Homework 4 is due May 13, 17:30.

Homework submission A pdf copy of your own solutions to Problems 1–3 should be submitted at SUCourse.

Problem 1 Consider a variation of the Longest Common Subsequence (LCS) problem where k ($k > 2$) sequences are given as input and the goal is to find their LCS.

(a) How would you modify the recursive formula and the top-down LCS algorithm that we have studied in class for two sequences, to find an LCS for k sequences? Please explain.

(b) What is the asymptotic time complexity of your algorithm? Please discuss: is it expected or a surprise?

Problem 2 Given a tree (V, E), design an algorithm using dynamic programming to find a minimum cardinality set $X \subseteq V$ of vertices such that the following holds: for every edge $\{u, v\} \in E$, $u \in X$ or $v \in X$.

(a) Please describe how you design your algorithm (i.e., optimal substructures, recursive solution, construction of solution, etc.), and include a pseudocode of your algorithm.

(b) Analyze the asymptotic time complexity of your algorithm.

(c) How does the computational complexity of this problem change, if a graph is given as an input instead of a tree? Please discuss.

Problem 3 (a) Define the Minimum Leaf Spanning Tree (MLST) problem: What are the input and the output?

(b) Describe a real-world application of MLST.

(c) Prove that MLST is NP-complete.