First Name: ______________________
Last Name: ______________________
MSU Net ID: _____________________

Exam (Version A) for CSE 480 (2019) KEY

Instructions:

• DO NOT START THE EXAM UNTIL TOLD TO DO SO

• You need to answer 6 of the 7 questions.

• On one of the questions make a large slash through it, which indicates that it should not be graded.

• Answer the questions in the spaces provided on the page. If you run out of room for an answer, continue on the back of that page.

• Show us the crossed out question and present your MSU ID (or other form of ID) when you hand in your exam.

![Diagram of a US cell phone number structure](http://xkcd.com/1129/)

Figure 1: http://xkcd.com/1129/
Question 1: Transaction Modes .................................................. 50 points

After the statement is completed by the associated connection, write which connections are **holding** each type of lock in the table. If no connection holds a type of lock, leave it blank. A connection can only hold **one lock** at a time. If an action can’t be granted the necessary locks, cross out that action and proceed as if that action wasn’t in the schedule (don’t perform a rollback). The first line indicates the prior state of the database (only Transactions B and C have begun).

<table>
<thead>
<tr>
<th>ID</th>
<th>SQL statement</th>
<th>Shared</th>
<th>Reserved</th>
<th>Exclusive</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prior State Of Database</td>
<td>B C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>INSERT INTO ...</td>
<td>C</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>COMMIT TRANSACTION;</td>
<td></td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>BEGIN TRANSACTION;</td>
<td></td>
<td></td>
<td>B</td>
</tr>
<tr>
<td>B</td>
<td>COMMIT TRANSACTION;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>UPDATE ...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>BEGIN IMMEDIATE TRANSACTION;</td>
<td></td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>INSERT INTO ...</td>
<td></td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>COMMIT TRANSACTION;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>SELECT ...</td>
<td></td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>INSERT INTO ...</td>
<td></td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>BEGIN IMMEDIATE TRANSACTION;</td>
<td></td>
<td>A G</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>CREATE TABLE ...</td>
<td></td>
<td>A</td>
<td>G</td>
</tr>
<tr>
<td>G</td>
<td>COMMIT TRANSACTION;</td>
<td></td>
<td>A</td>
<td>G</td>
</tr>
<tr>
<td>H</td>
<td>BEGIN TRANSACTION;</td>
<td></td>
<td>A G</td>
<td></td>
</tr>
</tbody>
</table>
Question 2: Closures .................................................. 50 points
Here’s a relation (R), its attributes and its functional dependencies (F):

\[ R(\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta) \]
\[ \gamma \epsilon \rightarrow \beta \zeta \gamma \]
\[ \zeta \gamma \rightarrow \alpha \]
\[ \eta \rightarrow \alpha \]

(a) (10 points) Which of the following are in the attribute set closure \( \{\gamma \beta \zeta\}^+ \)?

- \( \{\eta\} \)
- \( \{\zeta \gamma\} \)
- \( \{\alpha\} \)
- \( \{\beta \delta\} \)
- \( \{\delta\} \)
- \( \{\zeta \alpha\} \)

(b) (10 points) Which of the following are in the functional dependency closure of F (\( F^+ \))? 

- \( \sqrt{\gamma \epsilon \rightarrow \beta \zeta \gamma} \)
- \( \sqrt{\gamma \epsilon \delta \rightarrow \beta} \)
- \( \sqrt{\zeta \gamma \rightarrow \delta} \)
- \( \sqrt{\delta \rightarrow \delta} \)
- \( \sqrt{\gamma \epsilon \rightarrow \zeta} \)
- \( \sqrt{\zeta \gamma \rightarrow \alpha \gamma} \)

(c) (20 points) Which of the following are superkeys?

- \( \{\delta \epsilon \eta\} \)
- \( \{\gamma \delta \epsilon \eta\} \)
- \( \{\beta \gamma \delta \epsilon \eta\} \)
- \( \{\gamma \epsilon \zeta \eta\} \)
- \( \{\alpha \beta \gamma \delta \epsilon \zeta \eta\} \)

(d) (10 points) Which of the following are keys?

- \( \{\delta \epsilon \eta\} \)
- \( \{\gamma \delta \epsilon \eta\} \)
- \( \{\beta \gamma \delta \epsilon \eta\} \)
- \( \{\gamma \epsilon \zeta \eta\} \)
- \( \{\alpha \beta \gamma \delta \epsilon \zeta \eta\} \)
- \( \{\alpha \beta \gamma \epsilon \zeta \eta\} \)
Question 3: Functional and Multi-Valued Dependencies .......................... 50 points

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>0</td>
<td>255</td>
<td>0</td>
<td>$1</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>Gray</td>
<td>128</td>
<td>128</td>
<td>128</td>
<td>$3</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>Olive</td>
<td>128</td>
<td>128</td>
<td>0</td>
<td>$3</td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>Aqua</td>
<td>0</td>
<td>255</td>
<td>255</td>
<td>$4</td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>Magenta</td>
<td>255</td>
<td>0</td>
<td>255</td>
<td>$5</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>Magenta</td>
<td>255</td>
<td>0</td>
<td>255</td>
<td>$5</td>
<td>F</td>
<td></td>
</tr>
</tbody>
</table>

(a) (10 points) Which of the following columns are functionally determined by column E?

- A
- √ B
- √ C
- D
- E
- F

(b) (10 points) Which of the following functional dependencies hold?

- √ F → F
- √ A → B C D
- √ B → C
- E → B C
- C D → C
- E F → A B C D

(c) (10 points) Which of the following multi-valued dependencies hold?

- √ F →→ F
- √ A →→ B C D
- √ B →→ C
- E →→ B C
- C D →→ C
- E F →→ A B C D

(d) (10 points) Which of the following are keys?

- √ {E F}
- {A }
- {A E F}
- {A F}
- {A B C D E F}
- {B D F}

(e) (10 points) Which of the following normal forms is this relation?

- √ First Normal Form
- Boyce-Codd Normal Form
- Third Normal Form
- Forth Normal Form

Also all of the above is acceptable.
Rubric: Completely correct gets full credit. One wrong checkbox gets half credit (5 points).
Question 4: SQL statement ........................................... 50 points

I have been tasked with determining the average rating of supervisors in my organization. I have a table called *employees* with the following columns: name (the name of the employee), rating (the rating that employee gave for their supervisor), id (a unique id for each employee), supervisor (the id of that employee's supervisor). You need to write a query that returns the name and average rating of each supervisor (ordered from highest rated to least).

Here is some example data:

Table: *employees*

<table>
<thead>
<tr>
<th>name</th>
<th>rating</th>
<th>id</th>
<th>supervisor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Josh</td>
<td>55</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Abdol</td>
<td>NULL</td>
<td>2</td>
<td>NULL</td>
</tr>
<tr>
<td>John</td>
<td>30</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Jon</td>
<td>62</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Tian</td>
<td>40</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>Yash</td>
<td>54</td>
<td>6</td>
<td>6</td>
</tr>
</tbody>
</table>

And the desired results for this example data:

<table>
<thead>
<tr>
<th>name</th>
<th>average_rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Josh</td>
<td>62.0</td>
</tr>
<tr>
<td>Abdol</td>
<td>55.0</td>
</tr>
<tr>
<td>Yash</td>
<td>54.0</td>
</tr>
<tr>
<td>Jon</td>
<td>35.0</td>
</tr>
</tbody>
</table>

Fill in the blanks when the clause should be used in the SELECT statement:

```sql
SELECT _____ supervisor.name, avg(supervised.rating) _____
FROM _____ employees AS supervisor _____
INNER JOIN _____ employees AS supervised _____
ON _____ supervisor.id = supervised.supervisor _____
WHERE _____
GROUP BY _____ supervisor.id _____
HAVING _____
ORDER BY _____ avg(supervised.rating) DESC _____;
```
Question 5: Converting E/R Diagram ........................................... 50 points

Below is an E/R diagram representing a database of course offerings. Be sure to combine relations when possible.

(a) How many tables are needed to create relations?

i. (15 points) With the Object-Oriented method? 17

ii. (15 points) With the Nulls method? 2

iii. (15 points) With the Entity-Relation method? 6

(b) (5 points) If most of the courses offered aren’t any of the subclasses, which subclass method is the least space-efficient?

Nulls
Part a:
Correct answer gets full points. Many students also chose to merge Course and Offering (16, 1, 5) which is wrong, but we didn’t penalize if they were consistent.

Part b: Only correct answer gets points.
Question 6: Locks ................................................................. 50 points

(a) (10 points) There are three transactions \( (T_a; T_b; T_c) \) with \( T_a \) starting first, then \( T_b \), then \( T_c \). Draw a hypothetical Waits-For Graph that could occur under a Wait-Die deadlock resolution strategy, but not a Wound-Wait one.

**Solution:** Any edge pointing from a older to a younger transaction (i.e. \( T_a \rightarrow T_b \), but not edges going the other way.

(b) (10 points) With the same three transactions, draw a hypothetical Waits-For Graph that could occur under a Wound-Wait deadlock resolution strategy, but not a Wait-Die one.

**Solution:** Any edge pointing from a younger to a older transaction (i.e. \( T_b \rightarrow T_a \), but not edges going the other way.

(c) (10 points) With the same three transactions, draw a hypothetical Waits-For Graph that could occur not under a Wound-Wait deadlock resolution strategy, nor a Wait-Die one.

**Solution:** Including both edges from younger to older and older to younger. Would also accept impossible graphs (e.g. self-loops).

(d) (10 points) If \( T_b \) got rollbaked as the result of deadlock resolution, what would then be the youngest transaction?

\[ T_c \]

(e) (10 points) What does every Waits-For Graph have in common during the execution of an optimistic scheduler?

**Solution:** No edges!
Question 7: Conflict-Equivalence ........................................ 50 points

Give a conflict-equivalent serial schedule for the following 9 action schedule. If such a schedule doesn’t exist, prove the nonexistence of such a schedule with a Precedence Graph.

S: $w_1(L); r_2(J); r_1(M); w_4(J); w_3(K); r_3(J); r_2(M); w_4(L); w_2(K);

Show your work here:

**Solution:**

Conflicts:

- $w_1(L); w_4(L); \Rightarrow T_1 < T_4$
- $r_2(J); w_4(J); \Rightarrow T_2 < T_4$
- $w_4(J); r_3(J); \Rightarrow T_4 < T_3$
- $w_3(K); w_2(K); \Rightarrow T_3 < T_2$

Precedence Graph:

No possible serial schedule given the cycle: $T_2 \rightarrow T_4 \rightarrow T_3 \rightarrow T_2$

Serial schedule:

1. ____________  4. ____________  7. ____________
2. ____________  5. ____________  8. ____________
3. ____________  6. ____________  9. ____________
Rubric

- Precedence Graph correct: -0
- Precedence Graph correct but did not detect cycle: -25
- Precedence Graph missing 1 conflict: -35
- Precedence graph detected but wrong precedence: -10 points per every wrong precedence.
- Precedence graph detected but wrong arrows: -5