Department of Computer Science

Prof. Dr. Nikolaus Augsten
Jakob-Haringer-Str. 2
5020 Salzburg, Austria
Telefon: +43 662 8044 6347
E-Mail: nikolaus.augsten@plus.ac.at

Paris Lodron Universität Salzburg

Similarity Search in Large Databases — Exam
Wintersemester 2023/2024 — 31.01.2024

Name: ______________________ **Student ID:** ______________

Hints

- Check whether you received all pages of the exam (9 pages).
- Write your name or your student ID on each sheet of the exam and hand in all pages.
- All answers are expected to be written on the exam sheets.
- Clearly highlight and enumerate additional pages that are used for longer answers. Match your text with the according exercise.
- Only use pencils that are permanent and non-red colored.
- Use the notation and techniques discussed in the lecture.
- Exercises with more than one solution are not graded.
- You are allowed to use one A4 sheet with your personal notes (both sides, hand written or printed).
- Exam duration: 90 minutes

Signature ______________________

Grading — Filled by the examiner

Exercise	1	2	3	4	5	6	7	8	Summe
Total points	2	2	2	2	2	2	2	2	16
Points reached									

Exercise 1 - *Finding Matchings.* 2 Points

For a given distance function δ, the distance matrix d_{ij} for two sets of objects $A = \{a_1, a_2, a_3\}$ and $B = \{b_1, b_2, b_3\}$ is given in Figure 1.

Figure 1: Distance matrix d_{ij} for two sets of objects A and B.

	b_1	b_2	b_3
a_1	5	3	2
a_2	1	3	4
a_3	6	2	1

Compute the result tuples for the following matching strategies:

a) Threshold Matching with $\tau = 2$.

b) k-Nearest Neighbor Matching with $k = 1$.

c) Global Greedy Matching.

Exercise 2 - *String Edit Distance*. 2 Points

Compute the edit distance between the strings `lights` and `tight`. Use the matrix produced by the dynamic programming algorithm to derive the shortest edit scripts and represent them with the gap representation.

Exercise 3 - *q-Grams*. 2 Points

Consider the edit distance computation between two strings. One edit operation affects at most q q-grams.

Provide an example where two edit operations affect strictly less than $2q$ q-grams.

Exercise 4 - *Complexity of the Tree Edit Distance Algorithm.* 2 Points

Prove that the number of key roots of a tree is equal to the number of leaves, i.e., $|kr(T)| = |leaves(T)|$.

Exercise 5 - *Forest Distance Matrix.* 2 Points

Consider ordered trees T_1 and T_2 in Figure 2, forest distance matrix fd, and tree distance matrix td for the trees T_1 and T_2.

a) Compute the left-most leaf descendant arrays l_1, l_2 for trees T_1, T_2, respectively.

b) Fill the missing values for d_i and d_j into the forest distance matrix fd for the circled keyroot nodes in trees T_1 and T_2.

c) Circle the cell in the forest distance matrix that stores the distance between the prefixes $T_1[3..4]$ and $T_2[1..3]$.

d) Cross all cells in the forest distance matrix and/or the tree distance matrix required to compute the distance between the prefixes $T_1[3..4]$ and $T_2[1..3]$.

fd	$\vec{d_j}$					
$d_i \downarrow$						

td	1	2	3	4	5
1					
2					
3					
4					
5					
6					

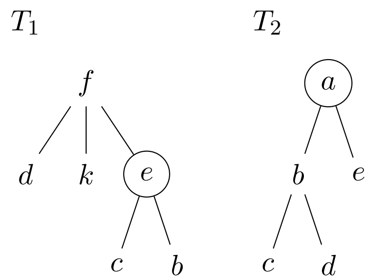

Figure 2: Two ordered trees T_1 and T_2.

Exercise 6 - pq-Gram Distance. 2 Points

For the ordered trees T_1 and T_2 in Figure 4:

a) List the pq-gram profiles of T_1 and T_2 with $p = 2$ and $q = 3$.

b) Compute the pq-gram distance of T_1 and T_2.

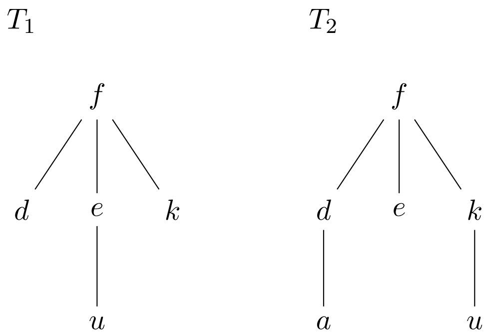

Figure 3: Two ordered trees T_1 and T_2.

Exercise 7 - *Binary Branch Distance and Lower Bound.* 2 Points

For the ordered trees T_1 and T_2 in Figure 4:

a) Represent T_1 and T_2 as normalized binary trees and compute the binary branch distance.

b) Based on the binary branch distance, what is the smallest value that the edit distance between T_1 and T_2 can adopt?

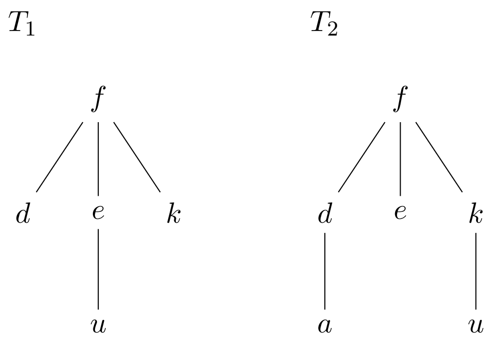

Figure 4: Two ordered trees T_1 and T_2.

Exercise 8 - *Dice Prefix Signature.* 2 Points

Consider the collection $R = \{s_1, s_2, s_3, s_4\}$ of sets in Figure 5. Build an inverted index by addressing the following tasks:

a) Find a suitable ordering technique and sort the sets accordingly.

b) Given a prefix of 2 for all sets, illustrate the inverted list index.

$$
\begin{aligned}
s_1 &= \{A, C, B, D, F, E\} \\
s_2 &= \{D, E, F, B, A\} \\
s_3 &= \{B, D, F, G\} \\
s_4 &= \{C, B, G\}
\end{aligned}
$$

Figure 5: Set collection $R = \{s_1, s_2, s_3, s_4\}$.