COMP322-Spring-2015

Home	Office Hours	Turnin Guide	HJlib Info	edX site	Other Resources

COMP 322: Fundamentals of Parallel Programming (Spring 2015)

Instructor:	Prof. Vivek Sarkar, DH 3131		
Co-Instructor:	Dr. Eric Allen	**Graduate TAs:**	Prasanth Chatarasi, Peng Du, Xian Fan, Max Grossman
	Please send all emails to comp322-staff at rice dot edu	**Undergraduate TAs:**	Matthew Bernhard, Nicholas Hanson-Holtry, Yi Hua,
			Yoko Li, Ayush Narayan, Derek Peirce,
Cross-listing:	ELEC 323		Maggie Tang, Wei Zeng, Glenn Zhu
		Course consultants:	Vincent Cavé, John Greiner, Shams Imam
Lectures:	Herzstein Hall 210	**Lecture times:**	MWF 1:00pm - 1:50pm
Labs:	DH 1064 (Section A01), DH 1070 (Section A02)	**Lab times:**	Wednesday, 07:00pm - 08:30pm

Course Syllabus

A summary PDF file containing the course syllabus for the course can be found here . Much of the syllabus information is also included below in this course web site, along with some additional details that are not included in the syllabus.

Course Objectives

The primary goal of COMP 322 is to introduce you to the fundamentals of parallel programming and parallel algorithms, by following a pedagogic approach that exposes you to the intellectual challenges in parallel software without enmeshing you in the jargon and lower-level details of today's parallel systems. A strong grasp of the course fundamentals will enable you to quickly pick up any specific parallel programming system that you may encounter in the future, and also prepare you for studying advanced topics related to parallelism and concurrency in courses such as COMP 422.

The desired learning outcomes fall into three major areas (course modules):

1) Parallelism: creation and coordination of parallelism (async, finish), abstract performance metrics (work, critical paths), Amdahl's Law, weak vs. strong scaling, data races and determinism, data race avoidance (immutability, futures, accumulators, dataflow), deadlock avoidance, abstract vs. real performance (granularity, scalability), collective & point-to-point synchronization (phasers, barriers), parallel algorithms, systolic algorithms.

2) Concurrency: critical sections, atomicity, isolation, high level data races, nondeterminism, linearizability, liveness/progress guarantees, actors, request-response parallelism, Java Concurrency, locks, condition variables, semaphores, memory consistency models.

3) Locality & Distribution: memory hierarchies, locality, cache affinity, data movement, message-passing (MPI), communication overheads (bandwidth, latency), MapReduce, accelerators, GPGPUs, CUDA, OpenCL.

To achieve these learning outcomes, each class period will include time for both instructor lectures and in-class exercises based on assigned reading and videos. The lab exercises will be used to help students gain hands-on programming experience with the concepts introduced in the lectures.

To ensure that students gain a strong knowledge of parallel programming foundations, the classes and homeworks will place equal emphasis on both theory and practice. The programming component of the course will mostly use the Habanero-Java Library (HJ-lib) pedagogic extension to the Java language developed in the Habanero Extreme Scale Software Research project at Rice University. The course will also introduce you to real-world parallel programming models including Java Concurrency, MapReduce, MPI, OpenCL and CUDA. An important goal is that, at the end of COMP 322, you should feel comfortable programming in any parallel language for which you are familiar with the underlying sequential language (Java or C). Any parallel programming primitives that you encounter in the future should be easily recognizable based on the fundamentals studied in COMP 322.

Prerequisite

The prerequisite course requirements are COMP 182 and COMP 215. COMP 322 should be accessible to anyone familiar with the foundations of sequential algorithms and data structures, and with basic Java programming. COMP 321 is also recommended as a co-requisite.

Textbooks

There are no required textbooks for the class. Instead, lecture handouts are provided for each module as follows. The links to the latest versions on Owlspace are included below:

- Module 1 handout (*Parallelism*)

- Module 2 handout (*Concurrency*)
- Module 3 handout (*Distribution and Locality*)

You are expected to read the relevant sections in each lecture handout before coming to the lecture. We will also provide a number of references in the slides and handouts.

There are also 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

Past Offerings of COMP 322

- Spring 2014 (Rice University)
- Spring 2013 (Rice University)
- Fall 2012 (Harvey Mudd College CS 181E, half-semester class, co-instructor: Prof. Ran Libeskind-Hadas)
- Spring 2012 (Rice University)
- Spring 2011 (Rice University)
- Fall 2009 (Rice University)

Lecture Schedule

lec36-slides

Week	Day	Date (2015)	Topic	Assigned Reading	Assigned Videos (Quizzes due by Friday of each week)	In-class Worksheets	Slides	Work Assigned	Work Due
1	Mon	Jan 12	Lecture 1: The What and Why of Parallel Programming, Task Creation and Termination (Async, Finish)	Module 1: Sections 0.1, 0.2, 1.1	Topic 1.1 Lecture, Topic 1.1 Demonstration	worksheet1	lec1-slides		
	Wed	Jan 14	Lecture 2: Computation Graphs, Ideal Parallelism	Module 1: Sections 1.2, 1.3	Topic 1.2 Lecture, Topic 1.2 Demonstration, Topic 1.3 Lecture, Topic 1.3 Demonstration	worksheet2	lec2-slides		
	Fri	Jan 16	Lecture 3: , Abstract Performance Metrics, Multiprocessor Scheduling	Module 1: Section 1.4	Topic 1.4 Lecture, Topic 1.4 Demonstration	worksheet3	lec3-slides	**Homework 1**	**Lecture & demo quizzes for topics 1.1, 1.2, 1.3, 1.4**
2	Mon	Jan 19	No lecture, School Holiday (Martin Luther King, Jr. Day)						
	Wed	Jan 21	Lecture 4: Parallel Speedup and Amdahl's Law	Module 1: Section 1.5	Topic 1.5 Lecture, Topic 1.5 Demonstration	worksheet4	lec4-slides		
	Fri	Jan 23	Lecture 5: Future Tasks, Functional Parallelism	Module 1: Section 1.6 (self-study), Section 2.1	Topic 1.6 Lecture, Topic 1.6 Demonstration, Topic 2.1 Lecture, Topic 2.1 Demonstration	worksheet5	lec5-slides		**Lecture & demo quizzes for topics 1.5, 1.6, 2.1**
3	Mon	Jan 26	Lecture 6: Finish Accumulators	Module 1: Section 2.3	Topic 2.3 Lecture , Topic 2.3 Demonstration	worksheet6	lec6-slides		
	Wed	Jan 28	Lecture 7: Data Races, Functional & Structural Determinism	Module 1: Sections 2.5, 2.6	Topic 2.5 Lecture , Topic 2.5 Demonstration, Topic 2.6 Lecture , Topic 2.6 Demonstration	worksheet7	lec7-slides	**Homework 2**	**Homework 1**
	Fri	Jan 30	Lecture 8: Map Reduce	Module 1: Section 2.4	Topic 2.4 Lecture , Topic 2.4 Demonstration	worksheet8	lec8-slides		**Lecture & demo quizzes for topics 2.3, 2.4, 2.5, 2.6**
4	Mon	Feb 02	Lecture 9: Memoization	Module 1: Section 2.2	Topic 2.2 Lecture , Topic 2.2 Demonstration	worksheet9	lec9-slides		
	Wed	Feb 04	Lecture 10: Loop-Level Parallelism, Parallel Matrix Multiplication, Iteration Grouping (Chunking)	Module 1: Sections 3.1, 3.2, 3.3	Topic 3.1 Lecture, Topic 3.1 Demonstration, Topic 3.2 Lecture, Topic 3.2 Demonstration, Topic 3.3 Lecture, Topic 3.3 Demonstration	worksheet10	lec10-slides		

	Fri	Feb 06	Lecture 11: Barrier Synchronization	Module 1: Section 3.4	Topic 3.4 Lecture , Topic 3.4 Demonstration	worksheet11	lec11-slides		**Lecture & demo quizzes for topics 2.2, 3.1, 3.2, 3.3, 3.4**
5	Mon	Feb 09	Lecture 12: Iterative Averaging Revisited, SPMD pattern	Module 1: Sections 3.5, 3.6	Topic 3.5 Lecture, Topic 3.5 Demonstration, Topic 3.6 Lecture, Topic 3.6 Demonstration	worksheet12	lec12-slides		
	Wed	Feb 11	Lecture 13: Java's ForkJoin Library			worksheet13	lec13-slides		**Homework 2**
	Fri	Feb 13	Lecture 14: Data-Driven Tasks and Data-Driven Futures	Module 1: Section 4.5	Topic 4.5 Lecture, Topic 4.5 Demonstration	worksheet14	lec14-slides	**Homework 3** **hw_3.zip**	**Lecture & demo quizzes for topics 3.5, 3.6, 4.5**
6	Mon	Feb 16	Lecture 15: Abstract vs. Real Performance			worksheet15	lec15-slides		
	Wed	Feb 18	Lecture 16: Phasers, Point-to-point Synchronization	Module 1: Sections 4.2, 4.3	Topic 4.2 Lecture, Topic 4.2 Demonstration, Topic 4.3 Lecture, Topic 4.3 Demonstration	worksheet16	lec16-slides		
	Fri	Feb 20	Lecture 17: Pipeline Parallelism, Signal Statement, Fuzzy Barriers	Module 1: Sections 4.4, 4.1	Topic 4.4 Lecture, Topic 4.4 Demonstration, Topic 4.1 Lecture, Topic 4.1 Demonstration,	worksheet17	lec17-slides		**Lecture & demo quizzes for topics 4.1, 4.2, 4.3, 4.4**
7	Mon	Feb 23	Lecture 18: Classification of Parallel Programs		Topic 4.6 Lecture, Topic 4.6 Demonstration	worksheet18	lec18-slides		
	Wed	Feb 25	Lecture 19: Midterm Summary, Take-home Exam 1 distributed				lec19-slides	**Exam 1**	
	Fri	Feb 27	No Lecture (Exam 1 due by 4pm today)						**Lecture & demo quizzes for topic 4.6, Exam 1**
-	M-F	Feb 28-Mar 08	Spring Break						
8	Mon	Mar 09	Lecture 20: Critical sections, Isolated construct, Parallel Spanning Tree algorithm	Module 1: Sections 3.5, 3.6	Topic 5.1 Lecture, Topic 5.1 Demonstration, Topic 5.2 Lecture, Topic 5.2 Demonstration, Topic 5.3 Lecture, Topic 5.3 Demonstration	worksheet20	lec20-slides		
	Wed	Mar 11	Lecture 21: Eureka-style Speculative Task Parallelism			worksheet21	lec21-slides		
	Fri	Mar 13	Lecture 22: Read-Write Isolation, Atomic variables		Topic 5.4 Lecture, Topic 5.4 Demonstration, Topic 5.5 Lecture, Topic 5.5 Demonstration, Topic 5.6 Lecture, Topic 5.6 Demonstration	worksheet22	lec22-slides	**Homework 4** **hw_4_eureka.zip**	**Homework 3, Lecture & demo quizzes for topics 5.1 to 5.6**
9	Mon	Mar 16	Lecture 23: Actors		Topic 6.1 Lecture, Topic 6.1 Demonstration, Topic 6.2 Lecture, Topic 6.2 Demonstration, Topic 6.3 Lecture, Topic 6.3 Demonstration	worksheet23	lec23-slides		
	Wed	Mar 18	Lecture 24: Actors (contd)		Topic 6.6 Lecture, Topic 6.6 Demonstration	worksheet24	lec24-slides		
	Fri	Mar 20	Lecture 25: Concurrent Objects, Linearizability of Concurrent Objects		Topic 6.4 Lecture, Topic 6.4 Demonstration, Topic 6.5 Lecture, Topic 6.5 Demonstration, Topic 7.4 Lecture	worksheet25	lec25-slides		**Lecture & demo quizzes for topics 6.1 - 6.6, 7.4**
10	Mon	Mar 23	Lecture 26: Intro to Java Threads		Topic 7.1 Lecture	worksheet26	lec26-slides		
	Wed	Mar 25	Lecture 27: Java Threads (contd), Java synchronized statement		Topic 7.2 Lecture	worksheet27	lec27-slides		

	Fri	Mar 27	Lecture 28: Java synchronized statement (contd), advanced locking		Topic 7.3 Lecture	worksheet28	lec28-slides		**Lecture & demo quizzes for topics 7.1, 7.2, 7.3**
11	Mon	Mar 30	Lecture 29: Safety and Liveness Properties		Topic 7.5 Lecture	worksheet29	lec29-slides		
	Wed	Apr 01	Lecture 30: Dining Philosophers Problem		Topic 7.6 Lecture	worksheet30	lec30-slides		
-	Fri	Apr 03	Midterm Recess						**Lecture & demo quizzes for topic s 7.5, 7.6**
12	Mon	Apr 06	Lecture 31: Task Affinity with Places			worksheet31	lec31-slides		
	Wed	Apr 08	Lecture 32: Apache Spark framework for cluster computing			worksheet32	lec32-slides		
	Fri	Apr 10	Lecture 33: Message Passing Interface (MPI)			worksheet33	lec33-slides		**Homework 4 (now due by 11:59pm on April 12th)**
13	Mon	Apr 13	Lecture 34: Message Passing Interface (MPI, contd)			worksheet34	lec34-slides	**Homework 5** **hw_5_boruvka.zip**	
	Wed	Apr 15	Lecture 35: PGAS languages			worksheet35	lec35-slides		
	Fri	Apr 17	Lecture 36: Memory Consistency Models			worksheet36	lec36-slides		
14	Mon	Apr 20	Lecture 37: GPU Computing			worksheet-37	lec37-slides		
	Wed	Apr 22	Lecture 38: Fortress language				lec38-slides		
	Fri	Apr 24	Lecture 39: Course Review (lectures 20-37), Last day of classes				lec39-slides		**Homework 5 (automatic extension till May 1)**
-	Tue	May 5	**Scheduled final exam during 0900-1200 (Herzstein Hall Amphitheatre)**						

Lab Schedule

Lab #	Date (2015)	Topic	Handouts	Code Examples
1	Jan 14	Infrastructure setup, Async-Finish Parallel Programming	lab1-handout	lab_1.zip
2	Jan 21	Abstract performance metrics with async & finish	lab2-handout	lab_2.zip
3	Jan 28	Futures and Data Race detection	lab3-handout	lab_3_futures.zip and lab_3_datarace.zip
4	Feb 04	Real Performance from Finish Accumulators and Loop-Level Parallelism	lab4-handout and lab4-slides	lab_4_forall.zip and lab_4_hjviz.zip
5	Feb 11	Loop Chunking and Barrier Synchronization	lab5-handout and lab5-slides	lab_5_onedimavg.zip
6	Feb 18	Futures vs. Data-Driven Futures	lab6-handout and lab6-slides	lab_6_ddfs_and_futures.zip
7	Feb 25	Unix / Command line Basics, Phasers	lab7-handout and lab7-slides	lab_7.zip
-	Mar 04	No lab this week — Spring Break		
8	Mar 11	Eureka-style Speculative Task Parallelism	lab8-handout	lab_8_eureka.zip
9	Mar 18	Isolated Statement and Atomic Variables	lab9-handout	lab_9.zip

10	Mar 25	Actors	lab10-handout	lab_10_actors.zip
11	Apr 01	Java Threads	lab11-handout and lab11-slides	lab_11_threads.zip
12	Apr 08	Java Locks	lab12-handout and lab12-slides	
13	Apr 15	Apache Spark	lab13-handout	32big.zip
14	Apr 22	Message Passing Interface (MPI)	lab14-handout	

Grading, Honor Code Policy, Processes and Procedures

Grading will be based on your performance on five homeworks (weighted 40% in all), two exams (weighted 20% each), weekly lab exercises (weighted 10% in all), and class participation including worksheets, in-class Q&A, Piazza participation, and online quizzes (weighted 10% in all).

The purpose of the homeworks is to train you to solve problems and to help deepen your understanding of concepts introduced in class. Homeworks are due on the dates and times specified in the course schedule. Please turn in all your homeworks using the subversion system set up for the class. 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.

As in COMP 321, all students will be given 3 slip days to use throughout the semester. When you use a slip day, you will receive up to 24 additional hours to complete the assignment. You may use these slip days in any way you see fit (3 days on one assignment, 1 day each on 3 assignments, etc.). The only requirement for use of your slip days is that you e-mail the instructors *prior* to the time the assignment is due. On group projects, each student in the group must use a slip day in order to extend the deadline for the assignment. When slip days are used, you should clearly indicate so at the beginning of the assignment writeup. Other than slip days, no extensions will be given unless there are exceptional circumstances (such as severe sickness, not because you have too much other work). Such extensions must be requested and approved by the instructor (via e-mail, phone, or in person) before the due date for the assignment. Last minute requests are likely to be denied.

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 homework 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 (as shown here). Exams 1 and 2 test your individual understanding and knowledge of the material. Exams are closed-book, and collaboration on exams is strictly forbidden. Finally, it is also your responsibility to protect your homeworks and exams from unauthorized access.

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.