Virginia Tech. **CS 4604 – Introduction to DBMS**
Computer Science **Spring 2013, Prakash**

Homework 2: More RA and SQL

(due February 27th, 2013, 9:05am, in class—hard-copy please)

Reminders:

a. Out of 100 points.
b. Rough time-estimates: ~3-5 hours.
c. Please type your answers. Illegible handwriting may get no points, at the discretion of the grader. Only drawings may be hand-drawn, as long as they are neat and legible.
d. There could be more than one correct answer. We shall accept them all.
e. Whenever you are making an assumption, please state it clearly.

Q1: GPAs [20 points]

Suppose you are given a relation `Grades(PID, GPA)` that lists a student's PID and GPA. Write a SQL query to find the GPA that occurs most often in the relation. If multiple GPAs occur equally often, your query should output all of them.

Q2: Social Network Friends [25 points]

We are in-charge of an online social network MyBook. Consider the relation `MyBookFriends(Id, FriendId)`, which is a giant table for each user on MyBook. The relation `MyBookFriends` keeps track of all friends of that user. Together, the two attributes comprise the (only) key for this relation. Researchers studying social networks are interested in counting the number of people who have k friends, for every possible value of k.

(25 points) Write a SQL query that operates on `MyBookFriends` and returns a relation `Counts(NumFriends, NumIds)`. If this relation stores the tuple (k, l), then it means that there are l distinct users (Id values) in `MyBookFriends`, each of who has exactly k friends. *For no points: imagine you run this query on a real social network like Facebook, and then plot the values with k on the x-axis and l on the y-axis---what is the shape of the plot you expect? Uniform? Linear? Non-linear? Any other particular function?*

Hints

As you can imagine, you do not know beforehand all the different values of k (or l) that should appear in `Counts`. Hence your SQL query should be able to figure out all these values automatically and correctly.

Q3: Recommending Collaborators [30 points]

Consider ResearchLink, an online social network for researchers, which also helps one find possible collaborators. We are given a simple table with schema:

`ResInt (pid, topic)` - Person with `pid` has `topic` as a research interest

Clearly a person can have multiple research interests (like 'Databases' and 'Artificial Intelligence'). The goal of this question is to write a relational algebra query that gives us pairs of researchers with exactly the same research interests (as we can recommend them as collaborators). This query is tricky---so let's try to solve it by the following steps.

Warm-up

Q3.1. (5 points) Write a relational algebra query to find all person-person-interest triplets `(pid1, pid2, topic)` such that person `pid1` has `topic` as a research interest, but person `pid2` doesn't. Call this the `PPT` view.

Full query

Q3.2. (10 points) Given the earlier view `PPT`, write the full relational algebra expression to find all pairs of people with the exact same set of research interests. Remove mirror pairs and self-pairs.

Verification with SQL

Q3.3. (10 points) Give the equivalent SQL query of Q3.2. (include the SQL query in hard copy) Again, make sure you remove mirror pairs and self-pairs.

Q3.4. (3 points) Run the SQL query of Q3.3 on the SQLite3 sample database at http://courses.cs.vt.edu/~cs4604/Spring13/homeworks/hw2/cs4604-hw2.db and paste the output.

Q3.5. (2 points) Make a query-hw2.txt file such that running it on SQLite3 gives the output of Q3.3 i.e. doing the following

```
your-machine% sqlite3 cs4604-hw2.db < query-hw2.txt
```

returns the desired pairs. Please e-mail the file to Qianzhou (qiand12 AT vt.edu) and make sure you use subject *CS4604 HW2 Query*.

Hints

Sanity check, for the database file: running the command

```
your-machine% sqlite3 cs4604-hw2.db 'select count(*) from ResInt;'
```

should give

```
30
```

Q4: Crypt-arithmetic [25 points]

This exercise is designed to help you think out of the box on the use of database programming for solving problems. You are given the crypt-arithmetic puzzle:

```
   SEND
+  MORE
  -----
  MONEY
```

The goal of the puzzle is to substitute numbers (from zero to nine) for letters, so that the addition works out. There are some constraints your solution should respect:

1. The same number should be used for a given letter, throughout. For example, if you guess "5" for the letter E, then E should get the value "5" at all the places it occurs.
2. Different letters should get different numbers, e.g., you cannot assign "4" to both E and to M.
3. None of the numbers SEND, MORE, or MONEY have any leading zeroes, i.e., they do not begin with a sequence of zeroes.

Explain how you will solve this puzzle by creating database tables and writing a query.

Q4.1. (5 points) The schema of the tables you use.

Q4.2. (10 points) Your SQL query.

Q4.3. (10 points) The solution you get for the puzzle when you use an SQL interpreter and RDBMS to solve this puzzle. Copy-paste the output you get.

Hints

The SQL query may be quite long so you may find it useful to create the query in a text file and use the source command (or equivalent) in your SQL interpreter to read in and execute the query.