15-415: Database Applications
Problem Solving Assignment 2

School of Computer Science
Carnegie Mellon University, Qatar
Spring 2014

Assigned Date: January 28, 2014
Due Date: February 07, 2014 by 12 Midnight

I. Relational Algebra [10 points]:

Since user-centered recommendation systems are gaining unprecedented popularity, it is a high time to create a "CMUQ venue recommendation system" which will record the preferences of current students for various recreational places in the Education City. The ultimate goal is to recommend venues to new students based on the similarity of their preferences to more senior students. We start with the following relation schemas:

PLACES (*category: string, bname: string, popularity: integer)*

- category - category of the venue (e.g., "Cafe", "Library").
- bname – building where the venue is located (e.g., "LAS", "CMUQ").
- popularity - popularity score of the place (between 1 and 10).

Table 1 presents an instance of PLACES. Note that a place is identified by its building and category.

STUDENT_LIKES *(sname: string, category: string)*

- sname - student's full name (e.g., Hubbly Bubbly). You may assume that every name in the database is unique.
- category - category of the venue of interest (e.g., "Cafe," "Library").

Table 2 presents an instance of STUDENT_LIKES.

Q1.1 What does the following relational algebra expression output? [2 points]

$$\pi_{bname}(\sigma_{popularity>5}(PLACES))$$

a. Names of buildings in which all corresponding places have popularity scores greater than 5.

b. Names of buildings whose corresponding places have at most one category with a popularity score greater than 5.
c. Names of buildings whose corresponding places have only one category with a popularity score greater than 5.
d. None of the above (hence, write-down your answer).

Q1.2 Write a relational algebra expression which returns all student names who have exactly one category of interest. [4 points]

Q1.3 Consider the following relational algebra expression and a relation instance of PLACES as shown in Table 1: [4 points]

$$(\pi_{category,bname}(PLACES)) \div (\pi_{bname}[\sigma_{category="Gym"}(PLACES)])$$

category	Bname	popularity
Café	Recreation Center	7
Library	LAS	5
Gym	Recreation Center	10
Library	CMUQ	8
Indoor Activity	Al-Awsaj	10
Gym	CMUQ	3
Bookstore	Student Center	5
Movies	Student Center	4
Café	CMUQ	4
Football Field	Al-Awsaj	10

Table 1: A PLACES relation instance *P*. A place is identified by its category and building, and has a popularity score.

Q1.3.1 What does the above expression entail (in English)? A possible answer is: "Select all (building, category) combinations that do not have a Gym". [1 point]

Q1.3.2 How many field(s) are (is) returned in the output? [1 point]

Q1.3.3 What are (is) the field(s) returned in the output? [1 point]

Q1.3.4 How many record(s) are (is) returned in the output? [1/2 point]

Q1.3.5 What are (is) the record(s) returned in the output? [1/2 point]

II. Tuple Relational Calculus (TRC) [15 points]:

For this question, we will consider the same relation schemas used in Q I.

Q2.1 Write a TRC expression that selects all the records from STUDENT_LIKES who like the "Gym". [1 points]

Q2.2 Consider the following TRC expression and an instance *S* of STUDENT_LIKES as shown in Table 2: [7 points]

$$\{S \mid S \in STUDENT_{LIKES} \wedge \exists S' \in STUDENT\_LIKES(S'.category = S.category \wedge S'.sname = 'Adrian\ McCarthy' \wedge S.sname \neq' Adrian\ McCarthy')\}$$

sname	category
Oliver Stark	Café
Noah Morgan	Gym
Fabiana Dallas	Gym
Timmy Jones	Library
Timmy Jones	Indoor Activity
Adrian McCarthy	Library
Adam Harrison	Café

Table 2: A STUDENT_LIKES relation instance *S*.

Q2.2.1 What does the above expression entail (in English)? A possible answer is: "Select (Adrian McCarthy, category) if he likes more than one category". [4 points]

Q2.2.2 How many field(s) are (is) returned in the output? [1/2 point]

Q2.2.3 What are (is) the field(s) returned in the output? [1/2 point]

Q2.2.4 How many record(s) are (is) returned in the output? [1 point]

Q2.2.5 What are (is) the record(s) returned in the output? [1 point]

Q2.3 Write a TRC expression that returns all categories that are liked by at most two Students. [7 points]

III. Domain Relational Calculus (DRC) [20 points]:

Consider the instances shown in Tables 1 and 2 and then answer the following:

Q3.1 Express Q2.1 as a DRC expression. [1 points]

Q3.2 Express Q2.2 as a DRC expression. [1 points]

Q3.3 Consider a freshman student, *Xau Yang*, who is new to Education City and would like to use CMUQ's recommendation system. Write DRC expressions that help achieve the following tasks: [18 points]

Q3.3.1 Since Xau is a new user, he is prompted to enter his categories of interest, and new entry(s) are (is) created in the STUDENT_LIKES table.

The recommender system works as follows: For each of Xau's categories of interest, the system should return those places for which the popularity of the category is greater than or equal to 5, if it exists. [9 points]

Q3.3.2 Xau is very happy with the system but has been feeling lonely. He would, therefore, like to meet students who share common interests with him and the recommender system should return the names of those students. [9 points]

IV. Relational Algebra, TRC, DRC, and SQL [55 points]:

Moving from CMUQ's recommendation system, we shall now consider a simplified version of a re-known social application, Twitter!. On a high-level, Twitter works as follows:

- Users post tweets that are short pieces of text
- Users may tag their tweets with zero or more tags of their own choice. A tag must begin with the hash tag sign '#'. For example, a user tweeting about the Database Applications course may decide to tag the tweet with #DBApps #ROCKS
- Users may follow zero or more other users. The tweets of the former are visible to the latter.

Given the above Twitter's description, we define the following relation schemas:

- Users(uname, city, street) – you may assume that uname is unique
- Follows(uname1, uname2) – user with uname1 follows user with uname2
- Tweets(tid, t_tittle, t_text) – tweet with tid has title t_title and content t_text
- UserTweets(uname, tid, ts) – user uname posted a tweet with tid at time ts
- TweetTags(tid, tag) – tweet with tid has tag in its list of tags.

We now would like to extract some useful information from the database and we leave this job to our database expert (you!). For each of the following questions, write (a) a relational algebra expression, (b) an RTC expression, (c) an RDC expression, and (d) an SQL query. State the reason clearly if an expression and/or query cannot be expressed.

Q4.1 Find all users (uname) who posted a tweet with tag #DBApps. [9 points]

Q4.2 Find all distinct tags ever used since the launch of Twitter. [9 points]

Q4.3 Considering a particular user 'Donald Trump', find all distinct tags of all tweets by users whom Donald follows. In other words, find Donald's reading interests. [9 points]

Q4.4 Find all users (uname and city) who follow users who follow user 'Donald Trump'. [9 points]

Q4.5 Find all users who read about IPhone 5 before its launch (i.e., before Sept 12, 2012). [9 points]

Q4.6 Find all users (uname and city) who follow at least everyone that user 'Donald Trump' follows. [10 points]