Variables, Branching Statements, and Loops

1. Write a program where the user enters a number, and the program prints out a right triangle of asterisks (*) where the number is the width of the base and the height. The triangle’s base must be at the top and the point must be at the bottom.

Example: If the user enters 5 the program will print out

```
*****
****
***
**
*
```

```java
import java.util.Scanner;
public static void main(String[] args)
{
    Scanner keyboard = new Scanner(System.in);
    System.out.println("Enter the size of the triangle");
    int number = keyboard.nextInt(); //get the base size
    //Put your code here

    for(int i=number;i>0;i--) //this one counts backwards
    {
        //each i corresponds to the number of * to be printed
        for(int j=i;j>0;j--)
        {
            System.out.print("*");
        }
        System.out.println(); //next line
    }
}
```
2. Write a program where the user enters a number, and then the program adds 10 from that number until it is greater than 50. It must print out the new value at each step.
Example if the user enters 14 the program will print out:
14
24
34
44

```java
import java.util.Scanner;
public static void main(String[] args) {
    Scanner keyboard = new Scanner(System.in);
    System.out.println("Enter a number");
    int number = keyboard.nextInt(); //get the number
    //Put your code here
    while (number <= 50) //until that number is greater than 50
    {
        System.out.println(number);
        number += 10; //keep adding 10
    }
}
```
3. Write a program that prints out each position per 1 second of a particle given an initial starting x and y position, velocities in the x and y direction, and a simulated number of positive, non-zero seconds. The equation for calculating positions are:

\[
\begin{align*}
    x_0 &= \text{given starting x position} \\
    y_0 &= \text{given starting y position} \\
    s &= \text{given number of simulate seconds} \\
    s_i &= \text{the current second} \\
    x &= x_0 + v_x s_i \\
    y &= y_0 + v_y s_i
\end{align*}
\]

where \( x \) is the position, \( v_x \) is the given velocity in the x direction, \( x_0 \) is the initial position, and \( s_i \) is the current number of simulated seconds. The previous applies the same for the y position. Make sure that for each simulated second it prints out on a separate line the current second, the current x position, and the current y position in the following format:

"Second: "<<current simulated second>>"x: "<<current x>>"y: "<<current y>>

Values denoted in “<< >>” represent variable values, and strings in quotations denote literal values (make sure to follow capitalization, punctuation, and spacing exactly).

For example, if the user entered \( x_0 = 5, y_0 = 2, v_x = 3, v_y = 2, s = 10 \), then the output would be:

```
Enter initial x and y positions, then x and y velocities, and finally a simulated number of seconds (strictly greater than 0)
5.0
2.0
3.0
2.0
10.0
Second: 0.0 x: 5.0 y: 2.0
Second: 1.0 x: 8.0 y: 4.0
Second: 2.0 x: 11.0 y: 6.0
Second: 3.0 x: 14.0 y: 8.0
Second: 4.0 x: 17.0 y: 10.0
Second: 5.0 x: 20.0 y: 12.0
Second: 6.0 x: 23.0 y: 14.0
Second: 7.0 x: 26.0 y: 16.0
Second: 8.0 x: 29.0 y: 18.0
Second: 9.0 x: 32.0 y: 20.0
Second: 10.0 x: 35.0 y: 22.0
```
public static void main(String[] args) {
    Scanner keyboard = new Scanner(System.in);
    System.out.println("Enter initial x and y positions, then x and y velocities, and finally a simulated number of seconds (strictly greater than 0)");
    //Write your solution here
    double x = 0.0;
    double y = 0.0;
    double vx = 0.0;
    double vy = 0.0;
    double s = 0.0;
    x = keyboard.nextDouble();
    y = keyboard.nextDouble();
    vx = keyboard.nextDouble();
    vy = keyboard.nextDouble();
    s = keyboard.nextDouble();
    double x0 = x;
    double y0 = y;
    for(double i = 0.0;i<s;i+=1.0) {
        x = x0 + vx*i;
        y = y0 + vy*i;
        System.out.println("Second: "+i+" x: " + x+" y: "+y);
    }
}
Arrays
No sorting algorithms will be on this exam.

1. Write a program that finds the minimum and maximum number in an array and then subtracts the maximum from the minimum and prints the result. The first line printed must be in the format:

```
<<maximum>>“-”<<minimum>>“=“<<result>>
```

Example if the array given was \{10,4,6,8,2\} the program would print out 10-2=8

```java
public static void main(String[] args)
{
    Scanner keyboard = new Scanner(System.in);
    int[] a = {10, 4, 6, 8, 2};
    //Put your code here
    //Initially sets the min and max to the first value
    int min = a[0];
    int max = a[0];
    for(int i=1;i<a.length;i++)
    {
        if(a[i]<min) //if the current index is less than the min
        {
            min = a[i]; //reset the min to the current index
        }
        if(a[i]>max) //if the current index is greater than the max
        {
            max = a[i]; //reset the max to the current index
        }
    }
    int result = max-min; //find the result
    System.out.println(max+"-"+min+"=“+result); //print it out
}
```
2. Write a program that multiplies all the number in an array and then prints out the result.

Example if the array given is \{2,4,6,8\} the program would print 384

```java
public static void main(String[] args)
{
    Scanner keyboard = new Scanner(System.in);
    int[] a = \{2,4,6,8\};
    //Put your code here
    int result = a[0]; //keeps track of the results
    for(int i=1;i<a.length;i++)
    {
        result *= a[i]; //adds the current index to the result
    }
    System.out.println(result); //prints the result
}
```
3. Write a program that goes through an array and then changes every instance of an even number into a 0, and then prints out the resulting array. Each value should be separated by a space (" ") in the printout.
Example if the array given is {1, 2, 3, 4, 5, 6, 7, 8} the program will print out 1 0 3 0 5 0 7 0

```java
public static void main(String[] args)
{
    Scanner keyboard = new Scanner(System.in);
    int[] a = {1, 2, 3, 4, 5, 6, 7, 8};
    //Put your code here
    //Looks for the even numbers in a loop
    for(int i=0; i<a.length; i++)
    {
        //it is even if it is divisible by 2 and the remainder is 0
        if(a[i]%2 == 0)
        {
            a[i] = 0;
        }
    }
    //Prints the results
    for(int i=0; i<a.length; i++)
    {
        System.out.print(a[i]+" ");
    }
}
```