CSE 177 / EECS 277 – DATABASE SYSTEMS IMPLEMENTATION

Project Description

This semester-long project requires the implementation of a single-user relational database system with a limited `SQL`-like language. The project is divided into five (5) stages that build on top of each other and follow the path of a `SQL` query from the user interface all the way down to the storage, and back through the relational algebra operators with the result displayed on the user's screen.

The five (5) stages of the project are:

1. Database Catalog (Week 5) **100 points**
2. Query Compiler (Week 8) **200 points**
3. Single-Table Relational Algebra Operators (Week 11) **200 points**
4. Join Operators (Week 13) **100 points**
5. Complete Database Server (Final exam) **200 points**

The project counts for **800 points** out of a total of **1000 points** available in the class. It is the single main factor in grading. There are only weekly quizzes beyond the project. As such, expect to spend the majority of your time in this class coding. This is a hands-on experience, targeted to familiarize you with the system-level code development process. However, we provide you with a large repository of sample code that implements the majority of the low-level functionality. We also designed the entire infrastructure of the system in order to make your coding targeted to the required parts. Essentially, almost everything is in place (from a design perspective). All you have to do is to plug-in your code in locations clearly identified by the requirements. Nonetheless, feel free to redesign and change anything you like/feel/need to.

CSE 177 teams consist of 4-5 students. For each stage of the project, one member is designated as the leader. Each student serves as the leader one time. The leader manages the coding for that stage and runs the presentation in the lab. However, this does not mean that the leader is the only one who implements the entire stage. The entire team has to participate in each stage and the final presentation. The same score is typically assigned to the entire team.

EECS 277 teams consist of one (1) student who has to do all the work.

Code

The coding is entirely in `C/C++` over a `Linux` environment. The reason for this is very simple: all the **real** systems are developed in `C/C++`. Familiarity with debuggers such as `gdb` and `valgrind` helps with coding, but is not a requirement. `SQLite` database is used for storing the metadata of the system. This is something everyone who took *CSE 111 Database Systems* is familiar with. `lex (flex)` and `yacc (bison)` are used to write the query compiler. We provide you the code that handles a subset of the `SQL` language. The enhancements you are required to develop are minimal.

Discussion & Sharing

This is a team project. As such, all the coding has to be done by every team individually. Nonetheless, teams are allowed to discuss the solutions and share their ideas, as long as each team implements their code individually. The motto is: *"Discuss as much as you like, but implement by yourself."* Feel free to use the CatCourse discussion and chat functionality. This helps the instructor/TA to see your discussions and even provide help, when necessary. Plagiarism-detection tools such as MOSS will be run on the committed code.

Chat GPT

The use of Chat GPT or other tools for code generation such as GitHub Copilot is allowed. In fact, we are curious what code is generated and what prompts are provided as input. However, remember that every stage has to be presented and demoed in-person to the instructor/TA. Therefore, you have to show that you fully understand the code and are able to answer any question about the code.

GitHub

The code is hosted on GitHub. In order to get access, you have to create an account on GitHub (if you do not already have one) and share your username with the instructor. A separate branch will be created for every team. Every team has to work exclusively on their branch. Students have to get familiar with the GitHub development process in order to actively participate in the code development with their team.