CMPS 182, Final Exam, Spring 2017, Shel Finkelstein

Multiple Choice Questions (Part I)
Answered on Scantron Sheet

This first Section (Part I) of the Spring 2017 CMPS 182 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.

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: (44 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. Choices sometimes include **"All of the Above"** and **"None of the Above"**, so read answer choices carefully.

Question 1: We have a relation R(C1, C2, C3).

- C1's domain has 5 values; C1 can also be NULL. → 5 + NULL value = 6
- C2's domain has 2 values; C2 can't be NULL. → 2 = 2
- C3's has 9 values; C3 can also be NULL. → 9 + NULL value = 10

What the largest number of different tuples there can be in R?

a) 16
b) 18
c) 90
d) 120
e) 180

6 * 2 * 10 = 120

Question 2: If an instance of relation R(A,B) has 10 different tuples in it, and an instance of relation S(B,C,D) has 30 different tuples in it, then how many tuples are there in the result if the following SQL query is executed on those instances?

```
SELECT *
FROM R, S
WHERE R.B = S.B;
```

30 * 10 = 300

a) 0
b) Exactly 40
c) Exactly 300
d) Between 0 and 40
e) Between 0 and 300

Question 3: An instance of relation R(A, B) has m tuples in it; all of these tuples are exactly the same (duplicates). An instance of relation S(A, B) has n tuples in it, and these tuples are also all 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?

↳ both tables have same # of attributes, corresponding attributes have same datatype

m tuples UNION ALL n tuples

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

True, m+n

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

ATOMICITY: all-or-none proposition
CONSISTENCY: any changes to values in an instance are consistent w/ other values in the same instance
ISOLATION: Limits ways concurrent transactions affect each other
DURABILITY: ability of system to recover committed transactions despite system failure

Question 4: We discussed ACID properties of transactions. What does Durability (the "D" in ACID) refer to 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.
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 5: We have a relation Employees(name, age, salary) where name is the primary key. Here are two queries, q1 and q2.

```
q1:  SELECT COUNT(*) FROM Employees;
q2:  SELECT COUNT(DISTINCT age) FROM Employees;
```

The answers to these two queries are integers, which we'll call x1 and x2. Which statement is correct about the relationship between x1 and x2?

a) x1 > x2 is always true, and x1 can't equal x2.
b) x1 ≥ x2 is always true, and x1 can equal x2.
c) x1 < x2 is always true, and x1 can't equal x2.
d) x1 ≤ x2 is always true, and x1 can equal x2.
e) None of the Above.

Question 6: For the relation Employees(name, age, salary), which employee names will appear in the result to the following query?

```
SELECT e1.name
FROM Employees e1
WHERE e1.salary <= ANY
        ( SELECT e2.salary
          FROM Employees e2);
```

a) There will never be any employee names in the result.
b) The names of all of the employees will be in the result.
c) The result will be NULL.
d) The result will be UNKNOWN.
e) The query will cause a runtime error.

Question 7: The Students relation was created by the statement:

```
CREATE TABLE Students (
        studentID           INTEGER  PRIMARY KEY,
        name                VARCHAR(20),
        major               CHAR(4) DEFAULT ('CMPS'),
        age                 INTEGER,
        level               CHAR(10) NOT NULL
  );
```

Which statement about the attributes of Students is correct?

a) studentID can have the value 'UCSC6541'.
b) level can't be NULL, but all of the other attributes can be NULL.
c) An INSERT statement that doesn't supply a value for major will result in an error.
d) There can't be two different tuples in Students that have the same name.
e) None of the Above.

Question 8: What statement is true if you've created a view V that uses a table T?

a) You can execute SELECT statements on the view V, just as you can on a table.
b) If you DROP view V, table T is also dropped.
c) If you DROP view V, all data in V is deleted from table T.
d) Joining two views V1 and V2 is not permitted in SQL.
e) None of the Above.

Question 9: Employees(name, age, salary) is a relation. The salary for an employee named Lee whose age is 30 is 8000. What's is Lee's salary after the following is executed, assuming that no other transactions are being executed?

```
BEGIN TRANSACTION;
UPDATE Employees SET salary = 2 * salary WHERE age = 30;
UPDATE Employee SET salary = salary + 1000 WHERE salary > 9000;
ROLLBACK TRANSACTION;
```

a) 8000
b) 9000
c) 16000
d) 17000
e) None of the Above.

Question 10: Which statement (or statements) is true for the relational model?

a) Relational data references other data by matching values, not by pointers.
b) Structured Query Language (SQL) is the principle language used for the relational model.
c) Codd originated the relational model, defining relational algebra and relational calculus, which he proved are equivalent.
d) The relational model is largely declarative, with queries expressing what data you want to get, not how to get it.
e) All of the Above.

Question 11: What does "Physical Independence" mean for relational databases?

a) Queries can be asked on tables without knowing the details of how the data in those tables is stored.
b) Changing the datatypes of data in your tables (e.g., from CHAR(20) to INTEGER) won't affect your queries.
c) Updates always work on views, just as they do on tables.
d) No queries ever produce runtime errors.
e) All of the Above.

Question 12: For the relations Slopes(slopeid, slopename, color) and Activities(cid, slopeid, date), what does the following Relational Algebra query do?

projection selection cross product

$$\Pi_{\text{Slopes.slopename, Slopes.color}}(\sigma_{\text{Slopes.slopeid=Activities.slopeid}}(\text{Slopes} \times \text{Activities}))$$

a) Finds the slopename and color for slopes in Slopes that were used in at least one activity in Activities.
b) Finds the cid and slopename for every activity in Activities that occurred on a slope that's in Slopes.
c) Finds the slopename and color for every slope in Slopes.
d) For each slope in Slopes, finds the slopename and color, and counts the number of activities that occurred on that slope.
e) Find the slopeid for every activity that's in Activities.

Question 13: For a table Movies(title, year, length, studio), which index probably won't help execute the following query?

```
SELECT title, year
FROM Movies
WHERE year = 2000 AND studio = 'Paramount';
```

a) An index on title
b) An index on year → will help
c) An index on studio → will help
d) An index on (year, studio) in that order → will help
e) An index on (studio, year) in that order → will help

Question 14: Assume that the Employees Table has been created as follow:

```
CREATE TABLE Employees (
        name   CHAR(20) PRIMARY KEY,
        age    INTEGER NOT NULL DEFAULT 18,
        salary INTEGER NOT NULL,
        CHECK (age < 21 OR salary >= 8000)
 );
```

Which INSERT statement (or statements) will result in an error? Be sure to give the best answer.

a) INSERT INTO Employees(name, salary) VALUES ('Alpha', 9000);
b) INSERT INTO Employees(name) VALUES ('Beta'); → salary null
c) INSERT INTO Employees VALUES ('Gamma', 30, 6500); → age not < 21
d) Answers a), b) and c) will all result in errors.
e) Answers b) and c) will result in errors, but answer a) will not.

Question 15: Which of these is/are advantages of Stored Procedures?

a) Stored Procedures can improve performance by enabling operations to be performed on data without moving that data from the database to the client.
b) Stored Procedures support transactions, which can't be executed without Stored Procedures.
c) Stored Procedures allow you to use JDBC, which can't be used without Stored Procedures.
d) All of the Above.
e) None of the Above.

char float

Question 16: Sells(bar, beer, price) is a table, where bar and beer are CHAR(20) and price is FLOAT. Assuming that myCon is a connection, what error appears in the following JDBC, which is supposed to print the beers and prices for the Horseshoe Bar?

```
PreparedStatement stmt = myCon.prepareStatement(
   " SELECT beer, price FROM Sells WHERE bar = 'Horseshoe Bar' ");
ResultSet Menu = stmt.executeUpdate();
while (Menu.next()) {
   // For each value in result, get values of beer and price, and print them
   System.out.println(Menu.getString(1), Menu.getFloat(2));
}
```

a) Can't use *mycon.prepareStatement*
b) Need EXEC SQL before SQL statement
c) Should use *Menu.fetch*, not *Menu.next*
d) Should use *System.out.println(ResultSet.Float(1), ResultSet.getString(2));* not *System.out.println(Menu.getString(1), Menu.getFloat(2));*
e) Should use *stmt.executeQuery*, not *stmt.executeUpdate*

Question 17: Suppose that we have an instance of the relation Students(studentID, name, major, age, level), in which there are no students whose level is 'Senior', but there are students whose level isn't 'Senior'. For that instance, what will be the result of the following query?

```
SELECT s1.studentID
FROM Students s1
WHERE s1.age >= ALL (
    SELECT s2.age
    FROM Students s2
    WHERE s2.level = 'Senior'
);
```

empty

a) The result will be empty (no tuples).
b) The result will be one tuple with studentID NULL.
c) The result will be only the studentID values for Seniors.
d) The result will be the studentID values for every student in Students.
e) None of the Above.

Question 18: Which attributes and aggregates can appear in the HAVING clause of a SQL SELECT statement?

a) Any attributes of the tables in the FROM clause, as well as any aggregates of those attributes.
b) Only the attributes that appear in the WHERE clause of the query, as well as aggregates of any attributes in the FROM clause tables.
c) Only the attributes that appear in the GROUP BY clause of the query, as well as aggregates of any attributes in the FROM clause tables.
d) Only the attributes that appear in the GROUP BY clause of the query; aggregates cannot appear in the HAVING clause.
e) Only the attributes that appear in the GROUP BY clause of the query, but the only aggregates permitted in the HAVING clause are the aggregates that appear in the SELECT clause.

Question 19: For the following addressbook DTD:

```
<!DOCTYPE addressbook [
  <!ELEMENT addressbook (person*)>
  <!ELEMENT person
     (name, address, homephone?, ( workphone | mobile )+, email*)>
  <!ELEMENT name          (#PCDATA)>
  <!ELEMENT address       (#PCDATA)>
  <!ELEMENT homephone     (#PCDATA)>
  <!ELEMENT workphone     (#PCDATA)>
  <!ELEMENT workphone     (#PCDATA)>
  <!ELEMENT email         (#PCDATA)>
]>
```

Does the following data correspond to that DTD?

```
<addressbook>
    <person>
        <name> Thomas Edison </name>
        <address> Menlo Park, NJ 08820 </address>
        <workphone> (732) 555 1234 </workphone>
        <workphone> (848) 555 3557 </workphone>
    </person>
</addressbook>
```

a) Yes, the data corresponds to the DTD.
b) No, because there's no homephone.
c) No, because there's no mobile.
d) No, because there are 2 workphones.
e) No, because there's no email.

Question 20: The relation R(A, B, C, D) has Functional Dependencies A → B, B → C, and BC → D. Which statement is true? Keys: A , A^+ = ABCD

a) ABCD is a key, because the attribute closure $(ABCD)^+$ is all the attributes of R.
b) AC is a key for R, because the attribute closure $(AC)^+$ is all the attributes of R.
c) AC is a superkey for R, because the attribute closure $(AC)^+$ is all the attributes of R, but AC is not a key for R.
d) BC is a key for R, because the attribute closure $(BC)^+$ is all the attributes of R.
e) BC is a superkey for R, because the attribute closure $(BC)^+$ is all the attributes of R, but BC is not a key for R.

Question 21: If salary has the value NULL, which statement is correct?

a) The condition "NOT (salary=1000)" has the value TRUE.
b) The condition "NOT (salary=1000)" has the value FALSE.
c) The condition "salary=1000 OR NOT (salary=1000)" has the value TRUE
d) The condition "salary=1000 AND NOT (salary=1000)" has the value FALSE.
e) None of the Above.

Question 22: Suppose that R(A, B, C) is a relation, with Functional Dependencies A→B and B→C. If r is an instance of R, and (4, 5, 6) and (7, 8, 9) are tuples in r. Then:

a) (4, 8, 9) can't be a tuple in r.
b) (2, 5, 9) can't be a tuple in r.
c) (7, 5, 1) can't be a tuple in r.
d) (3, 8, 5) can't be a tuple in r.
e) All of the Above.