Workshop #11

Program: ***Wk11.cpp, GeometricObject.h, GeometricObject.cpp, DerivedCircle.h, DerivedCircle.cpp, DerivedRectangle.h, DerivedRectangle.cpp***
Description: Inheritance

Consider geometric objects. Suppose you want to design the classes to model geometric objects like circles and rectangles. Geometric objects have many common properties and behaviors. They can be drawn in a certain color, filled or unfilled. Thus, a general class `GeometricObject` can be used to model all geometric objects.

This exercise provides you the base class `GeometricObject` (in *GeometriObject.h* and *GeometricObject.cpp*), please complete a derived class called `Circle` and a derived class `Rectangle`. Both derived classes inherit from the `GeometricObject` class.

You need to create a pair of files (*DerivedCircle.h* and *DerivedCircle.cpp*) for the `Circle` class and another pair of files (*DerivedRectangle.h* and *DerivedRectangle.cpp*) for the `Rectangle` class.

For the `Circle` class, it has only one private data member called `radius` (type double). You need to define three different constructors to initialize the data members in the `Circle` class and the `GeometricObject` class. The default constructor only needs to initialize `radius` to 1 (It will call the default constructor in the `GeometricObject` class implicitly.). The first non-default constructor only needs to initialize `radius`. The second non-default constructor initializes not only `radius`, but also the `color` and `filled` in the `GeometricObject` class. It also defines a couple of public member functions: a get function and a set function to access radius (`getRadius()` and `setRadius()`), a get function to return the area (`getArea()`), a get function to return the perimeter (`getPerimeter()`) and a get function to return the diameter (`getDiameter()`). Like the `GeometricObject` class, the `Circle` class also has a `toString()` function, but it returns "`Circle` object". For the radius, your program needs to verify if it is greater than or equal to 0 in the `setRadius()` function. If the `radius` is smaller than 0, set up `radius` to 0. `Pi` = 3.14159.

For the `Rectangle` class, it has two private data members: `width` and `height` (type double). It also has three constructors. The default constructor needs to initialize both `width` and `height` to 1. The first non-default constructor takes two arguments to initialize `width` and `height`. The second-default constructor takes four arguments to initialize `width`, `height`, `color`, and `filled`. It also defines several public member functions. A get function and a set function to access `width` (`getWidth()` and `setWidth()`), a get function and a set function to access `height` (`getHeight()` and `setHeight()`,) and a get function to return the perimeter (`getPerimeter()`). Like the `GeometricObject` class, the `Rectangle` class also has a `toString()` function, but it returns a `Rectangle` object. For `width` and `height`, your program needs to verify if each one is greater than or equal to 0 in (`setWidth()` and `setHeight()`). If the `width` or `height` is less than 0, set up width and height to 0.

For both the `Circle` and `Rectangle` classes, you must specify **constant member functions** if they do not modify any data members.

Use the provided *wk11.cpp* to test your classes.

(*Continued on next page*)

Sample runs:

```
Geometric Object
 color: red filled: true
Circle object
 color: black filled: false radius: 5 area: 78.5397 perimeter: 31.4159
Rectangle object
 color: Orange filled: true width: 2.5 height: 3.2 area: 8 perimeter: 11.4
Geometric Object
Geometric Object
Geometric Object
```

Change the *`toString()`* function in the `GeometricObject` class to a pure function. The output should be as follows:

```
Geometric Object
 color: red filled: true
Circle object
 color: black filled: false radius: 5 area: 78.5397 perimeter: 31.4159
Rectangle object
 color: Orange filled: true width: 2.5 height: 3.2 area: 8 perimeter: 11.4
Geometric Object
Circle object
Rectangle object
```