**Term Project (2): Patient Management System**

CIST2742

**Program specifications**

This project is to develop a program that manages patient information and the procedures a patient received in a hospital. The patient information includes:

Patient ID  
First name, last name  
Address, city, state, ZIP code  
Phone number  
Name and phone of emergency contact

Procedure information include:

Patient ID  
Name of the procedure  
Date of the procedure  
Name of the practitioner who performed the procedure  
Chares for the procedure

Here are some sample procedure data

Procedure name: Physical Exam  
Date: Today’s date  
Practitioner: Dr. Irvine  
Charge: 250.00

Procedure name: X-ray  
Date: Today’s date  
Practitioner: Dr. Jamison  
Charge: 500.00

The program is to support these functionalities:

* Add a patient and his or her procedure(s)
* Add procedure for an existing patient
* Look up for a patient by patient ID or first and last name, display the patient information and his or her procedure(s), calculate total charges
* Change patient information
* Delete a patient and all his or her procedure(s)
* Importing data (patient info and procedures) into the system from CSV text files, e.g., the patient.txt and procedure.txt files provided
* Export data to text files as json documents: patient.json and procedure.json for data exchange

Present a menu that lets user perform the above tasks

**Some suggestions:**

* Develop two classes: Patient and Procedure
* Store Patient objects and Procedure objects in two dictionaries, with Patient ID as the key, and persist the dictionaries to separate data files, e.g., patient.dat and procedure.dat.
* On exit, the program saves (pickle) the dictionaries to the data files. On start, the program loads (unpickle) data from the files. If a file does not exist, create an empty dictionary.

**Extra points:**

Develop GUI for the program.

**Requirements:**

This is an individual project, each student is required to complete it independently

The system developed should work as specified

The program should be well-structured and modularized for code-reuse and easy maintenance

The program should be well-documents using docstring

The program should follow the conventions specified in PEP 8 -- [PEP 8 -- Style Guide for Python Code | Python.org](https://www.python.org/dev/peps/pep-0008/).

In addition to a working program, screenshots of test runs should be also submitted.