

Advanced Homework 3

Bridging the Editor/IDE Gap

Assigned: Friday, January 22, 11:00AM

Due: Before the end of the final office hours this week (Thursday, ~5:30PM)

Submission Instructions

To receive credit for this assignment you will need to stop by someone's office hours, demo your running code, and answer some questions.

For this assignment, you may use vim or Emacs, whichever you are more comfortable with.

1 Some Tools that Support Other Tools

Integrated Development Environments (IDEs), such as Visual Studio, Xcode, or Eclipse provide many features beyond simple text editing.¹ Some of the most useful features rely on an understanding of C/C++ code, allowing you to jump from a function's use to its definition, inspect the types of complex structures, and assist authoring by suggesting possible completions.

Vim and Emacs both support rudimentary autocompletion out-of-the-box. Try editing an existing file and start typing a function you've used before, then hit `ctrl-n` and `ctrl-p` in vim or `M-/` in Emacs. This autocompletion has no understanding of C, it simply does string matching with any other strings anywhere else in the open files.²

There are two utilities, `ctags/etags` (vim/Emacs respectively) and `cscope` that can be integrated with your editor to provide some contextually aware editing assistance. `ctags/etags` works pretty much out of the box. Somewhat annoyingly, `cscope` requires you to generate a list of files for it to process. You may find that the `find` utility, and its filetype filters in particular, are very helpful for generating this file list.

Find a project of reasonable complexity,³ set up `ctags/etags` and `cscope`, and integrate them with your editor of choice.

Submission checkoff:

- Show off `ctags` usage
 - What did you have to do to generate tags for `ctags/etags`?
 - How often do you need to generate tags (after what changes do you need to regenerate tags)?
 - What did you have to do to integrate `ctags/etags` with your editor?
- Show off `cscope` usage
 - How did you generate a file list for `cscope`?
 - How often do you need to run `cscope`?
 - What did you have to do to integrate `cscope` with your editor?
- Explain the difference between `ctags` and `cscope`
 - Is one strictly better than the other or is it useful in some cases to set up both? Why?

¹ Interestingly, "simple text editing" is often what IDEs are worst at out-of-the-box. Hence part 3 of this assignment.

² Vim also understands how to parse C `#include` directives and Python `import` statements, and will search those files as well.

³ If you don't have a good project, `sqlite` ([zip of source](#)) is generally regarded as a well-engineered example of code. However, I recommend trying on your own code first.

2 Extending Editors

ctags/etags and cscope are programs that run independently of your text editor. You can also extend your editor with plugins. For vim, plugins are written in the custom language vimscript; Emacs itself is a Lisp interpreter, and plugins are written in Emacs Lisp.

While writing a plugin is a great way to learn a lot about the internals of your editor, it can be a steep learning curve. Fortunately, the Internet has an enormous library of really interesting and powerful plugins. As an example, there are plugins that solve the annoyance from part 1, that will automatically regenerate tags whenever needed.

Find at least three plugins to install in your editor. You may find it is easier to first install a plugin manager and then install plugins.

Submission checkoff:

- For each plugin:
 - What does this plugin do?
 - Show off how you use it
 - Why did you install this plugin?

3 Everything Old is New Again

While advanced IDEs and new text editors come out remarkably often, rarely are their simple text editing and navigation capabilities as good as the “classic” editors this homework has focused on. For this reason, there are plugins (or in some cases its built-in) to provide vim-like or Emacs-like editing modes for nearly every other editor.

Pick your favorite “advanced” GUI editor (suggestions Visual Studio, Xcode, Eclipse, Sublime, or Atom – gvim doesn’t count) and set it up to edit text in vim or Emacs mode. You will likely need to install a plugin to do this.

Submission checkoff:

- Show off your plugin in action