Advanced Algorithms
Weekplan 1

Lecture

This lecture is about fundamental techniques for effective sequential searching for patterns in strings. The important concept of finite automata will be introduced.

Curriculum for week 1

"Introduction to Algorithms and Data Structures" by Cormen, Leiserson, Rivest, and Stein (CLRS). Sections 32.1, 32.2, 32.3 and 32.4.

Exercises

Prepare solutions to the following exercises in CLRS for the exercise session next week.

- 32.1-2
- 32.1-4
- 32.3-1
- 32.3-5
- 32.4-2
- 32.4-5

Mandatory assignment

The mandatory hand-in assignment consist of the following two exercises. They should be handed in no later than Sep. 12 at the start of the lecture. You are encouraged to discuss the exercises with your classmates but you have to hand in your solutions individually.

- CLRS exercise 32.4-1.
- **Subsequence Matching** Given strings $P[1..m]$ and $T[1..n]$ we say that P is a *subsequence* of T if there is a sequence $x_1, \ldots, x_k$ of indices such that $1 \leq i_1 < i_2 < \cdots < i_m \leq n$ and for all $j = 1, \ldots, m$ we have $P[j] = T[i_j]$. As an example, $P =$ `abc` is a subsequence of $T =$ `xyabddck` since the index sequence $3, 4, 7$ satisfies $P[1] = T[3]$, $P[2] = T[7]$, and $P[3] = T[7]$.

  In the *subsequence matching problem* we are given two sequences $P[1..m]$ and $T[1..n]$ and want to determine if P is a subsequence of T. Design an algorithm for this problem and analyse it. Make your algorithm as fast as possible.