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

Query Optimization — Exercise sheet 8

Exercise 1

Exercise 1 a)

Read about *computer algebra*, also known as *symbolic computation*.
Wikipedia is your friend:
`https://en.wikipedia.org/wiki/Computer_algebra`

Exercise 1 b)

Implement `ExhaustiveTransformation`. You may use the helper classes provided in the solution code.

Exercise 1 c)

Discuss ways implement non-exhaustive termination criteria for `ExhaustiveTransformation`.

Exercise 1 d)

Discuss how to adjust `ExhaustiveTransformation` to avoid cross products. (See script.)

Exercise 1 e)

BONUS
Read and present in the next exercise session:
`https://pdfs.semanticscholar.org/5d91/f42e767c167ecf188a46608802ba5fe52347.pdf`

Exercise 2

Update the classification table for join algorithms from one of the previous exercises with the following algorithms:

- Transformation-based approach
- Memoization
- Generate permutations
- Quick Pick

- Iterative Improvement
- Simulated Annealing
- TabuSearch

Exercise 3

Find reasonable plans for the following using queries using indices. Explicitly state the assumptions you make regarding the indices.

Exercise 3 a)

```
select a.v
from   a a
where a.w > 10
  and a.w < 30
```

Exercise 3 b)

```
select *
from a
where exists (select *
              from b
              where a.v = b.key)
```

Exercise 3 c)

```
select a.b, min(a.c)
from a
group by a.b
```

Exercise 4

For each of the following queries, write down one plans with explicit attribute accesses and another one with implicit attribute accesses.

Exercise 4 a)

```
select s.name, s.matrnr
from studenten s
where s.age>27 and
      count(s.hoert)=0
```

Exercise 4 b)

```
select s.name, s.matrnr
from studenten s
where s.age>27 and
      count(s.hoert)<5 and
      exists v in s.hoert:
         v.name="AO"
```