Midterm Exam (Version B)
CS 122A
Spring 2017

Max. Points: 100
(Please read the instructions carefully)

Instructions:
- The total time for the exam is 80 minutes; be sure to *budget your time* accordingly.
- The exam is closed book and closed notes but “open cheat sheet”.
- Read each question first, in its entirety, and then carefully answer each part of the question.
- If you don’t understand something, ask one of the exam proctors for clarification.
- If you still find ambiguities in a question, note the interpretation you are taking.

<table>
<thead>
<tr>
<th>QUESTION</th>
<th>TOPIC</th>
<th>POINTS</th>
<th>SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>E-R to Relational Translation</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>Relational Queries</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>E-R Design</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>4</td>
<td>Modeling Terms</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Relational Operators</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>FD Design Theory</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>TOTAL</td>
<td>All</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>
Question 1: E-R to Relational Translation (20 points)

(20 pts) Translate the following E-R schema into an appropriate set of SQL tables. Avoid having more tables than needed, and be sure that your translated design – expressed as a series of CREATE TABLE statements in SQL – includes any/all appropriate (i) primary keys, (ii) unique keys, (iii) NOT NULL constraints, (iv) FOREIGN KEY constraints, and (v) ON DELETE options.

```
CREATE TABLE Building (
    bldgid INTEGER, manager VARCHAR(35),
    address VARCHAR(60),
    PRIMARY KEY (bldgid)
);
CREATE TABLE Office(
    bldgid INTEGER, no INTEGER,
    sqft INTEGER, rooms INTEGER,
    floor INTEGER,
    PRIMARY KEY (bldgid, no),
    FOREIGN KEY (bldgid) REFERENCES Building(bldgid) ON DELETE CASCADE
);
CREATE TABLE Renter(
    rid INTEGER, name VARCHAR(35),
    phone VARCHAR(20),
    PRIMARY KEY rid
);
CREATE TABLE Rent(
    rid INTEGER, bldgid INTEGER,
    no INTEGER, rate DECIMAL(6,2),
    PRIMARY KEY (rid, bldg, no),
    FOREIGN KEY (rid) REFERENCES Renter (rid) ON DELETE CASCADE,
    FOREIGN KEY (bldgid, no) REFERENCES Office (bldgid, no) ON DELETE CASCADE
);
```

SCORE: ____________
Question 2: Relational Queries (20 points)

Write each of the following queries in the indicated relational query language against the Hoofers Sailing Club schema that we’ve been using in the lectures. Be sure to read each problem carefully!

Sailors(sid, sname, rating, age) Reserve(sid, bid, day) Boats(bid, bname, color)

(5 pts) Using the relational algebra, print the ids of boats that have been reserved by one or more sailors named Harris but no sailors named John.

\[ \pi \text{bid} (\text{Reserves} \bowtie (\sigma \text{sname} = 'Harris'(\text{Sailors}))) - \pi \text{bid} (\text{Reserves} \bowtie (\sigma \text{sname} = 'John'(\text{Sailors}))) \]

(5 pts) Using English, state the query to which the following would be the relational calculus answer:

\[ \{ b | b \in \text{Boats} \land ( \forall s \in \text{Sailors} ( (s.\text{age} = 40) \Rightarrow ( \exists r \in \text{Reserves} ((r.\text{sid} = s.\text{sid}) \land (r.\text{bid} = b.\text{bid})))) \} \]

Find the boats that have been reserved by all of the sailors whose is age = 40.

(5 pts) Using the relational algebra, print the names and colors of boats that have been reserved by a sailor named Kim or by a sailor named Ghim:

\[ \pi \text{bname, color}((\text{Boats} \bowtie \text{Reserves}) \bowtie (\sigma \text{sname} = 'Kim' \lor \text{sname} = 'Ghim' (\text{Sailors}))) \]

(5 pts) Using the relational calculus, print the names and colors of boats that have been reserved both by a sailor named Alex and by a sailor named Alexander:

\[ \{ (\text{name, color}) | \exists b \in \text{Boats}(\text{bname}=\text{t.name} \land \text{color}=\text{t.color} \land \\
\exists r1 \in \text{Reserves}(r1.\text{bid}=b.\text{bid} \land \exists s1 \in \text{Sailors}(s1.\text{sid}=r1.\text{sid} \land \text{s.name}='\text{Alex}' )) \land \\
\exists r2 \in \text{Reserves}(r2.\text{bid}=b.\text{bid} \land \exists s2 \in \text{Sailors}(s2.\text{sid}=r1.\text{sid} \land \text{s.name}='\text{Alexander}' )) \} \]

SCORE: ____________
Question 3: E-R Design (25 points)

(25 pts) You have been hired by a new startup, PhotoTweets, to help them with the data management aspects of a new online service. They plan to serve home-made photos to mobile users from a cluster of Windows servers. Your job is to design an E-R conceptual schema for their database. Draw your E-R design below, starting from the partial picture that we’ve provided. Be sure to capture all relevant entities, relationships, and attributes. Clearly label each relationship to indicate its cardinality constraints (e.g., 1:1, 1:N, N:M), participation constraints (using double lines where appropriate), and roles (if needed). Mark the primary key for each entity set (by underlining it). Here’s what you have been told by their lead application developer:

- Each member has a unique member id, a name, a unique e-mail address, and a birthdate.
- Members can become photo sharing friends by “liking” one another. (Note: “Likes” may not be symmetric, i.e., one member might “like” another member but the opposite might not hold.)
- Members can choose to post photos, and their postdate should be recorded. Each photo will be posted by one member, and every photo must indeed have an associated member who posted it.
- Photos have a unique id, a GPS location data, the date that it was taken on, a thumbnail preview, a size, and a file name. Photos also have a set of tags (e.g., “scenery”, “people”, “animal”) intended to help members when searching for photos of potential interest.
- To avoid use of the service by social outcasts, every member must be liked by at least one other member.
- Some photos are ads. Ad photos also have a sponsoring company name and a product name.

SCORE: ____________
Question 4: Modeling Terms (5 points)

(5 pts) Refer again to the description of the PhotoTweets data in Question 3. Match each of the modeling constructs in the left column below with their best-matching feature (from Question 3) in the right column below. Indicate each answer by writing the letter of the relevant description feature to the left of each modeling construct. *(Hint: You should end up using each description feature once.)*

<table>
<thead>
<tr>
<th>Modeling construct</th>
<th>Description feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>e</td>
<td>a. Member.email</td>
</tr>
<tr>
<td>d</td>
<td>b. Post.postdate</td>
</tr>
<tr>
<td>b</td>
<td>c. Member.id</td>
</tr>
<tr>
<td>a or c</td>
<td>d. Photo.tags</td>
</tr>
<tr>
<td>c or a</td>
<td>e. (Member.email, Member.name)</td>
</tr>
</tbody>
</table>

Question 5: Relational Operators (10 points)

(1 pt each) Given two relations R and S, where R has N_R tuples, S has N_S tuples, and N_R > N_S > 0, what are the minimum and maximum possible result cardinalities for the following relational algebra queries expressed in terms of N_R and N_S?

<table>
<thead>
<tr>
<th>Query</th>
<th>min size</th>
<th>max size</th>
</tr>
</thead>
<tbody>
<tr>
<td>R U S:</td>
<td><strong><strong><strong>N_R</strong></strong></strong>_</td>
<td><strong><strong><strong>N_R+N_S</strong></strong></strong>_</td>
</tr>
<tr>
<td>σ(attr=val (R)):</td>
<td><strong><strong><strong>0</strong></strong></strong>_</td>
<td><strong><strong><strong>N_R</strong></strong></strong>_</td>
</tr>
<tr>
<td>R – S:</td>
<td>____<strong>N_R-N_S OR 1</strong></td>
<td><strong><strong><strong>N_R</strong></strong></strong>_</td>
</tr>
<tr>
<td>R ⨝ S:</td>
<td><strong><strong><strong>0</strong></strong></strong>_</td>
<td><strong><strong><strong>N_S*N_R</strong></strong></strong>_</td>
</tr>
<tr>
<td>π(attr1, attr2 (S):</td>
<td><strong><strong><strong>1</strong></strong></strong>_</td>
<td><strong><strong><strong>N_S</strong></strong></strong>_</td>
</tr>
</tbody>
</table>

SCORE: ____________
Question 6: FD Design Theory (20 points)

Consider a relational table with schema \( R(a, b, c, d, e) \). Each of the sub-problems below is based on a different list of FDs for \( R \). For each sub-problem, list the candidate keys for \( R \) based on its given FD list and indicate which is the highest normal form – 1NF, 2NF, 3NF, or BCNF – that \( R \) is currently in based on the given FD list. If asked (as in one case below), normalize the design into BCNF (else stop at 3NF) by decomposing \( R \) into several relations that have the lossless join and dependency-preserving properties. (Reminder: Be sure to use parentheses wherever you are indicating a composite key.) Last and also least, answer the 1-point question at the very end as well.

(5 pts) \( (a, b) \rightarrow c, \ d \rightarrow e \)
- Candidate key(s) for \( R \):
  \((a,b,d)\)
- Highest normal form:
  1NF

(5 pts) \( (a, b) \rightarrow e, \ (a, b) \rightarrow c, \ b \rightarrow d, \ d \rightarrow b, \ (a, d) \rightarrow c \)
- Candidate key(s) for \( R \):
  \((a,b),(a,d)\)
- Highest normal form:
  3NF

(7 pts) \( (c, d) \rightarrow e, \ a \rightarrow b, \ b \rightarrow d, \ b \rightarrow c, \) candidate key is \( a \).
- Highest normal form:
  2NF
- Normalized BCNF or 3NF design:
  \((a,b),(b,c,d),(c,d,e)\)

(2 pts) Suppose that \( R \) contains a tuple \( (95, “Li”, “Database Concepts”, “CS”, 2017-05-06) \). If \( c \rightarrow d \) is a functional dependency on \( R \), give an example of another tuple that, if inserted into \( R \), would violate this constraint.
(95,”Li”,”Database Concepts”, ”Business”, 2017-05-06)

(1 pt) Give the largest possible superkey for \( R \): \( R(a,b,c,d,e) \)

SCORE: ____________