

# Working with Code

UIC Linux Users Group

September 8, 2010

## editors

what you use

nano/pico

vim

eclipse

## compiling

g++

javac

makefiles

## debugging

introduction

tools

tips

## resources

resources

## beginner logic

- ▶ concept: programing is hard and scary. we should give beginners tools with limited capabilities
- ▶ problem: beginners don't want to stay beginners
- ▶ learn to use a decent IDE with a decent debugger, decent code prediction, syntax highlighting as soon as you are able

# nano

- ▶ familiar interface
- ▶ included in almost every linux distribution

## example

```
nano -w hello.txt
```

# vim

- ▶ Vim is an advanced text editor that seeks to provide the power of the de-facto Unix editor 'Vi', with a more complete feature set.
- ▶ Also included in almost every linux distribution

## vim basics

- ▶ arrows/mouse : move around the file
- ▶ i : insert a : append
- ▶ d : delete/cut
- ▶ y : yank/copy
- ▶ wq : save and quit
- ▶ w : save
- ▶ u : undo
- ▶ redo : ctrl+r

## more vim

- ▶ `sp` : splits the window
- ▶ `m[letter]` : assigns a macro
- ▶ `v` : visual mode
- ▶ `ctrl+v` : visual block
- ▶ `gD` : takes you to a variable/method declaration
- ▶ `ctrl+n` : auto-completion



## configuring vim

- ▶ set autoindent
- ▶ set smartindent
- ▶ set number : displays line numbers in the left margin

# eclipse

- ▶ advanced and powerful IDE (Integrated Development Environment)
- ▶ Primarily used for Java development, however can be used for C/C++, PHP, Android development as well

# g++

- ▶ `g++ myprogram.cpp -o myprogram`
- ▶ `./myprogram`

# javac

- ▶ `javac myprogram.java`
- ▶ `java -cp myprogram`

# introduction

- ▶ debugging is the process of identifying the root cause of an error and correcting it
- ▶ debugging can take up to 50 percent of development time and can be the hardest part
- ▶ forces you to read and understand your code

# tools

- ▶ gdb: (GNU Project Debugger)
- ▶ ddd: graphical front end for gdb
- ▶ compiler!!
- ▶ eclipse has a very good debugger for your java code

## tips

- ▶ try using a binary search algorithm to find the source of the error
- ▶ talk to someone about your problem a.k.a. confessional debugging
- ▶ take a break from the problem (go for a walk, get some coffee, etc)
- ▶ make sure to save the original source (version control?)
- ▶ well formatted and organized code makes debugging a lot easier

## resources

- ▶ download eclipse: <http://www.eclipse.org/downloads/>
- ▶ VIM cheatsheet:  
<http://www.zalas.eu/uploads/wp/2010/05/vi-vim-cheat-sheet.gif>
- ▶ Guide to Faster, Less Frustrating Debugging:  
<http://heather.cs.ucdavis.edu/matloff/UnixAndC/CLanguage/Debug.html>
- ▶ Guide to Makefiles:  
<http://www.delorie.com/djgpp/doc/ug/larger/makefiles.html>