

March 14<sup>th</sup>, 2022

# **Sheet 01 - Introduction to Java**

### **Exercise 1 (Datatypes & Operators)**

[2 Points]

Take a look at the following expressions and operations. Compute the resulting value and data type. Additionally, explain how they came to be. You are advised to write a main method to test these expressions. Either use System.out.println() or the debugger to get the results. If an expression is invalid, explain why this is the case.

Submit a plain text file containing all your answers. A template (exercise1.txt) for this file can be found in OLAT.

#### Hint

A

Only submit **UTF-8** encoded text files. Use a modern text editor or an IDE (e.g., intellij, eclipse) to create that file. Some recommended editors are VIM, Emacs, Sublime, Atom, gedit or Notepad++.

- a) 5 \* 7 / 3
- b) 1 << 8 % 3
- c) (short) Integer.MAX\_VALUE
- d) 23 / (double) 11
- e) (double) (23/11)
- f) 42f

- g) 4e3D
- h) 0.1f == 0.1
- i) "Peter=Coffee+"+'chocolate'+2.0
- j) "Peter=Coffee+"+"chocolate"+2.0
- k) 1 == 24 % 3 && 4 > 7 || true
- I) 1 == 24 % 3 ? 4 : 7

#### **Submit**



at/ac/uibk/pm/gXX/zidUsername/s01/e01/exercise1.txt

# Exercise 2 (Loops)

[2 Points]

Add a static method named printArray in the provided file PrintArray. java. The function should print the provided two dimensional int-array. Each row is printed in a new line and each element is followed by the provided separator. The method should expect two parameters and returns nothing. The first parameter is a two-dimensional int-array of arbitrary size and the second is a String denoting the separator.

The main method

```
public static void main(String[] args) {
   int[][] array = {{1, 2, 3}, {4, 5}, {6, 7, 8, 9}};
   printArray(array, " | ");
}
```

should produce the following output:

```
1 | 2 | 3 |
4 | 5 |
6 | 7 | 8 | 9 |
```

Implement printArray accordingly! Do not use any loops except foreach-loops. Simply hard coding the expected output is not allowed!

#### Hint

A

You can print to stdout with System.out.println and System.out.print the first metod adds a '\n' to the end, the second does not.

### Submit

1

at/ac/uibk/pm/gXX/zidUsername/s01/e02/PrintArray.java

### **Exercise 3 (Overloading & Call by Value)**

[2 Points]

Create a class called TimesTwo and implement the methods requested by a) and b). Call both methods in the main method and answer all questions of c) in a text-file.

- a) Write a static method called timesTwo that multiplies the given int-value by two and prints it afterwards
- b) Write a static method called timesTwo that multiplies each int-value in the given int-array by two and then prints the whole array afterwards.

#### Hint

A

You can use the method Arrays.toString() to print the whole array.

- c) Investigate and explain:
  - (i) Why is it possible to declare two methods with the same name?
  - (ii) Is it possible to define two methods with the same name and arguments but different return values?
  - (iii) Remember: Java only supports call by value. Observe the variables passed to each of the timesTwo methods before and after the method call in the main-method. Did the values change? Explain why!

Submit

at/ac/uibk/pm/gXX/zidUsername/s01/e03/TimesTwo.java

at/ac/uibk/pm/gXX/zidUsername/s01/e03/exercise3.txt

# **Exercise 4 (Max Difference)**

#### [2 Points]

Write a method called maxDifference that finds and returns the largest difference between the values of a given int-array.

values	maxDifference(values)
{1}	0
{1, 1}	0
{1, 2}	1
{1, 10}	9
{10, 1}	9
{-1, 1}	2
{1, 3, 5}	4
{11, 3, 8}	8

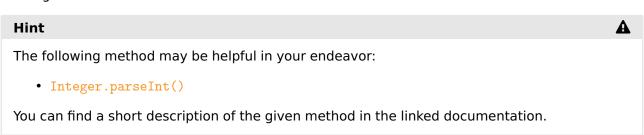


# **Exercise 5 (Pascal's triangle)**

### [2 Points]

Write a method called pascalsTriangle that computes and returns Pascal's triangle of height n. The two dimensional array returned by pascalsTriangle should contain rows 1 to n of Pascal's triangle. Each row should be a sub-array containing only the numbers of the corresponding level in Pascal's triangle.

The main-method prints Pascal's triangle of height n. The value for n is supplied as a command-line-argument.





**Important:** Submit your solution to OLAT and mark your solved exercises with the provided checkboxes. The deadline ends at 6:00 pm (18:00) on the day before the discussion.