

8.6 What is the printout of the following code?

```
public class Foo {
    private boolean x;
    public static void main(String[] args){
        Foo foo = new Foo();
        System.out.println(foo.x);
    }
}
```

8.6 Answer

false

8.10 Suppose that the class Foo is defined in (a). Let f be an instance of Foo. Which of the statements in (b) are correct?

(a)

```
public class Foo {
    int i;
    static String s;
    void imethod(){
    }
    static void smethod(){
    }
}
```

(b)

```
System.out.println(f.i);
```

```
System.out.println(f.s);
```

```
f.imethod();
```

```
f.smethod();
```

```
System.out.println(Foo.i);
```

```
System.out.println(Foo.s);
```

```
Foo.imethod();
```

```
Foo.smethod();
```

8.10 Answer

```
f.smethod();
```

8.18 Show the printout of the following code:

(a)

```
public class Test{
    public static void main(String[] args){
        int[] a = {1,2};
        swap(a[0], a[1]);
        System.out.println("a[0] = "+a[0]+" a[1] = "+a[1]);
    }
    public static void swap(int n1, int n2){
        int temp = n1;
        n1 = n2;
        n2 = temp;
    }
}
```

```
}
```

```
(b) public class Test{  
    public static void main(String[] args){  
        int[] a = {1,2};  
        swap(a);  
        System.out.println("a[0] = "+a[0]+" a[1] = "+a[1]);  
    }  
    public static void swap(int[] a){  
        int temp = a[0];  
        a[0]=a[1];  
        a[1] = temp;  
    }  
}
```

```
(c) public class Test{  
    public static void main(String[] args){  
        T t = new T();  
        swap(t);  
        System.out.println("e1 = "+t.e1+" e2 = "+t.e2);  
    }  
    public static void swap(T t){  
        int temp = t.e1;  
        t.e1 = t.e2;  
        t.e2 = temp;  
    }  
}
```

```
class T {  
    int e1 = 1;  
    int e2 = 2;  
}
```

```
(d) public class Test{  
    public static void main(String[] args){  
        T t1 = new T();  
        T t2 = new T();  
        System.out.println("t1's i = "+t1.i+" and j = "+t1.j);  
        System.out.println("t2's i = "+t2.i+" and j = "+t2.j);  
    }  
}  
class T {  
  
    static int i = 0;  
    int j = 0;  
  
    T(){  
        i++;  
        j = 1;  
    }  
}
```

8.18 Answer

(a)

```
a[0] = 1 a[1] = 2
```

(b)

a[0] = 2 a[1] = 1

(c)

e1 = 2 e2 = 1

(d)

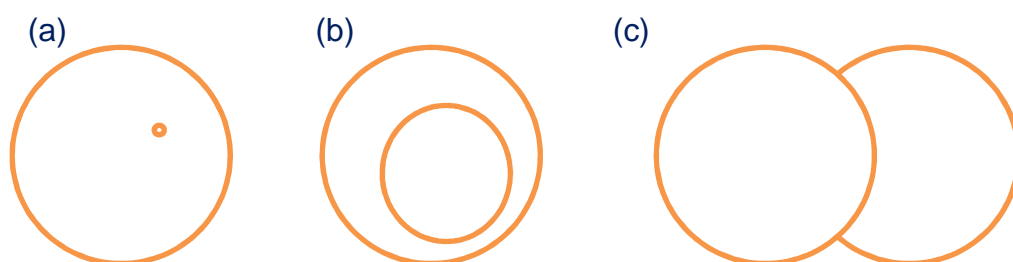
t1's i = 2 and j = 1

t2's i = 2 and j = 1

10.11 Define the Circle2D class that contains:

- Two double data fields named x and y that specify the center of the circle with get methods
- A data field radius with a get method
- A no-arg constructor that creates a default circle with (0,0) for (x,y) and 1 for radius
- A constructor that creates a circle with the specified x,y and radius
- A method getArea() that returns the area of the circle
- A method contains(double x, double y) that returns true if the specified point(x,y) is inside this circle (see fig (a))
- A method contains(Circle2D circle) that returns true if the specified circle is inside this circle (see fig (b))
- A method overlaps(Circle2D circle) that returns true if the specified circle overlaps with this circle. (see fig (c))

Draw the UML diagram for the class. Implement the class. Write a test program that creates a Circle2D object c1(new Circle2D(2,2,5.5)), displays its area and perimeter, and displays the result of c1.contains(3,3), c1.contains(New Circle2D(4,5,10.5)), and c1.overlaps(new Circle2D(3,5,2.3)).



10.11 Answer

Program

```
public class Circle2D {  
    public static void main(String[] args) {  
        Circle2D c1 = new Circle2D(2, 2, 5.5);  
        System.out.println("The area of c1 is " + c1.getArea());  
        System.out.println("The perimeter of c1 is " +  
c1.getPerimeter());  
    }  
}
```

```

        System.out.println("c1 contains (3, 3): " +
c1.coordContains(3.0, 3.0));
        System.out.println("c1 is inside Circle(4, 5, 10.5): " +
c1.circleContains(new Circle2D()));
        System.out.println("c1 overlaps Circle(3, 5, 2.3): " +
c1.overlaps(new Circle2D()));
    }

    private int x, y;
    private double x2, y2, radius, radius2;

    public Circle2D() {
        x = 0;
        y = 0;
        radius = 1;
    }

    public Circle2D(int x, int y, double radius) {
        this.x = x;
        this.y = y;
        this.radius = radius;
    }

    public double getArea() {
        return radius * radius * Math.PI;
    }

    public double getPerimeter() {
        return 2 * radius * Math.PI;
    }

    public boolean coordContains(double x2, double y2) {
        if ((x-x2)*(x-x2)+(y-y2)*(y-y2) < radius * radius)
            return true;
        else
            return false;
    }

    public boolean circleContains(Circle2D circle) {
        x2 = 4.0;
        y2 = 5.0;
        radius2 = 10.5;
        if ((x-x2)*(x-x2)+(y-y2)*(y-y2) < (radius-radius2)*(radius-
radius2))
            return true;
        else
            return false;
    }

    public boolean overlaps(Circle2D circle) {
        x2 = 3.0;
        y2 = 5.0;
        radius2 = 2.3;
        if ((radius-radius2)*(radius-radius2) < (x-x2)*(x-x2)+(y-
y2)*(y-y2) &&(x-x2)*(x-x2)+(y-y2)*(y-y2) <
(radius+radius2)*(radius+radius2))
            return true;
        else
            return false;
    }
}

```

Execution Result

```
The area of c1 is 95.03317777109125
The perimeter of c1 is 34.55751918948772
c1 contains (3, 3): true
c1 is inside Circle(4, 5, 10.5): true
c1 overlaps Circle(3, 5, 2.3): false
```

Class Diagram

