Implementation of DBMS
Exercise Sheet 12, Solutions
Klingemann, WS 2024 / 2025

1) Let `R` and `S` be relations, `p` a predicate with only `R` attributes, `q` a predicate with only `S` attributes and `m` a predicate with attributes from `R` and `S`. Show that the following rule holds. Use in your proof only the given rules in the box. Indicate in each step which rule you are using.

$$\sigma_{m \wedge p \wedge q}(R \bowtie S) = \sigma_m\big([\sigma_p(R)] \bowtie [\sigma_q(S)]\big)$$

Solution:

$\sigma_{m \wedge p \wedge q}(R \bowtie S)$

$= \sigma_{m \wedge p}(\sigma_q(R \bowtie S))$ (rule 1)

$= \sigma_m(\sigma_p(\sigma_q(R \bowtie S)))$ (rule 1)

$= \sigma_m(\sigma_p(R \bowtie [\sigma_q(S)]))$ (rule 4)

$= \sigma_m\big([\sigma_p(R)] \bowtie [\sigma_q(S)]\big)$ (rule 3)

Rules

Let `R` and `S` be relations, `p1` and `p2` arbitrary predicates and
`p` a predicate with only `R` attributes,
`q` a predicate with only `S` attributes

1) $\sigma_{p1 \wedge p2}(R) = \sigma_{p1}[\sigma_{p2}(R)]$

2) $\sigma_{p1 \vee p2}(R) = [\sigma_{p1}(R)] \cup [\sigma_{p2}(R)]$

3) $\sigma_p(R \bowtie S) = [\sigma_p(R)] \bowtie S$

4) $\sigma_q(R \bowtie S) = R \bowtie [\sigma_q(S)]$

2) Some familiar laws also apply for variants of joins, others do not. Tell, whether each of the following is true or not. Condition C involves only attributes of R. Give either a proof that the law holds or a counterexample.

Note, that ⟗ means the outerjoin (similar to the ordinary inner join but we also add for each relation the tuples that do not find a match in the other relation).

a) σ_C(R ⟗ S) = σ_C(R) ⟗ S

Solution:

This law does not hold. This results from the fact that tuples which do not fulfil the condition of a selection are simply gone whereas the outerjoin will produce tuples even if we do not have a matching tuple in the other relation. Consider the following example with the selection condition C being X = Y:

R:

X	Y
1	2

S:

Y	Z
2	3

Then we have

R ⟗ S:

X	Y	Z
1	2	3

$\sigma_{X=Y}$(R ⟗ S):

X	Y	Z
no tuple		

On the other hand we get:

$\sigma_{X=Y}$(R):

X	Y
no tuple	

$\sigma_{X=Y}$(R) ⟗ S:

X	Y	Z
null	2	3

b) (R ⟗ S) ⟗ T = R ⟗ (S ⟗ T)

Solution: This law does not hold, either. Consider:

R:

A	B
1	1

S:

C	D
2	2

T:

B	C
3	3

Then we have

R ⟗ S:

A	B	C	D
1	1	2	2

(R ⟗ S) ⟗ T

A	B	C	D
1	1	2	2
null	3	3	null

On the other hand we get:

(S ⟗ T):

B	C	D
null	2	2
3	3	null

R ⟗ (S ⟗ T):

A	B	C	D
1	1	null	null
null	null	2	2
null	3	3	null

3) We have a relation R(A, B, C, D). The tuples of R are stored in secondary storage in a random order. We want to create the result relation of the expression $\pi_{B,D}$ ($\sigma_{A=20}$(R)) and write it to secondary storage. We do this by sequentially reading all the blocks of R, apply both operations without writing the intermediate relation to disk and finally write the blocks of the result relation of the complete expression to disk. We assume that 10% of the tuples of R fulfil the selection condition. We further assume that the projection is eliminating duplicates and that for each tuple in the result relation of the selection there is one other tuple with the same values. We also have the following information:

- The relation R has 100000 tuples.
- The size of a block is 8192 bytes. Blocks have a header of 140 bytes.
- The sizes of attributes are 84 bytes for A, 20 bytes for B, 370 bytes for C and 120 bytes for D. Records of R have a header of 38 bytes. Records of the result relation of the given expression have a header of 22 bytes.
- Each block holding tuples is full of as many tuples as possible. We use unspanned storage for the records.

What is the cost in terms of number of I/Os?

Solution:
A record of R consists of the record header and the four attributes A, B, C and D. Therefore, the size of a record is (38 + 84 + 20 + 370 + 120) bytes = 632 bytes.
We have $\lfloor$((8192 – 140) bytes available for records/block) / (632 bytes/record)$\rfloor$ = 12 records/block and therefore, we need $\lceil$100000 records / (12 records/block)$\rceil$ = 8334 blocks to store the complete relation.
As 10% of the tuples of R fulfil the selection condition, the result relation of the selection consists of 0.1 * 100000 = 10000 records. As we assume that the projection is eliminating duplicates and for each tuple in the result relation of the selection there is one other tuple with the same values, the result relation of the projection has half the number of tuples, i.e., 5000. Each record of the result relation consists of the record header and the two attributes B and D. Therefore, the size of a record is (22 + 20 + 120) bytes = 162 bytes.
We have $\lfloor$((8192 – 140) bytes available for records/block) / (162 bytes/record)$\rfloor$ = 49 records/block and therefore, we need $\lceil$5000 records / (49 records/block)$\rceil$ = 103 blocks to store the complete result relation.
As we generate the result relation by reading all blocks of R and in the end write all blocks of the result relation to disk we have (8334 + 103) I/O’s = 8437 I/O’s.