2. Conjunctive Queries

Instructions: Work in groups of 2 students. You can write up your answers by hand (provided your handwriting is legible) or use a word processing system like Latex or Word. However, experience shows that Word is in general difficult to use for this kind of task. Please, include name and email address in your submission.

1. Classes of Conjunctive Queries

We view queries as functions that map database instances to relation instances. Consider the following classes of conjunctive queries, which are distinguished by the form of the rules by which they can be defined:

CQ: rules without equality "$=$" and disequality "$\neq$" atoms ("simple" conjunctive queries)

$\mathsf{CQ}_{=}$: rules that may have equality atoms, but no disequality atoms

$\mathsf{CQ}_{\neq}$: rules that may have disequality atoms, but no equality atoms

$\mathsf{CQ}_{=,\neq}$: rules that may have both, equality and disequality atoms (correspond to conjunctive queries as defined in the lecture)

CQ_{rep}: rules that may repeat variables in the head, but do not have equality and disequality atoms

CQ_{const}: rules that may have constants in the head, but do not have equality and disequality atoms

$\mathsf{CQ}_{rep,const}$: rules that may repeat variables and may have constants in the head, but do not have equality and disequality atoms.

Determine which inclusions hold between these classes and which not:

- To show that class C_1 is included in class C_2 (i.e., $C_1 \subseteq C_2$), indicate how any query in C_1 can be equivalently expressed by a query in C_2.

- To show that C_1 is not included in C_2 (i.e., $C_1 \nsubseteq C_2$), exhibit a query in C_2 for which you show that it cannot be expressed by a rule of the kind that defines queries in C_1.

Clearly, some inclusions are obvious. Note also that you can derive some other inclusions exploiting the fact that set inclusion is transitive.
For this exercise it suffices to sketch the proofs.

Hint: The following trivial lemma will be useful for your proof, since it allows you to exploit inclusions to conclude non-inclusions from other non-inclusions. With that lemma, you should be able to classify all these query tyles using no more then six non-inclusion proofs.

Lemma 1 *Let A, C, C', B be sets such that $A \subseteq C$, $C' \subseteq B$. Then*

$$C' \nsubseteq C \qquad \textit{implies} \qquad C' \nsubseteq A \quad \textit{and} \quad B \nsubseteq C.$$

As a consequence, you only have to prove some crucial non-inclusions, from which others will follow. Of course, you get the best leverage of the lemma if you prove non-inclusions "$C' \nsubseteq C$" for sets C', C, where C' has many supersets and C has many subsets.

(20 Points)

2. Unions of Conjunctive Queries

Show that adding union to simple conjunctive queries strictly increases the expressivity of the resulting query language. (Recall from the previous exercise that simple conjunctive queries have neither equality nor disequality atoms.)

Hint 1: Consider the query defined by the two rules

$$\begin{aligned} Q() &\ :\!-\ p(1) \\ Q() &\ :\!-\ p(2) \end{aligned}$$

and show that no query defined by a single rule is equivalent to it.

Hint 2: Assume there is an equivalent simple conjunctive query. Then consider several databases distinguished by the constants occurring in them.

(10 Points)

Submission: 12 April 2011, 10:30 am, at the lecture