COMP 322: Fundamentals of Parallel Programming

Vivek Sarkar
Department of Computer Science
Rice University
vsarkar@rice.edu
COMP 322 Course Information: Spring 2012

• “Fundamentals of Parallel Programming”
• Lectures: MWF, 1pm – 1:50pm
• Labs: 4pm – 5:20pm on Thurs (section A01) OR 3:30pm – 4:50pm on Weds (section A02)
• Instructor: Vivek Sarkar (vsarkar@rice.edu)
• Web site:
  • https://wiki.rice.edu/confluence/display/PARPROG/COMP322
  • Or do a web search on “comp322 wiki”
• Prerequisites: COMP 215 or equivalent
  — Parallel programming courses at other universities require more advanced prerequisites
• Cross-listing: ELEC 323
Scope of Course

• Approach

—Mid-level parallel programming --- “Simple things should be simple, complex things should be possible”

—Introduce students to fundamentals of parallel programming
  - Primitive constructs for task creation & termination, collective & point-to-point synchronization, task and data distribution, and data parallelism
  - Abstract models of parallel computations and computation graphs
  - Parallel algorithms & data structures including lists, trees, graphs, matrices
  - Common parallel programming patterns

—Use Habanero-Java (HJ) as pedagogical language for two-thirds of course, and then teach standard programming models (Java concurrency, MPI, CUDA) using HJ principles
What is Parallel Programming?

• Specification of operations that can be executed in parallel
• A parallel program is decomposed into sequential subcomputations called *tasks*
• Parallel programming constructs define task creation, termination, and interaction

Tasks A, B  Tasks C, D

Schematic of a Dual-core Processor
Why Parallel Computing Now?

• Researchers have been using parallel computing for decades as a specialized capability:
  — Problems too large to solve on one computer; use 100s or 1000s

• There have been higher level courses in parallel computing (COMP 422, COMP 522) at Rice for several years

• Why has Rice added a 300-level undergraduate course on parallel programming now?
  — Because the entire computing industry has bet on multicore parallelism
    - Number of cores in a single computer chip is projected to increase to ~ 100 by 2020
  — There is a desperate need for all computer scientists and practitioners to be aware of parallelism
  — Nationwide discussion on how to add parallel programming foundations to the undergraduate CS curriculum --- Rice is ahead of the curve
Domain Experts need high level parallelism-oblivious Programming Models.

Expertise gap between domain experts and concurrency experts.

Concurrency Experts use low-level Parallel Programming Models.
Domain Experts need high level parallelism-oblivious Programming Models

CS majors will bridge the gap with new parallel programming abstractions and algorithms

Concurrency Experts use low-level Parallel Programming Models

CS Majors to the Rescue
COMP 322 course evaluations: Spring 2011

• First time course was offered in current form (prior offering in Fall 2009 was targeted to juniors/seniors)

• Section A01
  (9 responses from 12 enrolled)

• Section A02
  (10 responses from 12 enrolled)
COMP 322 Course Information: Spring 2012

• “Fundamentals of Parallel Programming”

• Lectures: MWF, 1pm – 1:50pm

• Labs: 4pm – 5:20pm on Thurs (section A01) OR 3:30pm – 4:50pm on Weds (section A02)

• Instructor: Vivek Sarkar
  — Send email to vsarkar@rice.edu with any questions

• Web site:
  • https://wiki.rice.edu/confluence/display/PARPROG/COMP322
  • Or do a web search on “comp322 wiki”

• Prerequisites: COMP 215 or equivalent

• Cross-listing: ELEC 323