CSE 414 Final Examination

Monday, June 10, 2019, 2:30-4:20

Name: _______________________

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>Total:</td>
<td>200</td>
<td></td>
</tr>
</tbody>
</table>

- This exam is CLOSED book and CLOSED devices.
- You are allowed TWO, HAND-WRITTEN letter-size sheets with notes (both sides).
- You have 110 minutes;
- Answer the easy questions before you spend too much time on the more difficult ones.
- Good luck!
1 Relational Data Model

1. (25 points)

A marine biologist collects samples of micro-organisms from the ocean. She stores her data in a relational database with the following schema:

Sample(sid, lat, long, depth, technician)
Analysis(sid, oid)
Organism(oid, name)

- Sample represents a small amount of water taken from the ocean. The data records the latitude, longitude, and depth where the sample was taken (real numbers), and the name of the technician (of type text) who collected and analyzed the sample.
- Each sample is analyzed for organisms. Analysis lists all organisms found in that sample. sid and oid are foreign keys to Sample and Organism respectively (integers).
- Organism is a collection of names of organisms (e.g. marine microbes). The name is a text.
- sid, oid are integers.

Solution:

```sql
-- FOR TESTING ONLY!!!
drop table if exists Analysis;
drop table if exists Sample;
drop table if exists Organism;
create table Sample (sid int primary key, lat real, long real, depth int, technician text);
create table Organism (oid int primary key, name text);
create table Analysis (sid int references Sample, oid int references Organism);
```

(a) (5 points) Technicians are rewarded by the total number of organisms that they identify in all samples. Write a SQL query that computes, for each technician, the total number of organisms that they found in all samples. (If Alice finds, say, Trichodesmium, in two different samples, then you count this as 2.) Your query should return the technician’s name, the total number of organisms they identified, ordered decreasingly by this number.

Write your answer below:

Solution:

```sql
select x.technician, count(*)
from Sample x, Analysis y
where x.sid = y.sid
```
group by x.technician
order by count(*) desc;
(b) (5 points) The biologists is particularly interested in the geographical area with latitude from 20.5 to 22.3 and longitude from $-158.3$ to $-156.7$. (Yes, this is close to Hawaii!) Write a SQL query that returns the oids and names all organisms that were found in more than 100 samples from this area.

**Write your answer below:**

```sql
Solution:
select z.oid, z.name
from Sample x, Analysis y, Organism z
where x.sid = y.sid and y.oid = z.oid
  and 20.5 < x.lat and x.lat < 22.3
  and -158.3 < x.long and x.long < -156.7
group by z.oid, z.name
having count(*) > 100;
```
(c) (15 points) A deep-sea organism is one that lives only below a depth of 1000m. Write a SQL query that returns the names of all organisms that were found only at a depth greater than 1000m. (The depth is an integer and is represented in meters.) Write your answer below:

Solution:

```sql
select x.oid, x.name
from Organism x
where not exists
  (select *
    from Analysis y, Sample z
    where x.oid = y.oid and y.sid = z.sid and z.depth < 1000);
```
2 NoSQL, JSON, SQL++

The raw data that biologist used to populate her database came in JSON format directly from the ship where the samples were collected and analyzed. It had the following structure:

```json
"samples": [  
{ "sid": "s001",  
  "lat": "21.2",  
  "long": "-157.9",  
  "depth": "300",  
  "analysis":  
  { "technician": "Alice",  
    "organisms": [ {"oid": "o252", "name": "Trichodesmium"},  
                    {"oid": "o301", "name": "Crocosphaera"},  
                    ...  
                  ]  
  }  
},  
{ "sid": "s002",  
  "lat": "26.0",  
  "long": "-150.0",  
  "depth": "400",  
  "analysis":  
  { "technician": "Bob",  
    "organisms": [ {"oid": "o301", "name": "Crocosphaera"},  
                    {"oid": "o999", "name": "Prochlorococcus"},  
                    {"oid": "o777", "name": "Synechococcus"},  
                    ...  
                  ]  
  }  
},  
...  
]
```
Write tree SQL++ queries to convert the JSON data above into the relational schema.

(a) (5 points) Write the SQL++ query to construct the Sample table. Your query should return an output like:

```json
[ {
  "sid": "s001", "lat": "21.2", "long": "-157.9", "depth": "300", "technician": "Alice"},
  { "sid": "s002", "lat": "25.0", "long": "-150.0", "depth": "400", "technician": "Bob" },
...
]
```

Solution:
```
select x.sid, x.lat, x.long, x.depth, x.analysis.technician
from samples x;
```

(b) (5 points) Write the SQL++ query to construct the Analysis table. Your query should return an output like this:

```json
[ {
  "sid": "s001", "oid": "o252"},
  { "sid": "s001", "oid": "o301"},
...
  { "sid": "s002", "oid": "o301"},
  ...
]
```

Solution:
```
select x.sid, y.oid
from samples x, x.analysis.organisms y;
```

(c) (5 points) Write the SQL query to construct the Organisms table. Your query should return an output like this:

```json
[ { "oid": "o252", "name": "Trichodesmium"},
  { "oid": "o301", "name": "Crocosphaera"},
  { "oid": "o999", "name": "Prochlorococcus"},
  ...
]
```

Solution:
```
select distinct y.oid, y.name
from samples x, x.analysis.organisms y;
```
3  Datalog

3. (20 points)

At Gotham University\(^1\) each course has two prerequisites: for some courses both prerequisites are required, for other courses only one of the two prerequisites is required. The only exceptions are introductory courses, which don’t have prerequisites. Students can only take one course every quarter.

The schema is:

- `Course(cid, name, noQuarters)`
- `NoPrereq(cid)`
- `PrereqOneOfTwo(cid, cid1, cid2)`
- `PrereqTwoOfTwo(cid, cid1, cid2)`

(a) (5 points) The university requires that every course appear under either `NoPrereq` or under `PrereqOneOfTwo` or under `PrereqTwoOfTwo`. Write a datalog program that checks this constraint. Your program should return the `cid`'s and `name`'s of courses that do not occur in `NoPrereq` or `PrereqOneOfTwo` or `PrereqTwoOfTwo`. **Write your answer below:**

Solution:

```datalog
Listed(cid) :- NoPrereq(cid)
Listed(cid) :- PrereqOneOfTwo(cid, -, -)
Listed(cid) :- PrereqTwoOfTwo(cid, -, -)
Answer(cid, name :- Course(cid, name, -), not Listed(cid))
```

\(^1\)The *Wise Men of Gotham* are supposed to have feigned idiocy to avoid a Royal visit by King John. Gotham is also a fictional city in the comics of Batman.
(b) (15 points) Write a datalog program that computes, for each course, the smallest number of quarters need for a student to take sufficient prerequisites for that course, and the course itself. Your query should return pairs \( \text{oid}, \text{number-of-quarters} \).

For example, if the database is the following:

<table>
<thead>
<tr>
<th>cid</th>
<th>name</th>
<th>noQuarters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Math101</td>
<td>Math</td>
<td>3</td>
</tr>
<tr>
<td>Java102</td>
<td>Java</td>
<td>2</td>
</tr>
<tr>
<td>DB414</td>
<td>DB</td>
<td>3</td>
</tr>
<tr>
<td>ML446</td>
<td>ML</td>
<td>5</td>
</tr>
</tbody>
</table>

\[
\text{NoPrereq} \\
\text{PrereqOneOfTwo} \\
\text{PrereqTwoOfTwo}
\]

then your answer should be:

<table>
<thead>
<tr>
<th>cid</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>Math101</td>
<td>3</td>
</tr>
<tr>
<td>Java102</td>
<td>2</td>
</tr>
<tr>
<td>DB414</td>
<td>5</td>
</tr>
<tr>
<td>ML446</td>
<td>10</td>
</tr>
</tbody>
</table>

\text{Hint} if you were to compute the number of quarters that a student needs to take “Math101” and “Java102”, which don’t have any prerequisites, you could write:

\[
Q(c1+c2) :- \text{Course("Math101", -, c1), Course("Java102", -, c2)}
\]

Write your answer below:

\[
\begin{align*}
\text{Solution:} \\
Q(cid, q) &:- \text{NoPrereq(cid), Course(cid, -, q)} \\
Q(cid, q+q1) &:- \text{PrereqOneOfTwo(cid, cid1, cid2),} \\
&\quad Q(cid1,q1), \ Q(cid2,q2), \ q1<q2, \ \text{Course(cid, -, q)} \\
Q(cid, q+q2) &:- \text{PrereqOneOfTwo(cid, cid1, cid2),} \\
&\quad Q(cid1,q1), \ Q(cid2,q2), \ q1\geq q2, \ \text{Course(cid, -, q)} \\
Q(cid, q+q1+q2) &:- \text{PrereqTwoOfTwo(cid, cid1, cid2),} \\
&\quad Q(cid1,q1), \ Q(cid2,q2), \ \text{Course(cid, -, q)}
\end{align*}
\]
4 Query Execution and Optimization

4. (40 points)
(a) (10 points) Write a logical plan for the following query

with DepthCnt as
    (select x.depth, y.oid, count(*) as cnt
     from Sample x, Analysis y
     where x.sid = y.sid
     group by x.depth, y.oid)
select v.oid, v.name, max(u.cnt) as d
from DepthCnt u, Organism v
where u.oid = v.oid
group by v.oid, v.name;

You should turn in a relational algebra tree.
Write your answer below:

Solution:
(b) Indicate which of the optimization rules below are correct. All joins are natural joins:

i. (2 points)

Correct? [Yes/no]:

ii. (2 points)

Correct? [Yes/no]:

i. ___Yes___

ii. ___No___
iii. (2 points)

Correct? [Yes/no]:

iv. (2 points)

Correct? [Yes/no]:

Solution:
Note: The previous two questions were designed wrongly. The lower group-by on the right is applied to a key, so it is trivally a no-op. As stated, both equalities hold trivially, because the extra group by does nothing, so the answer is “YES” for both.

The correct design should have applied the new group on a join:

\[
\gamma_{z.oid, \max(d) \rightarrow m} (\gamma_{x.sid, \max(\text{depth}) \rightarrow d} ((\text{Sample } x) \bowtie (\text{Analysis } y)) \bowtie \text{Organism } z)
\]

\[
\gamma_{z.oid, \avg(d) \rightarrow a} (\gamma_{x.sid, \avg(\text{depth}) \rightarrow d} ((\text{Sample } x) \bowtie (\text{Analysis } y)) \bowtie \text{Organism } z)
\]

The first answer is “yes”, while the second is “no”, because the average of averages is not always the average. For example:

\[
\text{avg}(1, 1, 9, 11, 13) = 7 \neq \text{avg}(\text{avg}(1, 1), \text{avg}(9, 11, 13)) = \text{avg}(11) = 6
\]
v. (2 points)

\[
\begin{aligned}
&\text{Sample}(\text{id}, \text{lat}, \text{long}, \text{depth}, \text{technician}) \\
&\text{Analysis}(\text{sid}, \text{oid}) \\
&\text{Organism}(\text{oid}, \text{name})
\end{aligned}
\]

\[
\begin{aligned}
&\text{Sample}(\text{x}) \\
&\text{Analysis}(\text{y}) \\
&\text{Organism}(\text{z})
\end{aligned}
\]

\[
\begin{aligned}
&\sigma_{\text{depth} > 10000} \\
&\sigma_{\text{name} = 'X'}
\end{aligned}
\]

Correct? [Yes/no]: yes
(c) (10 points) Assume the following statistics on the database:

<table>
<thead>
<tr>
<th>Table</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample</td>
<td>100,000</td>
</tr>
<tr>
<td>Analysis</td>
<td>20,000,000</td>
</tr>
<tr>
<td>Organism</td>
<td>60,000</td>
</tr>
<tr>
<td>Sample, technician</td>
<td>100</td>
</tr>
<tr>
<td>Analysis, sid</td>
<td>10,000</td>
</tr>
<tr>
<td>Analysis, oid</td>
<td>45,000</td>
</tr>
<tr>
<td>Organism, name</td>
<td>15,000</td>
</tr>
<tr>
<td>Organism, name</td>
<td>15,000</td>
</tr>
</tbody>
</table>

Estimate the size of the answer to the following SQL query. You should make the usual uniformity, independence, and preservation of values assumption that we used in class:

```sql
select *
from Sample x, Analysis y, Organism z
where x.sid = y.sid and y.oid = z.oid
  and x.technician = 'Alice'
  and z.name = 'Synechococcus';
```

**Solution:**

\[
\frac{T(\text{Sample}) \cdot T(\text{Analysis}) \cdot T(\text{Organism})}{V(\text{Sample, sid}) \cdot V(\text{Organism, oid}) \cdot V(\text{Sample, technician}) \cdot V(\text{Organism, name})} = \frac{T(\text{Analysis})}{V(\text{Sample, technician}) \cdot V(\text{Organism, name})} = \frac{20,000,000}{100 \cdot 15,000} = 13.3
\]
(d) The depths of the samples are highly skewed: for each additional 100m, the number of samples is reduced by half. (That is, half of the samples have depth < 100m; half of the rest have depth < 200m; half of the rest have depth < 300m, etc.). There is an unclustered, B\(^+\) index on Sample\(\).\)\(\text{.depth}\). Indicate the optimal physical plan that the optimizer should choose for each of the two queries below. Choose between Table-scan and on-the-fly selection, or Index selection.

i. (5 points)

Q1: select * from Sample where depth < 100;

i. **Table scan**

Table scan or index selection?

ii. (5 points)

Q1: select * from Sample where depth > 8000;

ii. **Index Selection**

Table scan or index selection?
5 Parallel Query Processing

5. (20 points)

Consider the same relational database, and the same statistics:

\[
T(\text{Sample}) = 100,000 \quad T(\text{Analysis}) = 20,000,000
\]

We are storing and processing the data on \( P = 100 \) servers. The data is initially block partitioned, and we compute the following query:

```
select *
from Sample x, Analysis y
where x.sid = y.sid;
```

The query is computed distributively, on the 100 servers. You will be asked to estimate the number of answer tuples returned by each server. You only need to estimate the number of tuples in the final answer, not in any intermediate results. In case this estimate differs among servers, indicate the maximum number.

(a) **Plan 1:** \text{Sample} is broadcast to all 100 servers, then each server joins it with its local fragment of \text{Analysis}.

i. (5 points) Estimate the number of tuples/server assuming the data is uniform.

\textbf{Solution:} 200,000

ii. (5 points) Estimate the number of tuples/server assuming the data is skewed.

\textbf{Solution:} 200,000

(b) **Plan 2:** \text{Sample} and \text{Analysis} are hash-partitioned on the \text{sid} attribute on the 100 servers, then each server joins its local fragments.

i. (5 points) Estimate the number of tuples/server assuming the data is uniform.

\textbf{Solution:} 200,000

ii. (5 points) Estimate the number of tuples/server assuming the data is skewed.
Solution: 20,000,000
6 Conceptual Design

6. (40 points)

(a) (10 points) A large software company maintains a database of all its programmers. Some programmers are database administrators, and they have to report to some other programmer. The E/R diagram is given below. Write the CREATE TABLE statements for this E/R diagram, where name and rdbms are of type text, and pid is of type int. You answer should include keys and foreign keys where necessary.

Write your answer below:

Solution:

create table Programmer
    (pid int primary key,
     name text);
create table DBAdmin
    (pid int primary key references Programmer,
     rdbms text,
     reportsTo int references Programmer);
(b) (10 points) Consider a relation $R(A, B, C, D, E)$ satisfying the following FD’s:

\[
A \rightarrow B \quad C \rightarrow B \quad BD \rightarrow E
\]

Decompose $R$ into BCNF. In your final answer indicate the key of each relation.

**Solution:** Solution 1:
- in $R(ABCDE)$: $A+ = AB$ split into $R_1(AB), R_2(ACDE)$.
- in $R_2(ACDE)$: $CD+ = CDE$ split into $R_3(CDE), R_4(ACD)$.

Final answer: $R_1(AB), R_3(CDE), R_4(ACD)$.

Solution 2:
- in $R(ABCDE)$: $C+ = BC$ split into $R_1(BC), R_2(ACDE)$.
- in $R_2(ACDE)$: $CD+ = CDE$ split into $R_3(CDE), R_4(ACD)$.

Final answer: $R_1(BC), R_3(CDE), R_4(ACD)$.

Solution 3:
- in $R(ABCDE)$: $BD+ = BDE$ split into $R_1(BDE), R_2(ABCD)$.
- in $R_2(ABCD)$: $A+ = AB$ split into $R_3(AB), R_4(ACD)$.

Final answer: $R_1(BDE), R_3(AB), R_4(ACD)$.

Solution 4:
- in $R(ABCDE)$: $BD+ = BDE$ split into $R_1(BDE), R_2(ABCD)$.
- in $R_2(ABCD)$: $C+ = BC$ split into $R_3(BC), R_4(ACD)$.

Final answer: $R_1(BDE), R_3(BC), R_4(ACD)$. 
(c) Consider a relation $R(A, B, C, D, E, F, G)$ where $A$ is a key. We create a new table $S$ by running this query:

$$S: \text{select * from } R \text{ where } (B=C+2) \text{ and } (D+E=F) \text{ and } (G=7);$$

Indicate which of the following FDs are guaranteed to hold on $S$:

i. (2 points) $B \rightarrow C$
   
   Yes or no?  
   i. __Yes__

ii. (2 points) $C \rightarrow B$
   
   Yes or no?  
   ii. __Yes__

iii. (2 points) $D \rightarrow F$
   
   Yes or no?  
   iii. __No__

iv. (2 points) $F \rightarrow D$
   
   Yes or no?  
   iv. __No__

v. (2 points) $DE \rightarrow F$
   
   Yes or no?  
   v. __Yes__

vi. (2 points) $F \rightarrow DE$
   
   Yes or no?  
   vi. __No__

vii. (2 points) $F \rightarrow G$
   
   Yes or no?  
   vii. __Yes__

viii. (2 points) $G \rightarrow F$
   
   Yes or no?  
   viii. __No__

ix. (2 points) $CDE \rightarrow A$
   
   Yes or no?  
   ix. __No__

x. (2 points) $A \rightarrow CDE$
   
   Yes or no?  
   x. __Yes__
7 Transactions

7. (40 points)

(a) For each schedule below indicate whether it is conflict serializable and, if it is, indicate the equivalent serial schedule. Show your work by drawing the precedence graph.

i. (5 points)

\[ R_2(A), R_4(C), W_2(B), W_1(D), W_3(A), R_4(D), R_1(B), W_3(C) \]

**Solution:** Conflict serializable, in the unique order 2, 1, 4, 3.

ii. (5 points)

\[ R_2(A), R_1(D), W_4(B), R_3(C), W_3(D), R_2(B), W_4(C), W_1(A) \]

**Solution:** Not conflict serializable.

---

**Solution:** Both graphs:
Consider a database about the happiness status of Alice and Bob. Each morning Alice and Bob wake up happy or sad. Today their morning status is:

| A = happy | B = sad |

Every day at noon we run these two transactions, at about the same time:

T1: START
   READ(A,x) - read Alice
   WRITE(B,x) - update Bob
   COMMIT

T2: START
   READ(B,y)
   if y='sad' then WRITE(A,'sad')
   COMMIT

Indicate which of the following outcomes are possible if we run the transactions (1) under the REPEATABLE READ isolation level², or (2) under the READ UNCOMMITTED isolation level. Write yes or no in each entry below.

<table>
<thead>
<tr>
<th>Alice</th>
<th>Bob</th>
<th>REPEATABLE READS Possible?</th>
<th>READ UNCOMMITTED Possible?</th>
</tr>
</thead>
<tbody>
<tr>
<td>sad</td>
<td>sad</td>
<td>Solution: yes</td>
<td>Solution: yes</td>
</tr>
<tr>
<td>sad</td>
<td>happy</td>
<td>Solution: no</td>
<td>Solution: yes</td>
</tr>
<tr>
<td>happy</td>
<td>sad</td>
<td>Solution: no</td>
<td>Solution: no</td>
</tr>
<tr>
<td>happy</td>
<td>happy</td>
<td>Solution: yes</td>
<td>Solution: yes</td>
</tr>
<tr>
<td>DEADLOCK</td>
<td></td>
<td>Solution: yes</td>
<td>Solution: no</td>
</tr>
</tbody>
</table>

²This is the same as SERIALIZABLE on our static database.
(c) For each of the following statements indicate whether it is true or false:

i. (2 points) In a static database, every serializable schedule is conflict serializable.

   True or false?

   i. __False__

ii. (2 points) In a dynamic database, every serializable schedule is conflict serializable.

   True or false?

   ii. __False__

iii. (2 points) In a static database, every conflict serializable schedule is serializable.

   True or false?

   iii. __True__

iv. (2 points) In a dynamic database, every conflict serializable schedule is serializable.

   True or false?

   iv. __False__

v. (2 points) The two-phase locking protocol guarantees conflict serializability.

   True or false?

   v. __True__
vi. (2 points) The strict two-phase locking protocol guarantees conflict serializability.

True or false?  

vi. ____ True ____

vii. (2 points) Deadlocks can occur under the READ UNCOMMITTED isolation level.

True or false?  

vii. ____ True ____

viii. (2 points) Deadlocks can occur under the REPEATABLE READ isolation level.

True or false?  

viii. ____ True ____

ix. (2 points) Two transactions can hold the same SHARED LOCK at the the same time.

True or false?  

ix. ____ True ____

x. (2 points) Two transactions can hold the same EXCLUSIVE LOCK at the the same time.

True or false?  

x. ____ False ____