Parallel Programming
Course Introduction for On-campus Students

Oregon State University
Mike Bailey
mjb@cs.oregonstate.edu

What You Should Know on the Way In

Above all, you should be a good C programmer. Being comfortable with function calls, arrays, for-loops, structures, arrays of structures, structures of arrays, pointers, and linked lists is a must. It is strongly suggested that you not use this class as an opportunity to learn C for the first time.

On the math side. You should know algebra. There will be times when we have an equation that solves for “Y given X” and I will ask, “What if we already know Y, can we then go back and find X?”. It would be good if you can do that.

What We Will Be Covering

<table>
<thead>
<tr>
<th>Week</th>
<th>Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction, Sylabus, What this course is... and isn’t. Project notes, timing, graphing, Examples. Parallel programming background information. The three things we care about Parallel Processing for, Von Neumann architecture. Multithreading.</td>
</tr>
</tbody>
</table>

What this Course Is

This course is all about parallel programming on the “desktop” for applications that you are attempting to accelerate to improve user interaction and simulation and computational performance.

The goals of this course are to leave you “career-ready” (i.e., both work-ready and research-ready) for tasks that require desktop parallelism, both on a CPU and a GPU.

CS 475/575 topics include:

- Parallel computing: types, limitations
- Moore’s Law, Amdahl’s Law
- OpenMP
- Synchronization issues in parallel computing
- Cache issues in parallel computing
- SIMD
- GPU computing
- OpenCL
- CUDA

What this Course Isn’t

This course is not about supercomputers or clusters. A lot of the same principles that we will discuss about the desktop do apply to supercomputers and clusters so this will still be useful. However, if we have time, we will lightly touch on the Message Passing Interface, MPI, which is used in supercomputers and clusters.

What this Course Is

This course is all about parallel programming on the “desktop” for applications that you are attempting to accelerate to improve user interaction and simulation and computational performance.

The goals of this course are to leave you “career-ready” (i.e., both work-ready and research-ready) for tasks that require desktop parallelism, both on a CPU and a GPU. CS 475/575 topics include:

- Parallel computing: types, limitations
- Moore’s Law, Amdahl’s Law
- OpenMP
- Synchronization issues in parallel computing
- Cache issues in parallel computing
- SIMD
- GPU computing
- OpenCL
- CUDA

What this Course Isn’t

This course is not about supercomputers or clusters. A lot of the same principles that we will discuss about the desktop do apply to supercomputers and clusters so this will still be useful. However, if we have time, we will lightly touch on the Message Passing Interface, MPI, which is used in supercomputers and clusters.

What We Will Be Covering

Note: this schedule is approximate!
I will try to keep the schedule on the class web site up-to-date.
Class Textbook

There is no textbook for this class. The course material will consist of handouts and notes taken while watching the videos.

If you need further reference material, there are a bunch of links at the end of the class web site. You're not required to go look at any of these. They are just some links that I have found useful. They are there if you need them.