Chair of applied computer science III
Prof. Dr. Guido Moerkotte
Email: moerkotte@uni-mannheim.de
UNIVERSITY OF MANNHEIM

Query Optimization — Exercise sheet 2

Exercise 1

The following equivalences hold for 2-valued logic. In the context of database systems we have to deal with 3-valued logic (`NULL` values). Which of these equivalences holds for 3-valued logic? For the ones that don't hold, find a counter example. For the ones that hold, proof the equivalence by evaluating the truth table. Note that you only have to evaluate the cases where *unknown* occurs since you can assume that they hold for 2-valued logic.

Exercise 1 a)

$x \wedge \neg x = false$
and
$x \vee \neg x = true$

Exercise 1 b)

$(x \wedge y) \wedge z = x \wedge (y \wedge z)$
and
$(x \vee y) \vee z = x \vee (y \vee z)$

Exercise 2

Greedy algorithms are applied to solve optimization problems. The solution obtained is not necessarily globally optimal. Greedy algorithms construct a solution step-by-step where each step is locally optimal. Steps taken cannot be undone in subsequent steps. Greedy algorithms terminate once a complete solution is found.

Exercise 2 a)

Using your favorite book on algorithms or Wikipedia, read about the greedy algorithm technique.

Exercise 2 b)

Implement either `GreedyJoinOrdering-1` or `GreedyJoinOrdering-2` from Chapter 3.2.1 Heuristics in the script.

Exercise 2 c)

What types of query graphs can `GreedyJoinOrdering-1` and `GreedyJoinOrder-2` handle? What type of join trees do they generate?

Exercise 3

You are given the cardinalities for relations $R_i, i \in \{1, \ldots, 4\}$ together with their join selectivities:

$|R_1| = 10$, $|R_2| = 100$, $|R_3| = 1000$, $|R_4| = 25$
$f_{1,2} = 0,1$, $f_{2,3} = 0,2$, $f_{1,3} = 0,4$, $f_{3,4} = 0,1$

We already know that there is a large number of possible join trees (120, bushy and cross-products considered).

To see how costs vary from one join tree to another, compute the cost for some of the join trees for the cost functions C_{out}, C_{nlj}, C_{hj} and C_{smj}.
You will find their definitions in the script.
Note that the cost of a cross product $e_1 \times e_2$ is given by $|e_1| * |e_2|$.