Exercises Complexity Theory

Lecture 2

April 14, 2025

Only the exercises where points are given can be handed in.
(The maximum number of points per exercise is written in the margin.)

To be handed in on **Tuesday April 22, 2025**, in Brightspace under Assignment 2, **deadline: 11:00 AM.**

Exercise 1. For each of the following functions, $p(n)$, give the function $q(n)$ from the list such that $p(n) = \Theta(q(n))$.

$$f(n) = \sum_{i=1}^{n}(4i-4) \qquad g(n) = \sum_{i=1}^{n}\sum_{j=1}^{i} i \qquad h(n) = \sum_{i=1}^{\lfloor \log n \rfloor} n \qquad k(n) = \sum_{i=0}^{n} \frac{4}{2^i}$$

$$q(n) = \quad 1 \quad n \quad n(\log n)^2 \quad n^2 \log n \quad 2^n \quad n^n \quad \log n \quad n\log n \quad n^2 \quad n^3$$

Exercise 2. We have a recursive algorithm whose time complexity $T(n)$ satisfies

$$T(n) = 7T(n-2) + 5T(n-3) + f(n),$$

with $f(n) = \Theta(n^3)$. Find the smallest integer k for which $T(n) = \mathcal{O}(k^n)$ and prove that $T(n) = \mathcal{O}(k^n)$. (Also argue why your k is the smallest.)

Exercise 3. Consider the algorithm for computing the median of an array, as it has been described and discussed in the lecture. We adapt the algorithm by splitting the input array A randomly in $\frac{n}{7}$ groups of 7 elements.

(10) (a) Give the adapted algorithm and give an equation for $T(n)$, the time the algorithm takes on an array of length n.

(20) (b) Analyze the complexity of the algorithm by giving the recursion tree and derive an upper bound $f(n)$ for the running time $T(n)$, where n is the length of A. Is the algorithm still linear?

(10) (c) Prove that the adapted algorithm is $\mathcal{O}(f(n))$, where $f(n)$ is the function you've derived from the recursion tree in (b).

Exercise 4. In this exercise, use a recursion tree to give a good asymptotic upper bound for $T(n)$. Then verify your bound using the substitution method.

(20) (a) Let $T(n)$ be given by

$$T(n) = 3T(n-2) + d$$

for some constant d.

(20) (b) Let $T(n)$ be given by

$$T(n) = 2T(\frac{n}{8} + 8) + n^2.$$

Exercise 5. In this exercise, compute a function f for which $T(n) = \Theta(f(n))$ for the following T using the Master Theorem.

(5) (a) $T(n) = 16T(\frac{n}{2}) + \frac{3}{2}n^4$

(5) (b) $T(n) = 5T(\frac{n}{5}) + n\sqrt{n}$

(5) (c) $T(n) = 4T(\frac{n}{3}) + \log(n)$

(5) (d) $T(n) = 8T(\frac{n}{4}) + n\sqrt[3]{n}$