CS636: Analysis of Concurrent Programs

Swarnendu Biswas

Semester 2018-2019-II CSE, IIT Kanpur

Course Details



- CS636: Analysis of Concurrent Programs
- Semester 2018-2019-II
- Class hours: MW 3:30-4:45 PM KD 103
- Office hours: Tu 4-5 PM F 11-12 PM KD 302
- Webpage: https://www.cse.iitk.ac.in/users/swarnendu/courses/cs636/
- Discussion forum: **REGISTER** for CS636 on Piazza

Instructor Details

- Name: Swarnendu Biswas
- Office: KD 302
- Webpage: https://www.cse.iitk.ac.in/users/swarnendu
- Email: swarnendu@cse.iitk.ac.in



- Name: Nabhiraj Jain
- Office: KD 222
- Webpage: https://www.cse.iitk.ac.in/users/nabhiraj/
- Email: nabhiraj@cse.iitk.ac.in
- Name: Prafulla Saxena
- Office: KD 308
- Email: prafulla@cse.iitk.ac.in

Course Description

- We will discuss challenges and techniques with concurrent programming
 - Shared-memory programming, PThreads
 - Concurrency bugs
 - Synchronization primitives (locks)
 - Testing of concurrent programs
 - Memory consistency
 - Performance analysis

Prerequisites

- Some background in compilers, OS, and architecture would be handy
- Software and programming skills
 - It will mostly be programming assignments
 - Project WILL INVOLVE development with large infrastructures
 - For example, Intel Pin, LLVM, Soot, Jikes RVM
 - Goal: solve real-world research problems
 - Take steps to publish depending on progress
- Ability to read and criticize research papers

Guest Lectures

- We will probably have 2-4 lectures from other CS faculties
 - Possibly in March/April
 - Still working out the details
 - I will announce as soon as we finalize the content

Course Policies

- I am happy to clarify **EVERY REASONABLE** question
- Please be **ON TIME** to class!
- Please try to AVOID using laptops and/or mobile devices in class!
- Turn **OFF** your mobile phones!
- Email subjects **SHOULD** start with [CS636]
- Submitting your assignments late will mean losing points automatically. You will lose 10% for each day that you miss, for up to three days.

Evaluation

- Class/piazza participation 10%
- Paper reading and critiques 20%
- Assignments 50%
- Final exam 20%

- Class/piazza participation 10%
- Paper reading and critiques -20%
- Project 50%
 Final exam 20%

Evaluation 🔑

Class/piazza participation – 10%	Class/piazza participation – 10%
 Paper reading and critiques – 20% 	 Paper reading and critiques – 20%
• Ass • Fin Let me know your preference by 14 th Jan!	

Academic Integrity

- You MAY discuss concepts with classmates
- All assignments MUST be your own or your team's work when teamwork is permitted
- You MAY NOT search online for existing solutions related to the assignments, even as a reference
- Students caught CHEATING/PLAGIARIZING will automatically fail the course and will be reported to the institute





- Java Concurrency In Practice Brian Goetz et al.
- The Art of Multiprocessor Programming Maurice Herlihy and Nir Shavit
- Intel Threading Building Blocks: Outfitting C++ for Multi-Core Processor Parallelism James Reinders
- Automatic Parallelization: An Overview of Fundamental Compiler Techniques Samuel P. Midkiff
- A Primer on Memory Consistency and Cache Coherence Daniel J. Sorin et al.
- Parallel Computer Architecture: A Hardware/Software Approach D. E. Culler et al.
- An Introduction to Parallel Programming P. Pacheco
- Other handouts and research papers





- Java Concurrency In Practice Brian Goetz et al.
- The Art of Multiprocessor Programming Maurice Herlihy and Nir Shavit



- Parallel Compater Arenitecture. A maraware, sortware Approach D. L. cuner et al.
- An Introduction to Parallel Programming P. Pacheco
- Other handouts and research papers

What this course is not?

- This is not a programming tips and tricks course
 - We will discuss more generic abstract questions

- This is not an introductory course to any specific tool
 - However, we will probably make use of a few

Extending Collaborative Learning

• Make use of the office hours

Office hours: Tu 4-5 PM F 11-12 PM KD 302

• You are welcome to **PROVIDE** feedback anytime

Questions?