COSC 565/ COSC 475: Software Engineering I
Assignment 1
Due: 10th October, 2017

1) Can a software engineer become a certified Professional Engineering (PE)? Explain. Give advantages and disadvantages of certification in Software Engineering. [10]

2) Define the depth versus the breadth issue in software complexity. [10]

3) What are the eight principles for software engineering code of ethics recommended by the IEEE-CS/ACM Version 5.2 joint task force report? [10]

4) What are the four quadrants in a spiral model? Trace the requirements set of activities through each quadrant. [10]

5) Compare and contrast Agile and traditional methods [10]

	Agile	Traditional / Heavy
Requirements		
Design		
User involvement		
Documentation		
Communication		
Process complexity		
Overhead		

6) Using your knowledge of how an ATM is used, develop a set of use cases that could serve as a basis for understanding the requirements for an ATM system. [Write atleast 5 **Textual Use Cases]** [20]

7) Explain why change is inevitable in complex systems and give examples (apart from prototyping and incremental delivery) of software process activities that help predict changes and make the software being developed more resilient to change. [10]

8) Write a class BullsEye with methods [20]

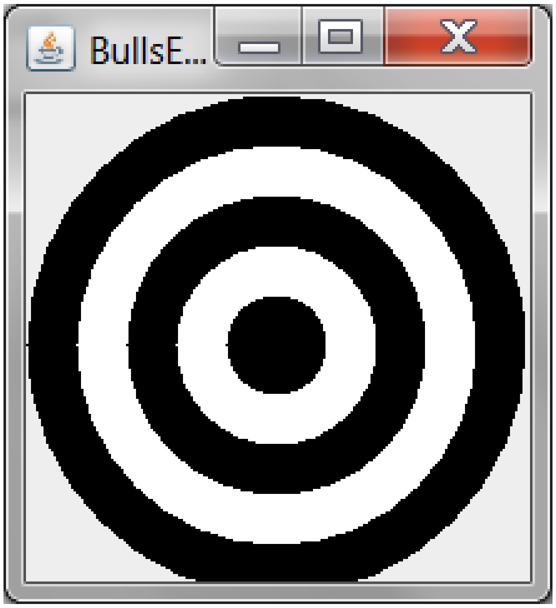
Use the following class as your main class:

```
import javax.swing.JFrame;

//Displays a "bull's eye".

public class BullsEyeViewer
{
   public static void main(String[] args)
   {
      JFrame frame = new JFrame();
      frame.setSize(220, 240);
      frame.setTitle("BullsEye");
      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      BullsEyeComponent component = new BullsEyeComponent();
      frame.add(component);
      frame.setVisible(true);
   }
}
```

Complete the following class in your solution:



```
import java.awt.Graphics2D;
import java.awt.geom.Ellipse2D;
import java.awt.Color;

/**
   Draws a bull's eye.
 */
public class BullsEye
{
   /**
      Creates a new instance of BullsEye.
      @param r the radius
      @param x the center x-coordinate
      @param y the center y-coordinate
    */
   public BullsEye(double r, double x,
double y)
   {
      . . .
   }

   /**
      Draws the bull's eye.
   */
   public void draw(Graphics2D g2)
   {
      . . .
   }

   . . .
}
```

Use the following class in your solution:

```
import javax.swing.JComponent;
import java.awt.Graphics;
import java.awt.Graphics2D;

/**
   Displays a bull's eye.
 */
public class BullsEyeComponent extends JComponent
{
   public void paintComponent(Graphics g)
   {
      Graphics2D g2 = (Graphics2D) g;

      int radius = 100;
      int xCenter = 100;
      int yCenter = 100;
      BullsEye be = new BullsEye(radius, xCenter, yCenter);
      be.draw(g2);
   }
}
```