CS234 Computer Science II

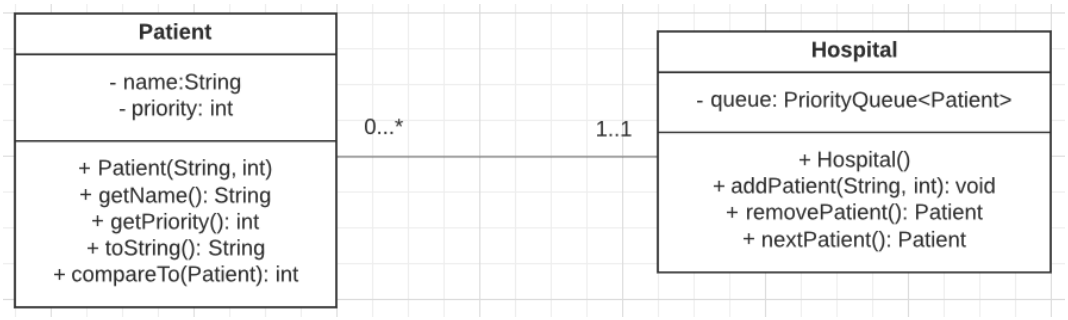
Lab 10
Total points: 100

Read the instructions carefully.

For this laboratory you need to create the definition of **two** Java **classes**. You will practice:

- Use of Priority Queues to store objects with priority.

For this lab you need to **create** the following **classes** (no main methods PLEASE!!)



How many Java files?

Diagram elaborated in Lucidchart. It is free! www.lucidchart.com

Important information.

Patient class

In this Hospital, the higher the priority number the higher the priority. For example, priority 10 is higher than priority 1. Therefore, a patient with priority 10 must be called first.

Patient(). Constructor to initialize an object. It receives the patient's name and priority.
getPriority(). Method to return the patient's priority.

getName(). Method to return the patient's name.

toString(). Method to print the content patient's object.

compareTo(). Method to implement the Comparable interface.

Hospital class

Hospital(). Constructor to initialize the Priority Queue of patients.

addPatient(). Method to add a new patient and their priority to the Priority Queue.

removePatient(). Method to remove a patient from the priority queue.

nextPatient(). Method to return the next patient in line

How to **test** your program?
You can implement a **tester program** like the following:

```
public static void main(String[] args) {

    Hospital StEllen = new Hospital();
    StEllen.addPatient("Eduardo", 3);
    StEllen.addPatient("Carlos", 4);
    StEllen.addPatient("Emma", 1);
    StEllen.addPatient("Sarah", 4);
    StEllen.addPatient("Carol", 5);

    System.out.println("Next patient is: " + StEllen.nextPatient());

    StEllen.removePatient();
    System.out.println("Next patient is: " + StEllen.nextPatient());
}
```

The **output** of the **tester** program is the following:

```
Next patient is: Carol (Priority: 5)
        The Doctor finished with the patient
Next patient is: Sarah (Priority: 4)
```

Please trace the tester program to **understand** the output.

Submission details:

Upload a **single ZIP** file.
Name your file as follows: **Lab10_Lastname_Firstname.zip**

Your **.zip** file must contain the following:

1. Your **.java** source file for your **class definitions** (.java files without the **main** method. No .class file). **Do not send the tester program**.
2. A **SINGLE PDF** with screenshots from your program running. Do not send .jpg files.

For this lab, you **do not need to submit the .txt** file with your instructions*. **Why?** Because I will use my tester programs to use your classes. Therefore, it is **extremely important** that your **class and method names** are **the same** as they are shown in the class diagram and in my tester program.

***Note:** If you want to use packages, you need to submit the .txt file indicating how to use your package (i.e., how the path should be created)
In each .java file, **write** as a multiline comment at the beginning of the file the following:

1. Your name

The **zip** file must be uploaded to Canvas.

I do not accept image files; it must be a PDF file.

Make sure to check the **due date** for this activity on Canvas. Try to submit it before the due date so you can have time to check for improvements. **No late submissions. I do not accept solutions via email or as comments on Canvas.**

Make sure you are **submitting the correct files**. I will grade the files uploaded to Canvas.

Make sure you **test** your **classes** with a similar **tester program** as the ones I am showing in this lab (i.e., a .java program with a main method where you create objects from your class).

Use the *javac* and *java* commands with the tester program **before** submitting your solution. Compiling the tester program should **implicitly compile the other java files**. You don't compile them one by one. Just the tester program.

Make sure to review the **grading rubric.**

Read all the instructions carefully.

If you have questions, contact me before making assumptions about what you need to do for solving this assignment.

It is YOUR responsibility to contact me if something from these instructions is not clear or ambiguous.

Ensure that your code is original and developed by you.