Homework 3

- 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/2020_1/Grading_Criteria_ADS.pdf

Problem 3.1 *Asymptotic Analysis*   (8 points)

Considering the following pairs of functions f and g, show for each pair whether or not it belongs to each of the relations $f \in \Theta(g)$, $f \in O(g)$, $f \in o(g)$, $f \in \Omega(g)$, $f \in \omega(g)$, $g \in \Theta(f)$, $g \in O(f)$, $g \in o(f)$, $g \in \Omega(f)$, or $g \in \omega(f)$.

(a) (2 points) $f(n) = 9n$ and $g(n) = 5n^3$,

(b) (2 points) $f(n) = 9n^{0.8} + 2n^{0.3} + 14 \log n$ and $g(n) = \sqrt{n}$,

(c) (2 points) $f(n) = n^2/\log n$ and $g(n) = n \log n$,

(d) (2 points) $f(n) = (\log(3n))^3$ and $g(n) = 9 \log n$.

Problem 3.2 *Selection Sort*   (13 points)

Insertion Sort was discussed during the lecture. Selection Sort is similar to Insertion Sort and works as follows. Given an array of elements, you always take the current element and exchange it with the smallest element that can be found on the right hand side of the current element. In doing so, you will gradually build up a sorted sequence on the left side (like in Insertion Sort), and in each iteration "attach" the smallest element from the remaining unsorted right side to it.

(a) (2 points) Implement Selection Sort.

(b) (3 points) Show that Selection Sort is correct (Hint: consider the loop invariant).

(c) (3 points) Generate random input sequences of length n as well as sequences of length n that represent **Case** A and **Case** B for the Selection Sort algorithm. **Case** A: the case which involves the most swaps (Hint: it is not a decreasingly ordered array). **Case** B: the case with the least swaps. Briefly describe how you generated the sequences (e.g., with a random sequence generator using your chosen language).

(d) (4 points) Run the algorithm on the sequences from (c) with length n for increasing values of n and measure the computation times. Plot curves that show the computation time of the algorithm in **Case** A, **Case** B, and **average case** for an increasing input length n. Note that in order to compute reliable measurements for the **average case**, you have to run the algorithm multiple times for each entry in your plot. You can use a plotting tool/software of your choice (Gnuplot, R, Matlab, Excel, etc.).

(e) (1 point) Interpret the plots from (d) with respect to asymptotic behavior and constants.

How to submit your solutions

You can submit your solutions via *Grader* at **https://grader.eecs.jacobs-university.de** as one generated ZIP file containing one PDF file and source code files and makefile.
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, February 24th, 23:00.