

Programming I

Lab 8

Exercise 1 (2 points)

For each of these code segments, show what the output is and briefly explain it especially with respect to the parameters. Do these without the help of DrJava. Submit your answers in a text file.

Segment 1

```
public class XYZ
{
    public static void main(String[] args)
    {
        int num1 = 5, num2 = 10;
        int result = XYZ.doSomething(num1, num2);
        System.out.println("The result is: " + result);
    }

    public static int doSomething(int a, int b)
    {
        System.out.println("a is " + a + " and b is " + b);
        if (a > b)
            return a;
        else
            return b;
    }
}
```

Segment 2

```
public class WXZ
{
    public static void main(String[] args)
    {
        for (int i = 5; i > 0; i--)
            WXZ.mystery(i);
    }

    public static void mystery(int num)
    {
        for (int i = 0; i < num; i++)
            System.out.print("* ");
        System.out.println(); // go to next line
    }
}
```

Submission Instructions: Programming exercises 1 and 4 require modifications to the BaseBoard.java file. Submit the BaseBoard.java file for each of these exercises in the corresponding **subfolder** on Labdata.

Programming Exercise 1 (3 points)

Using the methods to fill a circle developed in class and in Section 8.4 as a starting point, implement a method that fills a ring of a given thickness, color, location and radius (of the outer edge). You can think of drawing a ring as filling the area between the outer and the inner circles that define the ring. For instance to draw a ring of thickness 3 and an outer radius of 10 (inner radius of 7), the method would fill the cells that satisfies the compound condition:

$$(x-xshift)^2 + (y-yshift)^2 \leq 10^2 \quad \&\& \quad (x-xshift)^2 + (y-yshift)^2 \geq 7^2$$

inside the outer circle outside the inner circle

The method should be implemented in the Board class (in BaseBoard.java file) and should have the following header:

```
public void drawRing(int radius, int thickness, int xshift, int yshift, Color c)
```

To test your method, write a separate program with a main that uses drawRing to draw the Olympic logo. You are to turn in the modified BaseBoard.java file and the file containing your main method.

Programming Exercise 2 (2 points)

Write a program that finds the maximum of 3 integer values entered by the user. The program should prompt the user to enter the values in main, and then invoke a method `maxOfThree` with the three values passed as arguments to the method. The method will return the maximum value to main, which then displays it. The method header should be:

```
public static int maxOfThree(int a, int b, int c)
```

You may not use the method `Math.max` which returns the maximum of two values.

Programming Exercise 3 (3 points)

For this exercise, you are to write a program that computes income tax based on a graduated tax rate scheme:

if income is	tax rate is
$\leq 15,000$	0%
15,001-80,000	15% on earnings over \$15,000
80,001-200,000	25% on earnings over \$80,000
over 200,000	75% on earnings over \$200,000 (a bargain compared to historical norms)

Your program will prompt the user to enter his or her income in main. `main` will then invoke a static method called `computeTax`, which takes `income` as parameter and returns the tax owed. The main method finally prints out this value. As an example, a person earning \$16,000 will only pay taxes on the amount earned over \$15,000 (in this case \$1,000), so the tax owed is \$150.

Programming Exercise 4 (5 points)

You are to create the game *arrange*, which uses a 3x3 board with 8 pieces labeled 1 to 8, and an empty slot. The game starts with the pieces being placed randomly on the board. The player's objective is to arrange the pieces in order by repeatedly swapping a labeled piece with the empty slot. When the player wins, your program should notify the player (perhaps by changing the colors of the cells) and disable the processing of mouse events. See the following link for an example:

<http://javascript.internet.com/games/arrange.html>

You are to turn in the modified BaseBoard.java file and the file containing your main method.