CS 564 Midterm Exam
Fall 2015
Answers and Grading Key

A: ER and Relational Models [10%]

For the following questions, just answer (by circling) either True or False.

1. **[2%]** A multi-way relationship in ER can always be converted to a set of binary relationships.

2. **[2%]** It is possible to translate both a weak entity set and an ISA hierarchy in the ER model to the relational model.

3. **[2%]** The ER model has no mechanisms to capture keys that are not primary keys.

4. **[2%]** Given any ER diagram, there is always only one correct way to translate it to the relational model.

5. **[2%]** There are mechanisms to enforce referential integrity in SQL that cannot be depicted in the ER model.

B: Normalization and Dependency Theory [22%]

Consider a relation R with $2d$ attributes (d is an integer ≥ 2) that are named $A_1, A_2, \ldots, A_{2d}$. There is a set F of $2d$ FDs on R: $A_i \rightarrow A_{1+(i+1) \ mod \ 2d}$, for $i = 1 \ldots 2d$. Here, *mod* is the modulo operator (remainder after integer division, e.g., 17 mod 5 = 2). For example, suppose d = 2, the attributes are A_1, A_2, A_3, A_4 and F = $\{A_1 \rightarrow A_3, A_3 \rightarrow A_1, A_2 \rightarrow A_4, A_4 \rightarrow A_2\}$. Answer the following questions.

1. **[6%]** Suppose $d = 2$. Obtain a lossless-join and dependency-preserving decomposition of R to BCNF using the algorithm given in class. Make sure you show the projections of F on to each relation in your decomposition.

   ANSWER:

2. **[18%]** For the following, $d > 2$. Choose (by circling) the right option (only 1 is correct per question), and explain the reasoning behind your answer. If your answer is incorrect or your explanation is invalid/incomplete, no points will be awarded.

   (a) **[7%]** What is the number of keys in R?

   (i) 1 (ii) d (iii) $d+2$ (iv) $2d$ (v) d^2

   (b) **[9%]** In any possible lossless-join and dependency-preserving decomposition of R to BCNF, what is the number of relations with no non-trivial FDs in the projection of F^+ on to it?

   (i) 0 (ii) 1 (iii) 2 (iv) 3 (v) d

   ANSWER: (ii) 1**.**

C: SQL, Relational Algebra & Calculus [52%]

Throughout this part, we will be using the following database schema:

Product (pid, name, brand, price, color)

Sales (pid, buyer, store, date)
Sales.pid is a foreign key referring to Product.pid.

1. **[15%]** Consider the following SQL query Q:

```
SELECT    brand
FROM      Product
WHERE     color = 'green'
GROUP BY  brand
HAVING    COUNT(*) < 3
```

   (a) **[8%]** Is the query Q equivalent to the query below? If yes, explain why. If not, provide a database instance that is a counterexample.

```
SELECT  DISTINCT brand
FROM    Product p
WHERE   (SELECT  COUNT(*)
         FROM    Product q
         WHERE   q.brand = p.brand
                 AND q.color = 'green') < 3
```

   (b) **[7%]** Is the query Q equivalent to the query below? If yes, explain why. If not, provide a database instance that is a counterexample.

```
SELECT  brand
FROM    Product p
WHERE   color = 'green'
        AND (SELECT  COUNT(*)
             FROM    Product q
             WHERE   q.brand = p.brand
                     AND q.color = 'green') < 3
```

2. **[15%]** Express the following SQL query in both Relational Algebra and Tuple Relational Calculus:

```
SELECT  DISTINCT name
FROM    Product p
WHERE   color = 'green'
        AND NOT EXISTS (SELECT  *
                        FROM    Product q
                        WHERE   q.brand = p.brand
                                AND q.price > 100)
```

3. **[12%]** Write one SQL query for each of the following:

sales (sum of the prices of all sold products).

SELECT d.date
FROM (SELECT s.date, SUM(p.price) AS totalprice
FROM Product p, Sales s
WHERE s.store = "Madison" AND p.pid = s.pid
GROUP BY s.date) AS d
WHERE d.totalprice = (SELECT MAX (d.totalprice)
FROM d);

4. **[10%]** For the following questions, just answer (by circling) either True or False.

(a) **[2%]** An `INSERT` statement in SQL can only add one tuple to the database.

(b) **[2%]** The following SQL query does not return any tuples where price is NULL.

```
SELECT * FROM Product WHERE price <> 100;
```

(c) **[2%]** Suppose that we define the schema of Sales in SQL as follows:

```
CREATE TABLE Sales (
   pid      INTEGER,
   buyer    CHAR(20),
   seller   CHAR(20),
   date     DATE,
   FOREIGN KEY (pid) REFERENCES Product(pid)
                     ON UPDATE CASCADE);
```

When we update the pid of a product in Sales, the database will update the pid in the Product table.

(d) **[2%]** The following two queries are equivalent:

```
SELECT    pid
FROM      Product
ORDER BY  price DESC
LIMIT     1;
```

```
SELECT    pid
FROM      Product
WHERE     price = (SELECT MAX (price) FROM Product);
```

(e) **[2%]** The following is not a valid SQL query:

```
SELECT    brand, MAX (price)
FROM      Product
GROUP BY  brand
HAVING    price > 100;
```

D: DISKS AND BUFFER MANAGEMENT **[16%]**

1. **[6%]** For the following questions, just answer (by circling) either True or False.

   (a) **[2%]** Most disks have multiple disk blocks within the same disk sector.

   (b) **[2%]** Sequential disk access is almost always slower than random disk access.

   (c) **[2%]** Given a disk block, either its "next" block or its "previous" block or both will always be on the same track as the given disk block.

2. **[10%]** In this question, you have to count the number of page I/Os (both reads and writes) for the given page access sequence and buffer replacement policy.

   There is only 1 query process. A page "Request" means the query process wants to read that page's values for computations. A page "Modify" means the query process is modifying the values on that page. A page "Release" means the query process has finished using that page and notifies the buffer manager accordingly.

   The number of page frames in the buffer pool is 3. Initially, all buffer frames are free and clean, and none of the pages to be accessed are in RAM. Note that a buffer frame can be considered for replacement if and only if its pin count is 0. Also note that the dirty bit is set to 1 upon a "Modify" on a page.

   Access Sequence: Request A, Release A, Request B, Request C, Modify B, Modify C, Release B, Request A, Release A, Request D, Request B

   What is the total number of page I/Os with LRU policy? Explain each page I/O in your answer.