This homework is to be done **individually**.
The assignment is OPTIONAL – if you don’t submit it, your homework grade will be calculated as the weighted average of homeworks 1 through 4. If you do want credit (worth 5% of the homework grade), then the assignment must be typed and submitted before the due date via the “submit” program:

1) Create a directory called hw5 in your home directory
2) Put your answers in "answers.txt" or "answers.pdf", inside this directory
3) Run "submit hw5"
4) Only .txt and .pdf files will be accepted.

Don’t submit if you think it will lower your overall homework grade!
Note, even if you don’t submit this for a grade, you are highly encouraged to do the problems before the final, as a study aid.

**Questions** [8 questions, 100 pts total]

1. [20pts] Write SQL statements to generate the relational tables required to represent the following ER diagram. You do not need to worry about normalization, but you do need to show all “not null”, primary key, and foreign key constraints:
2. [10 pts] Consider relation R = (A, B, C, D, E, F) and the set of functional dependencies:
   - AB → C
   - BC → A
   - BC → D
   - D → E
   - CF → B

   List the Candidate Keys of R (there are two).

3. [10 pts] Using Armstrong’s Axioms, show that the F.D.s in Question 2 imply that
   AB → E. For each step be sure to indicate which axioms and other F.D.s you are using.

4. [10 pts] Is relation R (from Question 2) in BCNF? If yes, say why. If not, then
decompose R into two or more relations that are all in BCNF. Your decomposition must
be lossless, but does not need to preserve dependencies.

5. [20 pts] Consider the following schedule:
   r1(A), w1(A), r1(B), r2(A), w3(B), r2(C), r4(B), r4(D), w4(D), r2(D)

   Is the schedule serializable? If so, show an equivalent serial transaction order. If not,
describe why not. (note: “r1(A)” means transaction 1 read item A).

For questions 6-8, consider the execution of the ARIES recovery algorithm given the
following log (assume a checkpoint is completed before LSN 0 and the Dirty Page Table
(DPT) and Transaction Tables for that checkpoint are empty):

<table>
<thead>
<tr>
<th>LSN</th>
<th>Log Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Update: T1 writes P1</td>
</tr>
<tr>
<td>20</td>
<td>Update: T2 writes P3</td>
</tr>
<tr>
<td>30</td>
<td>T1 commit</td>
</tr>
<tr>
<td>40</td>
<td>Update: T3 writes P4</td>
</tr>
<tr>
<td>50</td>
<td>Update: T2 writes P1</td>
</tr>
<tr>
<td>60</td>
<td>T1 end</td>
</tr>
<tr>
<td>70</td>
<td>Update: T3 writes P2</td>
</tr>
<tr>
<td>80</td>
<td>T2 abort</td>
</tr>
<tr>
<td>X</td>
<td>crash, restart</td>
</tr>
</tbody>
</table>

6. [10 pts] During Analysis: a) What log records are read? b) What are the contents of the
Dirty Page Table (DPT) and the transaction table at the end of the analysis stage?

operations are redone (assuming no updates made it out to disk before the crash)?

8. [10 pts] During Undo: a) What log records are read? b) What operations are undone? c) Show any new log records that are written (for CLRs, be sure to show the undoNextLSN)?