

## Informatik 1, Lab 04

- **Submission deadline: 2023-06-05, 23:59**
- You will get a “pass” or “failed” after a discussion on 2023-06-08 or 2023-06-15, depending on which group you belong to

### Exercise 4-1

### Project Euler Problem 4

Solve problem 4 of Project Euler. <https://projecteuler.net/problem=4>, which reads

A palindromic number reads the same both ways. The largest palindrome made from the product of two 2-digit numbers is  $9009 = 91 \times 99$ .

Find the largest palindrome made from the product of two 3-digit numbers.

**Hint:** given an integer `i`, you can convert it to a `String` by either `i + ""` or `String.valueOf(i)`. For example:

```
jshell> int i = 42;  
i ==> 42  
  
jshell> String s = i + "";  
s ==> "42"  
  
jshell> String t = String.valueOf(i);  
t ==> "42"
```

### Exercise 4-2

### Collatz Sequence

According to Project Euler Problem 14, <https://projecteuler.net/problem=14>, the *Collatz Problem* is as follows:

The following iterative sequence is defined for the set of positive integers:

$$\begin{aligned} n &\mapsto n/2, (n \text{ is even}) \\ n &\mapsto 3n + 1, (n \text{ is odd}) \end{aligned}$$

Using the rule above and starting with 13, we generate the following sequence:

$$13 \mapsto 40 \mapsto 20 \mapsto 10 \mapsto 5 \mapsto 16 \mapsto 8 \mapsto 4 \mapsto 2 \mapsto 1$$

It can be seen that this sequence (starting at 13 and finishing at 1) contains 10 terms. Although it has not been proved yet (Collatz Problem), it is thought that all starting numbers finish at 1.

- a) Write a method `lengthOfCollatzSequence(int n)`, which calculates the length of the Collatz sequence of positive integer `n`.

b) Which starting number, under one **hundred**, produces the longest sequence?

#### Exercise 4-3

#### Palindromes

Write a method `boolean isPalindrome2(String s)` which checks if parameter `s` is a palindrome. The method should be case insensitive, and ignore all non-alphabetic characters.

- `isPalindrome2("ab")` should return `false`
- `isPalindrome2("aa")` should return `true`
- `isPalindrome2("aA")` should return `true`
- `isPalindrome2("a bA")` should return `true`
- `isPalindrome2("a,b.A")` should return `true`
- `isPalindrome2("Eine Note betone nie.")` should return `true`
- `isPalindrome2("Murder for a jar of red rum")` should return `true`

**Hint:** You cannot use any method defined in the Java class `String` besides `length` and `charAt`.

#### Exercise 4-4

#### 2-Digits

If we consider two adjacent digits as a 2-digit number, then the string "166837" contains five 2-digit numbers: 16, 66, 68, 83, 37. The largest two-digit number contained in the string is therefore 83.

Write a method `int largest2Digit(String s)` to return the largest 2-digit number contained in parameter `s`. You can assume that `s` consists of only characters between '0' and '9'.

#### Exercise 4-5

#### Statistics

Given an  $n$ -element array  $a = \{a_1, a_2, \dots, a_n\}$

- Its *mean*  $\bar{a}$  is defined as  $\bar{a} = \frac{1}{n} \sum_{i=1}^n a_i = \frac{a_1 + a_2 + \dots + a_n}{n}$ .
- Its *variance*  $\theta^2(a)$  is defined as  $\theta^2(a) = \frac{1}{n} \sum_{i=1}^n (a_i - \bar{a})^2 = \frac{(a_1 - \bar{a})^2 + (a_2 - \bar{a})^2 + \dots + (a_n - \bar{a})^2}{n}$ , where  $\bar{a}$  is  $a$ 's mean.

- Write a method which takes a parameter of type `int[]` and returns its average
- Write a method which takes a parameter of type `int[]` and returns its variance.

#### Exercise 4-6

#### Int to Array

Write a method `int[] intToArray(int n)` which returns an array containing the digits that `n` consists of. For example

- `intToArray(5)` should return `{5}`
- `intToArray(15)` should return `{1, 5}`
- `intToArray(155)` should return `{1, 5, 5}`
- `intToArray(3155)` should return `{3, 1, 5, 5}`