Fall 2023 B461 Assignment 4

Released: 16 Oct, 2023
Due: Oct 26, 2023

This assignment relies on the lectures:

- Query Optimization
- Aggregate functions
- Functions and Expressions
- Queries with quantifiers

To turn in your assignment, you will need to upload to Canvas a single file with name `assignment4.sql` which contains the necessary SQL statements that solve the problems in this assignment. The `assignment4.sql` file must be so that the AI's can run it in their PostgreSQL environment. You should use the `SQL Script file` to construct the `assignment4.sql` file. (Note that the data to be used for this assignment is included in this file.) In addition, you will need to upload a separate `assignment4.txt` file that contains the results of running your queries and `assignment4.pdf` file that contains the theoretical answers(Optimization steps) with conditions.

1 Database schema and instances

For the problems in this assignment, we will use the following database schema:

```
Person(pid, pname, city)
Company(cname, headquarter)
Skill(skill)
worksFor(pid, cname, salary)
companyLocation(cname, city)
personSkill(pid, skill)
hasManager(eid, mid)
Knows(pid1, pid2)
```

In this database we maintain a set of persons (`Person`), a set of companies (`Company`), and a set of (job) skills (`Skill`). The `pname` attribute in `Person` is the name of the person. The `city` attribute in `Person` specifies the city in which the person lives. The `cname` attribute in `Company` is the name of the company. The `headquarter` attribute in `Company` is the name of the city wherein the company has its headquarter. The `skill` attribute in `Skill` is the name of a (job) skill.

A person can work for at most one company. This information is maintained in the `worksFor` relation. (We permit that a person does not work for any company.) The `salary` attribute in `worksFor` specifies the salary made by the person.

The `city` attribute in `companyLocation` indicates a city in which the company is located. (Companies may be located in multiple cities.)

A person can have multiple job skills. This information is maintained in the `personSkill` relation. A job skill can be the job skill of multiple persons. (A person may not have any job skills, and a job skill may have no persons with that skill.)

A pair (`e`;`m`) in `hasManager` indicates that person `e` has person `m` as one of his or her managers. We permit that an employee has multiple managers and that a manager may manage multiple employees. (It is possible that an employee has no manager and that an employee is not a manager.) We further require that an employee and his or her managers must work for the same company.

The relation `Knows` maintains a set of pairs (`p1`; `p2`) where `p1` and `p2` are `pids` of persons. The pair (`p1`; `p2`) indicates that the person with `pid p1` knows the person with `pid p2`. We do not assume that the relation `Knows`

is symmetric: it is possible that (p1; p2) is in the relation but that (p2; p1) is not.

The domain for the attributes pid, pid1, pid2, salary, eid, and mid is integer. The domain for all other attributes is text.

We assume the following foreign key constraints:

- pid is a foreign key in worksFor referencing the primary key pid in Person;
- cname is a foreign key in worksFor referencing the primary key cname in Company;
- cname is a foreign key in companyLocation referencing the primary key cname in Company;
- pid is a foreign key in personSkill referencing the primary key pid in Person;
- skill is a foreign key in personSkill referencing the primary key skill in Skill;
- eid is a foreign key in hasManager referencing the primary key pid in Person;
- mid is a foreign key in hasManager referencing the primary key pid in Person;
- pid1 is a foreign key in Knows referencing the primary key pid in Person;
- pid2 is a foreign key in Knows referencing the primary key pid in Person.

The file Assignment4Script.sql contains the data supplied for this assignment.

2 Optimizing RA expressions

In this section, you are asked to use the RA expressions obtained by translation in your previous assignment(**Assignment 3**).

You are required to show the intermediate steps that you took during the optimization.

You can use the following notation to denote relation names in RA expressions:

P, P_1, $P_2, \cdots$	`Person`
C, C_1, $C_2, \cdots$	`Company`
S, S_1, $S_2, \cdots$	`Skill`
W, W_1, $W_2, \cdots$	`worksFor`
cL, cL_1, $cL_2, \cdots$	`companyLocation`
pS, pS_1, $pS_2, \cdots$	`personSkill`
hM, hM_1, $hM_2, \cdots$	`hasManager`
K, K_1, $K_2, \cdots$	`Knows`

Note: Please make note of the following example, and use it as a template to construct your answers for this section. You should write all the steps and RA expressions in Latex or a word editor. Images of handwritten notes will NOT be accepted.

Example 1 *Consider the query* "Find each (p, c) pair where p is the pid of a person who works for a company c located in Bloomington and whose salary is the lowest among the salaries of persons who work for that company.

A possible formulation of this query in Pure SQL is

```
select w.pid, w.cname
from   worksfor w
where  w.cname in  (select cl.cname
                    from   companyLocation cl
                    where  cl.city = 'Bloomington') and
       w.salary <= ALL (select w1.salary
                        from   worksfor w1
                        where  w1.cname = w.cname);
```

The RA expression for the pure SQL query(that you have from Assignment 3) in standard notation is as follows:

$$\pi_{W.pid,W.cname}(\mathbf{E} \cap (W - \mathbf{F}))$$

where

$$\mathbf{E} = \pi_{W.*}(W \bowtie \sigma_{city=\textbf{Bloomington}}(cL))$$

and

$$\mathbf{F} = \pi_{W.*}(W \ltimes_{W.salary>W_1.salary \wedge W_1.cname=W.cname} W_1).$$

Note: You have to start directly with the RA expression(of the Pure SQL query) for the following questions, and show the optimization steps. Finally, the last step should be the optimized RA expression.

Step 1 *Observe the expression* $\mathbf{E} \cap (W - \mathbf{F})$. *This expression is equivalent with* $(\mathbf{E} \cap W) - \mathbf{F}$. *Then observe that, in this case,* $\mathbf{E} \subseteq W$. *Therefore* $\mathbf{E} \cap W = \mathbf{E}$, *and therefore* $\mathbf{E} \cap (W - \mathbf{F})$ *can be replaced by* $\mathbf{E} - \mathbf{F}$. *So the expression for the query becomes*

$$\pi_{W.pid,W.cname}(\mathbf{E} - \mathbf{F}).$$

Step 2 *We now concentrate on the expression*

$$\mathbf{E} = \pi_{W.*}(W \bowtie \sigma_{city=\textbf{Bloomington}}(cL)).$$

We can push the projection over the join and get

$$\pi_{W.*}(W \bowtie \pi_{cname}(\sigma_{city=\textbf{Bloomington}}(cL))).$$

Which further simplifies to

$$W \ltimes \sigma_{city=\textbf{Bloomington}}(cL).$$

We will call this expression $\mathbf{E}^{opt}$.

Step 3 *We now concentrate on the expression*

$$\mathbf{F} = \pi_{W.*}(W \ltimes_{W.salary>W_1.salary \wedge W_1.cname=W.cname} W_1).$$

We can push the projection over the join and get the expression

$$\pi_{W.*}(W \ltimes_{W.salary>W_1.salary \wedge W_1.cname=W.cname} \pi_{W_1.cname,W_1.salary}(W_1)).$$

We will call this expression $\mathbf{F}^{opt}$.

Therefore, the fully optimized RA expression is

$$\pi_{W.pid,W.cname}(\mathbf{E}^{opt} - \mathbf{F}^{opt}).$$

I.e.,

$$\pi_{W.pid,W.cname}(W \ltimes \sigma_{city=\textbf{Bloomington}}(cL) - \pi_{W.*}(W \bowtie_{W.salary>W_1.salary \wedge W_1.cname=W.cname} \pi_{W_1.cname,W_1.salary}(W_1))).$$

1. "*Find the pid of each person that knows at least 2 people, such that at least 1 of them works at Apple or Netflix* ."

   Optimize the RA Expression that you came up with for the above query in Assignment 3 and mention at-least 2 conceptually different rewrite rules you used. [`10 pts`]

2. "*Return the the pair (p, c) where p is the pid of a person, and c is the cname of the company where p works, such that (1) p is managed by someone who has at-least 2 skills and (2) p does not know anyone that lives in Seattle*."

   Optimize the RA Expression that you came up with for the above query in Assignment 3 and mention at-least 2 conceptually different rewrite rules you used. [`10 pts`]

3. "*Return each skill that is the skill of at least 2 persons, such that at least 1 of them lives in Bloomington*"

   Optimize the RA Expression that you came up with for the above query in Assignment 3 and mention at-least 2 conceptually different rewrite rules you used. [`10 pts`]

4. "*Return the pair (p, s) where p is the pid of a person that works at a company headquartered in MountainView and s is the minimum salary among all people that know p*."

   Optimize the RA Expression that you came up with for the above query in Assignment 3 and mention at-least 2 conceptually different rewrite rules you used. [`10 pts`]

5. "*Return each cname such that*
   (1) at least 1 person working there has the OperatingSystems skill
   (2) at least 2 persons working there live in different cities"

Optimize the RA Expression that you came up with for the above query in Assignment 3 and mention at-least 2 conceptually different rewrite rules you used. [`10 pts`]

3 Solving queries using Aggregate Functions

Formulate the following queries in SQL. You must use aggregate functions in ALL these queries and must not use set predicates where it is mentioned explicitly. You can use views, temporary views, parameterized views, and user-defined functions and expressions.

6. Find each pair (c, p) where c is the city and p is the pid of the person that lives in c, and earns the lowest salary among all persons living in c. You must not use set predicates in this query. [10 pts]

7. Let p1 be a person and N be the set of skills of Netflix employees. Find the pid and name of each person p1 if p1 has less then 2 of the skills in N i.e. the combined set of job skills of persons who work for Netflix.

$$\{s \mid s \textit{ is a jobskill of an employee of } Netflix\ \}$$

[10 pts]

8. Find each pid of a person who knows at least two people who (a) work for Apple and (b) who make less than 60000.You must not use set predicates in this query. [10 pts]

4 Queries with quantifiers

Use the idea of Venn diagrams to write SQL queries for the following queries with quantifiers.
To get full credit in these problems, you must write appropriate views and parameterized views for the sets A and B that occur in the Venn diagram with conditions for these queries. (See the lecture on Queries with Quantifiers.)

Make the following query using the COUNT function:

9. Find the pid and name of each person who knows at least 3 people who each have at most 2 managers. [`10 pts`]

10. Find the cname of each company that employs an even number of persons who have at least 2 skills. [`10 pts`]