Carnegie Mellon University
15-826 – Multimedia Databases and Data Mining
Fall 2014, C. Faloutsos
Homework 0,
Due Date: Sep 23rd, at classroom 3:00pm
Prepared by: Jay-Yoon Lee

Reminders

- All homeworks are to be done **INDIVIDUALLY**.
- For code submission to blackboard submit a single file ([andrew-id]-hw0.tar.gz, e.g. jaylee-hw0.tar.gz) that contains all the codes for the questions wihtout the data. Write the code assuming that the data folder and src folder are in the same folder. (e.g. HW0/data/, HW0/src)
- Expected effort for this homework (approximate times):
  - 2-6 hours
    - 1-4 hours to write and debug all sql queries
    - 1-2 hours to run the queries and record the answers.

DBMS and SQL [100pts]

Problem Description: For each question in this part, provide both the SQL statement(s) and the resulting answer(s), unless specified otherwise. You'll be working with *Marvel dataset* which lists the appearances of Marvel characters in comic books. [1] [2] [3]
Hint: Please use SQLite3; version 3.6.20 is already on the andrew cluster machines. You may use your own machine and your own sqlite3 installation, as long as your submitted code runs correctly on the andrew cluster machines.

Implementation Details: Write sql code for the following queries.

1. **[0 pt]** *Data preparation*

   - Download the Marvel-data.tar.gz[2] from `https://www.andrew.cmu.edu/~jaylee/Marvel-data.tar.gz` which consists of
     - `marvel.txt:` characterID (1st column) and comcID (2nd column).
     - `marvelCharacters.txt:` charcterID and the name of the character.
     - `marvelComicBooks.txt:` comicID and the name of the comic book.
   - Also download the makefile folder at `http://www.cs.cmu.edu/~christos/courses/826.F14/HOMEWORKS/HW0/makefile-hw0.tar.gz`.
     Implement .sql files inside the folder so that when you hit make, the answers can be printed.

[1] Marvel data reference: `http://bioinfo.uib.es/~joemiro/marvel.html`.
[2] You can download the data (`Marvel-data.tar.gz`) available at `https://www.andrew.cmu.edu/~jaylee/Marvel-data.tar.gz`
[3] make file: `http://www.cs.cmu.edu/~christos/courses/826.F14/HOMEWORKS/HW0/makefile-hw0.tar.gz`

Print answers for question 2, 3 on one sheet of paper with '[course-id] [hw#] [question#] [andrew-id] [your name]'

2. **[10 pt]** *Create Database and Count*
   Create and load the following table using the extracted text file.

   - `marvel(charId INTEGER, comicId INTEGER)`
   - `marvelchar(charId INTEGER, charName VARCHAR)`
   - `marvelcomics(comicId INTEGER, comicName VARCHAR )`

   Report the number of rows for each table using hw0.2.sql in the order of marvel, marvelchar, marvelcomics. Check if the number of characters and comics match the number of lines the corresponding text files.
   (Hint1: check `.help` and `.import` in SQLite3)
   (Hint2: To check the number of line in text file, use $\mathsf{wc} - \mathsf{l}$ code.)

3. **[10 pt]** *Distinct elements in a Table* Using `hw0.2.sql`
   (a) Get the distinct number of characters form 'marvel' databse and check with your previous answer. **[5 pt]**
   (b) Get the distinct number of comic books form 'marvel' databse and check with your previous answer.**[5 pt]**

   (Hint: DISTINCT() command)

Print answers for question 4, 5 on one sheet of paper with '[course-id] [hw#] [question#] [andrew-id] [your name]'

4. **[15 pt]** *The most popular character A*
   Find character A who appeared most frequently over all comic books?
5. **[20 pt]** *Popularity distribution of characters* Our objective is to plot the popularity distribution of comic characters appearance in comic books. For that we define few terminologies:

   - p, `the popularity of characters:` the number of comic books that each characters appeared in.
   - f, `the frequency of popularity` p`:` count of comic characters that has the popularity p.

   Attach the plot the f vs. p in log-log scale (y-axis: f and x-axis p should both be in log scale). `Example:` If 'superman' appeared on 1000 comics, p for superman should be 1000. And if 'superman', 'spiderman', 'wolverine' are only characters that has $p = 1000$, then $p = 1000$ has $f = 3$.

Print answers for question 6, 7 on one sheet of paper with '[course-id] [hw#] [question#] [andrew-id] [your name]'

6. **[20 pt]** *Count of pairs*
Count the coappearing pairs in comic books excluding self-pairs, and mirror-pairs. (*Hint:* 1: Use condition `node1>node2`, a usual trick to exclude self-pairs and mirror pairs; *Hint:* 2: Join is better than nested select; *Hint:* 3: Avoid creating an intermediate table of pairs - it may be too slow).
`Example:` Excluding the mirror pairs and self pairs, Table 1 has 4 count of pairs total.

char1	char2	comicBook	pair type
Spiderman	Superman	sh1	
Spiderman	Superman	sh2	
Superman	Wolverine	sh3	
Captain America	Thor	sh4	
Wolverine	Superman	sh3	mirror pair with row 3
Spiderman	Spiderman	sh1	self pair

Table 1: Example of pairs, self-pairs and mirror-pairs.

7. **[25 pt]** *Query Optimization*
   (a) **[4 pt]** *Report* the wall-clock running time of your query for the question 6, using, e.g., the `time` Linux/Unix command.
   (b) **[8 pt]** *Index impact: Report* the wall-clock time of your query for the question 6, again with indices on the column `charID`.
   (c) **[8 pt]** *Index impact: Report* the wall-clock time of your query for the question 6 again with indices on the joint column `comicID` .
   (follow the syntax shown here).
   (d) **[5 pt]** *Query optimization:* Use the `explain select` for the last three sub-questions, i.e., with, and without indices. Based on the `explain select` and time you measured, select fastest query for question 6.
      i. No index
      ii. Join with index on charId
      iii. Join with index on comicId

   Hint: See `http://www.sqlite.org/draft/eqp.html` for a (rough) description of the output of `explain`.

What to turn in:

- **Code:**
  Submit a tar file to blackboard in a file name `[andrew-id]-hw0.tar.gz` (without the data such as marvel.db or Marve-data/), with all the SQL queries and commands you need to answer the questions above. Make sure it runs: we will grade it using

  ```
  make all
  ```

  (organize your code files following the provided makefile-hw0 folder.)
- **Answers:**
  Submit hardcopy at classroom that contains **1) question number**, **2) answers**, and **3) SQL queries**. After finishing up, `make all` will provide you all answers except plot for question 5. For grading purpose, please group answers in pairs of (2,3), (4,5), and (6,7) respectively. `Make sure you put '[course-id] [hw#] [question#] [andrew-id] [your name]' on top of every page you submit.`