Homework Assignment #4
(Initial)

CS 589 Principles of Database Systems: Spring 2011

This assignment is due Thursday, 4 March, at the beginning of class. You may do this assignment with a partner. If you do so, turn in a single submission with both names. You may seek help from your partner, the instructor and the class mailing list, but not other sources.

4A (10 points): Give a relational algebra expression that computes I_P for the Datalog program P for `Local(In, Loc, Num)` given in class. You can restrict your attention to `Local` facts. That is, given a set of facts f, your expression should return all `Local` facts in $I_P(f)$.

Note: You may use expressions in project, for example $\pi_{A,B,E=A+2B}(r)$.

4B a. (10 points) Provide a syntactic condition for "strong safety" of Datalog programs with computable predicates that guarantees a program returns only a finite number of facts (equivalently, has a finite minimum model). Explain how your condition guarantees finiteness.

b. (5 points) Show that the program for `Local(In, Loc, Num)` fails to satisfy your condition.

c. (10 points) Give a variant of the `Local` program that is strongly safe under your condition.

4C. Consider the following program P_{Fill} that determines if tiles can fill an M-by-N area.

```
Fill(M, N) :- Tile(M, N).
Fill(M, N) :- Fill(N, M).
Fill(M, N) :- Fill(M, N1), Fill(M, N2),
  Sum(N1, N2, N), LessEq(M, 10), LessEq(N, 10).
```

a. (10 points) Suppose the extensional database is

```
Tile(2, 4).
Tile(1, 3).
```

Show that P_{Fill} with this database implies `Fill(2, 7)` by giving a set of ground instances of rules that derive this fact.

b. (10 points) Explain why P_{Fill} with the facts above cannot derive `Fill(5, 5).`

c. (5 points) Show that there is a way to arrange tiles of the sizes above into a 5-by-5 area with no gaps.

4D. Refer to program P_{Fill} above.

a. (10 points) If we remove

```
Fill(M, N) :- Fill(N, M).
```

and add

```
Fill(M, N) :- Tile(N, M).
```

do we get an equivalent program? Support your answer.

b. (15 points) If we remove

```
Fill(M, N) :- Fill(N, M).
```

and add

```
Fill(M, N) :- Tile(N, M).
Fill(M, N) :- Fill(M1, N), Fill(M2, N),
   Sum(M1, M2, M), LessEq(M, 10), LessEq(N, 10).
```

do we get an equivalent program? Support your answer.

4E (to come) Problem on negation and recursion.