Marburg University

Department of Computer Science
Prof. Dr. Sebastian Wild

Date: 2026-04-24
Version: 2026-04-30 17:56

Exercise Sheet 1 for Advanced Data Structures (Summer 2026)

Hand In: *Until 2026-05-01 18:00, on ILIAS.*

Problem 1

40 points

Suppose you have a playlist containing n songs, represented as a sequence $s_1, \dots, s_n$. Each song has a unique label which never changes; furthermore it also has a quality score, initially 0.

Slice-move: Given $1 \le i \le j \le n$ and $k \notin (i, j]$, remove $s_i, \dots, s_{j-1}$ from the playlist and put it after s_k.

Reverse: Given $1 \le i \le j \le n$, reverse $s_1, \dots, s_{j-1}$.

Change-quality: Given $1 \le i \le j \le n$ and δ, increase the quality of songs $s_i, \dots, s_{j-1}$ by δ.

Rank-query: Given i, output the label of s_i as well as the currenty quality of s_i.

Give a data structure which implements all of these operations. Your data structure should furthermore work in (expected) $O(\log n)$ time for every operation.

Problem 2

30 points

Consider randomised skiplists with unbounded pointer tower height. Prove that the maximal height that occurs is at most $O(\log n)$ with high probability. In particular: for every α, the probability that this height exceeds $\alpha \log_2(n)$ is at most $1/n^{\alpha-1}$.