CS345 Final Examination

Friday December 13, 1996, 12:15–3:15 pm

Directions

The exam is *open book*; any written materials may be used. Answer all 11 questions on the exam paper itself. The total number of points is 200. You may make additional comments at the bottoms of pages or at the end if you feel the need to clarify or justify your answers. Do not forget to **sign the pledge** below.

I acknowledge and accept the honor code.

____

Print your name here: ____

1	2	3	4
5	6	7	8
9	10	11	

Problem 1: (20 points) Here are two conjunctive queries with negation:

Q_1: `q(X) :- n(X) & n(Y) & a(X,Y) & NOT a(Y,X)`
Q_2: `q(X) :- n(X) & n(Y) & a(X,X) & NOT a(Y,Y)`

a) Is $Q_1 \subseteq Q_2$? Verify or give a counterexample.

b) Is $Q_2 \subseteq Q_1$? Verify or give a counterexample.

Problem 2: (20 points) Suppose we have global "concepts":

$likes(D, W)$ = "drinker D likes wine variety W."

$freq(D, B)$ = "drinker D frequents bar B."

$serves(B, W)$ = "bar B serves wine W."

Let there also be the following view definitions:

```
v1(D,B) :- freq(D,B)
v2(D,B) :- likes(D,W) & serves(B,W)
v3(B,W) :- serves(B,W) & likes(D,W) & freq(D,B)
```

a) Suppose we are given the query

Q: `q(D,W) :- freq(D,B) & serves(B,W)`

Find all the minimal solutions that can be built from the three views above and are contained in the query Q.

b) Show that each of your solutions in (a) is contained in Q by expanding them and then describing containment mappings. You may abbreviate $freq$, $likes$, and $serves$ by f, l, and s, respectively.

c) Are any of your solutions equivalent to Q? Justify your answer.

Problem 3: (25 points) Test the queries below for containment $Q_2 \subseteq Q_1$:

```
Q1: p(X) :- a(X,Z) & a(Z,Y) & X != Z
Q2: p(X) :- a(X,Z) & a(Z,Y) & Y != Z
```

Note they differ only in the last subgoals: $X \neq Z$ for Q_1 and $Y \neq Z$ for Q_2.

a) Give equivalent "rectified" rules; i.e., rewrite them so no variable appears more than once in an ordinary (not arithmetic-comparison) subgoal.

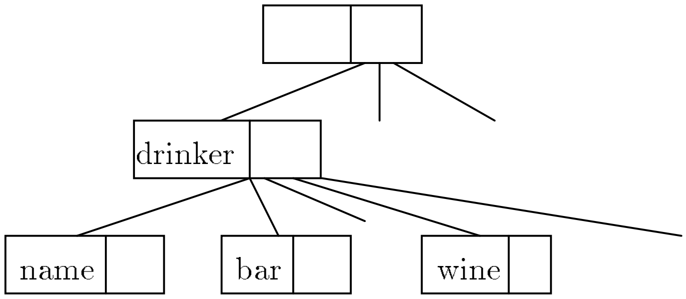
b) The Gupta-Zhang-Ozsoyoglu test requires that we find all possible containment mappings from Q_1 to Q_2 (ignoring the arithmetic subgoals). What are these mappings?

c) What logical statement must be verified in the G-Z-O test to check $Q_2 \subseteq Q_1$?

d) Is your condition from (c) true or not? Justify briefly.

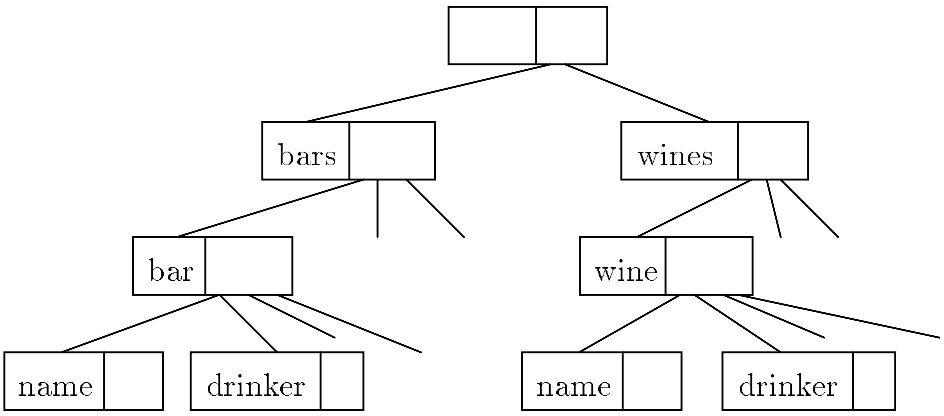
e) We can also test containment by the Klug approach, where we consider all partitions of the variables of Q_2 and then all orders for the blocks. In this case, since only $\neq$ is involved, it is not even necessary to consider orders of blocks. For *each* possible partition, indicate what conclusion can be drawn about whether $Q_2 \subseteq Q_1$.

Problem 4: (20 points) Suppose we have a source `s1` that has information about drinkers, the bars they frequent, and the wines they like. This source may be thought of as structured as an OEM object, consisting of a root object, some number of subobjects labeled *drinker*, and for each of the latter objects, a subobject labeled *name*, some number of subobjects labeled *bar* (whose value is the name of a bar), and some number of subobjects labeled *wine* (whose value is the name of a wine. You may assume *name* is a key for drinkers, for bars, and for wines. This structure is suggested below:



a) We wish to create a mediator `m1` that will export an OEM object looking like that of `s1` and pictured above. More precisely, we wish `m1` to export those parts of the object at `s1` that match the description and picture above, since we can never know what other subobjects might be present at `s1`. Write an MSL program that defines `m1`.

b) We wish to create another mediator `m2` that exports a representation of the information of `s1` in a different form, suggested by:



That is, the root of `m2` has two subobjects labeled *bars* and *wines*. A *bar* object has a subobject labeled *name* (the name of the bar) and some number of subobjects labeled *drinker* (the names of the drinkers who frequent that bar). Similarly, a *wine* object has a subobject labeled *name* (of the wine) and some number of subobjects labeled *drinker* (the names of the drinkers who like that wine). Write an MSL program to define `m2`.

Problem 5: (15 points) We defined the weak-instance window function as a series of operations on particular instances of the relations involved (i.e., particular databases). However, in simple cases, we can figure out in advance the result of these operations. Suppose that the relations are AB and BC. For each of the following dependencies, give an expression of relational algebra to compute $[AC]$ (according to the weak-instance definition).

a) The only dependency is $B \to\to A \mid C$.

b) The only dependency is $B \to C$.

Problem 6: (15 points) Find all the minimal stable models for the following rules:

```
p :- NOT q
q :- NOT p
r :- p & NOT s
s :- p & NOT r
```

Problem 7: (15 points) The database schema consisting of the relations ABC, BCD, and ACD is cyclic. Give an example of a particular database that is locally consistent but not globally consistent.

Problem 8: (15 points) The recursive Datalog program P is defined by the rules:

r_1: `p(X,Y,Z) :- a(X,Y) & a(Y,Z)`
r_2: `p(X,Y,Z) :- p(X,W,Y) & p(W,Y,Z)`
r_3: `p(X,Y,Z) :- p(X,Y,W) & p(Y,W,Z)`

Let the conjunctive query Q be

Q: `p(X,Y,Z) :- a(X,Z) & a(Z,Y) & a(Y,Z)`

Show that $Q \subseteq P$ by the method of canonical databases.

a) Give the canonical database D.

b) Complete the test by showing how to derive the head of Q by applying P to your database D from part (a). Indicate the rule used at each step.

Problem 9: (15 points) For the rules

r_1: `p(X,Y,Z) :- a(X,Y) & a(Y,Z)`
r_2: `p(X,Y,Z) :- p(X,W,Y) & p(W,Y,Z)`

Draw the rule/goal graph if the initial goal node (query) is p^{bff}.

Problem 10: (20 points) Apply the simplified magic-sets transformation (no 0th supplementary predicates, and no supplementary predicates before EDB subgoals) to the Datalog program

```
p(X,Y) :- a(X,Y)
p(X,Y) :- p(X,W) & b(W,Z) & p(Z,Y)
```

if the query is $p(0, A)$ (i.e., p^{bf}). Note that in this case, bf is the only binding pattern for p, so the rules do not need to be split or rectified.

Problem 11: (20 points) Suppose that we have the following global concepts.

1. Predicates $course(C)$, $student(S)$, and $grade(G)$, representing the classes course, student, and grade, respectively.
2. Predicate $taking(C, S)$, which means that student S is taking course C.
3. Predicate $gradeOf(S, C, G)$, meaning that student S has grade G in course C.

Let Q be the query

Q: `ans(S,G) :- course(C) & student(S) & student(sally) & grade(G) &`
`taking(C,sally) & taking(C,S) & gradeOf(S,C,G)`

i.e., find the grades of the students who are taking a course that Sally is also taking.

a) Suppose that each of these global concepts is supported by a database consisting of the relation $csg(C, S, G)$, meaning that student S is taking course C and getting grade G. Write suitable definitions for each of the global concepts in terms of the csg relation. In this and following parts, you may abbreviate the predicates $course$, $student$, $grade$, $taking$, and $gradeOf$ as c, s, g, t, and go, respectively.

b) Expand query Q so it is written in terms of the relation csg.

c) Minimize your query from (b).