2. Exercise

Timo Bergerbusch 344408
Thomas Näveke 311392
Shu Xing 381176

30.10.2017

1 Exercise 2.1

2.1.1 **Definition 1 (relational completeness)** *A query language is relationally complete if it is at least as expressive as relational algebra.*

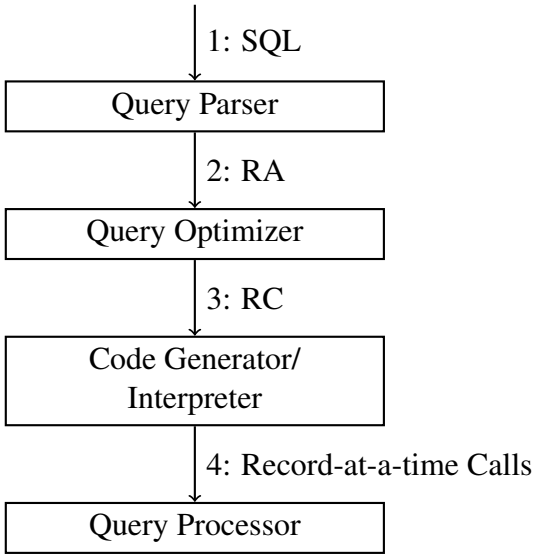
Methods with Example table A($\underline{x_1}$,y_1) and B($\underline{y_1}$)

method	RA (Example)	keyword	SQL(Example)
selection	$\sigma_{x_1>0}(A)$	WHERE	SELECT * FROM A WHERE A.$x_1 > 0$
projection	$\pi_{x_1}(A)$	SELECT	SELECT x_1 FROM A
cartesian product	A $\times$ B	FROM, CROSS	SELECT * FROM A, B
union	$\sigma_{x_1>0}(A) \cup \sigma_{x_1<0}(A)$	UNION	SELECT * FROM A WHERE $x_1 > 0$ UNION SELECT * FROM A WHERE $x_1 < 0$
difference	A - A	EXCEPT, NOT IN	SELECT * FROM B WHERE y_1 NOT IN A

2.1.2 **1. Beispiel**
SELECT SUM(salary) FROM works;

2. Beispiel
SELECT COUNT(city) FROM located;

2.1.3 Solution Graph:



2.1.4 Overview:

operator	operation type
selection	unary operator
projection	unary operator
cartesian product	set & binary operator
union	set & binary operator
difference	set & binary operator

2 Exercise 2.2

2.2.1 **relational algebra**:

$\pi_{pname}(\rho_p(works) \bowtie_{p.cname=b.cname \wedge p.salary>b.salary}$
$\rho_b(works \bowtie_{works.pname=boss.mname} \rho_{pname1 \leftarrow pname}(boss)))$

tuple relational calculus:

$\{< w.pname > | \ w \in works \wedge \exists m \in boss(w.pname = m.pname \wedge \exists w' \in works(w'.pname = m.pname \wedge w.cname = w'.cname \wedge w.salary > w'.salary))\}$

domain relational calculus:

$\{pname \mid \exists cname, salary1, salary2, mname \wedge works(pname, cname, salary1) \wedge works(mname, cnan$
$boss(pname, cname) \wedge salary1 > salary2\}$

2.2.2 **relational algebra**:

$\pi_{pname}(\rho_{c_1}(works) \bowtie_{c_1.pname=c_2.pname \wedge c_1.cname!=c_2.cname} \rho_{p_2}(works))$

tuple relational calculus:

$\{< w_1.pname > | \ w_1 \in works \wedge \exists w_2 \in works(w_1.pname = w_2.pname \wedge w_1.cname \neq w_2.cname)\}$

domain relational calculus:

$\{pname \mid \exists cname_1, cname_2, salary_1, salary_2 \wedge works(pname, cname_1, salary_1) \wedge$
$works(pname, cname_2, salary_2) \wedge cname_1 \neq cname_2\}$

2.2.3 **relational algebra**:

$\pi_{pname}(works) - \pi_{pname}(\rho_{p_1}(works) \bowtie_{p_1.salary<p_2.salary} \rho_{p_2}(works))0$

tuple relational calculus:

$\{< w_1.pname > | \ w_1 \in works \wedge \neg(\exists w_2 \in works(w_1.cname = w_2.cname \wedge w_2.salary > w_1.salary))\}$

domain relational calculus:

$\{pname \mid \exists cname, salary_1 \wedge works(pname, cname, salary_1) \wedge \neg(\exists pname_2, salary_2 \wedge works(pname_2, cname, salary_2) \wedge salary_2 > salary_1)\}$

2.2.4 **relational algebra**:

$\pi_{c_1.cname}(\rho_{c_1}(located) \bowtie_{c_1.city \neq c_2.city} \rho_{c_2}(\sigma_{cname='IBM'}(located)))$

tuple relational calculus:

$\{< c_1.cname > | \ c_1 \in located \wedge \forall c_2 \in located(c_2.cname \neq' IBM' \vee c_1.city \neq c_2.city)\}$

domain relational calculus:

$\{cname \mid \exists city \wedge located(cname, city) \wedge \neg located('IBM', city)\}$

3 Exercise 2.3

2.3.1 with $N = 15000$ and $B = 8$: $\lceil \frac{N}{B} \rceil \Rightarrow \lceil \frac{15000}{8} \rceil = 1875$

2.3.2 Number of passes given by: $1 + \lceil \log_{B-1} \lceil \frac{N}{B} \rceil \rceil$
with $N = 15000$ and $B = 8 \Rightarrow 1 + \lceil \log_{8-1} \lceil \frac{15000}{8} \rceil \rceil = 1 + \lceil \log_7 1875 \rceil = 5$

2.3.3 $2 = 1 + \lceil \log_{B-1} \lceil \frac{15000}{B} \rceil \rceil \Leftrightarrow 1 = \lceil \log_{B-1} \lceil \frac{15000}{B} \rceil \rceil \Leftrightarrow \frac{1}{2}(1 + \sqrt{600001}) \leq B \Rightarrow 123 \leq B$
So we need at-least 123 Buffer to be able to sort the file in two passes.

2.3.4 Since we only use 2+1 Pages in a Two-Way-Sort: $B = 3$ and $N = 15000$
$\Rightarrow 1 + \lceil \log_{3-1} \lceil \frac{15000}{2} \rceil \rceil = 14$ passes and $\lceil \frac{15000}{2} \rceil = 7500$ runs