Object Oriented Programming

Home Work 06 **Marks 10**

Instructions

Work on this home work individually. Absolutely NO collaboration is allowed. Any traces of plagiarism would result in a ZERO marks in this homework and possible disciplinary action. Tasks should be coded in **C++**.

Due Date

Paste the solution of the problem (source code .cpp file only) labeled with your complete roll number in SEM – HW 06 and SEA – HW 06 folders for SE Morning and SE Afternoon sections respectively on Tuesday, April 05, 2016 before 05:00 PM. These folders are available at \\printsrv\Teacher Data\Umair Babar\Students.

ADT: Circle

Write a **Circle** class having following functionalities

1. The class should have following **four private data members**.
    1. An **integer** named **x** that holds the **x-axis** of a circle.
    2. An **integer** named **y** that holds the **y-axis** of a circle.
    3. A **float** named **radius** that holds the **radius** of a circle.
    4. A **constant double** named **PI** that holds the **pi's value** i.e. **3.14159**.

    Value should only be assigned to data members **x and y** when it is **greater than or equal –50 and lesser than or equal to 50**, **0** otherwise and to **radius** when it is **greater than or equal 1 and lesser than or equal to 10**, **5** otherwise.
2. Provide the implementation of **mutators** for **x, y** and **radius** data members of the class.
3. Provide the implementation of **accessors** for all the data members (x, y, radius and PI) of the class.
4. Provide the implementation of following **constructors** and a **destructor**
    1. A **constructor** that accepts **Circle's x, y** coordinates and **radius** as arguments and assigns them to the appropriate member variables.
    2. A **constructor** that accepts **Circle's x**, **y** coordinates as arguments and assigns them to the appropriate member variables. The **radius** field should be assigned the default value.
    3. **A constructor** that accepts **Circle's x** coordinates and **radius** as arguments and assigns them to the appropriate member variables. The **y** coordinates should be assigned the default value.
    4. A **default constructor** that initializes all the data members of the class with **default values**.
    5. A **copy constructor** to initialize a circle's object with already existing object.
    6. A **destructor** that do nothing except displaying a simple message "Destructor executed…" on the screen.
5. Provide the implementation of following member functions
    1. **setCircle** method accepts **Circle's x**, **y** coordinates and **radius** as arguments and assigns them to the appropriate member variables.
    2. **getCircle** method to **initialize the data** of a circle **taken** from the user.
    3. **putCircle** method to display the **information** of a particular circle.
    4. **getArea** method calculate and return the **area** of a circle that is $PI * radius^2$
    5. **getDiameter** method calculate and return the **diameter** of a circle that is $radius * 2$
    6. **getCircumference** method calculate and return the **circumference** of a circle that is $2 * PI * radius$ that is
    7. **addCircle** method should accept **two circle objects** and return there **sum**.
    8. **isEqual** method should accept **two circle object** and return **true** if they are having same state, **false** otherwise.
    9. **findCircle** method should accept an **array of Circle objects** and return the **index** of the array which is equal to left hand side object, **–1** otherwise.
    10. **updateObjects** method should accept an **array of Circle objects** with its **size** and **update the radius** of all those objects to the **radius** of left hand side object exist in the array having same **x, y coordinates** as of left hand side object.
6. Once you have written the class, write **main** function and test its functionality by creating some objects of **Circle**.

NOTE: - No submission will be accepted after the due date and time.

Best of Luck