CMPS 180, Final Exam, Fall 2017, Shel Finkelstein

Multiple Choice Questions (Part I)
Answered on Scantron Sheet

Test Form letter: A

This first Section (Part I) of the Winter 2017 CMPS 180 Final is multiple choice and is double-sided. Answer all multiple choice questions on your Scantron sheet. You do not have to hand in this first Section of the Exam, but you must hand in the Scantron sheet, with your Name, email and Student ID on that Scantron sheet. Please be sure to use a #2 pencil to mark your choices on this Section of the Final.

Name and Student ID must also be filled in by shading letters/numbers on the form. You must also mark the **version** ("Test Form letter **A**") of the Final that you took. The box for Test Form letter is at the top of the Scantron sheet, just to the left of the multiple choice questions.

The separate second Section (Parts II and III) of the Final is not multiple choice and is single-sided, so that you have extra space to write your answers. If you use that extra space, please be sure to write the number of the problem that you're solving next to your answer. Please write your name, email and student id on the second Section of the Exam, which you must hand in. You may use any writing implement on this Section of the Exam.

At the end of the Final, please be sure to hand in **both your Scantron sheet for this first Section of the Exam** and also the separate second Section of the Exam, and show your UCSC id when you hand them in.

Part I: (40 points, 2 points each)

Answer the questions in Part I on your Scantron sheets, which should have your name, email and UCSC id on them. Select the **best answer** for each of the following. For some questions, a choice is **"All of the Above"**, so read answer choices carefully.

Question 1: The relational model has physical independence. What does "physical independence" mean for the relational model?

a) When you write queries, you don't have to know how the tables used are stored, or what indexes exist.
b) If you change what's a table and what's a view, then your queries will still work.
c) You can port a Relational Database System on different hardware and operating systems and it will still run.
d) You can also write your queries using a Navigational Model, with links.
e) None of the Above.

Question 2: If an instance of relation R1(A,B,C) has 11 different tuples in it, and an instance of relation R2(B,C,D) has 7 different tuples in it, then how many tuples will there be in the result if the following SQL query is executed on those instances?

```
SELECT *
FROM R1, R2
WHERE R1.B = R2.B AND R1.C = R2.C;
```

a) 0
b) Exactly 18
c) Exactly 77
d) Between 0 and 18
e) Between 0 and 77

Question 3: We discussed ACID properties of transactions. What does Durability (the "D" in ACID) mean for transactions?

a) Transaction execution is as if they were executed one at a time.
b) Transactions happen completely or not-at-all.
c) If a transaction commits, its changes are permanent, even if there are failures, although they can be changed by later transactions.
d) Business rules are always maintained by the database system.
e) Uncommitted (dirty) values from one transaction are never read by any other transaction.

Question 4: Assume that R is a relation, *attribs* is some attributes of R, and *cond* is a condition on R. When will the following equation be true?

$$\sigma_{cond} (\pi_{attribs} (R)) = \pi_{attribs} (\sigma_{cond} (R))$$

a) It will always be true.
b) It will be true only when R is a binary relation, with two attributes.
c) It will be true only when there are two attributes in *attribs*.
d) It will be true only when the conditions in *cond* refer to just the attributes in *attribs*.
e) None of the Above.

Question 5: An instance of relation R(A,B) has m tuples in it, all exactly the same, and an instance of relation S(A,B) has n tuples in it, all of which are exactly the same as the tuples in R(A,B). If R and S are Union-Compatible, then how many tuples are there in the result of the following query?

```
( SELECT * FROM R )
    UNION ALL
( SELECT * FROM S );
```

a) 1
b) m + n
c) m * n
d) min(m, n)
e) max(m, n)

Question 6: Let R(A,B,C,D) be a relation, where (A, B) is the Primary Key, C can be NULL, and D is Unique and also can't be NULL.

Assume that A's domain has 3 different values, B's domain has 5 different values, C's domain has 9 different values, and D's domain has 12 different values. What is the maximum number of different tuples that can be in any instance of R?

a) 8
b) 9
c) 10
d) 12
e) 15

Question 7: Which statement is false for Relational Algebra Operations, where R is a relation and C1 and C2 are conditions on the attributes of R?

a) $\sigma_{C1}(\sigma_{C2} (R)) = \sigma_{C1 \text{ AND } C2}(R)$
b) Intersection is Commutative (for union-compatible relations): $R \cap S = S \cap R$
c) Union is Associative (for union-compatible relations): $(R \cup S) \cup T = R \cup (S \cup T)$
d) Minus is Associative (for union-compatible relations): $(R - S) - T = R - (S - T)$
e) Product distributes over union, assuming that S and T are union-compatible relations: $R \times (S \cup T) = (R \times S) \cup (R \times T)$

Question 8: In JDBC, what's the difference between a Statement and a PreparedStatement?

a) You supply the query or update when you create a PreparedStatement, but not when you create a Statement.
b) You supply the query or update when you create a Statement, but not when you create a PreparedStatement.
c) You use ExecuteUpdate on a PreparedStatement, and ExecuteQuery on a Statement.
d) You use ExecuteQuery on a PreparedStatement, and ExecuteUpdate on a Statement.
e) There's no such thing as a PreparedStatement in JDBC.

Question 9: What difference between FALSE and UNKNOWN in the three-valued logic used by SQL is the major justification for the need for UNKNOWN?

(The correct answer must both be true and be the major justification for the need for UNKNOWN.)

a) UNKNOWN is the same as NULL, but FALSE isn't.
b) When you execute a SQL query, if the result for some tuples is FALSE, those tuples don't contribute to the answer, but if the result is UNKNOWN, then they do contribute to the answer.
c) The negation of FALSE is TRUE, but the negation of UNKNOWN is UNKNOWN.
d) TRUE OR UNKNOWN is UNKNOWN, but TRUE or FALSE is TRUE.
e) FALSE AND UNKNOWN is FALSE, but FALSE OR UNKNOWN is UNKNOWN.

Question 10: Here's a query Q on the relation Employees(name, age, salary):

```
Q:      SELECT e1.name
        FROM Employees e1
        WHERE EXISTS
                ( SELECT e2.salary
                  FROM Employees e2
                  WHERE e1.salary > e2.salary AND e2.age >= 65 );
```

And here are 3 other queries, which are different only on the line shown in **bold**:

```
Q1:     SELECT e1.name
        FROM Employees e1
        WHERE e1.salary >
                ( SELECT e2.salary
                  FROM Employees e2
                  WHERE e2.age >= 65 );
```

```
Q2:     SELECT e1.name
        FROM Employees e1
        WHERE e1.salary > ANY
                ( SELECT e2.salary
                  FROM Employees e2
                  WHERE e2.age >= 65 );
```

```
Q3:     SELECT e1.name
        FROM Employees e1
        WHERE e1.salary > ALL
                ( SELECT e2.salary
                  FROM Employees e2
                  WHERE e2.age >= 65 );
```

Which query (or queries) are equivalent to Q?

a) Just Q1
b) Just Q2
c) Just Q3
d) Just Q1 and Q2
e) Just Q1 and Q3

Question 11: Students(student_id, name, address, age, major) is a table where student_id is the Primary Key. If (98765, 'Eliza Doolittle', 'Higgins Place', 18, 'ENGL') is a tuple in that table, and the following is executed, with no other work going on:

```
BEGIN TRANSACTION;

UPDATE Students
SET age = age + 1
WHERE name = 'Eliza Doolittle';

UPDATE Students
SET major = 'CMPS'
WHERE age = 19;

ROLLBACK TRANSACTION;
```

then afterwards, what will be in the tuple that has student_id 98765?

a) (98765, 'Eliza Doolittle', 'Higgins Place', 18, 'ENGL')
b) (98765, 'Eliza Doolittle', 'Higgins Place', 19, 'ENGL')
c) (98765, 'Eliza Doolittle', 'Higgins Place', 18, 'CMPS')
d) (98765, 'Eliza Doolittle', 'Higgins Place', 19, 'CMPS')
e) None of the Above

Question 12: Suppose that Customers(cust_id, name, spentvalue) is a table and GoodCustomers is a View defined on Customers that selects only the Customers whose spentvalue is more than 1000. What happens when the statement:

```
DROP VIEW GoodCustomers
```

is executed?

a) The view GoodCustomers goes away, but the Customers table is unchanged.
b) The view GoodCustomers goes away, and all the tuples in Customers whose spentvalue is more than 1000 are deleted.
c) Both the view GoodCustomers and the Customers table go away.
d) The statement does nothing if there are any tuples in the Customers table.
e) The statement does nothing if there are any tuples in the GoodCustomers view.

Question 13: Employees(name, age, salary) is a relation, and the salary for an employee named Smith is 5000. Two different transactions T1 and T2 are executed by different people at approximately the same time, both with Isolation Level READ UNCOMMITTED. T1 and T2 are the only transactions executing. T2 commits, but T1 rolls back. What is Smith's salary after that happens?

T1:

```
BEGIN TRANSACTION;
UPDATE Employees SET salary = salary + 1000 WHERE name='Smith';
ROLLBACK TRANSACTION;
```

T2:

```
BEGIN TRANSACTION;
UPDATE Employees SET salary = 2 * salary WHERE name='Smith';
COMMIT TRANSACTION;
```

a) Must be 10000
b) Must be 11000
c) Must be 12000
d) Must be either 10000 or 12000
e) Could be something other than 10000, 11000 or 12000

Question 14: Sells(bar, beer, price) and Beers(name, manufacturer) are tables in a database schema, with Primary Key underlined. You look at a million different instances of the database. Here are two statements:

i. Sells.beer must be a Foreign Key corresponding to Beers.name if (for all million instances) every beer in Sells.beer also appears in Beers.name
ii. Beers.name must be a Foreign Key corresponding to Sells.beer if (for all million instances) every beer in Beers.name also appears in Sells.beer

a) Both statements are True.
b) The first statement is True and the second statement is False.
c) The first statement is False and the second statement is True.
d) Both statements are False.
e) All of the Above.

Question 15: In OLAP, what is a dependent attribute?

a) A key in a Dimension table.
b) An attribute in a Fact table that is not in a key of one of the Dimension tables.
c) The result of doing Roll-Up on a table.
d) The result of doing Drill-Down on a table.
e) None of the Above.

Question 16: What is a difference between XML and the Relational Model?

a) XML documents are not in First Normal Form, because XML allows arbitrarily many appearances of elements in a document, but the Relational Model requires First Normal Form.
b) Relational tables are not in First Normal Form, because relational tables allow arbitrarily many appearances of elements in a document, but XML requires First Normal Form.
c) Relational requires that data be in Boyce-Codd Normal Form, but XML does not.
d) Relational requires that data be in Third Normal Form, but XML does not.
e) None of the Above.

Question 17: Why is a reason (or reasons) that having a relational schema in Boyce-Codd Normal Form (BCNF) may be better than not having a schema in BCNF?

a) BCNF schemas reduce redundancy.
b) BCNF schemas avoid the Update Anomaly.
c) BCNF schemas avoid the Insert Anomaly.
d) BCNF schemas avoid the Delete Anomaly.
e) All of the Above.

Question 18: What are Armstrong's Axioms?

a) Transformations which relational systems use to optimize a query.
b) The 5 relational operators that Ted Codd used to define Relational Algebra.
c) Rules for generating functional dependencies from other functional dependencies.
d) An algorithm to determine whether a decomposition is a Lossless Join decomposition.
e) None of the Above.

Question 19: Employees(name, age, salary) is a table in which name is the Primary Key. How could a tuple-based CHECK require that if someone's age is NULL then their salary must be less than 5000?

a) CHECK (age = NULL OR salary < 5000)
b) CHECK (age IS NULL OR salary < 5000)
c) CHECK (age <> NULL OR salary < 5000)
d) CHECK (age IS NOT NULL OR salary < 5000)
e) CHECK (age IS NOT NULL AND salary < 5000)

Question 20: In this question, Employees(name, age, salary) is a table in which name is the Primary Key and age has a default value of 21. What will happen when we execute the following statement, assuming that there was no tuple with name 'Shaw' before the statement was executed?

INSERT INTO EMPLOYEES(name) VALUES ('Shaw');

a) There will always be an error because values for age and salary weren't supplied.
b) There will always be an error because the value for salary wasn't supplied. Not supplying a value for age won't be a problem because it has a default value.
c) There might be an error because the value for salary wasn't supplied, but that will happen only if salary is not allowed to be NULL.
d) There will never be an error.
e) None of the Above.