CSC 370 — Database Systems
Assignment No. 3

Note 1 **This assignment is to be done individually.**

A note on Academic Integrity and Plagiarim

Please review the following documents:

- Standards for Professional Behaviour, Faculty of Engineering:
  `https://www.uvic.ca/engineering/assets/docs/professional-behaviour.pdf`
- Policies Academic Integrity, UVic:
  `https://www.uvic.ca/students/academics/academic-integrity/`
- Uvic's Calendar section on Plagirism:
  `https://www.uvic.ca/calendar/undergrad/index.php#/policy/Sk_0xsM_V`

Note specifically:

> Plagiarism
> Single or multiple instances of inadequate attribution of sources should result in a failing grade for the work. A largely or fully plagiarized piece of work should result in a grade of F for the course.

A program you submit will be considered a **piece of work**. You are responsible for your own submission, but you could also be responsible if somebody plagiarizes your submission.

- This assignment is worth 1% of your total course mark.

Objectives

We made it! This is the last assignment of the term. After completing this assignment, you will have experience:

- Learn how the DBMS estimates the number of tuples of a result.

Knowing how to estimate the number of matching tuples of a query is a requirement to be able to know if and index is to be used.

Preliminaries

Before the DBMS computes a query it needs to estimate the number of tuples that will be returned. For a select of one table with a WHERE clause, this is done by computing the probability that any tuple t satisfies the WHERE clause. This probability is known as the selectivity of the clause. The selectivity of the clause is then multiplied by the number of tuples and the result is the expected number of tuples that the query returns. For example, given the query:

```
select * from productions where year = 2010;
```

`EXPLAIN` returns the following information:

```
[local]:ubuntu@imdb=# explain select * from persons where personname = 'Ryan Reynolds';
                                  QUERY PLAN
------------------------------------------------------------------------------
 Index Scan using idxpersonsname on persons  (cost=0.56..24.64 rows=5 width=32)
   Index Cond: (personname = 'Ryan Reynolds'::text)
(2 rows)
```

Note the number of rows: 5. This number is computed by multiplyng the selectivity of (personname = 'Ryan Reynolds') by the number of tuples in the relation (rounded to closest integer, the number of rows to return). Estimating that the query will return 5 tuples, the DBMS decides to use the index as the access path of this query.

Your task, should you choose to accept it

For this assignment use the database imdb.

Part A

Assume the following queries.

1. `select * from  productions where year IS NULL;`
2. `select * from  productions where year = 2014;`
3. `select * from productions where year > 1990 and year <=1992`
4. `select * from  productions where year IS NULL and year = 2014;`

For each of these queries:

1. Using EXPLAIN ANALYZE, record the number of tuples that postgres estimates each query will return and the actual number of tuples returned.
2. Compute, for each of the 4 queries:
   (a) Using **only the information in pg_stats and pg_class** compute the selectivity of the where clause
   (b) using this selectivity, compute the expected number of matching tuples

   Show all your work.

Part B

For the following query:

```
explain select * from productions where year > 1980 and year < 1985;
```

There is an index on year (b+tree, dense). Why does this query ignore the index and instead uses a sequential scan of the heap? Answer this question with one paragraph (there is no need to do calculations).

Hints:

- Do not plagiarize. You can talk to other students about what you are doing, but you are not allowed to share any code.
- Use section 68.1 of postgresql 10 manual:
  `https://www.postgresql.org/docs/10/row-estimation-examples.html`
  Yes, it is a bit outdated, but the information is valid. As of version 10, postgresql uses 100 buckets, not 10; it also stores 100 most frequent values, not 10.
- You only need to use pg_stats and pg_class (this includes the number of tuples in the relation).
- Note that, with respect to a given attribute, tuples in the relation will be divided into three types: null values (the attribute is null), most-frequent-values (the attribute matches one of these values), and histogram (the attribute is not null and does not match any most-frequent value). They are mutually exclusive; in other words, the counts in the histogram do not include null values nor most-frequent-values.
- For these questions, there is no need to sub-divide a bucket (if applicable).
- The selectivity of the conjunction of two clauses is the product of their selectivity.
- You results do not have to be identical to the estimation, but very close.

What to submit

See Brightspace for information on how to submit.