CS145 Final Examination

Friday, December 10, 1999, 8:30–11:30 AM

Directions

The exam is *open book/notes*; any written materials may be used.

For each of the 33 questions, circle the letter (a), (b), (c), or (d) of your chosen answer. Do not circle more than one answer. If you wish to change your answer, please indicate clearly what your "final answer" is.

Score = 3 times number right minus the number wrong, so random guessing nets you nothing on the average. One point is awarded for putting your name on the paper, so the maximum score is 100.

If you wish to explain or demonstrate your solution to a problem for partial credit, you may use page bottoms or the backs of the pages (but warn us on the front). Please use this option sparingly, e.g., if you think the question is flawed or open to multiple interpretations, because we shall only be awarding partial credit in rare situations.

You have approximately 5.4 minutes per question. Use your time wisely, and do not spend too much time on any one question.

Do not forget to **sign the pledge** below.

I acknowledge and accept the honor code.

____

Print your name here (**1 pt.**): ____

In each of the first 11 questions, you are asked to compare two queries Q_1 and Q_2. You must tell whether the queries are:

1. The same, meaning that for every database the answers to the two queries are the same *as bags*. That is, the same tuples are produced by each query, and a tuple is produced the same number of times by each query. The order in which tuples are produced is not to be considered.

2. Completely different, meaning that there are databases where Q_1 produces more of some particular tuple, and other databases where Q_2 produces more of some particular tuple. Note that the query producing the smaller number of copies of a tuple may produce zero copies of that tuple.

3. One is contained in the other. for instance, Q_1 is contained in Q_2 if on every database, Q_2 produces at least as many copies of each tuple as Q_1 does. Note that it is possible Q_2 produces one or more copies of a tuple while Q_1 produces none of that tuple.

General advice:

- Do not assume a query has an error and therefore produces nothing.
- Relations mentioned in the queries may have attributes not mentioned, but their existence should not affect the answer.
- Relations may have `NULL`'s.
- All queries should be assumed to be in SQL2 unless stated otherwise.

Question 1:

Q_1:

```
SELECT DISTINCT a
FROM R
WHERE b > 10;
```

Q_2:

```
SELECT a
FROM R
WHERE b > 10
GROUP BY a;
```

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

In the following two questions, assume the schemas of the relations are $R(a, b)$ and $S(b, c)$.

Question 2:

Q_1:

```
SELECT a
FROM R, S
WHERE R.b = S.b;
```

Q_2:

```
SELECT a
FROM R
WHERE b IN (
    SELECT b
    FROM S
);
```

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

Question 3:

Q_1:

```
SELECT a
FROM R NATURAL JOIN S
GROUP BY a
HAVING COUNT(*) < 2;
```

Q_2:

```
SELECT a
FROM R
WHERE b NOT IN (
    SELECT s1.b
    FROM S s1, S s2
    WHERE s1.b = s2.b AND s1.c <> s2.c
);
```

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

Question 4: Note the schema assumption of the previous two questions no longer applies.

Q_1:
```
SELECT a
FROM R
WHERE b >= ANY(SELECT d FROM S WHERE c>10);
```

Q_2:
```
SELECT a
FROM R
WHERE b >= ALL(SELECT d FROM S WHERE c>10);
```

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

Question 5:

Q_1:
```
(R NATURAL FULL OUTER JOIN S)
    UNION
(R NATURAL JOIN S);
```

Q_2:
```
(R NATURAL LEFT OUTER JOIN S)
    UNION
(R NATURAL RIGHT OUTER JOIN S);
```

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

The next two questions are based on the following ODL declarations:

```
interface R (extent Rext key name) {
    attribute string name;
    relationship Set<S> mySs inverse S::myRs;
}

interface S (extent Sext key name) {
    attribute string name;
    relationship Set<R> myRs inverse R::mySs;
}
```

Question 6:

Q_1:

```
SELECT ss.name
FROM Rext rr, rr.mySs ss
WHERE rr.name = "Sally";
```

Q_2:

```
SELECT ss.name
FROM Sext ss
WHERE EXISTS rr IN ss.myRs : rr.name = "Sally";
```

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

Question 7:

Q_1:

```
SELECT rr.name
FROM Rext rr, rr.mySs ss
WHERE rr.name = ss.name
```

Q_2:

```
SELECT rr.name
FROM Rext rr, Sext ss
WHERE rr.name = ss.name
```

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

Question 8: In the following question, assume R and S are *bags*, and the operators are bag operators.

$$Q_1: \ (R \cup S) - (R \cap S)$$

$$Q_2: \ (S - R) \cup (R - S)$$

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

Question 9: For the following relational algebra expressions, the relation schemas are $R(a, b)$ and $S(b, c)$.

$$Q_1: \ R \bowtie S$$

$$Q_2: \ \sigma_{R.b=S.b}(R \times S)$$

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

Question 10: The following queries are recursive Datalog, and as for all Datalog queries, the result is a set, not a bag.

Q_1:
```
Path(x,y) <- Arc(x,y)
Path(x,y) <- Path(x,a) AND Path(a,b) AND Path(b,y)
```

Q_2:
```
Path(x,y) <- Arc(x,y)
Path(x,y) <- Path(x,z) AND Path(z,y)
```

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

Question 11: In the following, the results of Q_1 and Q_2 should be taken to be the result of the final `SELECT * FROM R`. Assume that the schema of relation R is $R(a, b)$.

Q_1:

```
DELETE FROM R WHERE a=10;
INSERT INTO R VALUES(10,5);
SELECT * FROM R;
```

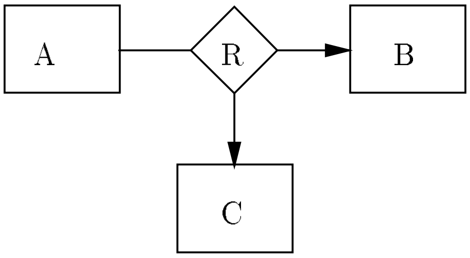
Q_2:

```
UPDATE R SET b=5 WHERE a=10;
SELECT * FROM R;
```

(a) Q_1 and Q_2 produce the same answer.
(b) The answer to Q_1 is always contained in the answer to Q_2.
(c) The answer to Q_2 is always contained in the answer to Q_1.
(d) Q_1 and Q_2 produce different answers.

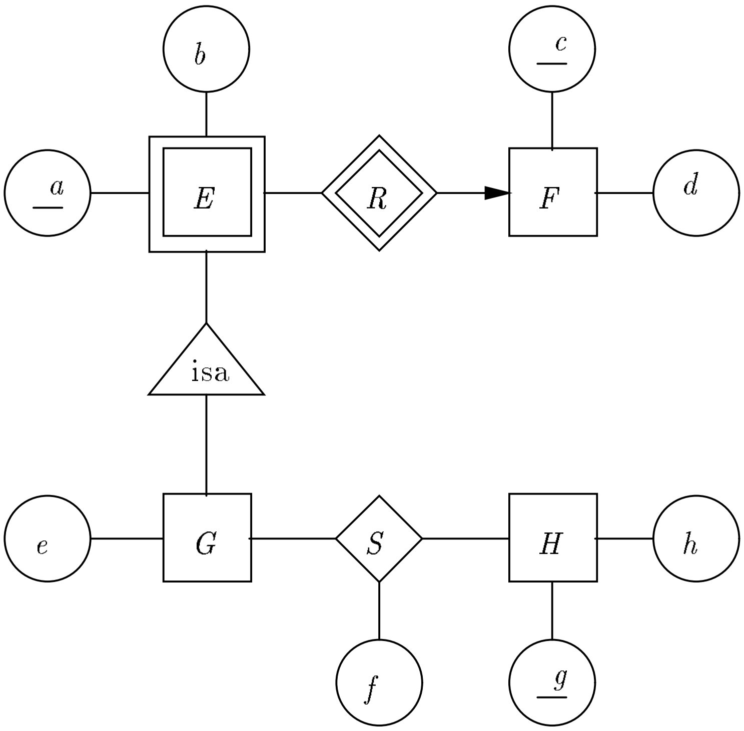
The remaining 22 questions do not have any specific form for the choices.

Question 12: Consider the following E/R diagram:



If A has 100 entities, B has 1000 entities, and C has 10 entities, what is the maximum number of triples of entities that could be in the relationship set for R?
(a) 100 (b) 1000 (c) 100,000 (d) 1,000,000

The following two questions are based on the E/R diagram below:



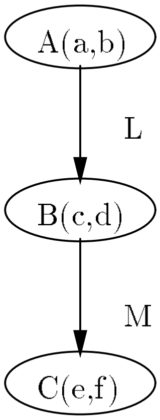
Question 13: If we translate this diagram to relations in the normal way for E/R diagrams described in the book and class notes, which relation schema would *not* be in the resulting database schema?

(a) $E(a, c, b)$ (b) $S(a, c, f, g)$ (c) $F(c, d)$ (d) $R(a, c)$

Question 14: If we instead used the "object-oriented" approach to translating this E/R diagram to relations, how many relation schemas would be different (when compared to the E/R-to-relations translation), either in their schema, or their set of tuples, or both?

(a) 1 (b) 2 (c) 3 (d) 4

Question 15: Below is a network diagram with three LRT's, $A(a,b)$, $B(c,d)$, and $C(e,f)$, plus the links L and M, as shown.



Imagine a relation $R(a,b,c,d,e,f)$ that contains a tuple (a,b,c,d,e,f) whenever there is a record $A(a,b)$ that is linked by L to a record $B(c,d)$, and the record $B(c,d)$ is linked by M to record $C(e,f)$. Suppose that a, c, and e are keys for the LRT's A, B, and C, respectively. Then the key(s) for R is/are:

(a) Only a (b) Only ace (c) Only e (d) Any of a, ac, or ace could be a key, depending on L and M.

Question 16: Given a relation $R(A,B,C,D,E)$ and functional dependencies $AB \rightarrow CE$, $B \rightarrow D$, and $D \rightarrow E$, which of the following FD's can *not* be inferred?

(a) $AD \rightarrow CE$ (b) $BC \rightarrow D$ (c) $AB \rightarrow A$ (d) $B \rightarrow E$

Question 17: Given a relation $R(A,B,C,D)$ with functional dependencies $A \rightarrow BC$, which of the following multivalued dependencies can *not* be inferred?

(a) $A \twoheadrightarrow BC$ (b) $A \twoheadrightarrow B$ (c) $A \twoheadrightarrow BD$ (d) None of the above; i.e., all are necessarily true.

Question 18: Suppose we have a relation $R(A,B,C,D,E)$ and the FD's $A \rightarrow DE$, $D \rightarrow B$, and $E \rightarrow C$. If we project R (and therefore its FD's) onto schema ABC, what is true about the key(s) for ABC?

(a) Only ABC is a key (b) Only A is a key (c) Only DE is a key (d) A, B, and C are each keys.

Question 19: Suppose functional dependency $B \to C$ holds in relation $R(A, B, C, D)$. Which additional FD will make R be in 3NF, but not BCNF?

(a) $D \to AB$ (b) $AC \to D$ (c) $CD \to B$ (d) $AD \to B$

Question 20: Consider relation $R(A, B, C, D)$ with FD's $A \to D$, $B \to D$, and $D \to BC$. Which of the following is true about the decomposition of R into relations with schemas AB and BCD?

(a) The decomposition is neither lossless nor dependency-preserving.
(b) The decomposition is lossless, but not dependency-preserving.
(c) The decomposition is dependency-preserving, but not lossless.
(d) The decomposition is both lossless and dependency-preserving.

Question 21: Which (if any) of the following SQL2 statements are legal? *Hint*: focus only on the use of `NULL`'s, not on some minor detail such as a missing semicolon should there be one. You may assume relation R has the schema $R(a, b)$.

(a) `SELECT a, NULL FROM R WHERE ISNULL(b);`
(b) `SELECT a, b FROM R WHERE b LIKE '%NULL%';`
(c) `SELECT a, b FROM R WHERE b=NULL;`
(d) None of the above; i.e., all have errors.

Question 22: Suppose $R(a, b)$ and $S(b, c)$ are relations. The Datalog program below:

```
P(x,y) <- R(x,z) AND S(z,y)
Q(x,y) <- P(x,z) AND P(z,y)
```

produces as its answer a relation for Q. Which of the expressions of relational algebra below *fails* to produce the same relation?

(a) $\pi_{a,f}\Big(\sigma_{c=d}\big((R \times \rho_{T(d,e)}(R)) \bowtie (S \times \rho_{U(e,f)}(S))\big)\Big)$
(b) $\pi_{a,e}\Big(R \bowtie S \bowtie \big(\rho_{T(c,d,e)}(R \bowtie S)\big)\Big)$
(c) $\pi_{a,c}\big(R \bowtie \rho_{T(b,d)}(S) \bowtie \rho_{U(a,e)}(R) \bowtie \rho_{V(e,c)}(S)\big)$
(d) None of the above; i.e., all three expressions produce the same value as Q.

The following two questions refer to this ODL definition of a class of "team" objects in which there are players and coaches.

```
interface Team (key name) {
    attribute string name;
    attribute Set<string> players;
    attribute Set<string> coaches;
}
```

We represent the class `Team` in the relational model by a relation

```
Teams(teamName, playerName, coachName)
```

If (n, P, C) represents a `Team` object, then relation teams has every tuple (n, p, c) such that p is in the set of players P, and c is in the set of coaches C.

Question 23: A key for the relation `Teams` is:
(a) `teamName`
(b) `teamName, playerName`
(c) `teamName, coachName`
(d) `teamName, playerName, coachName`

Question 24: From the information given above, a multivalued dependency that must hold in `Teams` is:
(a) $playerName \twoheadrightarrow coachName$
(b) $teamName \twoheadrightarrow coachName$
(c) $playerName \twoheadrightarrow teamName$
(d) $coachName \twoheadrightarrow teamName$

Question 25: Which statement about the relation $R(a, b)$ and the following SQL query:

```
SELECT a, MAX(b), MIN(b)
FROM R
GROUP BY a;
```

is true?
(a) This query cannot be written in relational algebra.
(b) This query can be written in relational algebra if and only if a is a key for R.
(c) This query can be written in relational algebra if and only if the functional dependency $b \rightarrow a$ holds for R.
(d) This query can be written in relational algebra.

The following two questions refer to this database schema. Remember that the language is SQL2, not Oracle SQL.

```
CREATE TABLE Stores(
    storeID INT CHECK(storeID IN (SELECT store FROM Sales)),
    location CHAR(30),
    PRIMARY KEY (storeID)
);
CREATE TABLE Items(
    name CHAR(30) PRIMARY KEY,
    price REAL CHECK(price >= 0.0)
);
CREATE TABLE Sales(
    store INT,
    item CHAR(30) REFERENCES Items(name),
    totalSales REAL,
    PRIMARY KEY (store, item)
);
CREATE ASSERTION BigSales CHECK(NOT EXISTS(
    SELECT * FROM Stores, Sales WHERE Stores.location = 'CA' AND
        Stores.storeID = Sales.store AND Sales.totalSales <= 1000000)
);
```

Question 26: Which of the following restrictions is enforced by the schema?

(a) No item has a price of 0.

(b) Every `storeID` that appears in the `Stores` relation must also appear as a `store` value in the relation `sales`.

(c) Every store has total sales of at least $1,000,000.

(d) None of the above; i.e., each could be false in some database that satisfies the constraints.

Question 27: Which of the following restrictions is enforced by the schema?

(a) Every `name` appearing in relation `Items` also appears in the `item` column of `Sales`.

(b) Every `item` appearing in `Sales` also appears in the `name` column of `Items`.

(c) No two tuples of `Sales` can agree on either `store` or `item`.

(d) None of the above; i.e., each could be false in some database that satisfies the constraints.

Question 28: Suppose we have the following table declarations:

```
CREATE TABLE A(w INT PRIMARY KEY);
CREATE TABLE B(x INT PRIMARY KEY
    REFERENCES A(w) ON DELETE SET NULL;
CREATE TABLE C(y INT REFERENCES A(w));
CREATE TABLE D(z1 INT REFERENCES B(x) ON DELETE SET NULL,
    z2 INT REFERENCES A(w) ON UPDATE CASCADE);
```

Consider the following scripts:

```
I.   DELETE FROM C; DELETE FROM B; DELETE FROM A; DELETE FROM D;
II.  DELETE FROM C; DELETE FROM D; DELETE FROM A; DELETE FROM B;
III. DELETE FROM B; DELETE FROM C; DELETE FROM D; DELETE FROM A;
```

Which of the above scripts will empty all four tables, without error?
(a) III only (b) I only (c) II and III only (d) I and III only

---

Question 29: For this question we have 2 relations: $R(a, b, c)$ and $S(d, e, f)$. Suppose we declare the following view:

```
CREATE VIEW theView AS
SELECT a, 'cs145' AS class, f
FROM R, S
WHERE R.b = S.e;
```

and then issue the following query Q:

```
SELECT a
FROM theView
WHERE theView.f > 10;
```

Which of the following queries produces the same answer as Q?
(a) `SELECT a FROM R,S WHERE R.class = 'cs145' AND S.f > 10;`
(b) `SELECT a FROM R NATURAL JOIN S WHERE S.f > 10;`
(c) `SELECT a FROM R, S WHERE R.b = S.e AND S.f > 10;`
(d) `SELECT a FROM theView,S WHERE S.f > 10;`

Question 30: The next question assumes that there is a relation $R(x, y)$ and a relation $S(u, v)$. The types of all four attributes x, y, u, and v may be assumed to be integers, so there are no type errors in what follows. We execute the following PL/SQL statement:

```
DECLARE
    a NUMBER;
    b NUMBER;
    CURSOR c IS
        SELECT r1.x, r2.x
        FROM R r1, R r2
        WHERE r1.y = r2.y;
BEGIN
    DELETE FROM S;
    OPEN c;
    LOOP
        FETCH c INTO a, b;
        EXIT WHEN c%NOTFOUND;
        IF a < b THEN
            INSERT INTO S VALUES(a, b);
        END IF;
    END LOOP;
    CLOSE c;
END;
.
run
```

Which of the following Datalog rules has the same effect as the above PL/SQL statement?

(a) `S(x,y) <- R(x,z) AND R(y,z) AND x<y`
(b) `S(a,b) <- R(x,y) AND R(x,z) and x<y`
(c) `S(a,b) <- R(a,b) AND R(c,d) AND a<c and b=d`
(d) `S(a,b) <- R(x,y) AND R(u,v) AND y=v`

Question 31: Consider the following Datalog rules, where f and g are EDB predicates:

```
a(X) <- b(X) AND NOT c(X)
b(X) <- d(X) AND NOT e(X)
c(X) <- b(X) AND NOT d(X)
d(X) <- e(X) AND NOT f(X)
e(X) <- f(X) AND NOT g(X)
```

The number of different strata (*not* the maximum stratum number) for the IDB predicates of this program is:

(a) 2 (b) 3 (c) 4 (d) Infinity

Question 32: Suppose we make the following declarations to the Oracle 8.0 DBMS that we have been using in class:

```
CREATE TYPE T AS OBJECT (a INT, b INT);
/
CREATE TABLE R (c INT, d T);
```

If R contains a single tuple, which of the following queries will *not* cause an error?

(a) `SELECT xx.d.a FROM R xx;`
(b) `SELECT xx.a FROM THE(SELECT * FROM R.d) xx;`
(c) `SELECT d.a FROM R;`
(d) `SELECT a FROM THE(SELECT d FROM R);`

Question 33: A is a unary (one-column) relation declared by

```
CREATE TABLE A(i INT);
```

Below is an *SQL3* trigger (it is not an Oracle trigger, and so there is nothing wrong with it modifying the same relation A on which the trigger is set).

```
CREATE TRIGGER Mystery
AFTER INSERT OR UPDATE ON A
REFERENCING OLD AS OldTuple, NEW AS NewTuple
WHEN (10 > (SELECT MAX(i) FROM A))
    UPDATE A SET i = i+1
FOR EACH ROW;
```

With A empty, we execute the statement:

```
INSERT INTO A VALUES(1);
```

The result of this statement is:

(a) A is left with a single tuple with value 10.
(b) A is left with a single tuple with value 2.
(c) A is left with tuples $2, 3, \ldots, 10$.
(d) The trigger never stops modifying tuples.
