

Exercise Sheet 5

for the lecture on

Advanced Programming and Algorithms

Submission until **Monday, 20th November, 12:30 pm.**

Discussion in the exercise classes on 20th, 21st and 24th November, 2023.

Problem 1 to hand in: *Data Structure*

Design the following methods in pseudocode (or in Python code as an extension of Problem 2 below). Provide the asymptotic worst-case running time for each method (without proof).

- a) Rewrite the methods that solve the INSERT KEY and SEARCH KEY problems in a doubly linked list.
- b) Provide a method that, given two doubly linked lists A and B , extends list A by list B . The output should be list A that now consists of all elements of both input lists, starting with A and ending in B :

Input: $A: a_1, a_2, \dots, a_n$

$B: b_1, b_2, \dots, b_n$

Output: $A: a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n.$

How can this be solved in $O(1)$?

- c) Finally, provide a method that, given two doubly linked lists A and B zips both lists. The resulting list A should now contain all elements, starting with the first element of A , then the first element of B , then the second element of A , etc.:

Input: $A: a_1, a_2, \dots, a_n$

$B: b_1, b_2, \dots, b_n$

Output: $A: a_1, b_1, a_2, b_2, \dots, a_n, b_n.$

Problem 2 as a programming exercise: *Lists and OOP*

Implement the singly linked list structure from the lecture. For the full learning effect, write all classes and methods yourself without using built-in methods.

Problem 3 for discussion: *zip iterator*

In Python there exists a method `zip` that given two lists $A = [a_1, a_2, \dots, a_n]$ and $B = [b_1, b_2, \dots, b_m]$, computes a list of tuples

$$[(a_1, b_1), (a_2, b_2), \dots, (a_{\min\{m,n\}}, b_{\min\{m,n\}})].$$

How can we use this function to generate a zipped list in the sense of Problem 1 c)?

Can this be done in one line of Python code?

Problem 4 as a programming exercise: *merge*

Write a function, that given two sorted singly linked lists (wrt some \leq relation), returns one sorted list (wrt the same relation) that contains the union of elements from both input lists.

What is the asymptotic running time of this function?