MPCS 51044

Current Homework

- Hard Deadline every week at 5:30 pm (just before start of class)
- Early Feedback and Review: Turn in by Friday 11 pm to get feedback by Sunday night. Resubmit as you need to.
- Email source code, header files you had in your local directory, input data, Makefiles, and READMEs only. In particular, no Visual Studio project directories or other IDE-specific files. Avoid file names with spaces, and definitely do not submit files whose names include single quotes ('), double quotes ("), ampersands (&), parentheses, or at signs (@). Underscores are fine. Dashes are acceptable.

Teaching Assistant Office Hours

- Every Monday at 4:30 pm in Young 111 (or 106), unless otherwise announced.

Compiling on Your Laptop - Quick Links

You can easily install a good C++ compiler on your own laptop or desktop machine.

- On Debian or Ubuntu, you can try `sudo apt-get install g++` (at the command line) and if you get g++ version 4.9 or higher, you should be all set.
- On a Mac, it may be best to use XCode and it should install a very nice version of clang once it's fully updated. Look for "clang –version" 3.4 or higher.
- On Windows, we recommend Visual Studio Community Edition, though for better standards conformance you may want to use a non-Microsoft compiler even on Windows. The Community Edition was released by Microsoft in the latter part of 2014 and is completely free for any non-commercial use and can be downloaded freely. Visual Studio Community Edition is better than the old 'Express' version, and actually gives you access to the full blown "Professional" environment, including developing binaries for 64-bit and/or 32-bit platforms as you desire or any other Windows-specific features you may want if you are Windows-oriented and has 'pretty good' C++11 support, but both GCC 4.8+ and clang 3.4+ tend to be better for standards conformance. HOMEWORK SUBMISSION NOTE: If you use Visual Studio, realize that you may be marked down for submitting your entire project directory as this will include some HUGE binaries.
- On Windows again, if you want better standards conormance, MinGW provides an entirely native GCC compiler for Windows. Link 1 - 64-bit Link 2 - 64-bit Link 3 - 32-bit only
- Intel C++ Compiler for Ubuntu - Licensing for Class Use Only - Click C++ Studio XE for Linux
- boost 1.60

CS Cluster Compiler Status and Compiling with g++ and

Clang

By logging into the CS cluster you can quickly have access to a compiler modern enough for most or all of the material in this class. The cluster currently has g++ 4.9.2 as well as an alternate Clang compiler (as of January 2016). * 2016 update - GCC:

```
To compile your program:
$ g++ -std=c++14 -Wall -Werror myfile.cpp -pthread

To compile **and run** your program with one command, you can use the
following:
$ g++ -std=c++14 -Wall -Werror myFile.cpp -pthread && ./a.out
```

* 2016 update - clang: Clang 3.6 is installed on the CS cluster as a secondary compiler and it includes std::thread support:

```
To compile your program:
$ clang++-3.6 -std=c++14 -Wall -Werror myfile.cpp -pthread

To compile **and run** your program with one command, you can use the
following:
$ clang++-3.6 -std=c++14 -Wall -Werror myFile.cpp -pthread && ./a.out
```

* Note that parameters to g++ and clang are identical. This is part of clang's design, to be interoperable with GCC. Also note that -Wall and -Werror are added, particularly because g++ has some oddities, like cleanly compiling methods with return values even when the method body has no return statement! For example, this code compiles cleanly on g++ without these settings: `struct X { int test() {} };`. That is very odd and very undesirable, so we turn on both -Wall and -Werror to get around these problems. Importantly, your code is expected to compile with these settings.

Homework Submission

Use subversion to submit your homework and see [this document](#).

Piazza and CHALK

We use Piazza and Chalk in this class. Chalk is where we have been distributing lecture slides and other lecture-related and homework-related material.

Log in with your CNET ID at chalk.uchicago.edu:

http://chalk.uchicago.edu/

Contact us if you have Questions

(Email addresses available in class.)

Questions and Answers

Q. For early feedback on homework, when do we have to submit?

`'A.'` By 11:30 pm Friday night. Receive feedback by Sunday night.

Q. When is the regular homework deadline?

`'A.'` Every week, before the start of class - homework cannot be accepted after the start of class, particularly since discussion of homework is the first thing on the class agenda!

Q. What development environment do you recommend?

`'A.'` The nice thing about C++ is that it is a standard, so you can use any conforming C++ compiler. If you or your employer has a preferred C++ development environment, you are encouraged to use that.

If you want specific suggestions, the CS cluster has a good environment that should meet all or at least most course needs. Some nice features of this environment include:

*g++, vim, and emacs are already installed. No need to install any software. *We as graders can better support you in your homework troubleshooting. *Eclipse with the CDT, and Code::Blocks are installed.

Homework conventions

Remember we are human. Sloppy uncommented code may not get the credit it deserves. We have been known not to give any credit on a problem simply because we didn't realize a solution was submitted. Calling out (by directory or filename) where your solutions are, including output where appropriate, etc. will help assure you get the grades you deserve. One helpful convention here is to place a readme.txt file in your directory to tell us all the problems you completed or if you got part of a problem but not all, to explain what you have. The readme is also a good place to put answers to verbal portions of questions as well as to show the output you got from your program.

From:

https://howto.cs.uchicago.edu/ - **How do I?**

Permanent link:

https://howto.cs.uchicago.edu/course:mpcs:51044?rev=1458755663

Last update: **2016/03/23 12:54**