Syllabus

CS 554: Data-Intensive Computing

http://www.cs.iit.edu/~iraicu/teaching/CS554-S23/

Semester: Spring 2023 (01/10/23 – 05/04/23)
Lecture Time: Tue/Thur, 11:25AM - 12:40PM
Location: Stuart Building 201

Professor:

- **Dr. Ioan Raicu** (iraicu@cs.iit.edu)
  - Office Hours: Thursday 12:45PM-1:45PM (SB 226B)

TA:

- **Alexandru Orhean (**aorhean@hawk.iit.edu**)**
  - Office Hours: Tuesday 12:45PM-1:45PM (SB 007)

Ioan Raicu

Alex Orhean

Overview

This course is a tour through various research topics in Blockchain technologies. We will cover the basics of distributed systems, to put blockchains in context of why it is a distributed systems problem. We will discuss the major consensus approaches: Proof of Work, Proof of Stake, and Proof of Space. Our discussions will often be grounded in the context of deployed blockchain technologies (Bitcoin, Ethereum, Chia). We will cover crypto mining, dive into block chain technologies with a focus on decentralization, scalability, performance, security, and tokenomics. We will also cover algorithmic trading and back-testing. We will explore solutions and learn design principles for building large blockchain-based systems to support digital currencies. Our readings and discussions will help us identify research problems and understand methods and general approaches to design, implement, and evaluate blockchain-based systems. The course involves lectures, discussions of research papers, and a major team (up to 3 students) project that is led by the students, including a written proposal, mid-term report, final report, and final oral presentation. This course is geared for junior/senior level undergraduates and graduate students in computer science.

Required Texts

None; readings will be from published research online material.

Prerequisites

CS450 (Operating Systems). Other courses that might contribute to having a better in depth understanding of this course are CS542, CS546, CS451, CS550, CS551, CS552, CS553, CS570, and/or CS595 (VMs). These courses are not required, and the introductory content of some of those courses will be repeated in this class. Many of these graduate courses are part of the Master of Computer Science Specialization in Distributed and Cloud Computing.

Knowledge of the following topics is recommended to get the most out of this class:

1. Programming (C, C++, or Java)
2. Networking
3. Operating systems
4. Architecture
5. Distributed systems

Computer Usage

Computer systems that can be used for development of projects (more information about access to these will be passed in the first several lectures):

- Chameleon (500+ node testbed at IIT/Argonne)
- Mystic (50+ node reconfigurable testbed at IIT)
- GIGI (16-node Raspberry Pi cluster at IIT with 1PB storage)

Projects

There will be a major semester long project that will require the implementation of a real/simulated system, a written report, and an oral presentation. Projects will focus on storage systems related topics. The students will be given several project ideas during the brainstorming sessions, and students must pick 1 project to develop further into a proposal (2~3 pages). Students are encouraged to work in teams of 2 or 3 on these projects, and multiple teams can work on the same project (independently). These projects will generally be of high interest to the DataSys Laboratory and are likely sub-projects from existing ongoing research projects. These projects could have some interaction with PhD students in the DataSys lab who are leading the projects at large. Students who obtain excellent results on their projects will be encouraged to submit their work as a publication to highly respected conferences and workshops.

Some major projects and tools you should familiarize yourself with are (as they might be useful to your projects):

- Operating systems: Linux
- Scripting: BASH
- Source control: GitHub
- Programming languages: C/C++, Java, Python
- Programming models: Multi-Threading (PThreads/CUDA/OpenMP/OneAPI), Network sockets
- File systems: NFS
- Data organization: Databases, XSearch
- Cloud middleware: OpenStack
- Key/Value Stores: ZHT, Casandra, MongoDB, MemCached, DynamoDB
- Message Queues: SQS, Kafka, Fabriq
- Virtualization: Oracle Virtual Box, XEN, VMWare, KVM

Schedule

Tentative skeleton schedule:

- 01/10/2023: Syllabus
- 01/12/2023: Introduction to Distributed Systems
- 01/24/2023: Introduction to Blockchain Technologies
- 01/31/2023: Project Brainstorming
- 02/07/2023: Group formation due
- 02/14/2023: Project Proposal due
- …
- 03/23/2023: Project Midterm Progress Report/Presentation due
- …
- 04/27/2023: Project Final Presentations
- 05/04/2023: Project Final Reports Due
- 05/10/2023: Grades Due

Late Policy

Assignments will be due at 11:59PM on the day of the due date, through BlackBoard. There will be a 15-minute grace period. There will also be a 4-day late pass, where students can submit late assignments without penalty; the late pass can be used in 1-day increments spread out over multiple assignments. Only project related assignments can use the late day passes. Paper reviews you write for reading assignments cannot use late day passes, and will incur a 20% penalty per day for late submissions. Any eligible late submissions (e.g. project proposal, project mid-term report, and project final report) beyond the grace period and beyond the 4-day late pass, will be penalized 20% every day it is late.

Mailing lists

This course will use BlackBoard to facilitate discussions and communication. BB should be the primary mechanism of communication between the students and the professor and the TAs.

Participation

There will be 25% points of the overall grade that will be received for participation. Students will be assigned research papers throughout the semester that they need to lead a discussion for (10% will go towards this). Online students will be able to lead these discussions through a Zoom session live during class. Paper reviews will also go towards participation points with 15% of the points.

Grades

There are no exams or homework.

Grading Policies:

- Project Proposal: 15%
- Mid-semester Progress Report: 10%
- Final Oral Presentation: 30%
- Final Project Report: 20%
- Participation: 25%

The following grading scale will be used.

- **A: 87% ~ 100%**
- **B: 75% ~ 86%**
- **C: 60% ~ 74%**
- **E: 0% ~ 59%**

Course Outcomes

When a student completes this course, s/he should be able to:

1. Understand the importance of data-intensive computing
2. Understand the difference between cluster, grid, clouds, and supercomputing.
3. Understand how to build large scale distributed systems
4. Understand applications that require data-intensive computing
5. Understand trends in many-core computing and challenges that will come with them
6. Build distributed systems
7. Be familiar with multiple programming models
8. Read and understand systems research papers
9. Make a formal presentation on a technical topic
10. Write up a formal report (or even a research paper) on the project