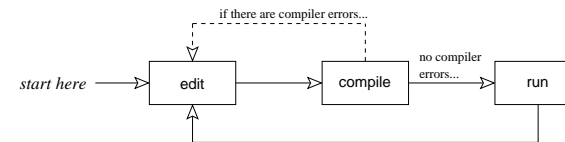


cis1.5 spring2009 lecture 1.2

- the software development cycle
- our first c++ program

the software development cycle

- an iterative process!
- three basic steps:
 1. edit
 2. compile
 3. run



our first c++ program: hello world

- typical first program in any language
- output only (no input)

```
/**
 * hello.cpp, 28jan09/sklar
 *
 * this program demonstrates output from a C++ application.
 *
 */
#include <iostream>
using namespace std;

int main() {
    cout << "hello world!\n";
} // end of main()
```

things to notice

- C++ is CASE sensitive
- punctuation is really important!
- *whitespace* doesn't matter for compilation
- **BUT** whitespace DOES matter for readability and your grade!
- the file name ends in .cpp
- about output
 - a computer screen is made up of *pixels* which are arranged in rows and columns
 - producing output on a computer screen is kind of like filling in the squares on a piece of graph paper
 - the *cursor* is the point on the screen at which the next character will be output (or input)
 - when a line fills up, the cursor moves to the next line...

components of hello.cpp

- `cout`
 - this is called a *method*
 - it is a standard part of the C++ language
 - it produces output on the computer screen
- *arguments*
 - also called *parameters*
 - those things that follow `cout`
 - << followed by a *string*, i.e., text in double quotes (")
 - escape sequences: `\n`, `\t`
 - tell `cout` what to write on the screen
- example

```
cout << "jingle bells\n";
```

let's try it!

1. EDIT

- Open the Terminal application (this should be in your dock...)
- At the "bash" prompt (probably looks like `student$`), type:

```
student$ pico hello.cpp
```

(*pico* is a "text editor")
- Type in the code for the "hello world" program (be mindful of punctuation and spelling and letter case!)
- Click on the **ctrl** and **X** keys at the same time to save your program and exit `pico`

2. COMPILE

- At the "bash" prompt (probably looks like `student$`), type:

```
student$ g++ hello.cpp -o hello
```
- Are there any error messages? If yes, then go back to EDIT and fix them; then do the above step again
- When your code is error free, you are ready to run your program!

3. RUN/TEST

- From inside the Terminal window, type the following:

```
student$ ./hello
```
- You should see the text that follows "cout", i.e., `hello world`
- Now play with it!
 - Go back to the EDIT step and change the message that your program displays
 - Each time you make a change, save your program and exit `pico`; compile it using `g++` and run it.
 - Don't make too many changes at once—always compile and test every little step. That way, you are more assured of maintaining code that runs :-)
 - Try adding another line with another "cout" command
 - Try deleting the `\n` characters

using an IDE

- an IDE is an "Integrated Development Environment"
- an IDE puts these steps together in one application
- there are many IDE's available
- **XCode** comes standard on the Mac with OSX
- **CodeBlocks** is free and can be used on both Mac and Windows (and Linux)
- we have both installed on the lab computers
- if you will be working on a Mac at home, my recommendation is to use XCode
- if you will be working on a PC at home, my recommendation is to use CodeBlocks
- *you can also just continue to use `pico` and the Terminal command line, as above (that's what I do!)*¹

¹On Windows, I think the equivalent of `pico` is called "edit", but I'll have to verify that...