KØBENHAVNS UNIVERSITET

Diskret Matematik og Formelle Sprog: Problem Set 2

Due: Wednesday February 24 at 23:59 CET.
Submission: Please submit your solutions via *Absalon* as PDF file. State your name and e-mail address close to the top of the first page. Solutions should be written in LaTeX or some other math-aware typesetting system with reasonable margins on all sides (at least 2.5 cm). Please try to be precise and to the point in your solutions and refrain from vague statements. *Write so that a fellow student of yours can read, understand, and verify your solutions.* In addition to what is stated below, the general rules in the course information always apply.
Collaboration: Discussions of ideas in groups of two to three people are allowed—and indeed, encouraged—but you should always write up your solutions completely on your own, from start to finish, and you should understand all aspects of them fully. It is not allowed to compose draft solutions together and then continue editing individually, or to share any text, formulas, or pseudo-code. Also, no such material may be downloaded from the internet and/or used verbatim. Submitted solutions will be checked for plagiarism.
Grading: A score of 120 points is guaranteed to be enough to pass this problem set.
Questions: Please do not hesitate to ask the instructor or TAs if any problem statement is unclear, but please make sure to send private messages sometimes specific enough questions could give away the solution to your fellow students, and we want all of you to benefit from working on and learning on the problems. Good luck!

1 Suppose that we are given an array $A = [5, 6, 4, 7, 3, 8, 2, 9, 1, 10]$ to be sorted in increasing order.

1a (30 p) Run insertion sort by hand on this array (as the lecturer has been doing in class), and describe in detail in every step of the algorithm how the elements in the array are moved.

1b (30 p) Run merge sort by hand on this array (as the lecturer has been doing in class), and show in detail in every step of the algorithm what recursive calls are made and how the results from these recursive calls are combined.

1c (20 p) Suppose that we are given another array B that is already sorted in increasing order. How fast do the insertion sort and merge sort algorithms run in this case? Is any of them asymptotically faster than the other as the size of the array B grows?

1d (20 p) Suppose that we are given a third array C that is sorted in *decreasing* order, so that it needs to be reversed to be sorted in the order that we prefer, namely increasing. How fast do the insertion sort and merge sort algorithms run in this case? Is any of them asymptotically faster than the other as the size of the array C grows?

2 Provide formal proofs of the following claims using mathematical induction.

2a (20 p) For all $K \in \mathbb{Z}^+$ it holds that $\sum_{s=1}^{K} s \cdot s! = (K+1)! - 1$.

2b (20 p) For all $s \in \mathbb{N}$ and all $k > 0$ it holds that $1 + sk \leq (1+k)^s$.

2c (30 p) For all $q \in \mathbb{N}$ it holds that $2^{4q+2} + 3^{q+2}$ is a multiple of 13.

2d (30 p) Let the sequence (T_i) be defined by

$$T_i = \begin{cases} 0 & \text{for } i = 1, \\ 1 & \text{for } i = 2, \\ T_{i-1} + T_{i-2} & \text{for } i \geq 3. \end{cases}$$

Prove that for all $i \geq 2$ it holds that

$$1 \leq \frac{T_{i+1}}{T_i} \leq 2 \, .$$