TUM

Computational Foundations I (WiSe 2023/2024) Task Sheet 3

Tasks marked with a star like **Optional Task*** are optional. Tasks marked like **Hard Task**$^+$ are given, but it is not expected that you solve them now. It is great if you learn to solve them during the lecture. Go back to them after a few weeks and see your own progress.

Learning Outcome: We start our way to master C and C++ from this sheet. This task sheet is a gentle start helping you to be proficient in basic C/C++ operations and structures such as functions, conditions, loop and input/output.

Task 4: C/C++ Practice I

First, we present an example showing you the format of every task and expected code.

Task 4.0 Hello, world! I

Write a program that outputs "Hello, world!".

— Input: None

— Expected output: Hello, world!

Possible implementation 1 (C)

```c
#include <stdio.h>

int main() {
    printf("Hello World!\n");
    return 0;
}
```

Possible implementation 2 (C++)

```cpp
#include <iostream>

int main() {
    std::cout << "Hello World!\n";
    return 0;
}
```

Task 4.1 Array

Define an array with 6 integers: 16, 7, 15, 47, -3, 0. Print this array to check.

— Input: None

— Expected output: 16, 7, 15, 47, -3, 0

Task 4.2 The statistics of the array

Consider the array defined in Task 4.1. Print the maximum, minimum, sum, median and average of this array. Hint: We may need to change data types during calculation.

−   Input: None

−   Expected output: 47, -3, 82, 11, 13.666...

Task 4.3 Even number filter for the array

Consider the array defined in Task 4.1. Print all even numbers. Hint: remainder is 1 or 0

−   Input: None

−   Expected output: 16, 0

Task 4.4 Refactor the even number filter

We expect exactly the same input and output of Task 4.4. However, a function called `isEven()` should be implemented. Then use `isEven()` in the `main()` function to achieve the same results as in Task 4.4. Your code should be like:

```cpp
#include <iostream>

bool isEven(int a) {
    // your implementation
    // return boolean values (true / false)
}

int main() {
    // use isEven() somewhere in your code
    return 0;
}
```

Hint: Think about PROCEDURE in Niki. We basically do the same thing, but in C/C++ (same idea but different syntax).

Task 4.5* Sort the array I

Consider the array defined in Task 4.1. Sort it using insertion sort and print the result.

−   Input: None

−   Expected output: -3, 0, 7, 15, 16, 47

Task 4.6* Sort the array II

Consider the array defined in Task 4.1. Sort it using bubble sort and print the result.

−   Input: None

−   Expected output: -3, 0, 7, 15, 16, 47

Task 4.7$^+$ Prime number filter for the array

Consider the array defined in Task 4.1. Print all prime numbers.

−   Input: None

−   Expected output: 7, 47

Task 4.8 Create your own array

Input a series of integers with a space in between. Read these integers and use them to define an array. Print the array to check. Hint: `scanf()` or `cin()`

−   Input: 1 2 3 4 5

−   Expected output: 1 2 3 4 5 (as long as the output is the same as input, then it would be fine)

Task 4.9 Triangle or not

Input three values and judge whether it can form a triangle in a plane.

Example 1

−   Input: 1 2 3

−   Expected output: No

Example 2

−   Input: 3 2 3

−   Expected output: Yes

Task 4.10 Hello, world! II

Try to print a "Hello, world" with a frame as shown below. Please use a loop to implement instead of manully controlling the number of hyphen (-). Hint: `strlen()`

```
+---------------+
| Hello, world! |
+---------------+
```

Task 4.11 Hello, world! III

Try to print a "Hello, world" with a frame and your name as shown below. The name should be taken from user inputs.

−   Input: firstname lastname

−   Expected output:

```
+-----------------------------------+
| Hello, world and firstname lastname! |
+-----------------------------------+
```