Examination examples

Databasteknik (5 hours)

1. **Relational Algebra & SQL (4 pts total; 2 pts each).**

   Part A Consider the relations $R(A, B)$, and $S(C, D)$. Of the following three equivalences between expressions of relational algebra, which are true?

   (a) $\pi_{A,B}(R \times S) = R$

   (b) $\pi_{A,B,D}(R \bowtie_{B=C} S) = R \bowtie \rho_{T(B,D)}(S)$

   (c) $R - \rho_{T(A,B)}(S) = \rho_{T(A,B)}(S - \rho_{U(C,D)}(R))$

   Part B Suppose we are given a relation Empls(employeeID, personNum, name, managerID) and want to find the ID's of all employees who do *not* manage any employee named Sandra. Here are two possible queries:

```
  I. SELECT managerID
     FROM   Empls
     WHERE  NOT EXISTS(SELECT * FROM Empls WHERE name = ’Sandra’);

 II. SELECT managerID
     FROM   Empls
     WHERE  NOT(employeeID =
                  SOME(SELECT employeeID FROM Empls WHERE name =
’Sandra’));
```

   Which, if any, of the two queries will correctly get the desired set of employees?

---

2. **Relational Database Design (6 pts total).** Suppose we have a relation $R(A, B, C, D, E)$ with the following set of functional dependencies: $A \to B$, $C \to D$, $BD \to E$, and $E \to C$.

   (a) **(1 pt)** Indicate which are the *candidate keys* for R.

   (b) **(1.5 pts)** Indicate which of the functional dependencies violate BCNF.

   (c) **(1.5 pts)** Indicate which of the functional dependencies violate 3NF.

   (d) **(2 pts)** Suppose we decompose R into several relations, one of which has schema ACE. What nontrivial functional dependencies hold for ACE? You need not mention a functional dependency if it follows from another that you mention by adding attributes to the left side, and you need not mention functional dependencies that have more than one attribute on the right.

---

3. **Object-Oriented and Object-Relational Databases (4 pts total)**

   a) What three kinds of user-definable database extensibility mechanisms available in an object-relational database system?

   b) Which of the above extensibility mechanisms are lacking or weak in a simple object-oriented database system (i.e. in an object store)?

4. **Security and Integrity (4 pts total)**

   a) How is authorization specified in modern relational databases?

   b) Why are views useful for authorization?

   c) When can a user transfer authorization rights to another user?

   d) What is ’domain constraints’ and what are they used for?

5. **Query Optimization (4 pts total)**

   a) What is selectivity and how is it used in cost-based query optimization?

   b) What is the worst case complexity of cost-based query optimization and how does one avoid this cost?

   c) Why does cost-based query optimization pay off despite its complexity?

   d) What is 'evaluation primitives'? Give some examples of evaluation primitives. How are they used by a cost-based query optimizer?

6. **Relational Algebra & SQL (6 pts total).** Consider the following relations for keeping track of customers, products, and purchases:

   product(pid,pname,type,vendor,price)
   bought(cid,pid)
   customer(cid,cname,age,sex)

   Write the following queries in the relational algebra and in SQL:

   (a) **(1+1 pts)** Find the name and price of each product that has been purchased by at least one male customer.

   (b) **(1+1 pts)** Find the name of each customer that has not purchased a product whose type is computer.

   (c) **(1+1 pts)** Find the name and sex of each each "supershopper" — a customer who has purchased all products costing more than 10000.

7. **Security and Integrity (6 pts total, 2 each)**

   (a) Why are views useful for authorization in modern relational databases?

   (b) How is referential integrity specified in modern relational databases?

   (c) What is 'domain constraints' in the relational data model and what are they used for?

8. **Query Optimization (6 pts total, 2 each)**

   Given is the relational table

   ```
   SALARIES(SSN,AMOUNT)
   ```

   where SSN is key with ordered primary index and AMOUNT has an ordered secondary index. The table has 1000000 tuples. Each data block holds 100 tuples.

   We state the query

   ```
   SELECT SSN, AMOUNT FROM SALARIES ORDER BY AMOUNT DESCENDING
   ```

   (a) Outline the cheapest execution plan (using evaluation primitives) for evaluating the above query.

(b) Approximately how many blocks does the above plan access? Show calculations and assumptions.

(c) Describe shortly the principles of cost-based query optimization.

9. **Active Databases (6 pts total, 2 each)**

   We have two tables

```
EXPLOYEE(SSN, DNO, SALARY), key SSN
DEPT(DNO, MGRSSN), key DNO
SSN: Social security number
DNO: Department number
SALARY: Employee (incl. manager) salary
MGRSSN: Social security number of department manager
```

(a) Outline code of how to use active database facilities to maintain a master table of the sum of all salaries per department:

```
SALSUM(DNO,SALSUM), key DNO
```

(b) Assume a table maximizing the total salary budget for each department:

```
BUDGET(DNO,TOTSAL), key DNO
```

Outline code for using same kind of active database facilities to guarantee that no department ever has higher salaries than budgeted.

(c) What extra functionality does an active relational database provide compared to a passive one?

10. **ODBC (2 pts total, 1 each)**

(a) What is ODBC?

(b) What is the difference between JDBC and ODBC?

11. **Relational Algebra & SQL (7 pts total).** Consider the following self-expanatory relational schema:

Student(id: int, name: string, nation: string, age: int)
Plays(id: int, sport: string)

(a) **(2 pts)** Can you write a SQL query that is equivalent to the following one but does not use the keyword DISTINCT? If so, give the query. If not, explain why not.

```
SELECT DISTINCT nation
FROM student
WHERE age > 25
```

(b) **(2 pts)** Can you write a SQL query that is equivalent to the following one but does not use a HAVING clause? If so, give the query. If not, explain why not.

```
SELECT sport
FROM student, plays
WHERE student.id = plays.id
GROUP BY sport
HAVING AVG(age) > 25
```

(c) **(3 pts)** The following question does not depend on the relational schema above. Consider the following SQL query:

```
SELECT p.a1
FROM   p, r1, r2
WHERE  p.a1 = r1.a1  OR  p.a1 = r2.a1
```

Under which conditions does this preceding query select values of `p.a1` that are either in `r1` or in `r2` ? Be precise in your answer and careful to examine all cases.

12. **Relational Database Design (5 pts total).** Suppose we have a relation $R(A, B, C, D, E)$ with functional dependencies $F = \{AB \rightarrow C, C \rightarrow E, E \rightarrow C, C \rightarrow D, AB \rightarrow E\}$.

(a) **(3pts)** Find a dependency preserving, lossless join decomposition of R into 3NF.

(b) **(2pts)** Is the decomposition $\{ABC, ADE, CE\}$ dependency preserving with respect to F? Explain briefly.

13. **Active Databases (6 pts total)**

a) What are the components of ECA in a relational database? (3 pts)

b) What extra functionality does an active relational database provide compared to a passive one? (2 pts)

c) Why is it often difficult to implement complex integrity constraints with triggers? (1 pt)

14. **Object-Oriented Databases (6 pts total)**

a) What is pointer swizzling? (1 pt)

b) What is an OID? (1 pt)

c) What facilities are better supported in relational databases than in first generation object-oriented databases (object stores)? (4 pts)

15. **Query Processing (4 pts total)**

A large company maintains a table of the effectiveness of their sales force:

```
SALES(SSN, SALES)
```

containing the SALES (in $) of each sales person identified with SSN, primary index on SSN and secondary index on SALES. There are 10000 sales persons and 100 tuples fit a disk block. The management need to regularly know the top ten sales performers and you are asked to design an application program to quickly get those stars.

a) What query would your application use and how would it be called to get the answer quickly? (2 pts)

b) What is the optimal query execution plan? (1 pt)

c) How many disk blocks are read? Motivate shortly. (1 pt)

16. **Relational calculus (4 pts total)**

    Express over relations

    ```
    PERSON(SSN, NAME, ADDRESS, PHONE)
    EMPLOYEE(SSN, SALARY, DEPT)
    ```

    the query below in domain calculus, and tuple calculus, respectively.

    ```
    SELECT NAME, SALARY
                 FROM PERSON, EMPLOYEE WHERE SALARY>100000 AND
                                             PERSON.SSN = EMPLOYEE.SSN
    ```