Department of Computer Science

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

PARIS LODRON UNIVERSITÄT SALZBURG

Similarity Search in Large Databases — Midterm
Winter Semester 2024/25 — 04.12.2024

Name: _____ **Student ID:** _____

Hints

- Check whether you received all pages of the exam (6 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 (single side, hand written or printed).
- Exam duration: 60 minutes

Signature _____

Grading — Filled by the examiner

Exercise	1	2	3	4	5	Sum
Total points	7	7	7	7	7	35
Points reached						

Exercise 1 — 7 Points

Suffix Array Blocking selects for a string a block key value (BKV) by forming all its suffixes of a length of at least l_{min}. For example, for $l_{\min} = 4$ and string `'albert'`, the following suffixes are formed: `'albert'`, `'lbert'`, `'bert'`.

ID	Name	ZIP	YoB
a_1	Huber	5034	1993
a_2	Mayr	5020	1974
a_3	Christen	5020	1998
a_4	Mayer	5034	1974
a_5	Gruber	5010	2011

ID	Name	ZIP	YoB
b_1	Gruber	5034	1998
b_2	Maier	5020	1974
b_3	Huber	5034	1949
b_4	Gruber	5020	2011
b_5	Chirsten	5020	1998
b_6	Huber	5034	1993

Solution of ER task: $\{(a_1, b_3), (a_1, b_6), (a_2, b_2), (a_3, b_5), (a_4, b_2), (a_5, b_1)\}$

Figure 1: Dirty-dirty entity resolution problem.

1. Assign all records in Figure 1 to blocks based on Suffix Arrays on attribute *name* with $l_{\min} = 4$.
2. Which record pairs are *false negatives*, i.e., pairs that match (according to the given solution) but have not been compared?

Exercise 2 — 7 Points

Search the pattern `'abba'` in the text `'ablaba'` allowing 1 error. Use the matrix produced by the algorithm to find all possible matches and represent them with the gap representation.

Exercise 3 — 7 Points

Compute the set of *positional q-grams* ($q = 3$) for the strings

$$x = \texttt{abvcbc} \quad \text{and} \quad y = \texttt{cbabvbvc},$$

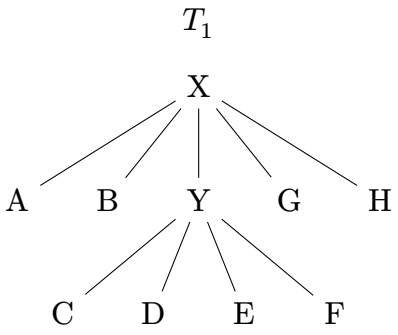
and list all pairs of *corresponding* q-grams for the edit distance threshold $k = 2$.

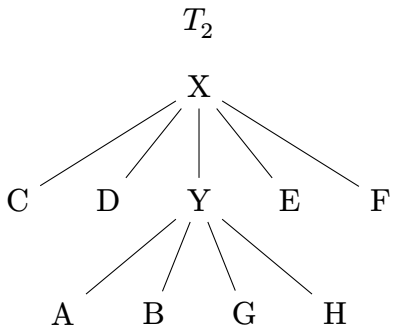
Exercise 4 — 7 Points

In the lecture, we defined the edit operations insertion, deletion, and rename on **ordered** trees. The same operations can also be defined on **unordered** trees with minor differences: As the trees are unordered, deletion does not preserve the order of the children. When inserting a node, the nodes detached from the parent node that are attached to a new node **do not** have to be consecutive, any subset of children can be detached.

Consider the following trees T_1, T_2:

1. Find the minimal number of **unordered** edit operations required to transform T_1 into T_2 if the trees are **unordered**.
2. Argue why strictly more **ordered** edit operations are required to transform T_1 into T_2 if the trees are **ordered**. (You do not have to and are encouraged not to use Zhang&Shasha.)





Exercise 5 — 7 Points

Consider the following definition of an edit distance $ed'(x, y)$ between strings x and y:

For any two strings x, y, $ed'(x, y)$ is equal to the minimum, over all strings z, of the number of deletions and replacements (and **no insertions**) needed to transform x into z and y into z.

Argue that ed' is equivalent to the (unit cost) edit distance defined in the lecture.

Hint: You may use the following property of the standard unit-cost edit distance:

$$ed(x, y) = \min_z(ed(x, z) + ed(y, z))$$

The following geometric interpretation of this statement might help. Note that the existing edit scripts (represented by arrows in the figure) might contain insertions in general. You have to argue that there is a z such that both edit scripts do not contain insertions.

