

## CSE 636: Test #1 (due 10/21/11)

Submit all the answers using `submit_cse636` as a **single pdf file**.

This is **individual work**. Duplicate solutions will be considered violations of academic integrity. Please write your name and the text “**The submitted solutions are my individual work.**” at the beginning of the submitted file.

### Problem 1 (15 pts)

Assume an undirected graph is represented as a set of facts of the form `node(x)` for a node  $x$ , and `edge(x,y)` and `edge(y,x)` for an edge  $\{x,y\}$ . A graph is *bipartite* if it does not contain a cycle of odd length.

1. Write a Stratified Datalog $\neg$  program  $P_1$  that checks whether a given graph is bipartite.
2. Give a stratification of the program  $P_1$ .
3. Test the program using `xsdb` on two graphs with at least 6 nodes per graph and report the results.
4. Can the program  $P_1$  be written without using negation?
  - If the answer to this question is positive, show the correct Datalog program.
  - If the answer is negative, identify the relevant property of Datalog programs, and show an example in which it is violated.

### Problem 2 (15 pts)

Assume XML documents are represented using the following Datalog predicates:

- `root(Id)`;
- `node(Id,TagName)`;
- `parent(ParentId,Num,ChildId)`

where `Id`, `ParentId`, `ChildId` are unique element node identifiers, `TagName` is the element name, and `Num` is the child’s sequential number (1,2,3,...). Other information present in those documents is ignored.

Consider the subset of XPath defined by the following grammar:

```

$$\begin{aligned} \langle \text{abspath} \rangle &::= \langle \text{axis} \rangle \langle \text{relpath} \rangle \\ \langle \text{relpath} \rangle &::= \langle \text{step} \rangle \\ \langle \text{relpath} \rangle &::= \langle \text{step} \rangle \langle \text{axis} \rangle \langle \text{relpath} \rangle \\ \langle \text{axis} \rangle &::= “/” \mid “//” \\ \langle \text{step} \rangle &::= \langle \text{name} \rangle \mid “*” \end{aligned}$$

```

Explain how you would generate from an XPath expression satisfying the above restrictions a Datalog program that always returns in the `query(X)` predicate the set of nodes satisfying the XPath expression. The nodes do not have to be returned in document order.

Show the Datalog program resulting from converting the following expression:

```
//a/*//b//*/c.
```

**Extra credit (10 pts):** Generalize your approach by allowing *conditions* on elements in the XPath expressions. The conditions may involve path expressions of the above kind and equality comparisons only. Describe the modified grammar and the extensions to your Datalog generation method.

**Problem 3 (20 pts)**

You are given two databases containing phone directories:

- The database A contains an *alphabetical* directory where each entry consists of a single subscriber name, subscriber address, and subscriber phone numbers (one or more), in that order. The entries are sorted alphabetically by name.
  - The database B contains a *reverse-lookup* directory where each entry contains a single phone number, subscriber name, and subscriber address, in that order. The entries are sorted by phone number.
1. Define the schemas of A and B using DTDs and XML Schema (with appropriate integrity constraints).
  2. Define in XQuery the mappings from A to B, and from B to A.
  3. Run the queries defining the mappings on a non-trivial example database (at least 10 entries) and report the results.