Homework 1 (by Deepa Parameswaran) - Solutions
Due: hard copy, in class at 1:30pm, on Tuesday, Feb. 3

VERY IMPORTANT: Deposit hard copy of your answers, in class. For ease of grading, please
1. Separate your answers, on different page(s) for each question (staple additional pages, if needed).
2. Type the full info on each page: your name, Andrew ID, course#, Homework#, Question# on each of the 4 pages.

Reminders:
• Plagiarism: Homework is to be completed individually.
• Typeset all of your answers whenever possible. Illegible handwriting may get zero points, at the discretion of the graders.
• Late homeworks: in that case, please email it
  – to all TAs
  – with the subject line exactly 15-415 Homework Submission (HW 1)
  – and the count of slip-days you are using.

For your information:
• Graded out of 100 points; 4 questions total
• Rough time estimate: ≈6 hours (1-2 hours for each question)

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity-Relationship Diagram</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>SQL Tables from the ER Model</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Relational Algebra for Job Portal</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Relational Calculus</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Total:</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>
Question 1: Entity-Relationship Diagram ............ [25 points]

GRADED BY: Deepa Parameswaran

On separate page, with ‘[course-id] [hw#] [question#] [andrew-id] [your-name]’

Consider a database to store information about a Research Organization. The database has the following properties:

- Every department has a title, and a unique department ID (departmentID).
- A department may have zero or more employees.
- Each employee belongs to exactly one department. We store the name of the employee and a unique employee ID for each employee (employeeID).
- Employees can be researchers or managers. For managers we store their annual bonus amount and for researchers their doctorate degree subject.
- People work in projects. Each Project has a unique project ID (projectID).
- Every project has exactly one manager and zero or more researchers.
- A manager can manage one or more projects but a researcher must work on exactly one project.

Given this description of the database and its constraints, we have created a mostly correct Entity-Relationship Diagram, shown in Figure [ ].

(a) [10 points] Find and correct any mistakes in the given ER diagram. Specifically, number and list them, like, e.g.
   1. delete: arrow, from x to y
   2. change to bold line: thin line, from z to w
   3. change to bold box: entity e

   Solution:
   1. Change the arrow Department to Employee, to a line from Department to Employee
   2. Make the line from Manager to Project bold
   3. Make the line from Project to Manager bold
   4. Change the line from Project to Manager to a bold arrow
   5. Make the line from Researcher to Project bold arrow
   6. Change the bold line from Project to Researcher to a line

(b) [5 points] There may also be some missing element(s). If none, say ’none’ - otherwise, add them to the picture, and list them, numbered. E.g.
   1. add: attribute a, to entity e
   2. add: bold line, arrow, from c to d.
   3. add: weak entity, f, with attributes . . .

   Solution:
   1. Underline attribute employeeID

Question 1 continues...
Figure 1: Almost correct ER diagram
(c) [10 points] List and number all the bold lines and all the arrows that are in the final, corrected version of the diagram. E.g.

1. **bold**, line, from Department to Employee
2. thin, **arrow**, from \(x\) to \(y\)

**Solution:**

1. Not Bold line Department to Employee
2. Bold with an arrow Employee to Department
3. Bold with an arrow from Researcher to Project
4. Not Bold line from Project to Researcher
5. Bold line Manager to Project
6. Bold with arrow Project to Manager

**Clarifications/Hints:**

- List your assumptions, if any. We will accept all reasonable assumptions.

**Grading info:**

-5 points if the student forgot to underline employeeID

---

**Question 1 continues...**
-0.5 points if the student mentioned the underline employeeID part in the first question
-4 points if the student does not mention all the lines and arrows in the final diagram
-0.5 point for extra changes to the ER diagram that are incorrect, but don’t take off points for that mistake more than once.
-0.5 in (a) if forgot to make line change the incorrect cardinalities
-3 for (b) if student knows that the attribute length is missing but adds it incorrectly
Consider a database for the New York art gallery. It records information about artists, paintings and exhibits. The constraints are exactly as shown in Figure 2. Paintings, Artists and Exhibitions have unique identifiers as shown in the Figure, with binary relationships among them as illustrated. To clarify:

- The line from “Exhibition” to “showcases”, is thick.
- The arrow from “Painting” to “paints”, is also thick.
- No other lines, boxes, or diamonds, are thick.

Figure 2: ER diagram for art exhibits: turn to SQL tables

(a) [20 points] Give the DDL statements, that correspond to the above ER diagram.
- Use proper data types (we’ll accept all reasonable choices).
- Avoid syntax errors (we’ll forgive missing semicolons).
• Specify your decisions with respect to **CASCADE** deletions. (E.g., 'I decided to reject deletions in Exhibition, when there are still participating artists')

• **Without** using CHECK statements, enforce as many as possible of the implied integrity constraints as you can.

**Solution:**

```sql
CREATE TABLE Exhibition (  
  exhibitionID INTEGER,  
  Exhibition_Date DATE,  
  Room_Number INTEGER,  
  PRIMARY KEY(exhibitionID) 
);
Grading info: It's OK to use other types e.g. integer for Exhibition_Date.
```

```sql
CREATE TABLE Painting (  
  paintingID INTEGER,  
  title CHAR(20),  
  medium CHAR(30),  
  Price INTEGER,  
  ArtistNumber INTEGER,  
  PRIMARY KEY(paintingID),  
  FOREIGN KEY(ArtistNumber) REFERENCES ARTIST ON DELETE CASCADE 
);
Grading info:  
- No penalty for ON DELETE NO ACTION  
- No penalty for having two tables for this, as long as that table does NOT take paintingID and artistID as composite primary key (which violates data integrity constraint)
```

```sql
CREATE TABLE ARTIST (  
  ArtistNumber INTEGER,  
  name CHAR(20),  
  SSN INTEGER UNIQUE NOT NULL,  
  PRIMARY KEY (ArtistNumber) 
);
Grading info: You should include SSN - either INTEGER, VARCHAR(9) or CHAR(9) or any other reasonable type, but no penalty whether UNIQUE, NOT NULL is present or not
```

```sql
CREATE TABLE SHOWCASES (  
)
```

Question 2 continues...
exhibitionID INTEGER,
paintingID INTEGER,
PRIMARY KEY (exhibitionID,paintingID),
FOREIGN KEY (exhibitionID) REFERENCES Exhibition
    ON DELETE CASCADE,
FOREIGN KEY (paintingID) REFERENCES Painting
    ON DELETE CASCADE
);

Grading info:

- No penalty for ON DELETE NO ACTION
- 5 point if you miss the 'SHOWCASES' table
- 0.5 point for every incorrect or missing attribute.
- 1 point for every incorrect table.
- 1 point if the answer does not say what constraints can't be enforced (-1 per part).  
- 2 points if primary key is missing or wrong.

(b) [5 points] Which of the implied IC (integrity constraints) of Figure 2 need CHECK statements to be enforced? List them all, or say none. For example, a (possibly, correct) answer could be:

- thin line, from “Painting” to “showcases”

Solution: thick line, “Exhibition” to “showcases”. This is the only one that we need to CHECK to see if there exist at least one painting associated with the given Exhibition.
Question 3: Relational Algebra for Job Portal ........ [25 points]

GRADED BY: Elomar de Souza

On separate page, with ‘[course-id] [hw#] [question#] [andrew-id] [your-name]’

Consider the relations of a Job Portal database as shown in Table 1. They describe a Job Portal, recording people, their skills and their endorsements (as in LinkedIn(TM)).

We have the following tables:

- **Member**: For each member we record the `userID` and `name`.
- **Skill**: For each Skill we record the `skillID` and the `skillName`.
- **Endorsement**: Each row shows which person has what skill, at what proficiency level `prof` (e.g., months of experience), and the count `n` of endorsements from other members.

For example, the first row of Table 1(a) means that user 'M103' (='John') masters skill 'S1' (='JAVA') with proficiency level '18', and he has received `n`=5 endorsements from other users.

```
<table>
<thead>
<tr>
<th>userID</th>
<th>name</th>
<th>skillID</th>
<th>skillName</th>
</tr>
</thead>
<tbody>
<tr>
<td>M101</td>
<td>Jack</td>
<td>S1</td>
<td>JAVA</td>
</tr>
<tr>
<td>M102</td>
<td>Jill</td>
<td>S2</td>
<td>.Net</td>
</tr>
<tr>
<td>M103</td>
<td>John</td>
<td>S3</td>
<td>Oracle</td>
</tr>
<tr>
<td>M104</td>
<td>Jane</td>
<td>S4</td>
<td>SAP</td>
</tr>
<tr>
<td>M105</td>
<td>Job</td>
<td>S5</td>
<td>R</td>
</tr>
</tbody>
</table>
```

(a) Member

```
<table>
<thead>
<tr>
<th>userID</th>
<th>skillID</th>
<th>prof</th>
<th>n</th>
</tr>
</thead>
<tbody>
<tr>
<td>M103</td>
<td>S1</td>
<td>18</td>
<td>5</td>
</tr>
<tr>
<td>M103</td>
<td>S2</td>
<td>46</td>
<td>5</td>
</tr>
<tr>
<td>M103</td>
<td>S4</td>
<td>17</td>
<td>5</td>
</tr>
<tr>
<td>M102</td>
<td>S4</td>
<td>10</td>
<td>4</td>
</tr>
<tr>
<td>M101</td>
<td>S4</td>
<td>15</td>
<td>1</td>
</tr>
<tr>
<td>M106</td>
<td>S4</td>
<td>21</td>
<td>1</td>
</tr>
<tr>
<td>M105</td>
<td>S4</td>
<td>31</td>
<td>3</td>
</tr>
<tr>
<td>M104</td>
<td>S5</td>
<td>45</td>
<td>4</td>
</tr>
</tbody>
</table>
```

(b) Skill

(c) Endorsement

Table 1: Relations of Job Portal database.

Given this database instance, answer the following questions:

(a) [2 points] Which of the following is the meaning of the expression \( \sigma_{n<5}(\text{Endorsement}) \)

1. It lists all the `n` values, that are less than 5, eliminating duplicates (i.e., \{1,3,4\} in our case).
2. It lists all Endorsement tuples ((`userID`, `skillID`, `prof`, and `n`) with less than 5 endorsements.
3. It lists the `n` value for each Endorsement tuple, and it rounds it down to 5, if higher than 5.
4. None of the above. The real answer is ...............
(b) [2 points] We want to list the mature skills, that is, the skillNames, for which there is at least one veteran (defined as prof > 36 months of experience). Which, if any, of the following expressions achieves that?

1. $\sigma_{\text{skillName}}(\pi_{\text{prof}>36}(\text{Skill} \bowtie \text{Endorsement}))$
2. $\pi_{\text{skillName}}(\sigma_{\text{prof}>36}(\text{Skill} \bowtie \text{Endorsement}))$
3. $\pi_{\text{skillName}}( (\sigma_{\text{prof}>36}(\text{Skill})) \bowtie \text{Endorsement})$
4. $\sigma_{\text{skillName}>36}(\pi_{\text{prof}}(\text{Skill} \bowtie \text{Endorsement}))$
5. None of the above. The real answer is .............

Solution: Answer #2

Grading info:
-1 for picking "None of the above" and writing a valid solution; -2 for picking the wrong answer

(c) For the following expression:

$$\sigma_{n<4}(\text{Member} \bowtie \text{Endorsement})$$

i. [0 points] Optional: describe in English what the expression does

Solution: List userID, name, skillName, proficiency and number of endorsements, of members who have received less than 4 endorsements

ii. [1 point] How many, and which are the columns (= attributes) in the answer?

Solution: 5 columns: userID, nameID, skillID, prof, and n.

iii. [3 points] How many tuples are in the answer?

Solution: 3

iv. [3 points] List all the tuples in the answer, as a table.

Solution:

<table>
<thead>
<tr>
<th>userID</th>
<th>name</th>
<th>skillName</th>
<th>prof</th>
<th>n</th>
</tr>
</thead>
<tbody>
<tr>
<td>M101</td>
<td>Jack</td>
<td>S4</td>
<td>15</td>
<td>1</td>
</tr>
<tr>
<td>M106</td>
<td>Jay</td>
<td>S4</td>
<td>21</td>
<td>1</td>
</tr>
<tr>
<td>M105</td>
<td>Job</td>
<td>S4</td>
<td>31</td>
<td>3</td>
</tr>
</tbody>
</table>

Grading info:
- ii. -1 for missing columns or having extra columns
- iii. -1 for being off by one on number of tuples; -3 for getting the wrong number of tuples by more than one
- iv. -1 for missing one tuple; -2 for performing wrong operation (e.g. $\times$ instead of $\bowtie$) or getting some columns and tuples wrong; -3 for all tuples and columns wrong

Question 3 continues...
(d) For the following expression:

\[ \pi_{\text{userID, skillID}}(\text{Endorsement}) \div \pi_{\text{skillID}}(\sigma_{\text{userID}=\text{M105}}(\text{Endorsement})) \]

i. [0 points] Optional: describe in English what the expression does

Solution: List userID of Members who possess all the skills that userID M105 has.

ii. [1 point] How many, and which are the columns (= attributes) in the answer?

Solution: One column: userID.

iii. [3 points] How many tuples are in the answer?

Solution: 5

iv. [3 points] List all the tuples in the answer, as a table.

Solution:

<table>
<thead>
<tr>
<th>userID</th>
</tr>
</thead>
<tbody>
<tr>
<td>M101</td>
</tr>
<tr>
<td>M102</td>
</tr>
<tr>
<td>M103</td>
</tr>
<tr>
<td>M105</td>
</tr>
<tr>
<td>M106</td>
</tr>
</tbody>
</table>

Grading info:

- ii. -1 for missing columns or having extra columns
- iii. -1 for being off by one on number of tuples; -3 for getting the wrong number of tuples by more than one
- iv. -1 for missing one tuple; -1 for missing column or having extra column; -2 for getting some columns and tuples wrong; -3 for all tuples and columns wrong

(e) For the following expression:

\[ \pi_{\text{E.userID, E1.userID}}(\rho_{E}(\text{Endorsement}) \bowtie_{\text{E.skillID}=\text{E1.skillID} \land \text{E.userID} > \text{E1.userID}} \rho_{E1}(\text{Endorsements})) \]

i. [0 points] Optional: describe in English what the expression does

Solution: Find pairs of userID, that may compete for the same job - ie., they share at least one skill. Again, no self- nor mirror-pairs

ii. [1 point] How many, and which are the columns (= attributes) in the answer?

Solution: 2 columns: E.userID, E1.userID.

iii. [3 points] How many tuples are in the answer?

Question 3 continues...
Solution: 10

iv. [3 points] List all the tuples in the answer, as a table.

Solution:

<table>
<thead>
<tr>
<th>E userID</th>
<th>E1 userId</th>
</tr>
</thead>
<tbody>
<tr>
<td>M106</td>
<td>M105</td>
</tr>
<tr>
<td>M106</td>
<td>M103</td>
</tr>
<tr>
<td>M106</td>
<td>M102</td>
</tr>
<tr>
<td>M106</td>
<td>M101</td>
</tr>
<tr>
<td>M105</td>
<td>M103</td>
</tr>
<tr>
<td>M105</td>
<td>M102</td>
</tr>
<tr>
<td>M103</td>
<td>M102</td>
</tr>
<tr>
<td>M103</td>
<td>M101</td>
</tr>
<tr>
<td>M102</td>
<td>M101</td>
</tr>
</tbody>
</table>

Grading info:

- ii. -1 for missing columns or having extra columns
- iii. -1 for being off by one on number of tuples; -3 for getting the wrong number of tuples by more than one
- iv. -1 for missing one tuple; -2 for getting some columns and tuples wrong; -3 for all tuples and columns wrong
Question 4: Relational Calculus .......................... [25 points]

GRADED BY: Hong Bin Shim

On separate page, with ‘[course-id] [hw#] [question#] [andrew-id] [your-name]’

We will again use the Job portal database from the last question (Table I). We start with questions on relational tuple calculus (RTC).

(a) For the following RTC expression

\[
\{ t \mid \exists e \in \text{Endorsement} \ (e.\text{skillID} = "S1" \land e.\text{userID} = t.\text{userID}) \}
\]

i. [0 points] Optional: describe in English what the expression does

**Solution:** List the members (userID) for people with skill "S1".

ii. [1 point] How many, and which are the columns (= attributes) in the answer?

**Solution:** There is one column: userID.

iii. [1 point] How many tuples are in the answer?

**Solution:** 1

iv. [2 points] List all the tuples in the answer, as a table.

**Solution:**

<table>
<thead>
<tr>
<th>userID</th>
</tr>
</thead>
<tbody>
<tr>
<td>M103</td>
</tr>
</tbody>
</table>

(b) For the following RTC expression

\[
\{ t \mid \exists e1 \in \text{Endorsement}, \exists e2 \in \text{Endorsement} \ \\
( e1.\text{skillID} = e2.\text{skillID} \\
\land e1.\text{userID} > e2.\text{userID} \\
\land t.\text{user1} = e1.\text{userID} \\
\land t.\text{user2} = e2.\text{userID}) \}
\]

i. [0 points] Optional: describe in English what the expression does

**Solution:** Find pairs of people that may compete, i.e., they share at least one skill - eliminate self-pairs and mirror-pairs

ii. [1 point] How many, and which are the columns (= attributes) in the answer?

**Solution:** 2 columns: user1, and user2. Equally correct: e1.\text{userID}, e2.\text{userID}.

iii. [1 point] How many tuples are in the answer?

Question 4 continues...
iv. [2 points] List all the tuples in the answer, as a table.

**Solution:**

<table>
<thead>
<tr>
<th>e1.userID</th>
<th>e2.userID</th>
</tr>
</thead>
<tbody>
<tr>
<td>M106</td>
<td>M105</td>
</tr>
<tr>
<td>M106</td>
<td>M103</td>
</tr>
<tr>
<td>M106</td>
<td>M102</td>
</tr>
<tr>
<td>M106</td>
<td>M101</td>
</tr>
<tr>
<td>M105</td>
<td>M103</td>
</tr>
<tr>
<td>M105</td>
<td>M102</td>
</tr>
<tr>
<td>M105</td>
<td>M101</td>
</tr>
<tr>
<td>M103</td>
<td>M102</td>
</tr>
<tr>
<td>M103</td>
<td>M101</td>
</tr>
<tr>
<td>M102</td>
<td>M101</td>
</tr>
</tbody>
</table>

(c) For the following RTC expression

\[
\{ t \mid \exists e \in \text{Endorsement}, \exists m \in \text{Member} \\
( e.userID = m.userID \\
\wedge t.name = m.name \\
\wedge e.skillID = "S1" \\
\wedge e.prof > 20 ) \} 
\]

i. [0 points] Optional: describe in English what the expression does

**Solution:** List user names with high proficiency in Java (="S1").

ii. [1 point] How many, and which are the columns (= attributes) in the answer?

**Solution:** One column name.

iii. [1 point] How many tuples are in the answer?

**Solution:** 0

iv. [2 points] Give, as a table, all of the tuples returned by the query.

**Solution:**

<table>
<thead>
<tr>
<th>name</th>
</tr>
</thead>
</table>

Question 4 continues...
The next questions are on relational domain calculus (RDC).

(d) For the following RDC expression
\{\langle u \rangle \mid \exists s, \exists p, \exists n (\langle u, s, p, n \rangle \in \text{Endorsement} \land n>4)\}

i. [0 points] Optional: describe in English what the expression does

**Solution:** List the userIDs that have more than 4 endorsements, in at least one skill of theirs

ii. [1 point] How many, and which are the columns (= attributes) in the answer?

**Solution:** One column: userID.

iii. [1 point] How many tuples are in the answer?

**Solution:** 1

iv. [2 points] List all the tuples in the answer, as a table.

**Solution:**

<table>
<thead>
<tr>
<th>userID</th>
</tr>
</thead>
<tbody>
<tr>
<td>M103</td>
</tr>
</tbody>
</table>

(e) For the following RDC expression:

\{\langle u_1, u_2 \rangle : \mid \exists s, \exists p_1, \exists n_1, \exists p_2, \exists n_2 (\langle u_1, s, p_1, n_1 \rangle \in \text{Endorsement} \\
\land \langle u_2, s, p_2, n_2 \rangle \in \text{Endorsement} \\
\land u_1>u_2)\}

i. [0 points] Optional: describe in English what the expression does

**Solution:** Find pairs of userIDs, that may compete for the same job - ie., they share at least one skill. Again, no self- nor mirror-pairs

ii. [1 point] How many, and which are the columns (= attributes) in the answer?

**Solution:** Two columns: u1.userID, u2.userID.

iii. [1 point] How many tuples are in the answer?

**Solution:** 10

iv. [2 points] List all the tuples in the answer, as a table.

**Solution:** The same as in the corresponding RTC question:

Question 4 continues...
(f) For the following RDC expression:

\[
\{\langle u_1 \rangle | \exists p_1, \exists n_1 (\langle u_1, "S4", p_1, n_1 \rangle \in \text{Endorsement} \\
\quad \land \forall u_2 (\exists p_2, \exists n_2 \\
\quad \quad ( (\langle u_2, "S4", p_2, n_2 \rangle \in \text{Endorsement}) \Rightarrow (n_1 \geq n_2)) ) \}\}
\]

i. [0 points] Optional: describe in English what the expression does

\textbf{Solution:} List the most endorsed member(s), wrt "S4"

ii. [1 point] How many, and which are the columns (= attributes) in the answer?

\textbf{Solution:} One column: userID.

iii. [2 points] How many tuples are in the answer?

\textbf{Solution:} 1

iv. [2 points] List all the tuples in the answer, as a table.

\textbf{Solution:}\n
<table>
<thead>
<tr>
<th>userID</th>
</tr>
</thead>
<tbody>
<tr>
<td>M103</td>
</tr>
</tbody>
</table>

End of Homework 1