

# IN3130 Exercise set 3

We start with a few short exercises on algorithm running times and running time analysis. This math is central to complexity theory, and important to have a good understanding of. As you know we usually use  $O$ -notation (more correctly, *asymptotic notation*) to indicate running times of algorithms. A short note on the course web page describes four variants of asymptotic notation:  $O$ ,  $\Theta$ ,  $\Omega$  and  $o$ .

## Exercise 1

- Show that  $n \cdot 3$ ,  $n+3$  both are  $O(n)$ .
- Show that  $2n \log n$  is  $O(n^2)$ .
- Is  $2^{n+1} = O(2^n)$ ?
- Is  $\frac{10n + 16n^3}{2} = O(n^2)$ ?

## Exercise 2

- What do we know about the running time of an algorithm if it is  $O(n!)$ ?
- What do we know about the running time of an algorithm if it is  $\Omega(n)$ ?
- What do we know about the running time of an algorithm if it is  $\Theta(2^n)$ ?
- What do we know about the running time of an algorithm if it is  $O(n^2)$ ?
- The statement "This algorithm has a running time of at least  $O(n^2)$ ." may seem odd. Does it make sense?

We continue with a few exercises on string search, partially from the textbook.

## Exercise 3

Spend some time repeating/discussing why/how the different shift strategies of Knuth-Morris-Pratt and simplified Boyer-Moore (Horspool) work. Pay attention to what parts of the pattern  $P$  and  $T$  overlap with what, and why that is necessary for a match to be possible.

## Exercise 4

Find the overlapping prefixes and suffixes (as defined in the Knuth-Morris-Pratt-algorithm) for the string "ababc"

## Exercise 5 (Knuth-Morris-Pratt, Exercise 20.3 in Berman & Paul)

Simulate CreateNext pages 637-8 in Berman & Paul – calculate the Next[]-array for the pattern "abracadabra".

## Exercise 5 (Horspool)

Simulate CreateShift page 639 in Berman & Paul – calculate the array Shift[a:z] for the patterns  $P_1 = \text{"announce"}$ , and  $P_2 = \text{"honolulu"}$ .

## Exercise 6

Draw uncompressed suffix trees for the strings "BABBAGE" and "BAGLADY". And check if "BAG" is a common substring. Can you make do with only one tree?

[end]