup doctor Schedule

Example of Additional Functionalities:

1. Group doctors by departments and schedule patients to doctors from the right department based on their symptoms (a symptom-specialty hash map can be used).
2. Add dedicated nurses to each doctor and schedule an available appointment for the doctor and a nurse.
3. Make a priority queue of patients based on the urgency of the patient case.