Live Lecture Chat Window Q&A
March 31, 2021

11:48:00 From Kuhn, Kurt Arthur J to Everyone : A bit an odd question: do you feel it's worth formatting my machine before and do a fresh restart before diving into parallel?

There are no odd questions, just odd answers. 😊 My personal paranoia is that I do nothing to alter my system once a quarter has started. Too risky.

12:13:49 From Name Withheld to Everyone : We are allowed to code in modern C++, correct?

I don't see why not. I won't look at your code unless it seems like something went wrong with your program.

12:13:54 From Abhi Balijepalli to Everyone : This assignment is just basically to see what OpenMP is right? Not really coding involved in this. We are changing a couple things in the code right?

That's correct – Project #0 is a thinly disguised excuse to get OpenMP working on whatever system you plan to use. That's why the project number doesn't even get the dignity of a positive number. Also why it's worth only 30 points.

12:14:37 From Taylor, Donald M to Everyone : Do we need to include the execution time or is it okay to just talk about megamults/sec?

MegaMults/Sec is fine.

12:15:15 From Mills, Aedan Phillip to Everyone : aww
12:15:16 From Jhala, Yesha to Everyone : Awww so cute!
12:15:17 From Oumar, Abdullah F to Everyone : awww
12:15:17 From Liu, Susan to Everyone : awwwwwwww
12:15:18 From Jaquelin Yang to Everyone : SO CUTE!!!
12:15:18 From Weiner, Christopher Charles to Everyone : awwww
12:15:19 From Patrick Sullivan to Everyone : and this is why I signed up for this class
12:15:19 From Lee, Dresden A to Everyone : awwwh!
12:15:19 From Bhavya Singh to Everyone : awww
12:15:20 From Scott to Everyone : daw
12:15:20 From Liu, Susan to Everyone : so cute!!!!!!
12:15:21 From DiRuscio, Heather Frances to Everyone : Hi Curie!!!
12:15:21 From Jacob Eckroth(He/him/his) to Everyone : aww
12:15:22 From Rhea, Conner to Everyone : so cute
12:15:22 From Alcaide, Tiffanie Charlyne Yu to Everyone : awwww
12:15:22 From Ganguly, Malini to Everyone : awe
12:15:23 From Shifflett, Jerred to Everyone : Man I miss walking dogs. And those ears
12:15:23 From Foster, Conner Sandow to Everyone : awwww
12:15:23 From Liu, Susan to Everyone : awwwwww
12:15:24 From Pearson, Dana George to Everyone : AW
12:15:24 From Mills, Arthur M to Everyone : Awwww!
12:15:25 From Oh, Patrick Heungshik to Everyone : this is the real reason I took this class
12:15:26 From Alice Li to Everyone : cute
12:15:26 From Samantha Rodarte to Everyone : adorable
12:15:26 From Yamamoto, Satoru to Everyone : she is gonna grade our assignments
12:15:26 From Ebert, Victoria to Everyone : Shes so cute omg
12:15:27 From Damle, Kedar Anant to Everyone : AWWW
12:15:27 From Molotkov, Alexander Scott to Everyone : Awww!
12:15:27 From ShengTse Tsai to Everyone : doggie!!!

12:15:29 From Jiayu Han to Everyone : awwwwwwwww
12:15:30 From James, April to Everyone : im in love...
12:15:32 From Molotkov, Alexander Scott to Everyone : This is the quality content I signed up for
12:15:32 From Tseng, Kuan-Lun to Everyone : Corgi is the best
12:15:33 From Liu, Susan to Everyone : lol
12:15:34 From Emily to Everyone : corgiiiii :3
12:15:35 From Phan, Theresa to Everyone : cute
12:15:39 From Carter, Luke Lawrence to Everyone : make that 20 pages of chat with just awws
12:15:40 From Jacob Eckroth(He/him/his) to Everyone : this is the best part of online classes
12:16:33 From Shelbi Wakamatsu to Everyone : ^^^

I now know that if you show a cute puppy on a class Zoom, the chat window explodes. If Curie had an ego, it would be in full bloom right now.

12:24:15 From Adlouni, Mohammad to Bailey, Mike(Direct Message) : Hi Professor Bailey, can one of these live sessions be closer to eob? Maybe 2 or 3pm pst? I understand the lectures are recorded but just wanted to ask :). Thank you for doing live lectures!

I picked MW 12:00-2:00 because that is when the class was scheduled for if we were in a classroom. This is the best way to avoid conflicts you all might have with other more-synchronous classes. I can consider something later in the day too. Let me work on that.

12:28:31 From Ben Wichser to Everyone : Thanks. I hadn't yet figured out makefiles. Tutorial appreciated.

There are very detailed tutorials out there, but here is what you *really* need to know. In the example we just looked at, the file called *Makefile* contained::

```
ArrayMult:      ArrayMult.cpp
                g++ ArrayMult.cpp  -o ArrayMult   -lm  -fopenmp
```

It is in the form:

```
WhatYouAreTryingToBuild:            ASpaceSeparatedListOfTheFile(s)YouNeedToBuildItWith
                                    IfYouHaveThoseFiles,HereAreTheCommandLine(s)ToBeExecuted
```

You then just type:

```
make ArrayMult
```

And if the date on ArrayMult.cpp is newer than the date on ArrayMult, it will execute the line(s). If not, it knows not to bother.

12:29:40 From Zhao, Miao T to Everyone :
https://www.gnu.org/software/make/manual/html_node/Introduction.html

Looks good.

12:30:34 From Weng, Chi-Chieh to Everyone : Do we only need to submit the pdf file?

Please submit both the PDF and your C/C++ file.

12:31:18 From Stachura, Ryan to Everyone : sorry I missed it, what command did he use to run the script?

The Makefile: make ArrayMult
The script: csh loop.csh

12:31:19 From Abhi Balijepalli to Everyone : Can we write the output into a .csv?

Yes! And I hope you do. Excel can slurp in either a CSV (comma-separated values) or tab-separated values file.

12:31:22 From Gutzmann, Melanie to Everyone : has anyone been able to submit? the canvas page is blank for me
12:31:34 From Carter, Luke Lawrence to Everyone : Teach
12:31:35 From Jacob Eckroth(He/him/his) to Everyone : you submit through teach.engr.oregonstate.edu

Yes. I know it is not Canvas, but with Teach, I can write a script to get all your PDF files at once and make them into one big PDF and grade that. We can get your scores back to you *way* faster than going one-at-a-time through Canvas. I have begged the Canvas people to give me a way to do something similar on Canvas. So far, the answer has been no.

12:33:00 From Gutzmann, Melanie to Everyone : idk why but using teach has me anxious that my files didn't submit correctly

I would work with you if that happened. Go be anxious about something else.

12:33:53 From Trieu, Huy to Everyone : I am getting errors when compiling on flip

Be sure you change the question marks into actual values. Other than that, it should compile right away on flip.

12:35:16 From Ricardo Wu to Everyone : Where to find execution time? I am trying to calculate for speed-up

The execution time, in seconds, is: (time1 – time0)

12:36:39 From Caruthers, Sean P to Everyone : Does anybody know if we are just turning in the PDF for this assignment, or do we need to include our script and such?

You don't need to turn in your scripts or Makefiles. From a grading-standpoint, I don't care how you get your code compiled and run. From a friend-standpoint, I *do* care that you not waste your valuable time, so I recommend the scripting.

12:40:01 From Lam, Brandon to Everyone : google chrome eating my resources as we speak

That's why it is a good idea to kill all apps, especially browsers, before you do your timing runs.

12:40:17 From Ben Wichser to Everyone : Can we use flip for all the projects?

You can use *flip* for all the OpenMP projects and the SIMD project. *flip* doesn't have any GPUs, so we will use other machines for the GPU projects.

12:40:39 From Zinn Morton to Everyone : I gamed the system by timing my code at 4am

😊

12:41:11 From John Teeter to Everyone : But there is no "problem" with running a large sample size to get consistent results? Other than burning energy/cpu time?

That's correct, no problem.

12:41:41 From Sean Wilton to Everyone : Can we just schedule the job to run in the dead of night? Haha
12:41:47 From Martin, Garrett to Everyone : yea you can use the at command
12:41:50 From Martin, Garrett to Everyone : echo "cc -o foo foo.c" | at 1145 jan 31

Good approach!

13:08:40 From Akins, Sean to Everyone : What's the inherent costs / overhead at the CPU on thread switching?

It's not huge, but it is there, especially on non-hyperthreaded systems.

13:09:14 From Jacob Eckroth(He/him/his) to Everyone : can I make a thread for every single iteration of a for loop

You *can*, but the issue is if you *should*. That would be an awful lot of wasted overhead. Think of it as deliberately adding friction into a mechanical system.

13:11:37 From Gibbs, Matthew Leigh to Everyone : how common are hyperthreaded systems?

Very common. Basically every-other generation of Intel CPUs has hyperthreading.

13:11:47 From Abhi Balijepalli to Everyone : How do I check if my machine has hyper threading?

For Windows, see: https://www.qualitestgroup.com/resources/knowledge-center/how-to-guide/determine-system-hyperthreaded-windows/
For Linux, see: https://stackoverflow.com/questions/29605260/how-to-check-whether-hyper-threading-is-enabled-or-not-in-ubuntu
For Mac, see: https://superuser.com/questions/1101311/how-many-cores-does-my-mac-have

13:13:02 From Shultz, Caleb Ethan to Everyone : Is Intel or AMD better at threading or are they essentially the same

I would guess they are essentially the same. The threading package you use and the machine/OS you run on probably have a bigger impact on how good your threading is.

13:13:21 From Ricardo Wu to Everyone : Does this mean re-entrant probably the best idea for tip#1?

The best idea for Tip #1 is don't keep internal or global state. Make each thread keep its *own* state.

13:21:26 From Taylor, James Edward to Everyone : volatile will also load the value from memory every time you access it instead of storing it in a register once and reusing it

Correct. It will also prevent the compiler optimizer from eliminating certain pieces of code.

13:26:20 From Damle, Kedar Anant to Everyone : Quiz / tests have any section to write code ? or just multi-choice ?

Strictly multiple choice.

13:27:32 From Jordan to Everyone : I was curious: how do we make multithreading more deterministic? do you have to work around the OS's scheduler, or spend special directives to it? Or are there other factors to why it's so non-deterministic, like hardware stuff?

You try to write the code so that the order of execution doesn't matter. That's the good news. The bad news is that doing that often cripples the parallelism.

13:27:34 From Prather, Eric Andrew to Everyone : During the Monday lecture, you mentioned a difference between task-based and thread-based approaches to parallel programming. I understand that we are focusing on the former; but is this always the best way to look at it?

Task-based seems to be the way that the newer threading packages are all going: OpenMP, TBB, DPC++.

13:27:45 From Akins, Sean to Everyone : How do you prioritize thread scheduling at the CPU level?

OpenMP doesn't let you do that. pthreads does, however.

13:29:20 From Villegas, Samuel D to Everyone : Is encapsulation possible with threads?

If we are worried about making it so that multiple threads cannot try to write to the same variable simultaneously, we sometimes make that variable an array instead and do something like this:

```
A[   omp_get_thread_num()  ] = 3.;
```

13:30:19 From Cho, Yongsung to Everyone : For Q5 on project#0, should I write just one digit about Parallel Fraction?

One decimal digit on reporting the parallel fraction is fine.

13:31:25 From Anthony F to Everyone : For Q1 and Q2, when we calculate time1 - time0. Are we averaging out the times given that NUMTRIES will be 10+ ?

Don't average. Go with either the maximum performance or the minimum elapsed time (same result).

13:33:15 From Alice Li to Everyone : do we need to record all the execution times for 1 thread and 4 threads? And if so, in the pdf?

No, just the highest performance or smallest time duration.

13:34:57 From Ng, Chi Hin to Everyone : What should be the principle (or smart practice) for deciding how many threads to use for a program? (eg based on the complexity of instructions / data size / as many as possible etc.)

It's very difficult to do just by looking at the code. Usually we run benchmarks. For instance, look at Slide #17 of the Data Decomposition noteset. The best performance is not with 4 threads, not with 8 threads, not with 16 threads, but with ***6*** threads! Go figure.

13:35:02 From Hawkins, Matthew R to Everyone : Does that mean if we were using a dynamic array for project 0, there wouldn't be a performance difference?

That's correct. Memory is memory, whether it holds global variables or holds dynamically-created variables.

13:51:17 From ryan to Everyone : For our multi-threading in Project#0, is it considered SIMD or MIMD?

It would be considered MIMD. SIMD allows us to execute one instruction and have 4 (or more) floating-point operations happen as a result of that one instruction. You will get to experience that in Project #4.

13:53:49 From Nesbit, Aaron to Everyone : I think another good example for explaining using multiple threads is a producer-consumer type program, where each thread would be running a different function from the program on some shared set of data

Excellent point! A lot of the design patterns you have already studied can be multithreaded.

13:54:58 From Huayue Sun to Everyone : Will there be any new videos or material uploaded on Friday?

Live Lectures are only Monday and Wednesday.

14:00:06 From timliu to Everyone : If I ran project0 on a 1-core system, would there be no speed-up between 1 and 4 threads?
14:00:45 From timliu to Everyone : Because of overhead?

Could be no-speedup, but it is possible that 4 threads would be slightly *slower* because of the overhead.