

CS 475/575 Live Lecture

Chat Window

April 5, 2023

15:14:29 What about using matplotlib?

It doesn't really matter to me *how* you make the graphs, only that you do a good job of it.

15:22:00 Ah, so we don't need to turn in the bash scripts if we made them, just the cpp files and pdf?

That's correct.

15:40:16 I think you get it [Microsoft Office] free through the university

15:41:22 <https://softwarelist.oregonstate.edu/students>

15:41:45 <https://uit.oregonstate.edu/office365>

15:41:52 Do we need to have both graphs or will one suffice? [starting with Project #1]

Starting with Project #1, you will be turning in:

- One rectangular number table
- Two graphs
- One commentary

15:42:58 I did it with LibreOffice Calc and the Excel instructions worked perfectly 1:1

Excellent! Good to know. Thanks.

15:48:36 I got the best performance on my personal computer testing with 10 threads which has a 12 thread processor

Interesting. Yes, it is not always the max-cores that gives the best performance. That's why we do benchmarks.

16:32:14 Could an IDE profiler be useful in determining code that could be parallelable? [in determining the Parallel Fraction, Fp]

I'm not sure, but I suspect not.

16:35:34 So to make sure I understand conceptually, your recommendation to run a bash/make file that runs a loop that pushes different numbers (instead of typing in manually and recompiling) to our c file,

The Makefile approach is good for compiling for a *single* run. The scripting loop approach is good for running many combinations of parameters.

Could we do that for project 0? run a for loop to test 1 thread, then another to test 4 threads?

You can, but since it is only 2 runs, it might be easier just to edit the file in between runs.

16:36:56 Should we include any bash scripts we write in our homework submissions, or just keep it to cpp files and pdfs?

Just the .cpp and the PDF. Later, in GPU Land, you will also include the .cu or .cl files.

16:37:45 But how the bash/make actually works confuses me, so I need to figure it out now when it's a baby amount of work load LOL

There is no “magic” in either. The Makefile just has rules for compilation built into it. When you run the “make” program, it is just playing those rules back.

A script is actually a “program” in a different language, but those scripting languages have been specially designed to control the running of other jobs, in this case, lines that do compiling and running. But, the best part of the scripting, at least for us, is that it can compile and run all those different combinations of parameters in your program without you having to manually edit them.

16:38:11 Going back to the scripting stuff, if we're using visual studio to compile/run the C code (Win11), is using it with a script any different? I got it to compile/run using the debug/run button and the OpemMP setting under properties, but not sure how I'd make a script to use that (or is this where I need to go play with the PowerShell tutorial slide stuff?)

Yes, if you are on Windows, you can still take advantage of scripting to simplify your data-gathering, but this time it will be using PowerShell (see the PowerShell noteset and video). You will use Method #2 to get those parameters into your program (see the Scripting noteset) and will use the Visual Studio-compiled .exe program to do the running.

16:38:17 For Q1. Do we just need to say that we ran this on our machine or FLIP.

Just saying “I ran this on flip” is good enough, since we know what flip is. If you ran this on your own machine, we’d like to at least know what your CPU chip is. It gives context to the performance numbers on your graphs.

16:38:28 For proj0 is it just a simple commentary about the findings, then for later projects we include code?

You really don’t have to include code in your commentary ever, unless you did something exceedingly clever and you want us to know about it. Since you are also turning your .cpp file into Teach, we will have your code if we want to see it.

16:40:29 Are there instructions for logging into the flip machine? Is this done through ssh with a local shell or via an OSU website portal? (I'm on an Intel Mac)

They are on the Resources Page, but here they are again:

1. Open up a terminal window (command prompt window on Windows)
2. Type: **ssh flip.engr.oregonstate.edu**
3. Login

Find more information, go to: <https://it.engineering.oregonstate.edu/ssh-keygen>

16:41:48 Can you use putty to then ssh into flip?

Yes – *putty* gives you a terminal emulator window. I use something called *MobaXterm*, which our IT people recommended. It has more intelligence than putty.

16:43:39 I'm curious, could I have my current laptop give commands to an older laptop and use the processing capabilities from multiple systems?

Yes! We will cover that aspect of parallel programming nearer to the end of the quarter. That's what your Project #7 will be about. If you want a preview, take a look at the **MPI** noteset. That's how clusters, render farms, and supercomputers all work.