Graduate Students

look for other sources of information

make decisions, because all research problems are under-specified

evaluate their own work

write, write, write

question everything

"So long, and thanks for the Ph.D.!” a.k.a.
"Everything I wanted to know about C.S. graduate school at the beginning but didn't learn until later."

http://www.cs.unc.edu/~azuma/hitch4.html
CS620 Topics in Programming Languages

Instructor: Michelle Strout
mstrout@cs.arizona.edu
Gould-Simpson 707
Office hours: 3:30-4:30pm Thursday or by appt

class web page: http://www.cs.arizona.edu/classes/cs620/spring16

piazza: https://piazza.com/arizona/spring2016/csc620/home

D2L: http://d2l.arizona.edu
Plan for Today

Introductions

Announcements

– Read articles posted for today on class schedule and for Tuesday.
– Take quiz by Monday night. (Will send out once have emails).

Course Goals

Administrivia

– You are responsible for what is on the syllabus, the schedule page, the assignments page, and posted to piazza.

Paper Reviews

– Over the semester you need to write 8 reviews.
– Reviews are due before class on the day the paper is being discussed.

Projects, Paper presentations, and Programming Model presentations
Parallel Programming Models

Parallel programming models provide a way to specify algorithms for a parallel computer architecture.

Mechanisms
- Programming languages: Chapel, Julia, Go, CUDA, ...
- Libraries: Pthreads, TBB, Petsc, Halide, ...
- Language extensions: Cilk++, Chombo
- Pragmas: OpenMP, OpenACC, OpenCL, ...

Parallel Programming Models you have used?
Course Research Goals

Questions we will be exploring

– With the advent of multi-core and many-core, what key features should a parallel programming model include?
– How can we separate implementation concerns from the algorithm specification?
– How much should parallel programming models be specialized for particular domains?
– How can we compose programs from different domains?
– How does implementation of programming constructs impact design?
– How should programming models be evaluated?

Approach

– Read and critique influential papers
– Learn and present the key features in a variety of programming models.
– Research project where compare three programming models used to specify same computation
Research Skill Goals

Finding Interesting Research Problems
- effective paper reviewing skills
- presenting the crucial points in a paper
- synthesizing ideas between papers
- suggesting future research questions

Thesis Project Preparation
- how to break projects into manageable pieces
- how to leverage existing research tools
- presenting research in a succinct fashion
Administrative Matters

Turn to your syllabus
Tips for Reading a Research Paper (by Tia Newhall)

Read the paper three times

– First read the abstract, introduction, related work, and conclusion.
– Next read the entire paper writing down questions and vocabulary words you need to look up. Do some examples to understand concepts.
– Look up the vocabulary words and understand what the mean.
– Finally re-read the paper critically, answer the paper review questions.

Questions to answer in a paper review (see example review online)

– What problem did the paper address?
– Is it important/interesting? What was the context for the paper?
– What is the approach used to solve the problem?
– How does the paper support or otherwise justify the conclusions it reaches? How does it evaluate the approach?
– What problems are explicitly or implicitly left as future research questions? What are the limitations of the research?
Selecting a Paper to Present

Email papers you are interested in reading for the class (optional)

Read through the potential list of papers
- Will provide and cover Tuesday January 26\textsuperscript{th} in class.
- Read the abstract, introduction, and conclusion for all papers you are seriously considering.
- Consider selecting a paper related to your course project.

Schedule around your deadlines in this course and other courses

Paper selection will be first-come-first-serve
- Paper selection can start January 28\textsuperscript{th}.
- I will start posting assignments as soon as requests are made.
- You must indicate which paper you would like to present by February 4 before class.
Project

Project Outline

– Learn about an existing code with poor performance from a scientist at UofA (<1000 SLOC).

– Rewrite that code in two to three different programming models.

– Compare and contrast the programming models.

Deliverables

– preliminary project proposal (1-2 paragraphs), approx Feb 8

– proposal presentation (5 minutes), Feb 25

– written proposal (2-3 pages), March 1

– verbal status report (15 minutes)

– code and usage documentation to scientists

– final presentation (20 minutes)

– final report (6-10 pages in a conference format)
Working with UofA Scientists

Analyzing asteroid orbit data
- Kat Volk, Planetary Scientist
- Input: positions and velocities over time of thousands of test particles from a simulation of the outer solar system
- Output: resonance of particle orbits with planetary orbits
- Code: Perl code
- Goal: reduce time for analysis and would like to check more angles

Computational Fluid Dynamics
- Christof Brehm, Aerospace and Mechanical Engineering
- Input: geometry and initial conditions (or free stream conditions)
- Output: flow quantities (pressure, velocity, temperature) as function of space and time
- Code: Fortran and MPI
- Goal: investigate benefits of hybrid computing, i.e., MPI+OpenMP, MPI+OpenACC/OpenCL, and MPI+OpenMP+OpenACC
Programming Model Presentations

Goal
– Learn about, do some examples, and present a programming model to the class.
– Select a paper that is primarily focused on the presented programming model.

Implementation
– Each student will pick one programming model and paper about it.
– Student will then present the programming model.

Logistics of where we are doing all this programming?
– Will be using some machines in department for debugging. We will help with installing the programming environment for any particular model.
– Also will learn how to use HPC resources on campus.
See code of conduct linked from syllabus.

- The Department of Computer Science is committed to providing and maintaining a supportive educational environment for all. We strive to be welcoming and inclusive, respect privacy and confidentiality, behave respectfully and courteously, and practice intellectual honesty. Disruptive behaviors (such as physical or emotional harassment, dismissive attitudes, and abuse of department resources) will not be tolerated. The complete Code of Conduct is available on our department web site. We expect that you will adhere to this code, as well as the UA Student Code of Conduct, while you are a member of this class.

Course culture

- When ask questions.
- Will provide 1-2 minutes for discussions with neighbor.
- Then will use index cards to call on people.
- For projects, will meet with scientists and other students working with same code, and myself. Can work with other students, but will be developing separate versions in different parallel programming models.
Next Time

Reading (posted on class schedule)

– How to Have a Bad Career in Research/Academia by David Patterson.
– Parallel Algorithm for Dense Matrix Multiplication by Patricia Ortega.

Class

– Dave Lowenthal will be covering parallel programming and example parallel programming models next week.

Due

– Quiz due Monday night (will be announced on piazza sometime tomorrow Friday 15th)
– Paper suggestions (optional)