CST0006 – Computer Programming Foundations

INTRODUCTION to WEEK02

In this lecture and lab section we are going to learn how to use IDEs, find out what they are and why they're preferred over the old text-based way of building software.

We will also be spending a lot of time going over some of the Git GUI clients that are available. Even though understanding how to do this on the command line will set you apart from those that don't. It is important to see why using one would still be nice.

IDEs – Integrated Development Environments

An Integrated Development Environment is a collection of software development tools, integrated into one application that programmers can use to aid them in developing software. These software tools can consist of:

- Source code editor usually with built-in syntax highlighting and sometimes advanced code completion function, grammar checkers and spell checkers.
- Interpreter, compiler and/or build tools. These tools are translation tools that will convert your human-readable source code into a language that can be understood by the computer. These tools sometimes work as you type to warn you of grammar issues, syntax issues and spelling mistakes.
- Debugging tools. Many IDEs have debugging tools that helps you go though your code, stopping after each statement, so that you can see what is actually happening in memory as your code is running.
- Source Code Management Systems. Some IDEs can also have an integrated source code management system that you can chose to use. Some of these tools are just as nice as their independent counterparts. I find most of them to be very bulky or clunky and can slow down the IDE to almost painful states.

It is my strong opinion that an IDE should give you "just enough" functionality to be awesome, otherwise the IDE becomes a nuisance to use. There are some IDEs like Microsoft's Visual Studio that try to be everything and offer every feature, but it's painfully slow to use and installing it takes forever! I don't think I've ever hated a piece of software as much as Visual Studio. It gets in your way instead of being a tool that aids you in developing software. It's like any tool, if you spend the whole time focusing on the tool instead of the project, it's a bad tool.

Thonny

Thonny is an open source, cross-platform, Python IDE that sells itself as a “Python IDE for beginners”. However, it’s just powerful enough and has enough features for it to probably be one the best Python IDEs you’ll ever use. I’m a huge fan of simplicity, and find that Thonny is even more simple to use than Pyzo, which is another simplistic Python IDE.

Code::Blocks

Code::Blocks is an open source, cross-platform, C and C++ IDE. It has a brilliant way of organizing your code into development projects. It is a little confusing at first, but once you understand how to do this, you’ll wish all of the other IDEs did this too. Code::Blocks was my preferred C/C++ IDE while I was in school, and it continues to outshine the others due to its speed and simplicity.

GIT GUI Clients

Git GUI clients are a visual way to point and click your way though the tracking of changes to your source code. Most people prefer to use a GUI client to track these changes. In some ways they offer a few benefits to tracking changes, in others they don’t offer anything extra than what you get on the command line. We will be using both types of interfaces in this course, because knowing what each command does and how they work will make using the GUI simpler.

We will be downloading and testing two Git GUI Clients that are cross-platform. In no why am I suggesting that these two are the best, but due to their popularity we will be trying both

GitAhead

GitAhead claims to be “The elegant git gui for development teams”, which I agree with. It is one of the simplest GUI clients I’ve used. Like I said before, a tool should be exactly what you need without extra bloat. This GUI client offers that.

https://www.gitahead.com

GitKraken

GitKracken claims to be “The legendary Git GUI client for Windows, Mac and Linux”, which I also agree with. This GUI client has quickly become one of the best and most feature rich GUI clients on the market today. It offers a lot of features in a pretty interface.

https://www.gitkraken.com