## **INFSCI 0017 – Fundamentals of Object-Oriented Programming (Fall 2018)**

## **Lab 8**

## Topics Reviewed

1. Classes & Objects
2. Class constructors
3. Visibility

## Grading and Submission

You are to write a complete Java program that meets the requirements outlined in the Lab 8 Tasks section.

Once you have completed the program, you should demonstrate your program for your Lab TA.

There will be 10 points for this lab, 2 points for each class described below

Note that if your program does not compile, the TA will not grade it.

## Lab 8 Tasks

**INFSCI 0017 – Fundamentals of Object-Oriented Programming (Fall 2014)**

**Homework 4**

1. Create a new Java project called EMR.
2. Use the class diagram(s) below to create a skeleton for your Java program. Start by creating classes, then add properties and then method stubs. Pay close attention to visibilities.

|  |
| --- |
| **EMR** |
|  |
| +main():void |

|  |
| --- |
| **Patient** |
| -patientID:String  -firstName:String  -lastName:String  -ssn:String  -gender:char  -diagnoses:Diagnoses  -medication:Medication  -symptom:String  -weight:double  -height:double |
| +Patient(firstName:String, lastName:String, ssn:String, gender:char, weight:double, height:double)  +calculateBMI():double |

|  |
| --- |
| **Doctor** |
| -employeeID:String  -firstName:String  -lastName:String  -ssn:String  -specialization:String |
| +Doctor(firstName:String, lastName:String, ssn:String)  +prescribe(patient:Patient):void |

|  |
| --- |
| **Medication** |
| -medicationID:int  -name:String |
| +Medication(medicationID:int, name:String) |

|  |
| --- |
| **Diagnoses** |
| -diagnosesID:int  -name:String |
| +Diagnoses(diagnosesID:int, name:String) |

1. **DO THIS ITEM LAST!  
   EMR** class is the driver class (the class with the main method, the class the *run*). The main method should do several things:
   1. Create an instance of Patient (a new variable of type Patient). You can make up the data that you pass to Patient’s constructor. Look at the EMR Driver class example at the end of this document.
   2. Ask patient to enter a symptom using JOptionPane.showInputDialog(…);
   3. Set Patient’s symptom property to provided value using appropriate setter.
   4. Create Doctor object
   5. Make the doctor *prescribe* to the patient (see classes Doctor and class Patient)
   6. Output a "profile" of the patient, including complete name, SSN, BMI, diagnoses and prescribed medication. If patient has no diagnoses, nor prescription (see method prescribe in class Doctor), the profile should show "none".
2. **Diagnosis** class:
3. Diagnosis constructor will initialize corresponding properties.
4. **Medication** class:
5. Medication constructor will initialize corresponding properties.
6. **Patient** class:
   1. Patient constructor will:
      1. Initialize corresponding properties
      2. Initialize patientID property based on a social security number (ssn property) with all dashes and spaces stripped followed by first letter of first name and first letter of last name.
   2. calculateBMI() method will:
      1. Calculate patient’s BMI (<https://www.cdc.gov/healthyweight/assessing/bmi/childrens_bmi/childrens_bmi_formula.html>)
      2. Return the bmi value
7. **Doctor** class:
   1. Doctor constructor will:
      1. Initialize corresponding properties
      2. Initialize employeeID property based on a social security number (ssn property) with all dashes and spaces stripped followed by first letter of first name and first letter of last name.
   2. prescribe(…) method will complete the following actions:
      1. Match the symptom of the patient parameter with diagnoses and medication (see table below for the match list).
      2. If symptom matches with diagnosis:
         1. Create Diagnosis object with appropriate parameters for the constructor.
         2. Create Medication object with appropriate parameters for the constructor.
         3. Set patient’s diagnosis and medication properties with recently created objects using corresponding setters.

**Hint**: use setters from the Patient object passed to the prescribe() method

* + 1. If symptom does not match with diagnosis:
       1. Set Patient’s diagnosis and medication properties to ***null*** using corresponding setters.
       2. Display a message telling patient that he/she cannot be diagnosed.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Symptom** | **Diagnosis** |  | **Medication** |  |
|  | **ID** | **Name** | **ID** | **Name** |
| Headache | 1 | Dehidration | 1 | Tylenol |
| Cough | 2 | Common cold | 2 | Cough drops |
| Fever | 3 | Influenza | 3 | Tamiflu |

**EMR Driver class example (the following code will go into main() method):**

***Note: if you are going to copy and paste this code, you’ll need to fix Microsoft’s weird quotes.***

Patient patient = new Patient("John", "Doe", "111-11-1111", 'm', 170, 70);

String symptom = JOptionPane.showInputDialog("Please enter your symptom:");

patient.setSymptom(symptom);

Doctor doctor = new Doctor("Jane", "Smith", "222-22-2222");

doctor.prescribe(patient);

String patientProfile = "Name: " + patient.getLastName() + ", " + patient.getFirstName() + "\n";

patientProfile = patientProfile + "SSN: " + patient.getSsn() + "\n";

patientProfile = patientProfile + "BMI: " + patient.calculateBMI() + "\n";

if(patient.getDiagnoses() != null){

patientProfile = patientProfile + "Diagnoses: " + patient.getDiagnoses().getName() + "\n";

}

else{

patientProfile = patientProfile + "Diagnoses: none\n";

}

if(patient.getMedication() != null){

patientProfile = patientProfile + "Medication: " + patient.getMedication().getName() + "\n";

}

else{

patientProfile = patientProfile + "Medication: none\n";

}

JOptionPane.showMessageDialog(null, patientProfile);