Question # 1

What is the output of the following programs?

Programs	Output

`public class FinalQ1A {`
`  public static void main(String[] args) {`
`  String a = "Final";`
`  change(a);`
`  System.out.println(a);`
`  }`
`  public static void change(String a) {`
`  a = "Midterm";`
`  }`
`}`
	`Final`

`public class FinalQ1B {`
`  public static void main(String[] args) {`
`  String[] a = {"Final"};`
`  change(a[0]);`
`  System.out.println(a[0]);`
`  }`
`  public static void change(String a) {`
`  a = "Midterm";`
`  }`
`}`
	`Final`

`public class FinalQ1C {`
`  public static void main(String[] args) {`
`  String[] a = {"Final"};`
`  change(a);`
`  System.out.println(a[0]);`
`  }`
`  public static void change(String[] a) {`
`  a[0] = "Midterm";`
`  }`
`}`
	`Midterm`

`public class FinalQ1D {`
`  public static void main(String[] args) {`
`  String[] a = {"Final"};`
`  change(a);`
`  System.out.println(a[0]);`
`  }`
`  public static void change(String[] a) {`
`  a = new String[]{"Midterm"};`
`  }`
`}`
	`Final`

`public class FinalQ1E {`
`  public static void main(String[] args) {`
`  String[] a = {"Final"};`
`  String[] b = a;`
`  b[0] = "Midterm";`
`  System.out.println(a[0]);`
`  }`
`}`
	`Midterm`

Question # 2

Define the static method **generateTest** that takes the array of **Question** objects as its argument. It selects 10 random questions, stores their indexes in an integer array of size 10, and returns the array.

```
public static int[] generateTest(Question[] q) {
      Random r = new Random();
      int[] tq = new int[10];
      for(int i = 0; i < 10; i++)
            tq[i] = r.nextInt(q.length);
      return tq;
}
```

Question # 3

Define a static method **copyArray** that takes a 2D integer array and returns a copy of the array. Do not use the clone method.

```
public static int[][] copyArray(int[][] x) {
      int[][] y = new int[x.length][];
      for(int i = 0; i < x.length; i++) {
            y[i] = new int[x[i].length];
            for(int j = 0; j < y[i].length; j++)
                  y[i][j] = x[i][j];
      }
      return y;
}
```

Question # 4

Consider the class Point:

```
public class Point {
	.
	public Point(double x, double y){}
	public double distanceFrom(Point other){ }
	.
}
```

Define a class Triangle as follows:

```
public class Triangle {
	// one instance variable; an array of 3 Point objects

		private Point[] p;

	// a default constructor

	public Triangle() {
		this.p = new Point[3];
	}

	// setPoint method that takes a Point and integer n to store it in the array at
	// the index n.

	public void setPoint(Point p, int n){
		this.p[n] = p;
	}

	// an instance method that returns the perimeter of the Triangle

	public double calculatePerimeter() {
		double sum = 0.0;
		for(int i = 0; i < p.length; i++)
			sum += p[i].distanceFrom(p[(i+1) % p.length]);
		return sum;
	}

}
```

Define a test class as follows

```
public class TriangleTest {
	public static void main(String[] args) {
		// create a default Triangle object

		Triangle t = new Triangle();

		// set the points (0, 0), (3, 4), (-3, 4)

		t.setPoint(new Point(0,0), 0);
		t.setPoint(new Point(3,4), 1);
		t.setPoint(new Point(-3,4), 2);

		// call the method to get the perimeter of the Triangle and print it

		System.out.println("The perimeter = " + t.calculatePerimeter());

	}
}
```