Database Management Systems (COP 5725)

Spring 2016

Instructor: Dr. Markus Schneider

TA: Kyuseo Park, Lin Qi

Exam 2

Name:	
UFID:	
Email Address:	

Pledge (Must be signed according to UF Honor Code)

On my honor, I have neither given nor received unauthorized aid in doing this assignment.

____

Signature

For scoring use only:

	Maximum	Received
Question 1	25	
Question 2	45	
Question 3	10	
Question 4	20	
Total	100	

Question 1 (Relational Algebra) [25 points]

Consider the following relation schema below (primary keys are underlined).

project (pno, pname, plocation, dnum) [dnum references dnumber in department]
work_on (essn, pno, hours) [essn references ssn in employee]
department (dnumber, dname, mgrssn, mgrstartdate)
[mgrssn refers to an ssn of a manager]
employee (ssn, fname, lname, bdate, addr, sex, salary, superssn, dno)
[superssn refers to an ssn of a supervisor, dno references dnumber in department]
dependent (essn, dependent_name, sex, bdate, relationship)
[essn references ssn in employee]

Write the following queries in Relational Algebra.

1. Find the names of all managers who have at least one dependent. [5 points]

2. Find the last names and ssn of all employees who work on project number 1 and on project number 2. [5 points]

3. Find the names of all employees who earn more than their supervisors. [5 points]

4. List the names of all employees whose department manager earns less than $25,000. [5 points]

5. Find the names of all employees who work on the project that is conducted by department 5. [5 points]

Question 2 (SQL) [45 points]

A) Consider the following table schemas:

branch (branch_name, branch_city, assets)
customer (customer_name, customer_street, customer_city)
account (account_number, branch_name, balance)
loan (loan_number, branch_name, amount)
depositor (customer_name , account_number)
borrower (customer_name, loan_number)

Write SQL statements for the following queries:

1. Find the number of depositors for each branch. [4 points]

2. Find all customers who have both an account and a loan at the Gainesville branch. [5 points]

3. Find the average balance of all customers who live in Gainesville and have more than three accounts. [5 points]

4. Find the name of the branch where the average account balance is greater than the other branches. [6 points]

B) Consider the following table schemas:

dept (deptno, dname, loc)
emp (empno, ename, job, hiredate, sal, deptno) (:employee's information)

Write SQL statements for the following queries:

5. Find the employees (empno, name, job, salary) whose job is the same as the employee with empno = 7777 and who receive less salary than the employee with empno = 8888. [7 points]

6. Find the employees (that is, output deptno, dname, ename, job, hiredate) who work in the sales department and were hired in 2015. (in ascending order by hiredate) (4 points)

7. Find the departments (that is, output deptno, number of employees) for which more employees work than for other departments. [7 points]

8. Find the employees (that is, output empno, name, deptno, hiredate) who have worked longer than others in each department. [7 points]

Question 3 (Relational Algebra and SQL) [10 points]

The division (or quotient) operator ÷ of the Relational Algebra does not have a direct equivalent in SQL. It identifies attribute values from a relation that are paired with all of the values from another relation. Without loss of generality, let **R** = $\{A_1 : C_1, \ldots, A_n : C_n, B_1 : D_1, \ldots, B_m : D_m\}$ and **S** = $\{B_1 : D_1, \ldots, B_m : D_m\}$ be two relation schemas. Let R be a relation with respect to **R** and S be a relation with respect to **S**.

1. Provide the formal definition of the division operator by means of a Relational Algebra expression that only makes use of the basic Relational Algebra operators. [4 points]

2. A first solution to mapping the definition in 1. into SQL is somewhat difficult to understand. It makes use of a doubly nested and negated SQL statement and is based on the *not exists* predicate. We are here not interested in this solution. Instead, we follow another idea that maps the different parts of the definition in 1. one by one into SQL and does not use the *not exists* predicate. Write down the corresponding SQL query. [6 points]

Question 4 (QBE) [20 points]

Consider the following database schema:

Drivers (did, dname, gender, age)
Reserve (did, cid, day, cost)
Cars (cid, cname, model, color, rid)
RentalCompany (rid, rname, revenue, rating)
IsMember(did, rid, join_time, member_type)

Primary key attributes are underlined.

Answer the following questions using QBE. Draw tables in your answer.

1. Find the oldest driver who is a member of the company ‘Budget’ and the company ‘Avis’. [7 points]

2. Find the name of the customer who has reserved a car named ‘A6’ on ‘01/03/2016’ from the company ‘Budget’. [6 points]

3. Update the member type to ‘VIP’ for those drivers who were members of company ‘Avis’ and have spent more than 1000 in renting (reserving) cars from Avis. [7 points]