Mid-Term Exam

⊕ Avoid asking questions - If something is unclear, document your assumptions and move on.

⊕ Do not forget to write your name on the first page. Initial each subsequent page.

⊕ Be neat and precise. We will not grade answers we cannot read.

⊕ If you have written something incorrect along with the correct answer, you should not expect to get all the points. I will grade based upon what you wrote, not what you meant.

⊕ You should draw simple figures if you think it will make your answers clearer.

⊕ Good luck and remember, brevity is the soul of wit.

<table>
<thead>
<tr>
<th>Num</th>
<th>Poss</th>
<th>Pts</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>
1. (15 points) You are a consultant, and your client wishes you to design a database for his pizza establishment. The database should contain customer data: name, a single phone number, possibly multiple credit cards, and all orders. Be sure to indicate keys, participation, cardinality, etc.

(a) (10 points) Draw an E-R diagram for the database. Document your assumptions.

**answer:**

```
CardID  Bank
         owns
         |
         |
         SSN  name  comments  addr
         has
         |
         |
         orderID  stuff
```

cards(CardID, Bank)
customers(SSN, name, comments, addr)
orders(orderID, stuff)
owns(cardID, SSN)
has(orderID, SSN)

goes to

cards(CardID, SSN, Bank)
customers(SSN, name, comments, addr)
orders(orderID, SSN, stuff)
(b) (5 points) Turn the E-R diagram into a relational schema. Combine and eliminate schemas where possible.
2. (30 points) Assume an SQL table of author-pairs, each pair signifying a paper written by two co-authors. We define an *Erdos number* as the **minimum** degrees of separation between any author and Erdos.

```sql
create table authors (A varchar(10), B varchar(10));
```

and the table is populated as follows:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>rance</td>
<td>erdos</td>
<td>“Weird Theory Stuff”</td>
</tr>
<tr>
<td>pete</td>
<td>rance</td>
<td>“Model Checking is Fun!”</td>
</tr>
<tr>
<td>gasarch</td>
<td>erdos</td>
<td>“The Cardinality of Esoterica”</td>
</tr>
<tr>
<td>erdos</td>
<td>biff</td>
<td>“The Math of the Cloud”</td>
</tr>
<tr>
<td>biff</td>
<td>hal</td>
<td>“Learning to Compute”</td>
</tr>
<tr>
<td>rance</td>
<td>biff</td>
<td>“Model Checking Blah Blah”</td>
</tr>
<tr>
<td>larry</td>
<td>john</td>
<td>“Vision Obscured”</td>
</tr>
<tr>
<td>hal</td>
<td>dave</td>
<td>“Linguistic Obscurita”</td>
</tr>
</tbody>
</table>

Erdos has an Erdos number of 0, “Gasarch” is a 1, and Pete is a 2.

(a) **(10 pts)** Write an SQL query returning a single column containing all authors with an Erdos number of 1.

```sql
answer:
(SELECT a FROM authors WHERE b=’erdos’)
UNION
(SELECT b FROM authors WHERE a=’erdos’);
```
(b) **(10 pts)** Re-write the previous query in relational algebra.

**answer:**
\[ \pi_B(\sigma_{A='erdos'}(authors)) \cup \pi_A(\sigma_{B='erdos'}(authors)) \]

(c) **(10 pts)** Write an SQL query returning a single column containing all authors with an erdos number of 2.
3. (40 pts) Consider the following relational schema and the listed functional dependencies.

\[ R(A, B, C, D, E) : F = A \rightarrow C, B \rightarrow CD, CD \rightarrow E, D \rightarrow C \]

(a) (5 points) The closure of \( B \) is:

answer: BCDE

(b) (5 points) List all candidate keys.

answer: AB

(c) (5 points) Show that \( C \) is/is-not extraneous in \( B \rightarrow CD \).

answer:

\[ F' = A \rightarrow C, B \rightarrow D, CD \rightarrow E, D \rightarrow C. \]

\((B)^+ \) under \( F' = B, BD, BDC \) (which includes C).
(d) (5 points) Show that $C$ is/is-not extraneous in $CD \rightarrow E$.
   answer:
   (D)+ under $F = D, DC, DCE$, which includes $E$ (beta).

(e) (5 points) Decompose $F$ into BCNF.
   answer:
   One approach is to use $A \rightarrow C$.
   This results in $(AC$, carries $A \rightarrow C$, key is $A$, in BCNF) and $(ABDE$, which carries no dependencies, key is $ABDE$, in BCNF).
(f) (5 points) Is your decomposition lossless? If not, what is lost?
   \textbf{answer}: All BCNF decompositions are lossless.

(g) (5 points) Does your decomposition preserve all dependencies? If not, what is lost?
   \textbf{answer}: We lose all but the first dependency.
(h) (10 points) Compute the canonical cover $F_c$.

**answer:** Taking out the extraneous attributes listed above, we are left with $A \to C, B \to D, D \to E, D \to C$. Combining, we get $A \to C, B \to D, D \to EC$

(i) (10 points) Decompose $F$ into 3NF. Is your decomposition lossless and does it preserve all dependencies? Why or why not?

**answer:** “AC, BD, CDE”, plus “AB”
4. (15 points) Use Armstrongs Axioms to prove the soundness of the pseudo-transitivity rule.

**answer:**
Given $A \rightarrow B, BC \rightarrow D$, then $AC \rightarrow D$

$A \rightarrow B$ (given)
$AC \rightarrow BC$ (augmentation)
$AC \rightarrow BC \rightarrow D$ (given, and transitivity)