Context

- Chips aren’t getting faster, they are getting **more computing cores**
- Gain in compute power by running on many machines, e.g., “the cloud”
- Thus, need to be able to write programs that are **parallel** and **distributed**: more complicated than sequential programs
Course goal

- How to build **correct** and **efficient** parallel and distributed programs
  - Focus on **Java** programming language
    - Concepts should apply to similar languages
  - Will also consider **Erlang**
    - Functional language, gaining popularity
  - Grab-bag, including **map/reduce** with **Hadoop**
Why these topics?

- **Java** concurrency: excellent concurrency libraries, mainstream language that you are familiar with
- **Erlang**: different concurrency programming idioms (favors immutability and message passing), rising popularity
- **Hadoop**: amazing and economical scalability
  - When appropriate, it works great
Textbook

- We will use this book the first half to 2/3 of the semester.
- The remainder of the semester will have separate lecture notes.
Projects vs. exams

- Projects and in-class programming exercises will help you learn ideas tested by exams
- I am considering allowing you to work in pairs; idea:
  - You may discuss, share code, etc. as indicated on the description, but only within the 2-person team
  - I will assume you understand the project solutions on the exams, which you must do alone
- **What do you think? See on-line survey; will post results to syllabus**

Monday, August 30, 2010
In-class exercises

- Small projects to cement ideas we are covering
  - Will go over answers the next class
- Will be on Mondays, last 30 minutes of class
- You are expected to complete and submit the assignment by the end of class
- Bonus materials for after class, if you wish
- Bring your own laptop or use one from laptop cart
- May work in small groups (no more than 4)
Overview / grade breakdown

- Three tests: two midterms, final exam (17.5% x 2, 25%)
- Five projects (5% each):
  - Project 1: Java Logging library / infrastructure
  - Project 2: Java Restaurant simulation
  - Project 3: Java Parallel maze generator / solver
  - Project 4: Erlang project TBD
  - Project 5: Hadoop map/reduce TBD
- 4-5 in-class programming exercises, discussions (15%)

*This breakdown depends on what is decided following the survey; see the syllabus*
Bonus prizes

- Some projects will have an optional competition
- The winner will get a cool (geek) T-shirt (of your choice)
- E.g., third project: fastest solver
- We’ll run it on a 24 processor machine. Linuxlab has several multi-processor machines.
Academic integrity

- All written work (including projects) must be done on your own or within your group, unless otherwise stated
  - Do not copy code from other students
  - Do not copy code from the web
- Work together on high-level project questions
- Work together on practice problems

Monday, August 30, 2010
Resources

- TA and Prof office hours posted on web site
- Use Forum (also linked) as first line of questioning
  - Helps everyone, may yield faster responses
- Read the on-line syllabus
  - More useful details and expectations

Monday, August 30, 2010
Overview of next two weeks

- Today: Chapter 1 of JCIP
- Wed: Chapter 2
- Mon: Labor day (no class)
- Wed: Chapter 12, first project posted