CSCI-B 561 ADVANCED DATABASE CONCEPTS
Assignment 2
Fall 2015
Indiana University,
Bloomington, IN

Ramprasad Bommaganty
rbommaga@indiana.edu

21th September 2015

Solutions

1. **Working in a company**

   Consider the relational schema given below, where eId of the relation dependent is a foreign key referring to empId of the relation employee. Assume that every employee has at least one associated dependent in the dependent relation.

   employee (empId,empName,empAge)
   dependent (depID,depName,depAge)

   (a) What does the following relational algebra query return

   $$\pi_{empId}(employee) - \pi empId(employee \bowtie_{(empId=eID)\wedge(empAge\leq depAge)} dependent)$$

   Answer: The query returns the Employee IDs of those employees whose age is always greater than that of their dependents. Since, it is assumed that each employee has at least one associated dependent, the case of the query returning employees who do not have a dependent whose age is less than theirs does not arise.

   (b) What is the optimized version(using least operations) of the relational algebra expression

   $$\pi_{A1}(\pi_{A2}(\sigma_{F1}(\sigma_{F2}(r))))$$

   where A1,A2 are sets of attributes in r with A1 $\subset$ A2, and F1, F2 are Boolean expressions based on the attributes in r?
   Answer:

   $$\pi_{A2}(\sigma_{F1\ AND\ F2}(r))$$

   is the optimized version of the given relational algebraic expression.

2. **Studying in SOIC**
Consider the following RDB schema: **Students**(sid,name,gender,age,dept) **Courses**(cid,name,credit) **Take**(sid,cid,term,year,grade)
The grade will be letters. e.g A=,A,....
GradePoints(grade,points)
This table maps a letter grade to a point in the scale of 0-4. For example, (A-, 3.7) is a tuple in it.

Write quries in relational calculus(RC).

   (a) Find the students who have taken at least one course from the theory area (B501, B502, or B503 as names) and passed, i.e., the grade is not 'F'.
   Answer:
   $Q(x) = \{S|\exists S \in Students\ \exists C \in Courses\ \exists T1 \in Take\ \exists T2 \in Take\ \exists T3 \in Take(S.sid = T1.sid \wedge C.cid = T1.cid \wedge (T1.cid \neq T2.cid \neq T3.cid) \wedge (T1.sid = T2.sid = T3.sid) \wedge C.name =' B501' \vee C.name =' B502' \vee C.name =' B503') \wedge T.grade \neq' F'\}$

   (b) List all the departments that for the first time have students who are taking 'B561'. (They never had any student taking 'B561' before the current semester but have students taking it now, i.e., in term 'FALL' and year 2015.)
   Answer:
   $Q(x) = \{P|\forall S \in Students\ \exists C \in Courses\ \exists T \in Take(C.name =' B561' \wedge T.term =' FALL' \wedge T.year =' 2015' \Longrightarrow (S.sid = T.sid \wedge C.cid = T.cid \wedge P.dept = S.dept))\}$

3. **Social in Clubs**
Consider the RDB schema below:

Likes(customer, activity) **Often**(customer,club) **Provide**(club, activity)
Write queries in relational calculus(RC), datalog and SQl as required.

   (a) Find customers who often go to some clubs('often' means there is a corresponding tuple in table Often) that provide some of their liked activities.
   Answer:
   Relational Calculus:
   $Q(x) = \exists y \exists z\ Often(x, y) \wedge Provides(y, z) \wedge Likes(x, z)$

   Datalog:
   $H(x, y) : -Provide(y, z) \wedge Likes(x, z)$
   $Q(x) : -Often(x, y), H(x, y)$

   SQL:
   SELECT DISTINCT O.Customer FROM Often O WHERE (SELECT * FROM Provides P, Likes L WHERE P.activity = L.activity)

   (b) Find customers who don't like some activities so that they don't often go to clubs where they can nd those activities.
   Answer:
   Relational Calculus: $Q(x) = \exists z \exists x \exists y\ Provide(z, x) \wedge \sim (Likes(x, y) \vee Often(x, z))$

   Datalog: $H(x) : -Likes(x, y) \vee Often(x, z)$
   $Q(x) : -Provide(x, y) \wedge \sim H(x)$

References

1. Database Systems, the complete book- Ullman.
2. Database Management Systems, Ramakrishnan
3. http://pages.cs.wisc.edu/ dbbook/openAccess/thirdEdition/slides/slides3ed-english/$Ch4_Domain_Calculus.pdf http://www.cs.princeton.edu/courses/archive/fall08/cos597A/Notes/relational_5calc_topost.pdf$

4. http://www.cs.princeton.edu/courses/archive/fall13/cos597D/notes/relational$_c$*alc.pdf*