

# Fundamental Algorithms 1

## Note:

An R-Exercise, in this course, will refer to material that you will (or should) have discussed in a lecture earlier in your studies – on important mathematical techniques, programming issues, basic algorithms, etc. → check respective text books, online material, etc., to recapitulate these topics, if necessary!

## R-Exercise 1

Prove (by induction over  $n$ ) that  $\frac{1}{3}n^2 + 5n + 30 \in O(n^2)$  for all  $n \in \mathbb{N}^+$ .

## R-Exercise 2

(a) Compare the growth of the following functions using the  $o$ -,  $O$ -, and  $\Theta$ -notation:

1.  $n \log n$
2.  $n^l$  for all  $l \in \mathbb{N}$
3.  $2^n$

Hint: use de l'Hôpital's rule for asymptotically larger or smaller!

(b) Try to give a simple characterization of the growth of the following expressions using the  $\Theta$ -notation:

$$1) \sum_{i=1}^n \frac{1}{i} \qquad 2) \log(n!)$$

Hint for  $\log(n!)$ : try to prove  $n^{\frac{n}{2}} \leq n! \leq n^n$  first!

## R-Exercise 3

Let  $l(x)$  be the number of bits of the representation of  $x$  in the binary system. Prove:

$$\sum_{i=1}^n l(i) \in \Theta(n \log n)$$

## R-Exercise 4

Prove that  $\Theta$  defines an equivalence relation on the set of functions  $\{f \mid f: \mathbb{N} \rightarrow \mathbb{R}\}$ . Use that  $(f, g) \in \Theta \Leftrightarrow f \in \Theta(g)$

## Note

*A Homework, in this course, is intended as a preparation of topics for the next lecture. Typically you will be asked to recapitulate (or learn) basic algorithms. In the lectures, we will **not** take the time to explain how these algorithms work, in detail. Instead, we will discuss certain properties or details of the algorithm or specific implementation.*

## Homework 1

Study the following basic algorithms for sorting:

**InsertionSort:** i.e., sorting a set of data by successively inserting individual items into a sorted list.

**MergeSort:** i.e., splitting a list into two halves, sorting the halves individually, and merging the sorted sublists  $\rightarrow$  in particular, study the **Merge** algorithm for combining two sorted lists into one.

You should understand how the algorithm proceeds to sort a given list of items.