ECS 165B Database Systems — Winter 2014
Dr. Bertram Ludäscher — January 8

Assignment 1 – Datalog and Recursive Queries

Due: Friday, January 17th (submission details via the class forum at piazza.com)

Practice Problem I (UNGRADED). *The following problem was covered in ECS-165A last quarter. The input data and sample solutions are available, so you might want to study those.*

Consider the following relational input schema (EDB):

```
city(CityId, CityPop, Lat, Long, CountryCode)
country(CountryCode, CName, Area, CountryPop, GDP, Capital)
country_continent(CountryCode, Continent)
borders(CountryA, CountryB, Length)
river_into_lake(rname, lname)
river_from_lake(rname,lname)
country_religion(country, religion)
udef_religion(religion)
```

Formulate the following queries in Datalog:

1. List all city information for cities with a population greater than 1 million.
2. What are `countryCode`s of countries that have cities with a population > 1 million?
3. List all African capital cities together with their city population and the country they are located in.
4. What countries have a border with another country?
5. What countries have at least two borders with another country?
6. What countries have no border with another country?
7. List all countries in Europe and Asia.
8. List all countries that are located in both Europe and Asia.
9. What countries are located on multiple continents?
10. Find the city (or cities) with the largest population.
11. What rivers flow in and out of the same lake?
12. What countries have all the religions that are in the `udef_religion` relation?[1]

Practice Problem II (UNGRADED). Consider a binary relation `parent(`C`,`P`)` which means that P is a parent of C. Write Datalog rules for the following relations:

1. `grandparent(`C`,`G`)`: G is a grandparent of C.
2. `ancestor(`C`,`A`)`: A is an ancestor of of C.
3. `samegen(`X`,`Y`)`: X and Y are in the same generation, i.e., (i) they share a parent P, or (ii) their parents P_1 and P_2 are themselves in the same generation.
4. `lca(`C`,`D`,`A`)`: A is the lowest common ancestor of C and D, i.e., A is an ancestor of both C and D, but there is no other shared ancestor A' of C and D which is "lower" than A.

[1] This is a table that holds the names of religions that a user is interested in.

Problem 1 (RA $\rightsquigarrow$ Datalog). Let $R(x, y, z)$, $S(x, y, z)$, and $T(x, y, z)$ be three relations. Write one or more Datalog rules that define the result of each of the following expressions in the relational algebra:[2]

1. $R \cup S$.
2. $R \cap S$.
3. $R - S$.
4. $(R \cup S) - T$.
5. $(R - S) \cap (R - T)$.
6. $\pi_{x,y}(R)$.
7. $\pi_{x,y}(R) \cap \rho_{U(x,y)}(\pi_{y,z}(S))$.

Problem 2 (RA $\rightsquigarrow$ Datalog). Let $R(x, y, z)$ and $S(x, y, z)$ be two relations. Write one or more Datalog rules that define theta-joins $R \bowtie_\theta S$ where θ is one of the following:

1. $x = y$.
2. $x<y$ `AND` $y<z$.
3. $x<y$ `OR` $y<z$.
4. `NOT` $(x<y$ `OR` $x>y)$.

For each of these conditions, interpret the arithmetic comparison as comparing an attribute of R on the left with an attribute of S on the right. For example, $x < y$ stands for $R.x < S.y$.

Problem 3. For each of the following Datalog rules, write an equivalent relational algebra expression:

1. `p(X,Y) :- q(X,Z), r(Z,Y).`
2. `p(X,Y) :- q(X,Z), q(Z,Y).`
3. `p(X,Y) :- q(X,Z), r(Z,Y), X<Y.`

Problem 4 (Recursion in PostgreSQL). Consider the Datalog rules for `ancestor(X,Y)`:

```
ancestor(X,Y) :- parent(X,Y).
ancestor(X,Y) :- parent(X,Z), ancestor(Z,Y).
```

Create a table `parent` in PostgreSQL, then define a recursive view that computes `ancestor`.
Hint: Use `WITH-RECURSIVE`.

[2] Here, $\pi_{x,y}(R)$ means the projection on the first and second column of R; another possible notation is $\pi_{1,2}(R)$.