Exercises: Relational Algebra

Exercise 2.1 Explain the statement: relational algebra operators can be composed. Why is the ability to compose operators important?

Exercise 2.2 Given two relations R1 and R2, where R1 contains N1 tuples, R2 contains N2 tuples, and N2 > N1 > 0, give the minimum and maximum possible sizes (in tuples) for the resulting relation produced by each of the following relational algebra expressions. In each case, state any assumptions about the schemas for R1 and R2 needed to make the expression meaningful:

(Assume set semantics)

1. R1 U R2
2. R1 ∩ R2
3. R1 – R2
4. R1 x R2
5. $\sigma_{a=5}(R1)$
6. $\pi_a(R1)$
7. $R1/R2$

Exercise 2.3 Consider the following schema:

Suppliers(*sid*: **integer**, *sname*: **string**, *address*: **string**)
Parts(*pid*: **integer**, *pname*: **string**, *color*: **string**)
Catalog(*sid*: **integer**, *pid*: **integer**, *cost*: **real**)

The key fields are underlined, and the domain of each field is listed after the field name. Therefore, sid is the key for Suppliers, pid is the key for Parts, and sid and pid together form the key for Catalog. The Catalog relation lists the prices charged for parts by Suppliers. Write the following queries in relational algebra:

1. Find the names of suppliers who supply red parts.
2. Find the sids of suppliers who supply red or green parts.
3. Find the sids of suppliers who supply red parts or are at 221 Packer Ave.
4. Find the sids of suppliers who supply both red and green parts.
5. Find the sids of suppliers who supply every part.
6. Find the sids of suppliers who supply every red part.
7. Find the sids of suppliers who supply every red or green part.
8. Find the sids of suppliers who supply every red part or supply every green part.
9. Find pairs of sids such that the supplier with the first sid charges more for some part than the supplier with the second sid.
10. Find the pids of parts supplied by at least two different suppliers.

Solutions

Answer 2.1 Every operator in relational algebra accepts one or more relation instances as arguments and the result is always a relation instance. So, the argument of one operator could be the result of another operator. This is important because this makes it easy to write complex queries by simply composing relational algebra operators.

Answer 2.2 The answer is presented in the following figure:

Expression	Assumption	Min	Max
R1 U R2	R1 and R2 are union-compatible	N2	N1+N2
R1 ∩ R2	R1 and R2 are union-compatible	0	N1
R1 – R2	R1 and R2 are union-compatible	0	N1
R1 x R2		N1 * N2	N1 * N2
$\sigma_{a=5}(R1)$	R1 has an attribute named a	0	N1
$\pi_a(R1)$	R1 has an attribute named a	1	N1
$R1/R2$	The set of attributes of R2 is a subset of the set of attributes of R1	0	0*

*When performing the division R1/R2, the set of attributes of R2 needs to be a subset of the set of attributes of R1. Imagine that R2 has two attributes y and z, while R1 has three - namely x, y, z. In order for a value of the attribute x to be contained in the result set of the division, it would need to appear in R1 with all the combinations of y and z that are present in R2. If N2>N1, this is impossible since the combinations of y and z in the R2 table will be more than the number of rows of R1.
Note that using set semantics, a combination of y and z cannot appear twice in R2.

Answer 2.3

1. RA:

$$\pi_{sname}\left(\pi_{sid}\left(\left(\pi_{pid}\sigma_{color='red'}Parts\right) \bowtie Catalog\right) \bowtie Suppliers\right)$$

SELECT *name*
FROM Suppliers, Catalog, Parts
WHERE Parts.*color* = ’red’ **AND**
Catalog.*pid* = Parts.*pid* **AND**
Suppliers.*sid* = Catalog.*sid*;

2. RA:

$$\pi_{sid}\left(\pi_{pid}\left(\sigma_{color='red' \vee color='green'} Parts\right) \bowtie Catalog\right)$$

SELECT *sid*
FROM Parts, Catalog
WHERE (Parts.*color = 'red'* **OR** Parts.*color = 'green')* **AND**
Parts.*pid* = Catalog.*pid*

3. RA

$$\rho\left(R1, \pi_{sid}\left(\left(\pi_{pid}\sigma_{color='red'} Parts\right) \bowtie Catalog\right)\right)$$
$$\rho\left(R2, \pi_{sid}\sigma_{address='221\ Packer\ Ave'} Suppliers\right)$$
$$R1 \cup R2$$

4. RA

$$\rho\left(R1, \pi_{sid}\left(\left(\pi_{pid}\sigma_{color='red'} Parts\right) \bowtie Catalog\right)\right)$$
$$\rho\left(R2, \pi_{sid}\left(\left(\pi_{pid}\sigma_{color='green'} Parts\right) \bowtie Catalog\right)\right)$$
$$R1 \cap R2$$

5. RA

$$\left(\pi_{sid,pid} Catalog\right) / \left(\pi_{pid} Parts\right)$$

6. RA

$$\left(\pi_{sid,pid} Catalog\right) / \left(\pi_{pid}\sigma_{color='red'} Parts\right)$$

7. RA

$$\left(\pi_{sid,pid} Catalog\right) / \left(\pi_{pid}\sigma_{color='red' \vee color='green'} Parts\right)$$

8. RA

$$\rho\left(R1, \left(\pi_{sid,pid} Catalog\right) / \left(\pi_{pid}\sigma_{color='red'} Parts\right)\right)$$
$$\rho\left(R2, \left(\pi_{sid,pid} Catalog\right) / \left(\pi_{pid}\sigma_{color='green'} Parts\right)\right)$$
$$R1 \cup R2$$

9. RA

$$\rho(R1, Catalog)$$
$$\rho(R2, Catalog)$$
$$\pi_{R1.sid,R2.sid}\left(\sigma_{R1.pid=R2.pid \wedge R1.sid \neq R2.sid \wedge R1.cost > R2.cost} (R1 \times R2)\right)$$

Alternative: $\pi_{R1.sid,R2.sid}\left(R1 \bowtie_{R1.pid=R2.pid \wedge R1.sid \neq R2.sid \wedge R1.cost > R2.cost} R2\right)$

10. RA

$$\rho(R1, Catalog)$$
$$\rho(R2, Catalog)$$

$$\pi_{R1.pid}\sigma_{R1.pid=R2.pid \land R1.sid \neq R2.sid}(R1 \times R2)$$

Alternative:

$$\pi_{R1.pid}\left(R1 \bowtie_{R1.pid=R2.pid \land R1.sid \neq R2.sid} R2\right)$$