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

Query Optimization Exercise sheet 9

Exercise 1

Exercise 1 a)

Discuss how to compute the break-even point when choosing between a table scan and an index scan.

Solution

Several factors play a role.

- Clustering of the index
- Relationship between the number of pages needed for the leaves of the index and the number of pages needed for the relation.
- caching properties for the pages on the index, e.g. the root of the index usually resides in the database buffer
- The type of predicate: point query, range predicate, ...

Using an index, the cost for the first access is $\log n$. After this, for range predicates, we have sequential access. If the table is not clustered by the index, we have to sort the TIDs obtained from the index to achieve good performance. If the the TIDs are *dense*, then we have (near) sequential for the disk reads.

Then, the break-even point corresponds to some selectivity of the predicate.

Exercise 1 b)

Let n be the number of pages needed for a relation. The time to access a page is $D_{pos} + D_{read}$. Where D_{pos} is the time to position the read/write head and D_{read} is the time it takes to read a page.

For the following numbers:

$$\begin{aligned} D_{pos} &= 5,0\,\frac{\text{ms}}{\text{page}} \\ D_{read} &= 0,5\,\frac{\text{ms}}{\text{page}} \\ n &= 110 \text{ pages} \end{aligned}$$

Compute the predicate selectivity s of the break-even point between random disk accesses and sequential disk accesses.

Solution

$$
\begin{aligned}
D_{pos} + n \cdot D_{read} &> s \cdot n \cdot (D_{pos} + D_{read}) \\
s &< \frac{D_{pos} + n \cdot D_{read}}{n \cdot (D_{pos} + D_{read})} \\
s &< 0,099
\end{aligned}
$$

That is, if the predicate selectivity is smaller than 10%, we should prefer to random disk accesses.

Exercise 2

Exercise 2 a)

Read about metaheuristics in computer science.
Wikipedia is your friend: `https://en.wikipedia.org/wiki/Metaheuristic`

Exercise 2 b)

Read the *Simulated Annealing* chapter in the script.

Exercise 2 c)

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

Exercise 3

Exercise 3 a)

What is the expexted number of distinct balls drawn, when drawing k times from an urn with m balls with replacement.

Solution

d := outcome of the above event.
$\mathbb{E}[d] = m * (1 - (1 - 1/m))^k)$

- $1/m$: probability of drawing ball i
- $1 - 1/m$: probability of not drawing ball i
- $(1 - 1/m)^k$ probability of not drawing ball i in k attempts

- $1 - (1 - 1/m)^k$: probability of drawing ball i at least once
- $m * (1 - (1 - 1/m)^k)$: previous probability holds for each ball. Multiply with the number of balls to get expexted value.

Exercise 3 b)

Describe the yao formula.
Compute the yao formula for the values $N = 1000, m = 100, k = 15$

Solution

Read the chapter *Counting the Number of Direct Accesses* in the script.
To compute the result, plug in the values in the formula.