Prof. Dr. Christoph Bockisch
MSc Steffen Dick
Fachbereich Mathematik und Informatik
AG Programmiersprachen und -werkzeuge

Philipps Universität Marburg

Tutorial exercises
Objektorientierte Programmierung: Wintersemester 2021/2022
Nr. 2

Task 2.1:

Look at the following method:

```
1  void smallestNumber(int a, int b, int c) {
2    if (a < b) {
3      if (a < c) {
4        System.out.println(a);
5      } else {
6        System.out.println(c);
7      }
8    } else if (b < c) {
9      System.out.println(b);
10   } else {
11     System.out.println(c);
12   }
13 }
```

The method *smallestNumber* takes 3 whole numbers and outputs the result to the console.

a) Think about the result that the method should output to the console when called like so (don't look at the code for this):
   i `smallestNumber(3,2,5);`
   ii `smallestNumber(3,3,5);`
   iii `smallestNumber(3,3,3);`

b) Calculate the result of the following calls to the method step by step:
   i `smallestNumber(3,2,5);`
   ii `smallestNumber(3,3,5);`
   iii `smallestNumber(3,3,3);`

c) Compare the expected results with the ones you calculated. Is this algorithm working correctly? If not, what could be improved upon?

Task 2.2: WHILE I pondered, weak and weary

Write a method that computes the length of a given number using a *while*-loop. The length of a number is the number of digits it uses (100 has a length of 3 and 0 of a length of 1).

Test your method using function calls (e.g. `countLength(100) == 3`).