

# Project Timing Demo Notes

## Summer 2022 - CS 2341

These are some notes about how the Project Check In and Timing Demos will run:

1. All the projects will be built and run from Dr. Fontenot's computer.
2. Demos will happen after class on Monday.
3. What you need to have completed:
  - a. AVL Tree as index for words from documents
  - b. Document Parser.
    - i. This should include Stemming and Stop Word Removal
    - ii. You DO NOT need to have functionality for indexing orgs and persons turned on for the speed check.
  - c. Functionality to parse all documents in a given folder, path provided as a command line argument.
  - d. Functionality to query the AVL tree based on 1 single word provided as a command line argument. You will print the document IDs that contain the word searched for.
4. During or before the the demos, I will:
  - a. pull your code from github (may be done ahead of time, but after deadline)
  - b. build it using cmake (as if it were being built in clion) in **Release** mode (See below for more info) and the compiler will support up to and including c++17.
    - i. If you're using any 3rd party library (RapidJSON included), the source needs to be included in your repo.
  - c. execute the project on a small (~10) document data set as proof of concept.
  - d. execute the project on increasingly large data sets as time permits
  - e. Timing data will be gathered using the **time** command.

Your program should execute with exactly 2 command line arguments:

1. a search term
  - a. After parsing all the documents at the path indicated in the 2nd command line arg, return a list of all document IDs containing this word.
  - b. You need to stem this word before looking for it in your index.
2. full path of folder to search - your program should expect an **absolute path** to a folder containing a set of json documents to parse.
  - a. **Ignore any files in this folder that don't end in .json.**
  - b. This folder will contain other folders like the original data set.

Example:

```
./a.out pathogenesis /home/mark/classes/2341/dataset01
```

If your computer username contains a space, DO NOT change your username to accommodate the path above. Simply use finder or Windows Explorer to create a new folder outside of the User home directory folder. You might need admin privileges to do this.

## Debug vs Release Mode

When you build your programs normally, the executables include extra info to allow you to use the debugger to step through your code. Also, no compiler optimization of your code is performed. These things make the execution slow.

You can also build your project in Release Mode, which removes any debugging symbols in the executable as well as cranks up the optimization of the compiler. You can add Release Build to your build configuration options by going to Clion Preferences > Build, Execution, Deployment > CMake. Then in the pane on the right side of the preferences dialog, click the + sign under Profiles to add a Release Profile.

