



Eclipse

Quick Review

- ▶ `scp 02piFI5.c` from `/home/CS300Public/2015` to the CS300 directory on your local machine
- ▶ Rename `02piFI5.c` to `02piPUNetID.c` (use your punetid)
- ▶ From the command line, compile your program
`gcc -Wall -g -o 02piPUNetID 02piPUNetID.c`
- ▶ Execute your program
`./ 02piPUNetID`
- ▶ Create a tarball
`tar czf 02piPUNetID.tar.gz 02piPUNetID.c`
- ▶ `scp` the tarball to your directory `Documents/CS300` on zeus
- ▶ Go to zeus and extract the tarball
`tar xzf 02piPUNetID.tar.gz`
- ▶ Compile and run the program to make sure it still works
- ▶ Submit your tarball
`submit cs300f15 02piPUNetID.tar.gz`



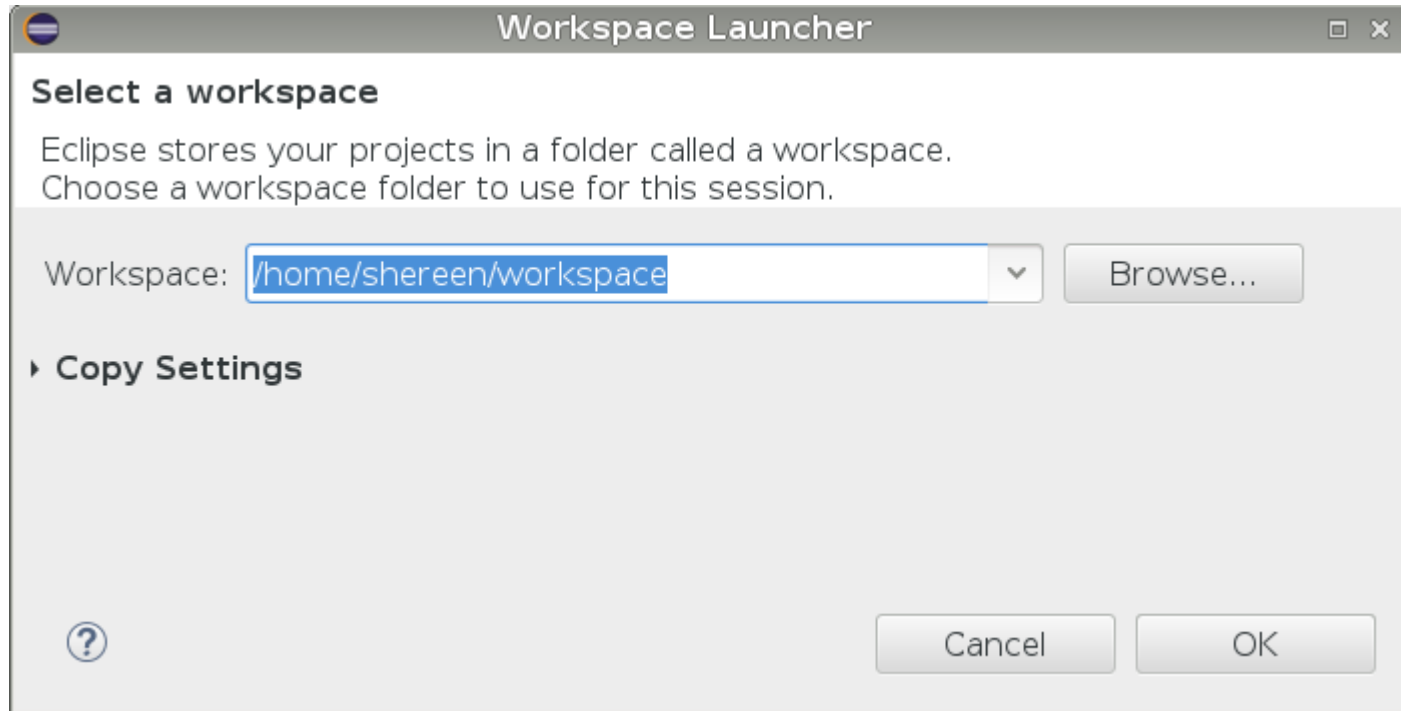
Eclipse 4.4 Luna

- ▶ Integrated Development Environment (IDE)
- ▶ Has a plugin architecture to add features
 - ▶ support for C development is via a plugin, CDT
 - ▶ <http://www.eclipse.org/cdt/>
- ▶ Can use the gcc compiler and gdb debugger
- ▶ Requires a Java Runtime Environment
- ▶ <http://www.eclipse.org/downloads>
 - ▶ Eclipse IDE for C/C++ Developers

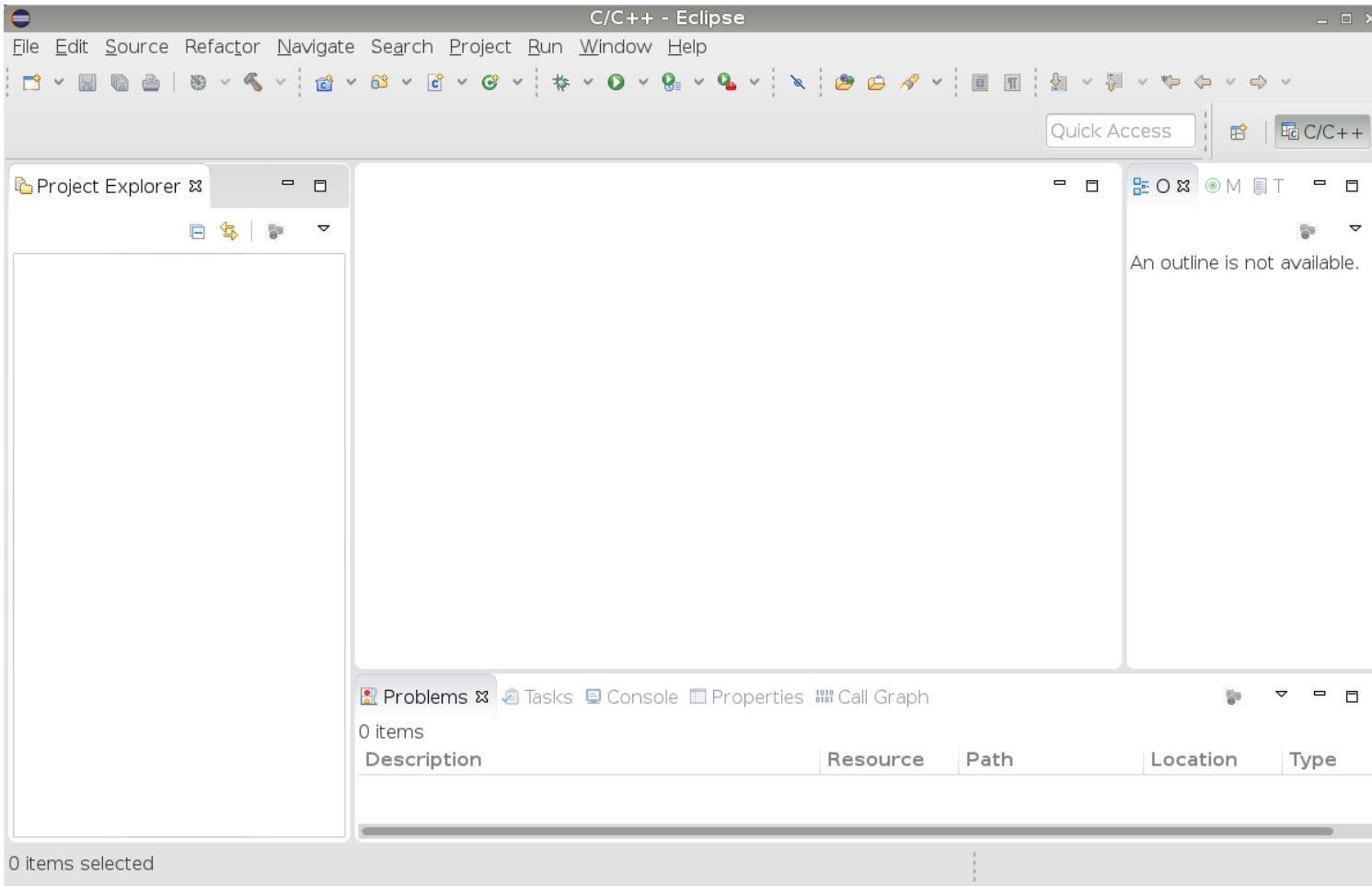
DANGER!

- ▶ Does Eclipse run on Windows?
 - ▶ Yes
- ▶ Can I write C code on Windows?
 - ▶ Yes, with the Cygwin suite installed
- ▶ Can I write C code on Windows for this class?
 - ▶ No

Start Eclipse



Select the perspective for coding

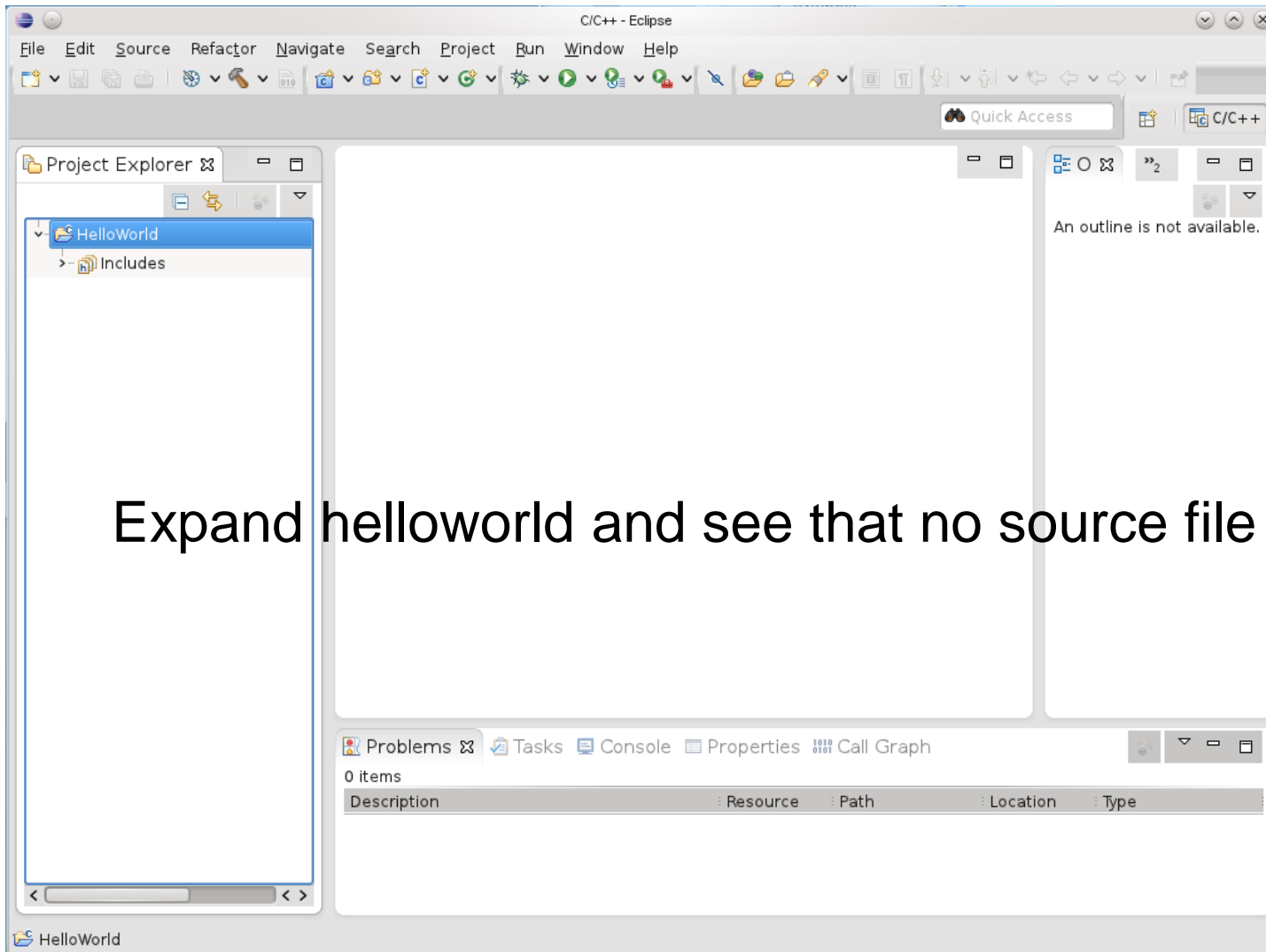


Make sure
the
perspective
is C/C++
not Java

Create a new **HelloWorld** project

- ▶ File → New → C Project
- ▶ Makefile Project → Empty Project → Linux GCC
- ▶ Name it “HelloWorld”
- ▶ Then click Next
- ▶ Then click Finish

A HelloWorld project



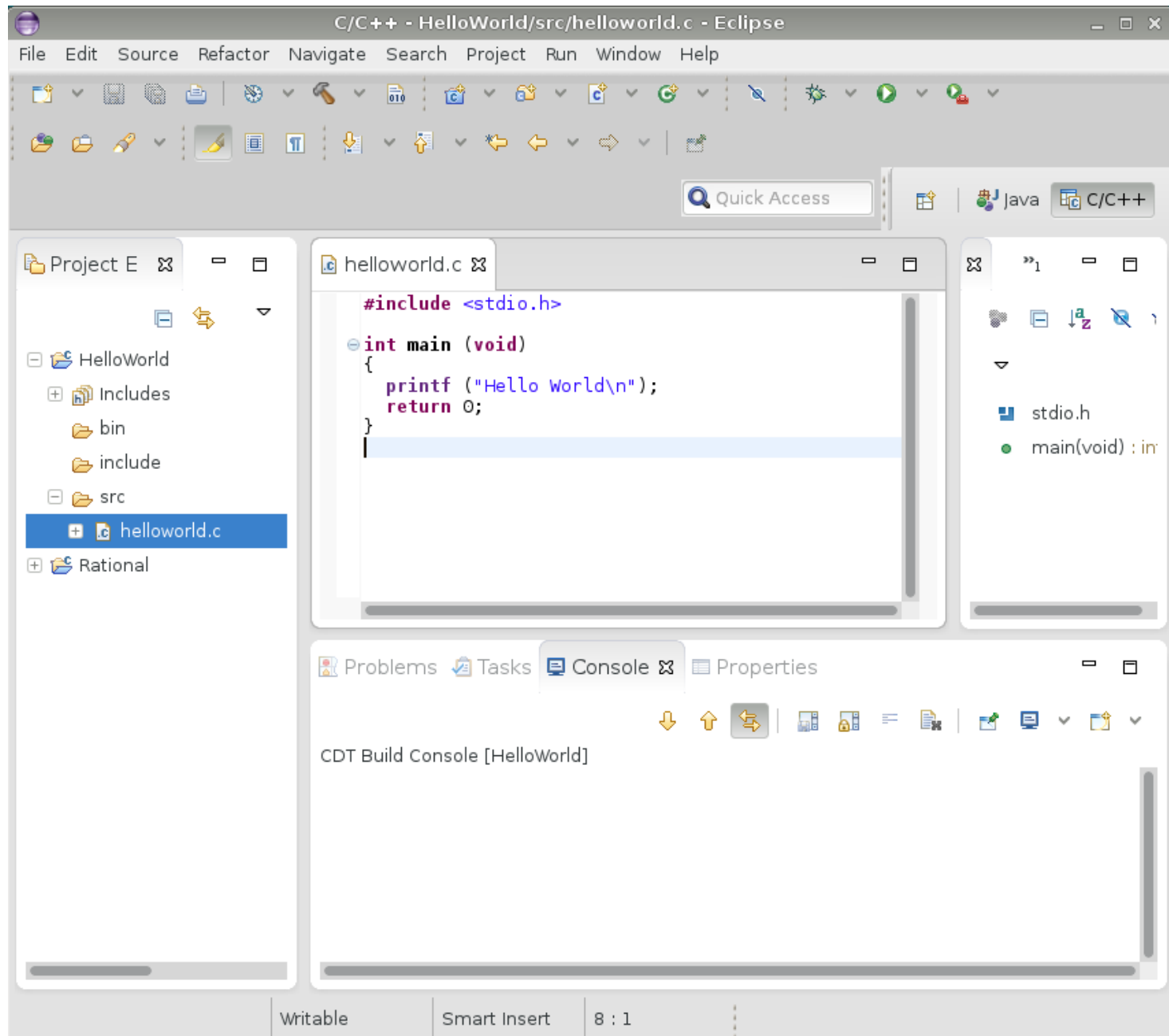
Create helloworld.c

- ▶ In HelloWorld project:
 - ▶ Create a folder called **src**
 - ▶ Create a folder called **bin**
 - ▶ Create a folder called **include**
 - ▶ Then create a new Source File called **helloworld.c** in the sources folder
 - ▶ Type in the following:

```
#include <stdio.h>

int main (void)
{
    printf ("Hello World\n");

    return 0;
}
```



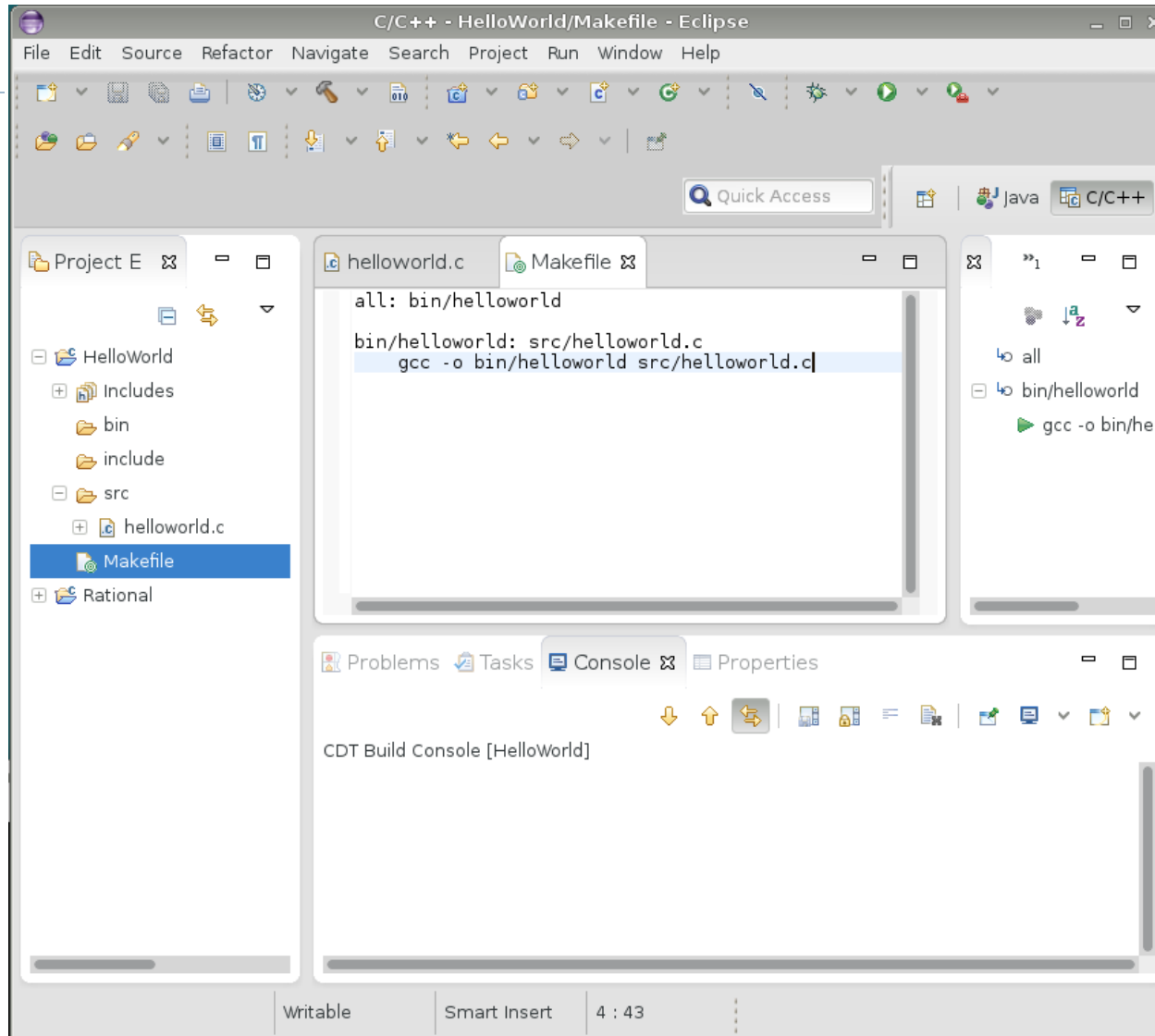
Create a Makefile

- ▶ You need to create a file called Makefile in the helloworld folder. A Makefile specifies rules of how the executable is to be created.
- ▶ Right Click HelloWorld → New → File
- ▶ Call the file Makefile
- ▶ The makefile text must be (a single tab character precedes gcc):

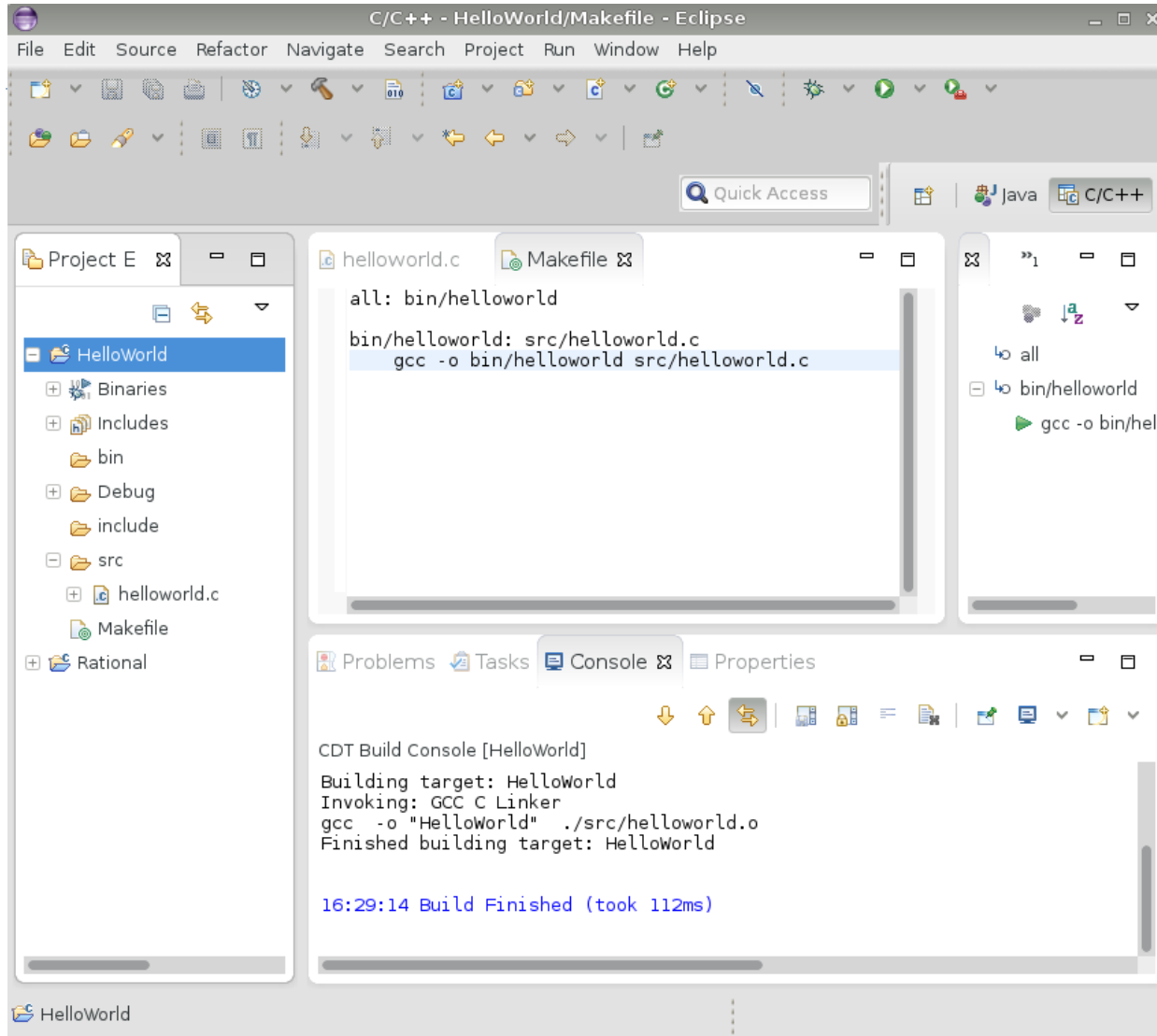
```
all: bin/helloworld
```

```
bin/helloworld: src/helloworld.c  
    gcc -o bin/helloworld src/helloworld.c
```

Makefile created

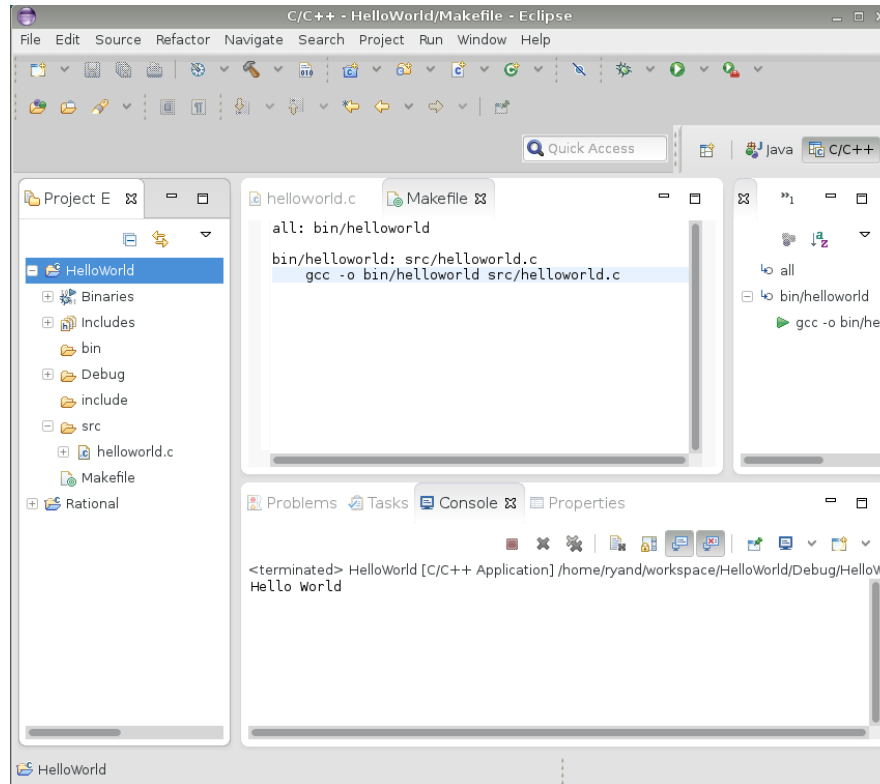


How to build your project ?



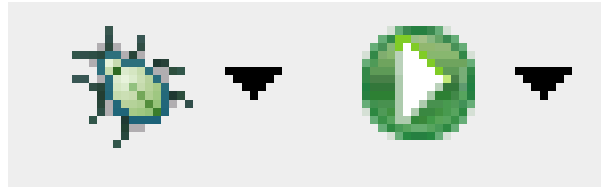
Click on HelloWorld, then Project → Build Project

How to run your program?



Right click on helloworld and Run As → Local C/C++ Application
Choose gdb/mi if given the option

Run versus Debug



Debug Run

Printing

- ▶ **Window | Preferences**
 - ▶ General | Appearance | Colors and Fonts
 - ▶ C/C++ | Editor | C/C++ Editor Text Font
 - ▶ Edit Default ... | Use Courier 10 Pitch, Size 8
- ▶ **This changes the font on the screen!**
 - ▶ You may want to change back after printing
- ▶ **Print doubled sided!**

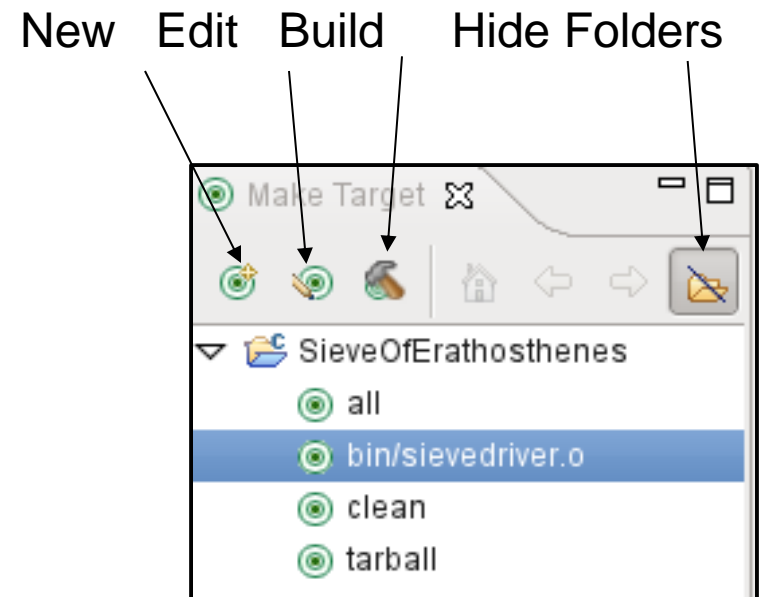
Coding Standards

- ▶ Copy the CodingStandardsProfile CS300PrefsF15.xml from the CS 300Public/2015 directory on zeus
 - ▶ Sets tabs, newlines, spacing to match the coding standards
 - ▶ Does not fix everything!
- ▶ Window | Preferences | C/C++ | Code Style | Formatter | Import
 - ▶ Select the CS300PrefsF15.xml file
- ▶ Open your .c file, then do one of the following:
 - ▶ Source | Format
 - ▶ Shift+Control+F

Add Make Target

- ▶ Open Makefile
- ▶ Open Make Target panel
 - ▶ Window | Show View | Make Target
- ▶ New Make Target
 - ▶ type name of existing target in the Makefile

This is just for your convenience.



Helpful Commands

- ▶ **F3** while cursor on function call
 - ▶ go to that function
- ▶ **Control-L**
 - ▶ go to line
- ▶ **Control-A**
 - ▶ select all
- ▶ **Control-I**
 - ▶ correct indentation

Be sure to look through the Source and Navigate menu!

Other tips

▶ Window | Preferences

- ▶ search for template to setup .c and .h file templates
 - ▶ you can add the file comment header automatically!
- ▶ search for margin
 - ▶ set the print margin column to 80!
- ▶ search for name style
 - ▶ to set naming conventions
- ▶ search for code analysis
 - ▶ setup error/warnings in code style