Information and Computer Science Department
Summer Semester 153
ICS 102 – Introduction to Computing I
Final Exam Key
Thursday, September 1, 2016
Duration: 120 minutes

Name:

ID#:

Question #	Max Score	Score
1	12	
2	20	
3	20	
4	20	
5	28	
Total	100	

Question # 1

What is the output of the following programs?

Programs	Output

`public class FinalQ1A`
`{`
`  public static void main(String[] args)`
`  {`
`  int[] a = {10};`
`  change(a);`
`  System.out.println(a[0]);`
`  }`
`  public static void change(int[] b)`
`  {`
`  b[0] *= 2;`
`  }`
`}`
	20

`public class FinalQ1B`
`{`
`  public static void main(String[] args)`
`  {`
`  int[] a = {10};`
`  change(a[0]);`
`  System.out.println(a[0]);`
`  }`
`  public static void change(int b)`
`  {`
`  b *= 2;`
`  }`
`}`
	10

`public class FinalQ1C`
`{`
`  public static void main(String[] args)`
`  {`
`  int[] a = {10};`
`  change(a);`
`  System.out.println(a[0]);`
`  }`
`  public static void change(int b[])`
`  {`
`  b = new int[]{20};`
`  }`
`}`
	10

`public class FinalQ1D`
`{`
`  public static void main(String[] args)`
`  {`
`  int[] a = {10};`
`  change(a);`
`  System.out.println(a[0]);`
`  }`
`  public static void change(int a[])`
`  {`
`  a = new int[]{20};`
`  }`
`}`
	10

Question # 2

Write a program that reads from the file "absences.txt" information about students. Each line of the file has id followed by number of absences. The program prints the students who have more than 9 absences.

```
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;
public class FinalQ2
{
	public static void main(String[] args)throws FileNotFoundException
	{
		Scanner file = new Scanner(new File("absences.txt"));
		int id;
		int absences;
		while(file.hasNextInt())
		{
			id = file.nextInt();
			absences = file.nextInt();
			if(absences > 9)
				System.out.println(id + " " + absences);
		}
	}
}
```

Question # 3

The standard deviation of a list of numbers n_1, n_2, n_3, and so forth is defined as the square root of the average of the following numbers:

$(n_1 - a)^2$, $(n_2 - a)^2$, $(n_3 - a)^2$, and so forth.

The number a is the average of the numbers n_1, n_2, n_3, and so forth.

Define a static method that takes the array of numbers as its argument and returns the standard deviation of the numbers in the array. The numbers in the array should be of type double.

```
public static double StDev(double[] n)
{
      double sum = 0;
      for(int i = 0; i < n.length; i++)
            sum += n[i];
      double a = sum / n.length;
      double sums = 0;
      for(int i = 0; i < n.length; i++)
            sums += (n[i] - a) * (n[i] - a);
      double sd = Math.sqrt(sums / n.length);
      return sd;
}
```

Question # 4

Write a static method **reverseRowsOrder** that takes a 2D integer array **x** and reverses the order of its rows. The first row becomes the last row and last row becomes the first row and so forth.

For example:

x ➔

2	3	4	5		
10	20	30			
12	13	14	12	13	14

x ➔

12	13	14	12	13	14
10	20	30			
2	3	4	5		

```
public static void reverseRowsOrder(int[][] x)
{
      int[] temp;
      for(int i = 0; i < x.length / 2; i++) {
            temp = x[i];
            x[i] = x[x.length - 1 - i];
            x[x.length - 1 - i] = temp;
      }
}
```

Question # 5

Consider the class Point:

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

Write a class Shape as follows:

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

      private Point[] p;


      // a constructor that has one parameter an array of points
      public Shape(Point[] p)

      {
            this.p = p;
      }

      // a method that returns the circumference of the shape

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

}
```

Write a test class as follows

```
public class ShapeTest
{
      public static void main(String[] args)
      {
            // create one shape object with the three points (0, 0), (3, 4), (-3, 4)

            Point[] p = {new Point(0,0), new Point(3,4), new Point(-3, 4)};

            Shape sh = new Shape(p);
            // call the method to get the circumference of the shape

            System.out.println(sh.getCircumference());
      }
}
```