

CS250 Homework 3

Date Assigned: Monday, April 20, 2009

Name: _____

Date Due: Friday, April 24, 2009 @ 5pm

Points: 20pts

Please give this solution to Professor Williams by 5pm on Friday. If he is not in his office, then place it under his office door.

I've placed an electronic copy of this document in the CS250 public folder. You can use this document to type up the solutions and print it out for Professor Williams.

Copy the project `circleInheritance` from the CS250 public folder on Turing to the desktop. Using this project, you are to do the following. For the questions that require code, write the code down below the question.

1. Create an object of type **Circle** that has an **x** value of 3, a **y** value of 8, and a **radius** of 2. List the constructors that are called in the order in which they are called.

2. What is the meaning of:

```
Circle::Circle (int x, int y, double radius) : Point(x, y)
{
    setRadius (radius);
}
```

3. How else can we write the above constructor with using **:Point(x, y)**?

4. Circle does not have direct access to Point's member called x. Give 2 different ways that Circle could have direct access to x.

5. Uncomment the function `operator>>` in the Point.h file. Is `operator>>` a member of Point? Why or why not?

6. Implement the `operator>>` function in class `Point` so that it displays the message: **Enter the point in the form (x , y)** and sets the x and y value to the numbers entered by the user. Test the function in main. Write the code here.

7. Is there a way to call Point's **print** function from Circle's **print** function? If so, implement it and write the code here.
8. Create a new class called **Cylinder** that is a subclass of **Circle**. Add any necessary variables and functions to this class. Implement all of the functions that you create and place all the code here.

9. Create an object of type **Cylinder** and read in the values from the user. After that, print out the object. Place your code here.

10. Create a pointer to a Point called **pPoint** and assign to it the object **cPoint**. What is the output after calling **pPoint->print();**?

11. Now, assign **pPoint** to the object **cCircle**. What is the output after calling **pPoint->print();**?

12. How can we change the output of the previous question so that it outputs the radius?

Good luck and have fun ☺