11:59:45 From Smith, Ryan Jeffery to Everyone: Thanks for telling us about the game jam, it ended up being really cool
12:03:22 From Smith, Ryan Jeffery to Everyone: https://itch.io/jam/beaver-game-jam/entries

Great! I’d like to hear more detail sometime.

12:03:08 From Jacob Eckroth(He/him/his) to Everyone: vaccination availability opened up to all adults though so maybe pizza days will be upon us soon

Yes, as of today, all of you have aged into the vaccine. Check the Benton and Linn county sites for appointments. But, get the vaccine! I want you all to be safe!

[More comments on the vaccine:]
12:04:58 From Jacob Eckroth(He/him/his) to Everyone: Keep signing up, this website sends you a text message if there are leftover at the end of the day sometimes https://hidrb.com/
12:05:01 From Burke, Caden Thomas to Everyone: reser gives away extras every Thursday and Friday. All you do is get in line and wait till they're done for the day
12:05:02 From Gutzmann, Melanie to Everyone: @Isaac we had to go to Newport to get appointments bc reser was full
12:05:45 From Ebert, Victoria to Everyone: You can also volunteer at the reser clinic and be put in a lottery to get extra vaccines

12:30:57 From Shuler, Patrick Logan to Everyone: how does C++ using intrinsics look on that performance graph? is it worse performing than the raw assembly?

Does a better job of getting into the function, so performs equal to or better than my assembly. 😄

12:34:03 From Wil Coiner to Everyone: Is the assembly in the example function machine/architecture agnostic?

The assembly is specific to g++ and Intel/AMD architecture. That’s why I don’t want to use it except for seeing how fast I can make it go.

12:43:14 From Shuler, Patrick Logan to Everyone: if anyone wants to have some more fun with intrinsics: https://software.intel.com/sites/landingpage/IntrinsicsGuide/#techs=SSE

First time I’ve ever heard the words “fun” and “intrinsics” in the same sentence.

12:56:29 From Coppinger, Travis James to Everyone: Do you want us to graph the agent we make?

Include it in the graph of all the other quantities.

13:00:37 From Lucian’s PC to Everyone: Would it be alright to gauge performance by using numnodes squared / (time1-time0) on project 2?

Better than all right – you must use NUMNODES\(^2\) in the numerator. Performance is always: total work done/elapsed time
13:02:58 From Cho, Yongsung to Everyone: In the Quiz#3, if in terms of 64-bit floating-point numbers, what is the size of a cache line? Could you please explain how to calculate it?

A 64-bit float (i.e., a double) occupies 8 bytes. Since there are 64 bytes in a cache line, that means there are 8 doubles per cache line.

13:07:25 From Jacky to Everyone: have you ever seen super-linear speedup in the projects? I got some speedup efficiencies > 1 in project 1 for some trial/thread combos on my own pc

“Can’t” really happen this way. My guess is there was a lot of load on the system when you took the 1-thread performance data. That makes the 1-thread performance look artificially bad, thus making the speedup look artificially good.

13:12:00 From Huy Trieu to Everyone: Are we to make functions for Deer(); / Grain(); Watcher();?

You could put all the deer code right in the parallel section pragma, but I like function calls better to keep the parallel sections part of the code looking cleaner. But, feel free to do it either way.

13:27:36 From Alcaide, Tiffanie Charlyne Yu to Everyone: So the Watcher function will not have any code under the first #pragma omp barrier because it doesn’t do anything?

Correct.

13:31:48 From Headrick, David Joshua to Everyone: I’ve read online that OpenMP isn't the best for games programming because it's constantly spawning new threads to compute things every frame. However, It looks like in project 3 we are keeping threads alive and using them for multiple "frames" of a simulation?

Yes, as you know from the small dataset size experiments, there is overhead to creating threads. So, a better strategy for game-ish applications is to create the threads once and feed them things to do via a stack or a queue.

And, thanks for bringing live dogs and guinea pigs to Office Hours!