Databasesystems 2
Forum: https://forum-db.informatik.uni-tuebingen.de/c/ss18-db2

Assignment 1 (24.04.2018)

Submission: Tuesday, 01.05.2018, 10:00 AM

1. [0 Points] **Introduction**

   (a) **Announcement**: The second assignment will be released next week Tuesday (01.05.2018).

   (b) **Before we are able to grade your team**, you have to complete one initial task first. In your team's GitHub Classroom repository, there exists a file called `README.md`. Add each team member's name, surname, *Matrikelnummer*, subject, field of study, forum username and student e-mail to the file and commit+push the changes.
   This task is not part of your submission and does not grant you any points, but is a **requirement** for your team to be graded in the first place. Please make sure the file always exists with the correct information present.

   (c) Each assignment submissions in "Datenbankensysteme 2" must be placed into a separate folder in the `root` directory of your team's GitHub Classroom repository. The name of the folder has to be in the format `solution<number>`. For this assignment, for example, your submission folder would be called `solution01`.

   (d) In general, the only accepted file formats are PDF of appropriate size (< 2MB), plain text files (.txt) and source files (.sql, .c, ...). Other formats may not be graded, unless stated otherwise. Your submitted code has to compile and work. If compiling, running and/or understanding your source code is paticulary complex, please write documentation accordingly.

   (e) Lastly, the usual rules for plagiarism and other academic integrity apply.

   (f) For any further questions about this lecture, assignment and other related topics, visit the forum at

   https://forum-db.informatik.uni-tuebingen.de/c/ss18-db2.

2. [20 Points] **FizzBuzz**
   In the lectures, we introduced `generate_series(...)` for both *PostgreSQL* and *MonetDB*. Solve the following tasks in (1) *PostgreSQL* and (2) *MonetDB*.

   (a) Write a simple *SQL* query which produces a table with a single column of type TEXT.
       Use `generate_series(...)` to generate numbers $i \in \{1, 2, \ldots, 100\}$. The resulting table should contain each i (as a string). However, note the two following exceptions:
       Whenever i is divisible by 3, the table should contain 'FIZZ' instead of i. Whenever i is divisble by 5, the table should contain 'BUZZ' instead of i. Lastly, if i is divisible by both 3 and 5, the table should contain 'FIZZBUZZ' instead of i.
       Order the output so that you obtain a result as shown here:

   ```
   '1'
   '2'
   'FIZZ'
     ⋮
   '14'
   'FIZZBUZZ'
   '16'
     ⋮
   '98'
   'FIZZ'
   'BUZZ'
   ```

   Note: Do **not** use user-defined functions or common table expressions as part of your solution.

   (b) Create a table `fizzbuzz(v text)` then populate it with the result of the query of (a).

   (c) Will the result of the following query **always** return the exact same result as the query in (a)?

   ```
   SELECT * FROM fizzbuzz;
   ```

   Explain briefly.

   (d) Table `fizzbuzz` is now persistently stored on your disk as a regular file. Name the path of this file on your system and describe the way you found it briefly.
       Consider the system catalogs table `pg_class`[1] in *PostgreSQL* and the *MAL* function `bat.info()` in *MonetDB*.

3. [10 Points] **Memory Scan**
   For this assignment, you are given the file `transfer.c` which has been introduced in the lecture. This C program measures the time needed to access the same chunk of memory (8 MB) over and over until the total amount of memory used equals 4 GB.
   Now, lower the size of the chunk of memory (see macro `SCANSIZE`). Each time you do, determine the time your program needs to complete and write it down.

   Explain how you can use these experiments to determine the size of your Level 1, Level 2 (and Level 3, if any) data caches of your CPU. Can you determine the cache level sizes on your system? What cache level sizes do your measurements suggest for your particular CPU?

---

[1] https://www.postgresql.org/docs/current/static/catalog-pg-class.html