Object-Oriented Programming (3190)

Homework 3

Spring 2023

1. [5pt] Create a ***Sphere*** class that inherits from a ***Circle*** class. We know that we can create a sphere object by turning a circle around its diameter.

a. Design an interface for a class named ***circle*** with one private data member: ***radius***. Define a parameter constructor and a destructor for the class and write member functions to find the perimeter and area of a circle using the following relationships. (**getPerimeter, getArea, print**)

perimeter = 2 * π * radius area = π * radius * radius

b. Define an interface file for a class named ***Sphere***. Also define a parameter constructor and a destructor for the class and write member functions to find the surface and volume of a sphere. (getSurface, getVolume, print)

c. Define an implementation file for the class Sphere using the following formulas to find the surface and the volume of a sphere; in the formulas, ***perimeter*** is the perimeter of the circle and area is the ***area*** of the circle defined in part (b).

surface = 2 * radius * perimeter volume = (4 / 3) * radius * area

d. Write an application file to test the ***Circle*** and ***Sphere*** classes. Ensure that your output matches the following output and test your code with at least two more test cases on your own.

Possible Output:

```
Run:
Circle:
Radius: 6
Perimeter: 37.70
Area: 113.10

Sphere:
Radius: 6.00
Surface: 452.39
Volume: 904.78
```

2. [5pt] Create a ***Cylinder*** class that inherits from a ***Circle*** class. We know that we can create a cylinder object by adding a height to the circle object.

a. Design an interface for a class named ***Circle*** as done in 1.(a).

b. Define an interface file for a class named ***Cylinder***. Also define a parameter constructor and a destructor for the class, and write member functions to find the surface and volume of a sphere. (**getSurface, getVolume, print**)

c. Define an implementation file for the class ***Cylinder*** using the following formulas to find the surface and the volume of a cylinder; in these formulas, perimeter is the perimeter of the circle and area is the area of the circle defined in part (b).

```
surface = height * perimeter + 2 * area
volume = height * area
```

d. Write an application file to test the ***Circle*** and ***Cylinder*** classes. Make sure that your output matches the following output and test your code with at least two more test cases on your own.

Possible Output:

```
Run:
Circle:
Radius: 6.00
Perimeter: 37.70
Area: 113.10

Cylinder:
Radius: 6.00
Height: 4.00
Surface: 376.99
Volume: 452.39
```

3. [5pt] Design a class name ***Square*** that defines a square geometric shape. The class must have a data member named ***side*** that defines the length of each side. Then define two member functions, ***getPeri*** and ***getArea***, to find the perimeter and area of the square shape. Now define a ***Cube*** that defines a cubic shape and inherits from the ***Square*** class. The class ***Cube*** needs no new data members, but it needs the member functions ***getArea*** and ***getVolume***. Provide the appropriate constructors and destructors for both classes. Provide ***print*** member function for both classes to show its information as output. Ensure that your output matches the following output and test your code with at least two more test cases on your own.

Possible Output:

```
Run:
Instantiation of a square
Size: 5
Perimeter: 20
Area: 25

Instantiation of a cube:
Size: 5
Surface: 150
Volume: 125
```

4. [5pt] Design a class named ***Rectangle*** with two private data members: ***length*** and ***width***. Define constructors and a destructor for the class and write member functions to find the perimeter and area of a rectangle. Then define a class named ***Cuboid*** (representing a box) that inherits from the class ***Rectangle*** with an extra data member: ***height***. Then write constructors and a destructor for the ***Cuboid*** class, and write member functions to find the surface and volume of the ***Cuboid*** objects. Provide ***print*** member function for both classes to show its information as output. Make sure that your output matches the following output and test your code with at least two more test cases on your own.

Possible Output:

```
Run:
Instantiation of a Rectangle object:
Length: 5 Width: 4
Perimeter: 18
Area : 20

Instantiation of a Cuboid object:
Length : 5 Width: 4
Perimeter: 18
Area: 20
Height: 8
Surface: 184
Volume: 160
```

End of Assignment.