Advanced Topics in Databases
Exercise 3

Task 1 Why is query optimization needed within relational DBMSs?
Considering different design options (tuple- vs operator-at-a-time, row vs. column-store), how do these design options affect the query optimization?

Task 2 Describe the process of executing a SQL-query.
What is the difference between logical and physical optimization?
Why are these two processing steps not sufficient to determine an efficient execution?

Task 3 Considering the deterministic and randomized approaches for join-order optimization, which of both categories would you use for the following use-cases:

(a) Optimization of complex queries
(b) Optimal results are needed
(c) Optimization of simple queries
(d) Reduce implementation effort for optimization

Task 4 Given the following query

```
SELECT *
FROM CUSTOMER C, ORDERS O, LINEITEM L, SUPPLIER S
WHERE C.C_ID = O.C_ID AND
      O.O_ID = L.O_ID AND
      L.S_ID = S.S_ID;
```

Provide all possible join orders considering only left deep trees without considering cross-joins.
For this, use the following notation: \(((C \bowtie S) \bowtie L)\)
What would change, if we consider cross joins during the optimization?
Task 5 Determine an optimal join order using the dynamic programming approach for the following query:

```
SELECT *
FROM CUSTOMER C, ORDERS O, LINEITEM L, SUPPLIER S
WHERE C.C_ID = O.C_ID AND
    O.O_ID = L.O_ID AND
    L.S_ID = S.S_ID;
```

The tables contain the following number of entries:

- CUSTOMER: 100
- ORDERS: 3000
- LINEITEM: 1000
- SUPPLIER: 200

The selectivity of joining the following tables are:

- C $\bowtie$ O: 0.1
- O $\bowtie$ S: 1
- O $\bowtie$ L: 0.5
- S $\bowtie$ C: 1
- L $\bowtie$ S: 0.2

Calculate the cost based on the following way: Result size + cost of input operators.
Example: $\text{(C } \bowtie \text{ O)} = \text{Cost C } + \text{ Cost O } + \text{ Cost (C } \bowtie \text{ O)} = \text{Cost C } + \text{ Cost O } + (\text{size C } \times \text{ size O } \times \text{ selectivity}) = 100 + 3000 + (100 \times 3000 \times 0.1) = 33,100$.

Use an iterative construction of the final result.
In each iteration, increase the number of tables contained in a result, (invalid and unneeded combinations can be skipped).

Task 6 What is the difference between the sequential variants of the dynamic programming approach for join-order optimization?
Which variant of the dynamic programming approach would you use in the following scenario:

(a) Optimization of star queries
(b) Optimization of linear queries with few tables
(c) Optimization of cyclic queries with many tables
(d) Optimization of clique queries

Good Luck!