Ctags and stags for Code Navigation

Quick Links: [Home (https://q.utoronto.ca/courses/419441/assignments/syllabus)](https://q.utoronto.ca/courses/419441/assignments/syllabus) [Lecture material (https://q.utoronto.ca/courses/419441/pages/lecture-material)](https://q.utoronto.ca/courses/419441/pages/lecture-material) [Lab assignments (https://q.utoronto.ca/courses/419441/pages/lab-assignments)](https://q.utoronto.ca/courses/419441/pages/lab-assignments) [Piazza Discussion ↪ (https://piazza.com/utoronto.ca/winter2026/ece353/home)](https://piazza.com/utoronto.ca/winter2026/ece353/home) This page brought to you by Prof. Ashvin Goel.

The idea of tags is to navigate around source code files by following symbols. This way you don't have to remember which file a given symbol is defined in. The instructions described below are for the `vim` and the `emacs` editors. (`emacs` is the better editor by far!:)

`vim` Instructions

For example, you might be in `test_thread.c`, looking at the variable `THREAD_NONE` being passed to the function `thread_yield`. You can look at how this variable is defined by typing `Ctrl-]`, which will take you to the `thread.h` header file. Then press `Ctrl-t` to go back.

Directions:

1. Generate a file called `tags` in the ~/ece353 directory. This file will contain information for all 'c' source and header files for the ECE353 labs. You will need to redo this step when you add new code to update the `tags` file.

   ```
   % cd ~/ece353
   % find . -name "*.[chS]" | xargs ctags
   ```

2. Add the following to `~/.vimrc`:

   ```
   set tagstack
   set tags=./tags,tags,$HOME/ece353/tags
   ```

   If you don't have an existing ~/.vimrc file, you will need to create it.

3. Use `Ctrl-]` to follow tags, and `Ctrl-t` to go back. Try ":help ctags" in vim for more info.

A few more suggestions:

- You can jump directly to a tag by typing ":t symbol" in command mode.
- When you jump to a tag, vim will sometimes tell you "tag 1 of 4 or more" (or something similar). In this case, typing ":tn" (n is for next) or ":tp" (previous) will move you along.

`emacs` Instructions

The usage is similar, except when the cursor is on a tag, you jump to the definition with `M-.` (meta and dot pressed simultaneously). The first time you try it, emacs will ask you about the location of the tag file. If you follow the instructions below, there will be a file named `TAGS` in `~/ece353`, so you can just select that directory and emacs will use the correct tags table.

In order to go back from where you came (upon typing `M-.`), use `M-*`.

If you have typed the first few characters of a symbol, `M-tab` will complete it from the tabs table, if there is a unique completion. This is only occasionally useful.

Directions:

1. Generate a file called `TAGS` in the current directory. This file will contain information for all 'c' source and header files, and assembly files. You will need to redo this step when you add new functions to update the `TAGS` file.

   ```
   % cd ~/ece353
   % find . -name "*.[chS]" | xargs etags
   ```

2. Use `M-.` to follow tags, and `M-*` to go back.

(BTW, emacs is the better editor by far!:)