COMP322-Spring-2011

NOTE: the most recent web site for COMP 322 is located here

COMP 322: Fundamentals of Parallel Programming (Spring 2011)

Instructor:	Prof. Vivek Sarkar, DH 3131	**Graduate TA:**	Sanjay Chatterjee
	Please send all emails to comp322-staff at rice dot edu	**Graduate TA:**	Raghavan Raman
		Undergrad TA:	Christopher Nunu
Assistant:	Amanda Nokleby, akn3@rice.edu, DH 3137	**Undergrad TA:**	Max Grossman
		Research Programmer:	Vincent Cave
Lectures:	Duncan Hall (DH) 1042	**Time:**	MWF 1:00-01:50pm
Labs:	Ryon 102	**Times:**	Tuesday 2:30-3:50pm (Sec 1), Wednesday 3:30-4:50pm (Sec 2)

Introduction

The goal of COMP 322 is to introduce you to the fundamentals of parallel programming and parallel algorithms, using a pedagogical approach that exposes you to the intellectual challenges in parallel software without enmeshing you in low-level details of different parallel systems. To that end, the main pre-requisite course requirement is COMP 211 or equivalent. This course should be accessible to anyone familiar with the foundations of sequential algorithms and data structures, and with basic Java programming. COMP 221 is also recommended as a co-requisite.

The pedagogical approach will introduce you to the following foundations 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 and data structures including lists, strings, trees, graphs, matrices
- Common parallel programming patterns including task parallelism, undirected and directed synchronization, data parallelism, divide-and-conquer parallelism, map-reduce, concurrent event processing including graphical user interfaces.

Laboratory assignments will explore these topics through a simple parallel extension to the Java language called Habanero-Java (HJ), developed in the Habanero Multicore Software Research project at Rice University. The use of Java will be confined to a subset of the Java 1.4 language that should also be accessible to C programmers --- no advanced Java features (e.g., generics) will be used. An abstract performance model for HJ programs will be available to aid you in complexity analysis of parallel programs before you embark on performance evaluations on real parallel machines. We will conclude the course by introducing you to some real-world parallel programming models including the Java Concurrency Utilities, Google's MapReduce, CUDA and MPI. The foundations gained in this course will prepare you for advanced courses on Parallel Computing offered at Rice (COMP 422, COMP 522).

Since the aim of the course is for you to gain both theoretical and practical knowledge of the foundations of parallel programming, the weightage for course work will be balanced across written assignments, programming assignments, and exams. Students interested in taking COMP 322 in Spring 2012 can find summary information here\.

Textbooks

There are no required textbooks for the class. You will be expected to read each lecture handout before coming to the lecture. We will also provide a number of references in the slides and handouts.

However, there are a few optional textbooks that we will draw from quite heavily. You are encouraged to get copies of any or all of these books. They will serve as useful references both during and after this course:

- Java Concurrency in Practice by Brian Goetz with Tim Peierls, Joshua Bloch, Joseph Bowbeer, David Holmes and Doug Lea
- Principles of Parallel Programming by Calvin Lin and Lawrence Snyder
- The Art of Multiprocessor Programming by Maurice Herlihy and Nir Shavit

Lecture Schedule

	Day	Date (2011)	Topic	Handouts	Slides	Homework Assigned	Homework Due

1	Mon	Jan 10	Lecture 1: The What and Why of Parallel Programming	lec1-handout	lec1-slides	HW1 (Written Assignment)	
2	Wed	Jan 12	Lecture 2: Task Creation & Termination using Async & Finish	lec2-handout	lec2-slides		
3	Fri	Jan 14	Lecture 3: Computation Graphs, Abstract Performance Metrics	lec3-handout (rev 1/14/2011)	lec3-slides	HW2 (Written Assignment)	HW1
-	Mon	Jan 17	School Holiday				
4	Wed	Jan 19	Lecture 4: Futures --- Tasks with Return Values	lec4-handout	lec4-slides		
5	Fri	Jan 21	Lecture 5: Parallel Array Sum and Array Reductions	lec5-handout	lec5-slides		HW2
6	Mon	Jan 24	Lecture 6: Data Races and How to Avoid Them	lec6-handout	lec6-slides	HW3 (Programming Assignment)	
7	Wed	Jan 26	Lecture 7: Parallel Prefix Sum, Forall parallel loops	lec7-handout	lec7-slides		
8	Fri	Jan 28	Lecture 8: Parallel Quicksort	lec8-handout (rev 1/28/2011)	lec8-slides		
9	Mon	Jan 31	Lecture 9: PRAM model, Amdahl's Law	lec9-handout	lec9-slides		
10	Wed	Feb 02	Lecture 10: Critical sections and the Isolated statement	lec10-handout (rev 2/3/2011)	lec10-slides		
-	Fri	Feb 04	No Lecture, School closed due to inclement weather				
11	Mon	Feb 07	Lecture 11: Abstract vs Real Performance, Work-sharing & Work-stealing schedulers	lec11-handout	lec11-slides		HW3
12	Wed	Feb 09	Lecture 12: Barrier Synchronization in Forall Loops	lec12and13-handout	lec12-slides	HW4 (Written Assignment)	
13	Fri	Feb 11	Lecture 13: Barrier Synchronization in Forall Loops (contd)	lec12and13-handout	lec13-slides		
14	Mon	Feb 14	Lecture 14: Point-to-point Synchronization and Phasers	lec14and15-handout	lec14-slides		
15	Wed	Feb 16	Lecture 15: Point-to-point Synchronization and Phasers (contd)	lec14and15-handout	lec15-slides		HW4
16	Fri	Feb 18	Lecture 16: Guest Lecture on Bitonic Sort (John Mellor-Crummey)		lec16-slides		
17	Mon	Feb 21	Lecture 17: Advanced Phaser topics	lec17-handout	lec17-slides		
18	Wed	Feb 23	Lecture 18: Midterm Summary		lec18-slides	Midterm Exam (Take-home)	
-	Fri	Feb 25	No lecture, Midterm Exam due today				Midterm Exam (Take-home)
-	M-F	Feb 28 - Mar 04	Spring Break				
19	Mon	Mar 07	Lecture 19: Java Atomic Variables	lec19-handout	lec19-slides		
20	Wed	Mar 09	Lecture 20: Java Concurrent Collections	lec20-handout	lec20-slides		
21	Fri	Mar 11	Lecture 21: Linearizability of Concurrent Objects	lec21-handout	lec21-slides	HW5 (Written Assignment)	
22	Mon	Mar 14	Lecture 22: Task Affinity with Places	lec22-handout	lec22-slides		
23	Wed	Mar 16	Lecture 23: Task Affinity with Places, contd.	lec23-handout	lec23-slides		
24	Fri	Mar 18	Lecture 24: Map Reduce	lec24-handout	lec24-slides		
25	Mon	Mar 21	Lecture 25: Dataflow Programming and Data-Driven Futures	lec25-handout	lec25-slides		HW5
26	Wed	Mar 23	Lecture 26: Dataflow Programming with Intel Concurrent Collections	lec26-handout	lec26-slides	HW6 (Programming Assignment)	
-	Fri	Mar 25	Midterm Recess				
27	Mon	Mar 28	Lecture 27: Java Threads	lec27-handout	lec27-slides		
28	Wed	Mar 30	Lecture 28: Java Threads (contd), synchronized statement		lec28-slides		
29	Fri	Apr 01	Lecture 29: Java synchronized statement with wait/notify		lec29-slides		
30	Mon	Apr 04	Lecture 30: Advanced locking in Java		lec30-slides		

31	Wed	Apr 06	Lecture 31: Java Executors and Synchronizers		lec31-slides		HW6
32	Fri	Apr 08	Lecture 32: Volatile Variables and Java Memory Model		lec32-slides	HW7 (Programming Assignment)	
33	Mon	Apr 11	Lecture 33: GPGPU programming with CUDA	lec33-handout	lec33-slides		
34	Wed	Apr 13	Lecture 34: CUDA contd.		lec34-slides		
35	Fri	Apr 15	Lecture 35: Liveness and Progress Guarantees		lec35-slides		
36	Mon	Apr 18	Lecture 36: Introduction to MPI		lec36-slides		
37	Wed	Apr 20	Lecture 37: Introduction to MPI (contd)		lec37-slides		
38	Fri	Apr 22	Lecture 38: Course Summary		lec38-slides	Final Exam (Take-home)	HW7
-	Fri	Apr 29					Final Exam (Take-home)

Lab Schedule ||

Lab #	Date (2011)	Topic	Handouts
1	Jan 11, 12	Infrastructure setup	lab1-handout
2	Jan 18, 19	Abstract performance metrics with async & finish	lab2-handout
3	Jan 25, 26	Data race detection	lab3-handout
4	Feb 01, 02	Points, regions, forall loops	lab4-handout
5	Feb 08, 09	Abstract vs Real Performance, Work-sharing & Work-stealing schedulers	lab5-handout
6	Feb 15, 16	Barriers and Phasers	lab6-handout
-	Feb 22, 23	No lab (midterm week)	
7	Mar 08, 09	Atomic Variables	lab7-handout
8	Mar 15, 16	Places	lab8-handout
9	Mar 22, 23	Data Driven Tasks	lab9-handout
10	Mar 29, 30	Java Concurrency I	lab10-handout
11	Apr 05, 06	Java Concurrency II	lab11-handout
12	Apr 12, 13	CUDA	lab12-handout
13	Apr 19, 20	MPI	lab13-handout

Grading, Honor Code Policy, Processes and Procedures

Grading will be based on your performance on homeworks (worth 50%) and exams (20% for first exam, and 30% for the second exam).

The purpose of the homeworks is to train you to solve problems and to help deepen your understanding of concepts introduced in class. Homeworks and programming assignments are due on the dates and times specified in the course schedule. Please turn in all your homeworks using the CLEAR turn-in system. Homework is worth full credit when turned in on time. A 10% penalty per day will be levied on late homeworks, up to a maximum of 6 days. No submissions will be accepted more than 6 days after the due date.

You will be expected to follow the Honor Code in all homeworks and exams. All submitted homeworks are expected to be the result of your individual effort. You are free to discuss course material and approaches to problems with your other classmates, the teaching assistants and the professor, but you should never misrepresent someone else's work as your own. If you use any material from external sources, you must provide proper attribution. Take-home exams, which are pledged under the Honor Code, test your individual understanding and knowledge of the material. Collaboration on exams is strictly forbidden. Finally, it is also your responsibility to protect your homeworks and exams from unauthorized access.

The work you submit for this class is expected to be the result of your own work and that of your homework partner. You are free to discuss course material and approaches to problems with your other classmates, the teaching assistants and the professor, but you should never misrepresent someone else's work as your own. If you use any material from online sources, you must provide proper attribution (as shown here) in your homework/programming assignment turnins. A tutorial on how and when to cite sources is here. You should explain what value you have added to work taken from online sources. Finally, it is also your responsibility to protect your work from unauthorized access. I will expect you to follow the Honor Code in this course.

Graded homeworks will be returned to you via email, and exams as marked-up hardcopies. If you believe we have made an error in grading your homework or exam, please bring the matter to our attention within one week.

Accommodations for Students with Special Needs

Students with disabilities are encouraged to contact me during the first two weeks of class regarding any special needs. Students with disabilities should also contact Disabled Student Services in the Ley Student Center and the Rice Disability Support Services.