Assignment#7: **Inheritance, polymorphism, Interfaces:**

- **This assignment is worth 25% of the final exam Grade**
- **Due date 07/27/2020 at 8:00AM**

---

1) Write a class called **Room**, which has three **private** instance variables:

   a) a double width, representing the width of the room in feet,

   b) a double length, representing the length of the room in feet, and

   c) an int floor, representing the building floor that the room is on.

2) Write a default constructor for the class Room that sets the width to 10, the length to 12.5, and the floor to 1.

3) Write get and set methods ("getters" and "setters") for the three instance variables. For the set methods for the width and length, only positive values should be set. If the input is 0 or a negative number, the variables should not be changed.

4) Write a constructor for the class Room that takes in two double parameters and an int, and sets length to the larger double, width to the smaller double, and floor to the int. Use the setters from part (b).

5) Override the default toString method for the class Room, so when called by an instance of Room created by the default constructor from part (a) would return the String:

   "Room properties:

   - Size: length = 12.0 x width = 10.0,

   - Floor#: 1"

6) Write an interface **measurable** that has only one method called computeAreaRoom().

7) The class **Room** has to implement the interface **measurable**.

8) Write a subclass of the class Room called **Classroom**, which also has a private instance variable of type int called numStudents, representing the maximum number of students that the classroom can hold.

9) Write a constructor for Classroom which takes in two double variables and two ints. The instance variable length should be set to the larger double, and the width should be set to the smaller double as in Room. The instance variable floor should be set to the first int, and the instance variable numStudents should be set to the second int. *Leave the instance variables as private in Room and use setters to access them.*

10) Override toString method for **Classroom** which uses the toString method for Room, followed by an additional String:

"- Type: classroom

- Capacity: numStudents students"

where numStudents is replaced by the instance variable value.

11) Write a subclass of the class Room called **officeroom**, which also has a private instance variable of type int called numEmployee, representing the maximum number of employees that the officeroom can hold.

12) Write a constructor for **officeroom** which takes in two double variables and two ints. The instance variable length should be set to the larger double, and the width should be set to the smaller double as in Room. The instance variable floor should be set to the first int, and the instance variable numEmployee should be set to the second int. *Leave the instance variables as private in Room and use setters to access them.*

13) Override the toString method for **officeroom** which uses the toString method for Room, followed by an additional String:

"- Type: office

- Capacity: numEmployee empoyees."

where numEmployee is replaced by the instance variable value.

14) Write the class **RoomsDemo** with the main method, and follow the instructions bellow:

a) Declare an array called **rooms** of three objects of type Room

- ..... **[] rooms =new** ..... **[3];**

b) Assign to the first element of the array an object of type Room with: width=8, length=15 and floor =1.

- **Rooms [.....] = new** .....

c) Assign to the second object an object of type Classroom with: width=30, length=50, floor =2 and 20 students.

- **Rooms [.....] = new** .....

d) Assign to the third object an object of type Officeroom with: width=20, length=25, floor =3 and 3 employees.

- **Rooms [.....] = new** .....

e) Use a for loop to get the following output (do not forget to display the areas):

```
Room properties:
 - Size: length = 15.0 x width = 8.0.
 - Floor#: 1
 - Area: 120.0 square feet.

Room properties:
 - Size: length = 50.0 x width = 30.0.
 - Floor#: 2
 - Type classroom
 - Capacity: 20 students
 - Area: 1500.0 square feet.

Room properties:
 - Size: length = 25.0 x width = 20.0.
 - Floor#: 3
 - Type: office
 - Capacity: 3 empoyees.
 - Area: 500.0 square feet.
```

Files to submit: Room.java, Classroom.java, Officeroom.java, Measurable.java and RoomsDemo.java

---

Grading Criteria:

1- Room.java (25pts)
2- Classroom.java (20pts)
3- Officeroom.java (20pts)
4- Measurable.java (10pts)
5- RoomsDemo.java (20pts)
6- Similar output (5pts)

Total points: 100pts
