public interface Measurable{
    double getMeasure();
}

public class SHAPE {
    public SHAPE(int w, int h)
    {
        this.boxWidth=w;
        this.boxHeight=h;
    }

    public double getArea(){
        return this.boxWidth * this.boxHeight;
    }
    private int boxWidth;
    private int boxHeight;
}

class LINE extends SHAPE implements Measurable{
    public LINE(int x0, int y0, int x1, int y1){
        super(Math.Abs(x1-x0), Math.Abs(y1-y0));
        this.x0=x0; this.y0=y0;
        this.x1=x1; this.y1=y1;
    }
    public double getMeasure () {
        this.Length=
        Math.Sqrt(Math.Sqr(x1-x0)+Math.Sqr(y1-y0));
        return this.Length;
    }
    public double getArea (){ return 0.0; }
    protected double Length;
    private int x0, y0, x1, y1;
}

final class VERTICAL extends LINE {
    static Main(string [] args){
        LINE edge = new LINE(5,10,5,100);
        SHAPE shape = new LINE(5,10,5,100);
        S.o.p(edge.getMeasure());
        S.o.p(shape.getArea());
    }
    final string label="LEFT";
}
Name ____________________________

Answer the following questions in plain English sentences.

1. Explain what the code in Line 1 above indicates:

2. Explain what the code in Line 2 above indicates:

3. Explain what the code in Line 3 above indicates:

4. Explain what the code in Line 4 above indicates:

5. Explain what the code in Line 5 above indicates:

6. Explain what the code in Line 6 above indicates:

7. Explain what the code in Line 7 above indicates:

8. Explain what the code in Line 8 above indicates:

9. Explain what the code in Line 9 above indicates:

10. Explain what the code in Line 10 above indicates:
11. Explain what the code in Line 11 above indicates:

12. Explain what the code in Line 12 above indicates:

13. Explain what the code in Line 13 above indicates:

14. Explain what the code in Line 14 above indicates:

15. Explain what the code in Line 15 above indicates:

16. Explain what the code in Line 16 above indicates:

17. Explain what the code in Line 17 above indicates:

18. Explain what the code in Line 18 above indicates:

19. Explain what the code in Line 19 above indicates:

20. Explain what the code in Line 20 above indicates:
21. What is the purpose of an interface?

22. What is the purpose of a JButton? How is it different from the buttons in 19 and 20?

23. What is the purpose of a JRadioButton? How is it different from the buttons in 18 and 20?

24. What is the purpose of a JCheckBox? How is it different from the buttons in 18 and 19?

25. What is the purpose of a try statement?

26. What is the purpose of a type cast statement?

27. What is the purpose of an Integer.parseInt() statement?

28. Why do we typically override the toString() function?

29. What is the purpose of a catch statement?

30. What is Object?
31. What is inheritance?

32. What is polymorphism?

33. What is data protection?

34. What is encapsulation?

35. What is the purpose of a `throw` statement?

36. Explain the Model part of the MVC architecture?

37. Explain the View part of the MVC architecture?

38. Explain the Controller part of the MVC architecture?

39. Why is the quote “Measure Twice, Cut Once.” Relevant to Computer Science?

40. What is an Abstract Class?
41. Write the code to create an array of 10 integers called Frog?

42. Write a loop to set the i\text{th} value in Frog to \text{i*\text{i}}?

43. Write the code to sum the values in Frog?

44. Write the code to create an ArrayList of Integer called Bird?

45. Write a loop to add ten even Integer(s) to Bird?

46. Write the code to sum the values in Bird?

47. Why would a class implement the Comparable interface??

48. What is big O notation used to describe??

49. (4pts) Suppose you wanted to override the toString() function in class LINE? (the class at the beginning of this test)
Give the code for the method: