Computational Logic
– Mock Exam –

Werner Nutt

01/06/10

- The exam comprises **six** questions, which consist of several subquestions.
- Each question is worth 15 points. The total mark for the test will be based on the **four** questions for which you achieved the highest mark.
- There is a total of 60 points that can be achieved in this exam. You will have 2 hours time to answer the questions.
- Please, write down the answers to your questions in the test booklet handed out to you.
- For drafts use the blank paper provided by the university.
- If the space in the booklet turns out to be insufficient, please use the university paper for additional answers and return them with the booklet.

Queries in Relational Algebra and Calculus

Suppose a boat club has a database with the schema

`Boat(bname, type, colour)`
`Reservation(mname, bname, day)`

which records information about the boats owned by the club and about which member has reserved which boat on which day.

Consider the following two queries:

1. "Which members have only reserved red boats? "
2. "Which members made reservations for every boat of type dinghy?"

Express each query

(i) in relational algebra

(ii) in relational calculus, that is, as an expression of the form

$$\{ x \mid \phi(x) \}$$

where x is the variable for which we want bindings and $\phi(x)$ is a logical formula with free variable x.

Safety and Domain Independence of Queries

Consider the following four queries expressed in relational calculus:

1. $\{\, x, y \mid \exists z\, \texttt{hasChild}(x, z) \lor \exists w\, \texttt{hasChild}(w, y) \,\}$
2. $\{\, x \mid \texttt{rich}(x) \land \forall y\, (\texttt{hasChild}(x, y) \to \neg\texttt{rich}(y)) \,\}$
3. $\{\, x \mid \texttt{rich}(x) \land \forall y\, (\neg\texttt{hasChild}(x, y) \to \texttt{rich}(y)) \,\}$

For each query, determine whether or not it is

- safe
- domain-independent.

For each query and each property, if your answer is "yes", briefly and informally explain your answer. If your answer is "no", provide an example showing that the query does not have the property in question.

Containment

In this question, we only consider relational conjunctive queries, that is, queries that do not contain comparisons.

Suppose q_0 is a fixed conjunctive query.

- The **container problem** for q_0 is the following decision problem:

  Given a conjunctive query q, decide whether $q_0 \subseteq q$.

- The **containee problem** for q_0 is the following decision problem:

  Given a conjunctive query q, decide whether $q \subseteq q_0$.

Prove or disprove the following statements:

1. For every conjunctive query q_0, there is a polynomial time algorithm to decide the *container problem* for q_0.

2. For every conjunctive query q_0, there is a polynomial time algorithm to decide the *containee problem* for q_0.

To prove a statement, a sketch of an algorithm together with a short argument why it is polynomial is sufficient. To disprove the statement, find a query q_0 for which the problem in question is NP-hard. Again, a proof sketch is sufficient to show the NP-hardness.

Expressiveness of Datalog

Consider a database containing facts about a binary relation `edge` and unary relations `red` and `blue`. Clearly, an instance of the database can be seen as a description of a directed acyclic graph, where some of the nodes are red and some nodes are blue.

In this question, we adapt the definition of cliques from undirected to directed graphs. We say that a directed graph is called a *clique* if any two vertices of the graph are connected by an edge. That is, for any two vertices v_1, v_2, there is an edge from v_1 to v_2 and an edge from v_2 to v_1. A graph has a clique of k nodes where $k > 0$ if there is a subgraph of k nodes that is a clique.

Below, there are four properties of directed graphs with red and blue nodes. Which of these properties can be expressed by Datalog queries?

(i) The graph is a clique.

(ii) The graph has a clique of exactly three nodes.

(iii) The graph has a clique of no more than three nodes (that is, the graph has a clique of k nodes, where $k \in \{1, 2, 3\}$).

(iv) The graph has a circle where red and blue nodes alternate (that is, where a red node is followed by a blue node and vice versa).

To show that a query is expressible, write down a datalog program that captures the property. (Note: the query predicate $q()$ of such a query has to be nullary, that is, it can either succeed or fail.) To show that a query is not expressible, sketch a proof. In that proof, you may use any of the results on the expressiveness of query languages that were shown in the course.

Datalog Queries

Consider the metro database of the lectures. That database contains a relation

`link`(`line`, `from`, `to`),

where an entry (l, $s1$, $s2$) means that station $s1$ and station $s2$ are connected by line l without a stop in between.

Consider the following three queries:

1. Which stations can be reached from Odeon with a trip involving only one line?
2. Which stations can be reached from Odeon with a trip involving no more than two lines?
3. Which stations can be reached from Odeon with a trip involving exactly two lines?

Answer the questions:

- Which of these queries is expressible in datalog and which not?
- Is there a query that can be expressed in stratified datalog$^{\neg}$, but not in datalog?

To show that a query is expressible, write down a datalog (or datalog$^{\neg}$) program that computes the query. To show that a query is not expressible, sketch a proof. In that proof, you may use any of the results on the expresivity of query languages that were shown in the course.

Magic Set Transformation

Suppose a binary EDB-relation par is given and the IDB-relation anc is defined by the following program P:

$$\begin{aligned}\texttt{anc}(x,y) &:- \texttt{par}(x,y)\\ \texttt{anc}(x,y) &:- \texttt{anc}(x,z),\texttt{anc}(z,y).\end{aligned}$$

1. For the query

   $$q(x) :- \texttt{anc}(\texttt{adam},x)$$

   transform the program P into a program P_{mag} using the magic set technique.

2. Give an instance $\mathbf{I}(\texttt{par})$ of par such that for $\mathbf{I}(\texttt{par})$ the number of tuples in the least fixpoint of P_{mag} is no more than 1 percent of the number of tuples in the least fixpoint of P. (An informal description of $\mathbf{I}(\texttt{par})$ suffices.)