CS 4321/5321 Project 5 (Preview)

Fall 2016

Due December 9th, 11:30 AM

This project is out of 75 points and counts for 22% of your grade.

1 Goals and important points

In this last Project, you will bring together the work you did in Projects 3 and 4 and you will add optimization functionality to your interpreter.

- you will add code that gathers statistics about the data, in order to allow informed optimization.
- you will improve your query plans by pushing selections in a more thorough way than previously.
- you will implement an algorithm that chooses an efficient join order for each query rather than following the order in the `FROM` clause.
- you will implement algorithms/heuristics to choose the best implementation for each selection and each join operator in your improved query plan.

You do not have to do any further optimizations, such as pushing projections. The goal in this Project is not to develop a production-quality optimizer, but to focus on a subset of important optimization techniques.

You will still support the same subset of SQL as in the previous Projects and should follow the same specification for sorting (`ORDER BY`). In addition, if a query has a `SELECT *` and multiple tables in the `FROM` clause, you should return columns in the order specified by the `FROM` clause. For example in `SELECT * FROM R,S,T WHERE ...` return the columns of `R` followed by the columns of `S` followed by `T`. This ordering happened "naturally" in previous projects because your join tree followed the `FROM` clause; in this project you will sometimes change the join order so you will need to enforce this behavior explicitly for `SELECT *`.

Complete instructions for Project 5 will appear later.