Homework 12

- Submit one ZIP file per homework sheet which contains one PDF file (including pictures, computations, formulas, explanations, etc.) and your source code file(s) with one makefile and without adding executable, object or temporary files.

- The implementations of algorithms has to be done using C, C++, Python or Java.

- The TAs are grading solutions to the problems according to the following criteria: https://grader.eecs.jacobs-university.de/courses/ch_231_a/2021_1/Grading_Criteria_ADS.pdf

Problem 12.1 *Longest ordered subarray*   (8 points)

Consider the array $A = (a_1, a_2, ..., a_n)$. We call a subarray a succession of numbers in A where the position of any value in the subarray in the array is always bigger than the value of the previous one. Use dynamic programming to find a subarray of maximal ordered length of the array A. Your algorithm should find one optimal solution if multiple exist.

Example

For $A = (8, 3, 6, 50, 10, 8, 100, 30, 60, 40, 80)$ the solution is: $(3, 6, 10, 30, 60, 80)$

Your program should take an input and generate an output as follows:

Sample input

```
8  3  6  50  10  8  100  30  60  40  80
```

Sample output

```
3  6  10  30  60  80
```

You can assume that the type of the input will be valid.

Your solution will be graded using multiple testcases, therefore following the input and output format is mandatory.

Problem 12.2 *Sum in triangles*   (12 points)

Consider a triangle formed from n lines ($1 < n \leq 100$), each line containing natural numbers in the interval $[0, 10000]$.

(a) (7 points) Use dynamic programming to determine the biggest sum of numbers existent on the path from the number in the first line and a number from the last line and print the respective path to the output. Each number in this path is seated to the left or to the right of the other value above it.

(b) (3 point) Analyze the runtime of your solution and compare it to the brute force approach.

(c) (2 point) Explain why a greedy algorithm does not work for this problem.

Example

The values are displayed for example in this manner:

```
        7
      3   8
    8   1   0
  2   7   4   4
4   5   2   6   5
```

For this example the resulting sum is 30.

Your program should take an input and generate an output as follows:

Sample input
```
7
3 8
8 1 0
2 7 4 4
4 5 2 6 5
```

Sample output
```
30
7 3 8 7 5
```

You can assume that the input will be valid.
Your solution will be graded using multiple testcases, therefore following the input and output format is mandatory.

How to submit your solutions

You can submit your solutions via *Grader* at **https://grader.eecs.jacobs-university.de** as a generated PDF file and/or source code files.
If there are problems with *Grader* (but only then), you can submit the file by sending mail to
`k.lipskoch@jacobs-university.de` **with a subject line that starts with CH-231-A**.

Please note, that after the deadline it will not be possible to submit solutions. It is useless to send solutions by mail, because they will not be graded.

This homework is due by Monday, May 3rd, 23:00.