UCLA
Computer Science Department
Fall 2015

Instructor: J. Cho

Student Name and ID: ______________________________

CS143 Midterm: Closed Book, 110 minutes

(IMPORTANT PLEASE READ **):**

- There are 5 problems on 12 pages for a total of 110 points. *You should look through the entire exam before getting started, in order to plan your strategy.*
- *Simplicity and clarity of your solutions will count.* You may get as few as 0 point for a problem if your solution is far more complicated than necessary, or if we cannot understand your solution.
- Unless otherwise indicated, assume the SQL92 standard and the notation that we learned in the class.
- If you need to make any assumption to solve a question, please write down your assumptions.
- To get partial credits, you may want to write down how you arrived at your answer step by step.
- You may use one-page double-sided cheat-sheet during exam. You are also allowed to use a calculator.
- Please, write your answers neatly. Attach extra pages as needed. Write your name and ID on the extra pages.

Problem	**Score**	
1	20	
2	30	
3	20	
4	10	
5	30	
Total	110	

Problem 1 (Queries): 20 points

Two queries are considered equivalent if they return exactly the same results for all database instances. For each of the pair of queries listed below, write "YES" if the two queries are equivalent and "NO" otherwise.

Assume the following for the two relations R and S referenced in the queries.

- Both R and S have two columns, $R(A, B)$ and $S(A, B)$.
- The attribute A is a key for each relation.
- The attribute B is not a key in either relation.
- NULL values are not allowed in any attribute.

Do not make any other assumptions about the data. To discourage randomly guessing the answers, we will give 4 points for each correct answer, deduct 1 point for each incorrect answer, and 0 point for each answer left blank.

Again, remember that A is a key, B is not, and NULL values are not allowed. These assumptions are crucial to derive the correct answers.

Note: you do not need provide an explanation to your answer. No partial point will be assigned to explanation.

1. Equivalent? ________

   (a) $\pi_A(R - S)$

   (b) $\pi_A(R) - \pi_A(S)$

   ANSWER:
   `No. Consider` $R = \{(1, 2)\}, S = \{(1, 3)\}$

2. Equivalent? ________

   (a) $\pi_A(R) - \pi_{R1.A}(\sigma_{R1.A<R2.A}(\rho_{R1}(R) \times \rho_{R2}(R)))$

   (b) `SELECT MAX(A) FROM R`

   ANSWER:
   `Yes.`

3. Equivalent? ________

   (a) `SELECT B FROM R GROUP BY B`

   (b) `SELECT DISTINCT B FROM R`

   ANSWER:
   Yes

4. Equivalent? ________

   (a)
   ```
   SELECT B
   FROM R
   WHERE NOT EXISTS(SELECT * FROM S WHERE R.B = S.B)
   ```

   (b) `(SELECT B FROM R) EXCEPT (SELECT B FROM S)`

   ANSWER:
   No. Consider $R = \{(1, 0), (2, 0)\}$ and $S = \{\}$.

5. Equivalent? ________

   (a)
   ```
   SELECT B
   FROM R R1
   WHERE B <= ALL (SELECT B FROM R R2 WHERE R1.A <> R2.A))
   ```

   (b) `SELECT MIN(B) FROM R`

   ANSWER:
   No. Consider $R = \{(1, 0), (2, 0)\}$.

Problem 2 (30 points)

Suppose the relation `R(A,B)` has n tuples and the relation `S(B,C)` has m tuples. As usual, underlined attributes are the keys (`A` is the key of `R` and `B` is the key of `S`). Assume that NULLs are not allowed for any attribute. Consider the relational algebra expression $\sigma_{R.B=S.B}(R \times S)$.

1. What is the minimum and the maximum number of tuples that may result from the above expression? Explain your answer briefly. (10 points)

   ANSWER:
   `Min: 0, Max:` n.

2. Now assume that `R.B` is a foreign key referencing `S.B` and `S.C` is a foreign key referencing `R.A`. What is the minimum and the maximum numbers of tuples from the above expression? Explain your answers briefly. (10 points)

   ANSWER:
   `Min:` n, `Max:` n.

3. Suppose the relation `T(A,B)` has m tuples and the relation `U(B)` has n tuples. Consider the relational algebra expression T/U. What is the minimum and the maximum numbers of tuples that may result from the above expression? Explain your answer briefly. (10 points)

   ANSWER:
   `Min: 0, Max:` $\lfloor \frac{m}{n} \rfloor$

Problem 3: 20 points

Assume the following database schema:

```
Manager(mid, name, department, salary)
Employee(eid, name, department, salary)
```

A person is either a manager or an employee, but not both. No attribute can have the NULL value. Every department's name is unique.

Now we want to enforce the following constraint on this database:

A mangers salary is higher than the salary of all employees in the same department.

1. What database modifications can potentially violate the constraint? In the following table write YES for the operations that can potentially violate the constraint. For example, if the deletion of a tuple from the `Manager` table may violate the constraint, write YES in the cell corresponding to the `Manager` row and the `DELETE` column. (10 points)

	INSERT	DELETE	UPDATE
`Manager`			
`Employee`			

ANSWER:
`Yes, No, Yes`
`Yes, No, Yes`

2. Is it possible to write one (or more) CHECK constraints and reject any modifications that will violate the constraint? If yes, show the CHECK constraint(s) portion of the relevant `CREATE TABLE` statements. For each constraint, write the table name in which the constraint is declared. If not, briefly explain why. (10 points)

   ANSWER:

```
Yes, To Manager table, add CHECK(salary > ALL (SELECT salary FROM Employee
WHERE Employee.department = Manager.department)).  To Employee table, add
CHECK (salary < ALL (SELECT salary FROM Manager WHERE Manager.department
= Employee.department)).
```

Problem 4 (Authorization): 10 points

Consider the following two tables in a database:

```
Student(sid, name, address, GPA)
Major(sid, dept)
```

`sid` is a key for `Student`, and `(sid, dept)` together is a key for `Major` (i.e., a student may have multiple majors). No attributes are permitted to be NULL. The owner (creator) of these tables is a user named Hennessy. Hennessy wants to grant to a user named Sahami the following permissions.

1. See the sids of those students who are majoring in “CS”.
2. Add CS as a student’s major (presuming the student is already in the database but not majoring in CS).

Is it possible to specify a command or sequence of commands that achieves this goal? If so, write down such command(s). If not, explain why not. Make sure to adhere to the SQL standard. You may lose points if your example/explanation is overly complicated. (10 points)

ANSWER:

```
CREATE VIEW CSMajor AS
SELECT * FROM Major WHERE dept='CS'
WITH CHECK OPTION;

GRANT SELECT, INSERT ON CSMajor TO 'Sahami';
```

Problem 5 (30 points)

Assume a disk of the following characteristics for this problem:

- 800GB total disk capacity
- 10 surfaces
- 10,000 tracks per surface
- 6000 RPM
- 10ms average seek time
- 8KB blocks (sectors are the same size.)

Assume that 1GB = 1,000,000,000B, 1MB = 1,000,000B, and 1KB = 1,000B.

1. Suppose that we are reading a file F that occupies exactly one entire track, and we want to estimate how long it takes to read the whole file sequentially. Do you need any any additional key parameter for doing this? Select one from the following choices:

   (a) the diameter of the disk surface — d
   (b) the average rotational latency — r
   (c) the transfer rate of the disk — t
   (d) none of the above

   Based on your choice above, how long does it take to read the file? You may use the symbol (after each choice) to represent the parameter that you think is useful in the estimation result. Assume that the disk head may not initially be on top of the track where the file is located. (5 points)

   ANSWER:
   `(d).  10ms + 5ms + 10ms = 25ms.`

2. Now, suppose that we want to estimate how long it takes to read a block residing in a random location. Do you need any additional key parameter for doing this? Select one from the following choices:

   (a) the diameter of the disk surface — d

   (b) the average rotational latency — r

   (c) the transfer rate of the disk — t

   (d) none of the above

   Based on your choice above, how long does it take to read a block? You may use the symbol (after each choice) to represent the parameter that you think is useful in the estimation result. (5 points)

   ANSWER:
   `(d). 800GB = 10 surfaces * 10,000 tracks * 8KB *` x `sectors/track.` x `= 1000 sectors/track. 10ms/1000 = 0.01ms per sector. time to read a block = 10ms + 5ms + 0.01ms = 15.01ms`

3. You need to store the following table in the above disk:

   `Customer(id LONG, name CHAR(32))`

   LONGs are 8 bytes and CHAR(32)'s are 32 bytes. Assume that tuples are not spanned across blocks. The table currently has 1 million tuples. What is the minimum number of blocks needed to store this table? (5 points)

   ANSWER:
   $\lfloor 8000/40 \rfloor = 200$ `tuples/block.` $\lceil 1,000,000/200 \rceil = 5,000 blocks needed$

4. To make the lookup of `Customer` table more efficient, we decide to create a B+tree index on the attribute `id`. The B+tree index is dense. Each B+node has spaces for 400 keys and 401 pointers (i.e., $n = 401$) One node of the B+tree is stored in one disk block. What is the minimum number of leaf nodes that are needed for the B+tree of the `Customer` table? (5 points)

   ANSWER:
   $n = 500$. # of leaves: $\lceil 1,000,000/400 \rceil = 2,500$.

5. Given the minimum number of leaf nodes, what is minimum height of the B+tree? Briefly explain how you arrived at your answer. (5 points)

   ANSWER:
   3. # of non-leaves: $\lceil 2,500/401 \rceil = 7$. root = 1.

6. Using the B+tree constructed above, how long would it take to execute the following query on average?

   ```
   SELECT * name FROM Customer WHERE id = 1024
   ```

   Assume that the root node of the B+tree is always cached in main memory and does not require a disk IO to read it. No other nodes are cached in main memory. (5 points)

   ANSWER:
   one IO for non-leaf, one IO for leaf, one IO for table $3 \times 15.01ms = 45.03ms$.

This page is intentionally left empty. You can use this page as scratch paper.