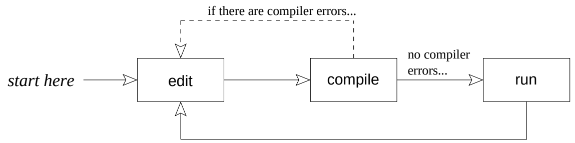
cisc1110 fall 2010 lecture I.2

- 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, 31aug2010/e.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 Code::Blocks application
   - Create an empty file
   - Type in your code
2. COMPILE
3. RUN/TEST

- 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 then compile it 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

- Code::Blocks is an IDE — an "Integrated Development Environment".
- There are many IDE's available.
- **XCode** comes standard on the Mac with OSX.
- **Code::Blocks** is free and can be used on both Mac and Windows (and Linux).
- **Eclipse** is a sophisticated and complex IDE that can be used with many different languages and is used in many workplaces.
- Code::Blocks is a good place to start using an IDE. It is installed on the lab computers and in the W.E.B.
- 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 Code::Blocks.

file system

- It is important to understand the how files are stored on your computer.
- You need to know what is stored *locally* (e.g., on your machine's hard drive) and what is stored *remotely* (e.g., on an email server).
- You also need to understand that local storage can either be on your machine's hard drive or on a removable drive, like a USB flash drive.
- The computer's *file system* defines how the computer can locate files.
- Using an IDE can make it confusing, so be careful to know where the files you create in your IDE are stored on your computer's file system.
- Files are organized into *folders*.
- Files are categorized according to *type*.
- The C++ files you write are called *source code* files. The file system interprets these as *plain text* files.
- Another type of file is a *binary* file. These are files that either require some special software to view them (e.g., Microsoft Word docs) or they are *executable applications* themselves.

- When you compile your program in Code::Blocks, you are creating a binary executable file.
- Note that binary executable files are specific to the computer architecture and operating system on which it was compiled.

variations on hello world

- add another line of output:

```
#include <iostream>
using namespace std;

int main() {
  cout << "hello world!\n";
  cout << "do you like my hat?\n";
} // end of main()
```

- use `endl` instead of `\n`:

```
#include <iostream>
using namespace std;

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