KING SAUD UNIVERSITY COLLEGE OF COMPUTER AND INFORMATION SCIENCES Computer Sciences Department		
CSC 111: **Introduction to Programming- I**	**Final Lab**	**1st Semester 1442**

Develop a java program to help a hospital in managing their patients. Consider the below UML and description:

Patient
- age : int - name : String - totalTestCost: double - bloodTests : String[] + **numPatients: int**
+ Patient (String nameP, int ageP , int numTests) +readTests () : void +countPositiveTests (): int +getAge(): double +getName():String +display():void

hospital
- Patients: Patient[]
+ main (String [] args): void **+displayOlderThan(int: age):void**

1. **Implement Class Patient**
   Attributes:
   - age: an int representing the age of the patient.
   - name: a String representing the name of the patient.
   - totalTestCost: a double representing the total cost of all blood tests done by the patient. All new patients have TotalTestCost equal to 0.0.
   - bloodTests: an array of Strings containing the names and results of blood tests the patient has done.
     - All blood tests have the format : *testName-result* : which is the name of the test followed by a hyphen followed by the result. The result can be either "positive" or "negative". Both name and result can be of any length and vary from one test to another. (Assume that the result will always be written in small letters)
   - numPatients: a class attribute that counts the number of patients created.

   Methods:
   - **Patient (String nameP, int ageP , int numTests):** A non-default constructor that receives the patient name, patient age, and number of blood tests he have done.
   - **readTests() :** This method reads from the user information about the patient's blood tests ; according to the number of blood tests conducted. For each blood tests it reads: its name (including the result) and price and saves the name in the blood tests array and adds the cost of the test to the totalTestCost.
   - **countPositiveTests ():** This method counts and returns the number of blood tests having a positive result.
   - **getAge():** Returns the age of the patient.
   - **getName():** Returns the name of the patient.

- **display() :** This method prints the information of the patient as follows:

  Name: <Patient name >
  age: <patient age > **years**
  Cost : <totalTestCost > **SR**
  Blood Tests Performed:
  <Blood tests names and results each in a separate line>

2. Create an application class named Hospital according to the above UML.

a. Create an array named ***Patients*** that can hold up to 100 patients.

b. Write the method ***displayOlderThan(int: age)*** This method displays the information of all patients older than the received age.

c. In the main method:
   i. Read the information of three patients from the user (including their blood tests) and add them to the array *patients*.
   ii. Display the information of all patients older than 50 years.
   iii. Read a patient name from the user, then count and display how many positive blood tests he has. If the patient is not found in the hospital, display an appropriate message (Assume names are unique).
   iv. Print the number of patients, DO NOT use a counter or loops in Hospital class.