Question 1: Probabilistic Databases - Implementation (1 P.)

The city hall office uses a OCR software to store handwritten forms in the database. For each form, the person filing it is parsed from the document.

The software sometimes extracts multiple syntactic different entries for the same persons from different forms. Thereby, it is not sufficient to use a `SELECT DISTINCT` statement to eliminate duplicates. Thankfully, the same company developing the OCR software, also develops a software to detect possible duplicate table entries.

Using this software, sets of tuple IDs can be extracted, which are probably the same person. The software also estimates the correctness of the values within the table rows.

As example data the following can be used:

FID	NAME	ZIP-Code	Date of Birth
1	Hoffman	66133	12.11.89
2	Mayer	67663	04.01.93
3	Hofmann	66123	11.11.89
4	Mayer	67653	01.04.1993
5	Meyer	67663	01.04.1993

Sets of same persons: FID = {1,3} and FID = {2,4,5}

FID	Correctness
1	0.6
2	0.3
3	0.4
4	0.2
5	0.5

Your task is to implement a probabilistic database to solve the following tasks:

a) Given a normal person table, create a probabilistic person table. List the possible worlds with their probabilities using the set representation.

b) The probabilistic table should support queries of the format:

- $q_{name}(x) \leftarrow R(\_, x, \text{ZIP}, \text{DoB})$
- $q_{zip}(x) \leftarrow R(\_, \text{Name}, x, \text{DoB})$
- $q_{dob}(x) \leftarrow R(\_, \text{Name}, \text{ZIP}, x)$

For example the queries:

- $q^1_{name}(x) \leftarrow R(\_, x, \_, 01.04.1993)$
- $q^2_{dob}(x) \leftarrow R(\_, 'Hofmann', 66133, x)$

In OLAT you will find a Java template which already supplies most of the boilerplate code and the example data. It will automatically check your implementation for correctness. Submit the code and its output after executing it with the specified sample data.

Required submission: Source code; Output after executing the code;

Question 2: Prob. Databases - Intensional Query Evaluation(1 P.)

Given a probabilistic database D with relations S and T.

$S=$

	A	B	
s_1	'a'	1	0.6
s_2	'b'	3	0.4
s_3	'c'	2	0.8

$T=$

	C	D	
t_1	1	'x'	0.5
t_2	2	'y'	0.3
t_3	1	'y'	0.8

Calculate all possible result tuples and their probabilities, using the intensional query evaluation, of the following query:

$$q^{rank}(z) \leftarrow S(x,y), T(y,z)$$

Determine the logic expressions (using the variables s_i and t_j) that describe the result tuples. Evaluate the truth table of each of these expressions and give the probability for each valid entry.

Hint: You can use the truth table generator found here: `https://web.stanford.edu/class/cs103/tools/truth-table-tool/`

Required submission: Results with probabilities; Intermediate Results;

Question 3: Extensional Query Evaluation and Safe Queries(1 P.)

R=

A	B	
a	1	p_1
a	1	p_2
b	2	p_3

S=

C	D	
2	x	q_1
1	y	q_2
1	x	q_3

Given the query

$$\exists b \, \exists a \; : \; R(a,b) \wedge S(b,c)$$

The intensional query evaluation returns the following results:

D	Expression	P
x	$(p_1 \wedge q_3) \vee (p_2 \wedge q_3) \vee (p_3 \wedge q_1)$	$1-((1-q_3*(1-(1-p_1)*(1-p_2)))*(1-q_1*(1-(1-p_3))))$
y	$(p_1 \wedge q_2) \vee (p_2 \wedge q_2)$	$1-(1-(q_2*(1-(1-p_1)*(1-p_2))))$

(a) Are the following plans safe (Do they return the correct probabilities)?

(i) $\pi_{\mathrm{D}}(\mathrm{R} \bowtie \mathrm{S})$

(ii) $\pi_{\mathrm{D}}(\pi_{\mathrm{B}}(\mathrm{R}) \bowtie \mathrm{S})$

Required submission: Decision Safe/Unsafe; Explanation;

(b) Use PostgreSQL to solve the following question. The following statements create two tables R and S and fills them with values. Given the query $Q(z) = R(z, x), S(x, y)$, provide an safe and an unsafe SQL query to compute the result. You may use the `prod` function, introduced in the lecture.

```
create table R(z char(8), x char(8), p float);
create table S(x char(8), y char(8), p float);
insert into R values('c', 'a1', 0.3);
insert into R values('c', 'a2', 0.4);
insert into R values('c', 'a3', 0.5);
insert into R values('c', 'a4', 0.6);
insert into S values('a1', 'b1', 0.3);
insert into S values('a4', 'b2', 0.4);
insert into S values('a2', 'b2', 0.5);
insert into S values('a2', 'b3', 0.3);
insert into S values('a2', 'b4', 0.4);
insert into S values('a3', 'b2', 0.5);
insert into S values('a4', 'b1', 0.6);
```

Hint:
(i) The correct intensional probability is ≈ 0.746.
(ii) To force a execution plan, you can execute sub-queries in an explicit `WITH .... AS` statement.

Required submission: Queries; Results of queries; Decision Safe/Unsafe;

Question 4: Data Layouts (1 P.)

Consider the following queries from the TPC-H benchmark:

```
select c_name, c_custkey, o_orderkey, o_orderdate, o_totalprice, sum(
   l_quantity)
from customer, orders, lineitem
   where o_orderkey in (
    select l_orderkey
    from lineitem group by l_orderkey
    having sum(l_quantity) > [QUANTITY]
)
and c_custkey = o_custkey and o_orderkey = l_orderkey
group by c_name, c_custkey, o_orderkey, o_orderdate, o_totalprice
order by o_totalprice desc, o_orderdate;
```

We assume every column requires 100 bytes per tuple.

a) What is the minimum amount of bytes that has to be read from the customer table, located on the hard disk, for row and column stores? **Required submission:** Calculation with intermediate results;

b) Provide a vertical partitioning (without redundancies) for the customer table and compare it (using bytes to read) against the row and column stores. **Required submission:** Vertical partitioning; Calculation with intermediate results;

Question 5: MapReduce (1 P.)

Given a text document, where each sentence is stored in a new line, like:

```
Computers are now used in almost all aspects of human activity.
One of their main uses is to manage information [...].
A large amount of data stored in a computer is called a database.
```

To analyze the text in this file, we create a directed graph $G = (V, E)$, with V being the set of words in the text. An edge $(x, y) \in E$ if the word x comes directly before word y in one sentence. For example, will the text above result in {("Computers","are"), ("are","now"),...}.

a) Give a map and a reduce job in pseudo code which creates such a graph, given an input text. The mapper is passed each line individually and the reducer can add an edge to the graph with the `add_edge(x, y)` function.

   How often will the map and reduce jobs be executed respectively?

   Required submission: Pseudo code; Number of invocations;

b) Extend your MapReduce functions to also track how often each word exists in the text and how often each word y follows word x. Use the function `add_weight(v, w)` to add weight w to vertex v in the graph (the word count). With function `add_edge(x, y, w)` you can add the weight to an edge.

   Required submission: Pseudo code;

Question 6: Potpourri (0 P.)

In the following we provide a collection of questions to help you prepare for the exam. Some questions are similar to questions we could ask in the exam, while others encourage you to gather a deeper understanding of a topic. This is of course not an exhaustive list of topics/questions we may use in the exam. These questions are also only a small part of the exam (see previous exams [1]).

Of course, the whole lecture is relevant to the exam, if not explicitly stated otherwise.

1) What do OLTP and OLAP mean and where are the differences?
2) Describe the advantages and disadvantages of compression in column stores?
3) For each multidimensional tree, can you find a case where you would prefer that tree to any of the others?
4) Why is it necessary to have all the information required to trace a TA written to persistent memory before committing? What "letter" of ACID does this affect?
5) Try to transfer the principle of MINDIST and MINMAXDIST estimations in the R-tree to other scenarios.
6) Does the threshold algorithm never perform worse than Fagin's algorithm? What measure for "better" or "worse" do you use?
7) How can you compute skyline queries, with *min* and *max* preferences, using only the *min* operator?
8) Which of the hashing techniques shown in the lecture would you use for a hash join?
9) Give an example of what can happen if the DBMS does not use WAL or does not follow the commit rule.
10) What is the "problem" with the intensional query evaluation in probabilistic databases?
11) Create a schedule, containing a lost update anomaly.
12) Describe how you would extend the kd tree to be able to store multiple data points per node. How would insertion, search and remove operations work?
13) For each of the three pruning strategies for the Nearest Neighbor search in the R tree, specify a case in which the strategy is very useful and one in which the strategy is not useful.
14) What is the difference between the following forumlas to calculate the allocation factor? 1. |Data tuples in primary file|/(|Bucket capacity|·|Buckets in primary file|) and 2. |All data tuples|/(|Bucket capacity| · |All Buckets|)
15) Describe the idea behind min-hashing and explain why the collision probability corresponds to the Jaccard coefficient of the sets involved.
16) Give an example, using two relations, with one correct and one incorrect plan regarding extensional query evaluation. What is the problem?
17) What is the difference between possible tuple and possible answer set semantic?
18) What is the #SAT problem? What is the difference to SAT?
19) How do you rate a procedure that is similar to the R-tree, but works with MB circles instead of MB rectangles ?

[1] https://kai.cs.uni-kl.de/pruefung/lv/610/

20) Describe the differences between transaction-consistent and fuzzy checkpoint procedures with regard to performance at the time of creation and restart.

21) Give an example of a BID database. What does BID stand for?

22) Why did we consider the extensional query evaluation and which problems does it have?

23) How would you choose a good pivot element in the GH tree and how would you define "good"?

24) Why is it useful to know how complicated it is to maintain materialized views?

25) What is the difference between SS2PL, S2PL and 2PL? Which scheduler is more restrictive?

26) Specify a way to map data from a higher dimensional space to a 1-dimensional space.

27) Look at the 5-layer model. Assign the topics of the lecture to the layers.

28) Describe how the inner nodes of an M-tree are structured.

29) Describe the difference between wound-wait and wait-die for a transaction t_i that conflicts with transaction t_j and t_i was started before t_j.

30) Describe the two phases of 2PL and specify a schedule that is not in Gen(2PL).

31) What do DSM and NSM mean?

32) Do we require undo/redo in the configuration "¬steal and ¬force"?

33) Can the same objects that you insert into an R tree be inserted into a kd tree?

34) For each larger topic (e.g., transactions, indices, ...) considered in the lecture, describe what influence their absence has on a DBMS.

35) Describe how you would implement the insert and delete operations for the GH tree.

36) Can TI databases be expressed using PC databases, if so why or how?

37) Describe what action and transaction consistency means and for which checkpoint procedures this is considered.

38) During the execution of a transaction, the database system crashes. Which "letters" of ACID are used now?

39) For each of the two prioritizations MINDIST and MINMAXDIST, specify a case in which one strategy is clearly superior to the other. Can you give a rule of thumb when to use which strategy?

40) Describe how a conflict is detected in timestamp ordering.

41) The log ring buffer has a capacity of 2^{15} entries. How many concurrent transactions can the database system handle?

42) Why are row stores not (well) suited for using compression?

43) What is better for read-intensive workloads using linear hashing: controlled splitting with low β_s, with high β_s, or uncontrolled splitting?

44) Compare the hashing algorithms with caching algorithms that you know from computer systems or similar lectures. What are the similarities and differences of the functionalities and requirements?

45) Why is it desirable to use a monotonous serialization class?

46) Which kind of index could be used to efficiently calculate k-skyband queries?

47) What are the worst-case complexities of searching in a quadtree and PR tree?

48) How/Why does wound-wait and wait-die prevent cycles in the WFG?

49) Why is the configuration "steal and ¬force" usually considered the best configuration for performance reasons?

50) In the R tree, the MBRs of all child nodes must be completely in the MBR of the parent node. Could this requirement also be weakened by saying that the child MBRs only have to intersect the parent MBR?

51) Is ACA absolutely necessary for ACID?

52) What logging method would you use if a) you have infinite memory available, but the system is so unstable that recovery needs to be done frequently, or b) recovery happens very rarely, but you can only use slow and limited permanent memory.

53) What is the structure of inner nodes in an R-tree?

54) Describe how the existence of secondary indexes affects transaction management and possible queries.

55) Why does the redo process start before the checkpoint if you are using fuzzy checkpoints?

56) Consider the following operation: $w_1(x)$, where x is a row of a table. What do you have to keep in mind to prevent multi-user anomalies; Keywords: indices and materialized views?

57) What does recoverable mean? Specify a schedule that is in RC and one that is not in RC (but in CSR).

58) A database stores two-dimenstional points. It does not support nearest neighbor search, but may return all points in a given rectangle. How can you use this database to efficiently answer skyline queries?

59) Discuss which consequences changes to the system configuration on slide 581 would have.

60) Why do we have to redo loser transactions, if we use the configuration used in the lecture? Under which circumstances would we not have to do that?

61) When is an aggregation function (in the context of Top-K algorithms) monotonous?

62) Which layout is usually better suited for OLTP, row or column store?

63) Describe a deadlock prevention method in which each transaction is assigned a priority $\in \mathbb{N}$ and in which the sum of the priorities of the restarted TA is to be minimized in order to avoid a deadlock.

64) What is Late Materialization?

65) Why is there the parameter m in the R-tree, which specifies that every node has at least $m \leq \lceil M/2 \rceil$ entries?

66) How do you create a serializability graph from a conflicting-step graph?

67) What goes wrong when using the FA algorithm with an aggregation function that is not monotonous?

68) Which conflict does the Thomas' Write Rule avoid?

69) What does MinDirtyPageLSN mean and where is it used?

70) Compare Quadtree, kd-Tree and Z-Curve in terms of two-dimensional area searches (i.e. "Which objects are in this rectangle?").

71) What is the best-case/worst-case for column stores vs. row stores?

72) Extend the timestamp-based approach presented in the lecture with versioning. Read operations are extended so that $r(x,t)$ indicates that the object x is read in the state from the time t. What does a write access look like? How many versions of an object must be stored? Is ACID still guaranteed?

73) Why are redo and undo phases idempotent and what does this mean?

74) Create a schedule that is in RC, but not in ACA. Describe what ACA means in your own words.

75) What does OCSR mean and what is the difference to the CSR class?

76) How relevant are the chapters on join implementation and optimization if you assume that all relations are stored in main memory?

77) Why are there different isolation levels when the "I" in ACID requires full serializability?

78) Use the illustrations in the script to describe exactly how the search for a key works in Extensible Hashing.

79) What are CLRs and why do we need them?

80) Is RC essential for ACID? And if so, for what part of ACID?

81) What are safe queries in probabilistic databases?

82) Find an example that shows that VSR is not monotonous.

83) What is the idea behind PAX?

84) Can we do without LSNs, if we store the transaction numbers in the logs?