Demo 5 / 13.10

Tasks

1. **Loops:** Download the example answer NumbersGraph.java of the previous Guru-task in demo 4 and modify it so that the subprogram called `drawPicture` will draw the highest value(s) of an array as a red circle, lowest as a green circle and the rest as black circles.

2. **Loops:** Create another (overloaded) subprogram `drawPicture` (same name as in the previous task) and do it either by starting from scratch or by adding to the previous one. To this subprogram you pass two parameters in addition to the array: a number for drawing all the values below it as green and an another number for drawing all the values higher or equal to it as red. Is it possible to make the task 1 with this subprogram? How about drawing a similar picture as in the Guru-task in the demo 4?

3. **Arrays**: Create a program which asks the user the average temperatures of six months (1-6) and draws a picture so that all the months below zero are drawn with a green circle and all months over 10 degrees with a red circle. Use the subprogram made in the previous task. We draw only six months because typing in all of them takes time and makes us bored. However, the program must be easily converted to ask and draw 12 months!

4. **Loops:** Create a subprogram `multiplication(n)` which prints the multiplication table of n with a for loop. For example `multiplication(3);` would print:

```
 1 * 3 =   3
 2 * 3 =   6
 3 * 3 =   9
 4 * 3 =  12
 5 * 3 =  15
 6 * 3 =  18
 7 * 3 =  21
 8 * 3 =  24
 9 * 3 =  27
10 * 3 =  30
```

   Create `multiplicationWhile(n,m)` which prints m amount of lines from the multiplication table of n. Implementation with `While` loop. Example for `multiplicationWhile(3,5):`

```
 1 * 3 =   3
 2 * 3 =   6
 3 * 3 =   9
 4 * 3 =  12
 5 * 3 =  15
```

   Do the same with Do while loop. What does `multiplicationDoWhile(6,0);` print if no special additions are made? What do you have to add?

5. **Tables:** Write a function subprogram `sum(numbers)` that sums the even numbers in a

table and substracts the odd ones from the sum. The function returns the final sum. Start off by writing the main program where you figure out how to call the subprogram.

6. **Tables:** Write a function subprogram `mode(numbers)` that looks for the most common number in a table. If two or more numbers seem to be most common the first encountered is counted as the most common. First think how you would solve this problem (think of an algorithm). You won't necessarily need an auxiliary table as you can solve this task with a few auxiliary variables. One of the middle numbers. Another is the average number. The third is the median, the middle cell of the table. This isn't mandatory yet. How do you calculate the median? Check Wikipedia for example.

B1. Write a function subprogram `beginsWithCapital(word)` that returns `true` if the string parameter begins with a capital letter. Using this, write a program that uses the console in the following way (hint: `Character-class`, `args`-table):

```
java demo5.BeginsWithCapital cat Chicken Dog fox Worm 2Hens
The ones that start with a capital letter:
 1. argument = Chicken
 2. argument = Dog
 4. argument = Worm
```

In Eclipse, for the F11 -run arguments can be given as follows:

```
Presuming the program has already been run in Eclipse and is called
BeginsWithCapital
  Run/Debug Configurations/BeginsWithCapital/Arguments
and then write the chosen parameters in Program Arguments.
```

GURU-tasks

G1-2 In task 2 someone might want to change the colors. In such a case, the list of parameters will become too large. One good solution is to make the thing that draws the circles into an object. Write the appropriate class so the following main program works:

```
public static void main(String[] args) {
    double numbers2[] = {1.9,3,2,2.4,1.2,2.6,3.2,3.1};
    Window win = new Window(600,400);
    win.scale(0,-1,numbers2.length,4);
    win.add(new Axis(100,100,0));

    TableAsCircles t2 = new TableAsCircles(luvut2);
    win.add(t2);
    t2.setMinColor(Color.BLUE).setMaxColor(Color.ORANGE).
       setColor(Color.PINK).setR(0.4).setMin(2).setMax(3);

    win.showWindow();

}
```

You should use the newest `Graphics.jar`.