

OAKLAND UNIVERSITY, School of Engineering and Computer Science

CSE 436/536 Concurrent and Multicore Programming, Summer 2016

Instructor: Yonghong Yan (www.secs.oakland.edu/~yan), yan@oakland.edu

Office: 534 Engineering Center; **Tel:** (248) 370 4087

Office Hours: After class or by appointment

Meeting Time: 5:30 pm – 8:50 pm Monday and Wednesday; Check <http://www.oakland.edu/important-dates> for other important date/deadline for Summer 2015 semester

Class Room: EC 550

Course Catalog Description:

This course will focus on concepts, theory, design and implementation of concurrent programs for multi-core computers, multi-core programming methodologies. Topics covered include mutual exclusion, memory model and thread-based parallelism, fork-join framework, locks, parallel control flow, concurrent data structures.

4.000 Credit hours

4.000 Lecture hours

Levels: Graduate, Post Bachelor, Graduate Professional Development, Doctoral & Ed Specialist, Undergraduate

Schedule Types: Main Campus Classroom, Lecture

Computer Science & Engineering Division Computer Science & Engineering Department

Requirements:

Minimum requirements include good reasoning and analytical skills and skills of C programming, e.g. the use of macro, pointer, array, struct, union, function pointer, and library for memory allocation and de-allocation (malloc and free). Knowledge or courses of computer architecture (memory hierarchy, cache, virtual address) and data structures will be necessary for performing well for the class. Knowledge or courses of programming languages and compilers, and operating systems will also help. Familiarity with Linux environment will be important for the assignments.

Course Objectives:

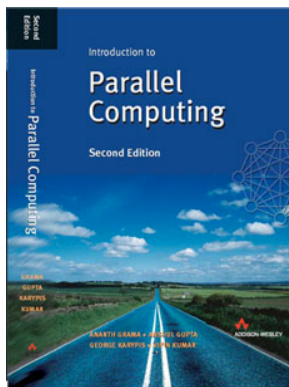
Upon completion of course, successful students shall be able to

1. Describe benefits and applications of concurrent and parallel programming.
2. Explain key concepts in parallel computer architectures, e.g. shared memory system, distributed system, NUMA and cache coherence.
3. Understand principles for concurrent program design, e.g. decomposition of works, task and data parallelism, processor mapping, mutual exclusion, locks.
4. Write parallel program using OpenMP, Cilkplus, CUDA, MPI programming models.
5. Perform analysis of parallel program problem.

Textbook and materials:

Most of the materials can be found from Internet and I will provide a list of links for those resources. There is no **required** textbook. The two recommended textbooks, if you really need ones, are:

Recommended Textbook:



Introduction to Parallel Computing, 2nd Edition

By Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar
Addison-Wesley, 2003

<http://www-users.cs.umn.edu/~karypis/parbook/>

Grade Assessment:

Course grade will be a combination of 4 homeworks (each 15% for total 60%), one final exam (40%) and 5% class participation.

Homework and Projects:

In those homeworks, you will apply theory and practice programming skills we discuss during the class. They require both correct programming and good description and discussion in the written report, which should present the approach you take and the results you find in your program. The goal for the project is to investigate specific challenging problem and develop solutions. You will be guided to study related work, identify the problem, develop solutions, perform experiments, analyze your results and report the findings. You will present your project to the class and your report should be in the form of publishable paper. Unless it is announced as teamwork, all assignments and class activities will be completed **individually**.

Behavioral Contract: (For lectures and homework/project)

- This class is offered in classroom with PCs and will include both lecture and lab practices; please read the following carefully, students will be asked to leave the labs if they do not follow this contract:
 - Digital devices such as: cell phones, tablets, ipod, mp3 players... etc need to be placed out of sight and must be set to silent. (I will keep my cell phone avail, for emergency preparedness)
 - No headphones are allowed.
 - Students are expected to use the PCs only for this class's material and applications.
 - Surfing the net for Social sites and being on Social Media are not allowed **during lecture hours**. Anyone who is not following this rule will be asked to leave the laboratory/lecture.

Attendance

- Attendance is required in all of the classes; it is part of a student's grade not an extra credit.
- **If you miss more than 2 classes you will not receive any credit for attendance. (The 2 classes include excused absences.)**
- Class participation is part of your grade make sure to participate in our class practices.

Late Policy: (PLEASE READ CAREFULLY)

Each assignment has a due date and a cutoff date, which is 3 days after due date. You will have a total of 7 "slip days" throughout the semester that you can use at your discretion to turn in programming assignments past the posted due date. Slip days are used in whole day increments. Once your slip days are consumed, late programming assignments will be penalized at 10% per day. Assignment submission will not be accepted after the cutoff date.

Public Website and Moodle:

- The public website is from <http://passlab.github.io/CSE436536> that we will use to post course public materials including notes, schedule, assignments, etc.
- The private website is located at <https://moodle.oakland.edu/moodle> for private use including assignment submission and other things that are not eligible for publishing on Internet.

Academic Conduct: *Expected conduct on assignments and exams*

Although students may discuss an assignment, each student should complete his or her assignment individually. Copying of another's assignment is not permitted. It is assumed that ALL work throughout the term is your own. Discussions during an exam or quiz are not permitted. Cheating during an exam or quiz is not permitted. It is assumed that ALL WORK THROUGHOUT THE TERM IS YOUR OWN! **Discussion of lab assignments are permitted but copying of assignments is not! Handing in a lab assignment or exam that was essentially copied from someone else does constitute as cheating.** All of the tests are closed book unless it was told otherwise. Obtaining help from notes, another individual or from hand held computing devices during an exam is regarded as cheating. The Oakland University Academic Conduct Policy can be found at <http://www4.oakland.edu/?id=1610&sid=75>. Cheating on examinations, plagiarism, falsifying reports/records, and unauthorized collaboration, access, or modifying of computer programs are considered serious breaches of academic conduct. The Oakland University policy on academic conduct will be strictly followed with no exceptions. See catalog under Academic Policies and Procedures.

Tentative class content and schedule:

Week	Date	Week date	Class	Content	Assignment
1	05/09	Monday	1	Introduction	
	05/09	Monday	2	Review of C programming, Compiler, Makefile, Linux and SSH access	
	05/11	Wednesday	3	Parallel algorithm design 1	
	05/11	Wednesday	4	Parallel program design 1 practice	Assignment 1
2	05/16	Monday		Rescheduled to 06/03 Friday and 06/10 Friday	
	05/16	Monday			
	05/18	Wednesday			
	05/18	Wednesday			
3	05/23	Monday	5	Parallel algorithm design 2	
	05/23	Monday	6	OpenMP	
	05/25	Wednesday	7	OpenMP	Assignment 1 due
	05/25	Wednesday	8	OpenMP practice and assignment 2	Assignment 2
4	05/30	Monday		Memorial day	
	05/30	Monday		Memorial day	
	06/01	Wednesday	9	Parallel program measurement and analysis	
	06/01	Wednesday	10	Parallel program measurement and analysis	
	06/03	Friday (reschedule)	11	Cilk	
	06/03	Friday (reschedule)	12	Cilk practice and performance analysis	
5	06/06	Monday	13	PThread	
	06/06	Monday	14	PThread and mutual exclusion	
	06/08	Wednesday	15	Parallel architecture	Assignment 2 due
	06/08	Wednesday	16	Parallel architecture and cache coherency	Assignment 3
	06/10	Friday (reschedule)	17	Manycore and GPU/CUDA	
	06/10	Friday (reschedule)	18	Manycore and GPU/CUDA	
6	06/13	Monday	19	Manycore and GPU/CUDA	
	06/13	Monday	20	GPU/CUDA practice and assignment 3	
	06/15	Wednesday	21	Manycore and Offloading	Assignment 3 due
	06/15	Wednesday	22	Distributed memory systems	Assignment 4
7	06/20	Monday	23	MPI	
	06/20	Monday	24	MPI practice and assignment 4	
	06/22	Wednesday	25	PGAS and UPC	
	06/22	Wednesday	26	Chapel and Final review	Last class
	06/25	Saturday		Last class day of Summer 01	Assignment 4 due
8	06/27	Monday		Final exam (6:30 - 9:30PM)	
	07/05	Tuesday		Grading completed	

Note: Assignment due 11:55PM on the date, cut-off date is 72 hours after the due date/time