Syllabus

CS 553: Cloud Computing

http://www.cs.iit.edu/~iraicu/teaching/CS553-S20/

Ioan Raicu

Alex Orhean

Poornima Nookala

Jaime Garcia

Semester: Spring 2020
Lecture Time: Monday/Wednesday, 11:25AM - 12:40PM
Location: John T. Rettaliata Engg Center 104

Professor:

- **Dr. Ioan Raicu** (iraicu@cs.iit.edu, 1-312-567-5704)
  - Office Hours (SB226B): Wednesday 12:45PM-1:45PM

Teaching Assistants

- **Alex Orhean** (aorhean@hawk.iit.edu)
  - Office Hours (SB007): Monday 12:45PM-1:45PM, Thursday 2:00PM-3:00PM
- **Poornima Nookala (**pnookala@hawk.iit.edu**)**
  - Office Hours (SB007): Tuesday 12:45PM-1:45PM, Friday 2:00PM-3:00PM
- **Jaime Garcia (**jcernudagarcia@hawk.iit.edu**)**
  - Office Hours (SB010): Monday 2:00PM-3:00PM, Thursday 12:45PM-1:45PM

Course Description

Cloud Computing is "A large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet." It has become a driving force for information technology over the past several years, and it is hinting at a future in which we won't compute on local computers, but on centralized facilities operated by third-party compute and storage utilities. Governments, research institutes, and industry leaders are rushing to adopt Cloud Computing to solve their ever-increasing computing and storage problems arising in the Internet Age. There are three main factors contributing to the surge and interests in Cloud Computing: 1) rapid decrease in hardware cost and increase in computing power and storage capacity, and the advent of multi-core architecture and modern supercomputers consisting of hundreds of thousands of cores; 2) the exponentially growing data size in scientific instrumentation/simulation and Internet publishing and archiving; and 3) the wide-spread adoption of Services Computing and Web 2.0 applications. This course is a tour through various topics and technologies related to Cloud Computing. Topics include distributed system models and

enabling technologies, computer clusters for scalable Computing, virtual machines and virtualization of clusters and datacenters, design of cloud computing platforms, cloud programming and software environments (Workflow Systems, MapReduce, Spark, Google App Engine, Amazon AWS, Microsoft Azure, and OpenStack), grid computing and resource management, P2P computing with overlay networks, ubiquitous computing with clouds and the Internet of things, and data-intensive distributed computing. The course involves lectures, projects, programming assignments, and exams. Prerequsites: CS450 or CS455.

Required Texts

We will be using the textbook Cloud Computing for Science and Engineering, by Ian Foster and Dennis B. Gannon. ISBN: 9780262037242 (Required)

Prerequisites

CS450 (Operating Systems) or CS455 (Data Communications). Other courses that might contribute to having a better in depth understanding of this course are CS451, CS542, CS546, CS550, CS551, CS554, CS562 and CS570. Many of these graduate courses are part of the Master of Computer Science Specialization in Distributed and Cloud Computing.

Detailed Course Topics

- Distributed System Models
- Parallel Computing
- Cloud Platform Architectures
- Cloud Programming
- Performance Evaluations

Assignments

There will be 10 assignments throughout the semester, each worth 5% to 10% of the total grade, and each taking about 1 to 2 weeks to complete. Some assignments might have extra credit components (up to 10% additional points). There will be 2 to 5 written assignments and 5 to 8 programming assignments. These assignments can be done individually, or in groups of 2 to 3 students. Students will be expected to have knowledge of Java, C and/or C++. It is expected that students know the basics of these languages. These assignments must all work in a Linux environment (in which they will be graded in).

Computer Usage

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

- **Amazon AWS (**https://aws.amazon.com**)**
- **Chameleon** (https://www.chameleoncloud.org)
- **IIT/CS Teaching Cluster** (https://fusion.cs.iit.edu)

Late Policy

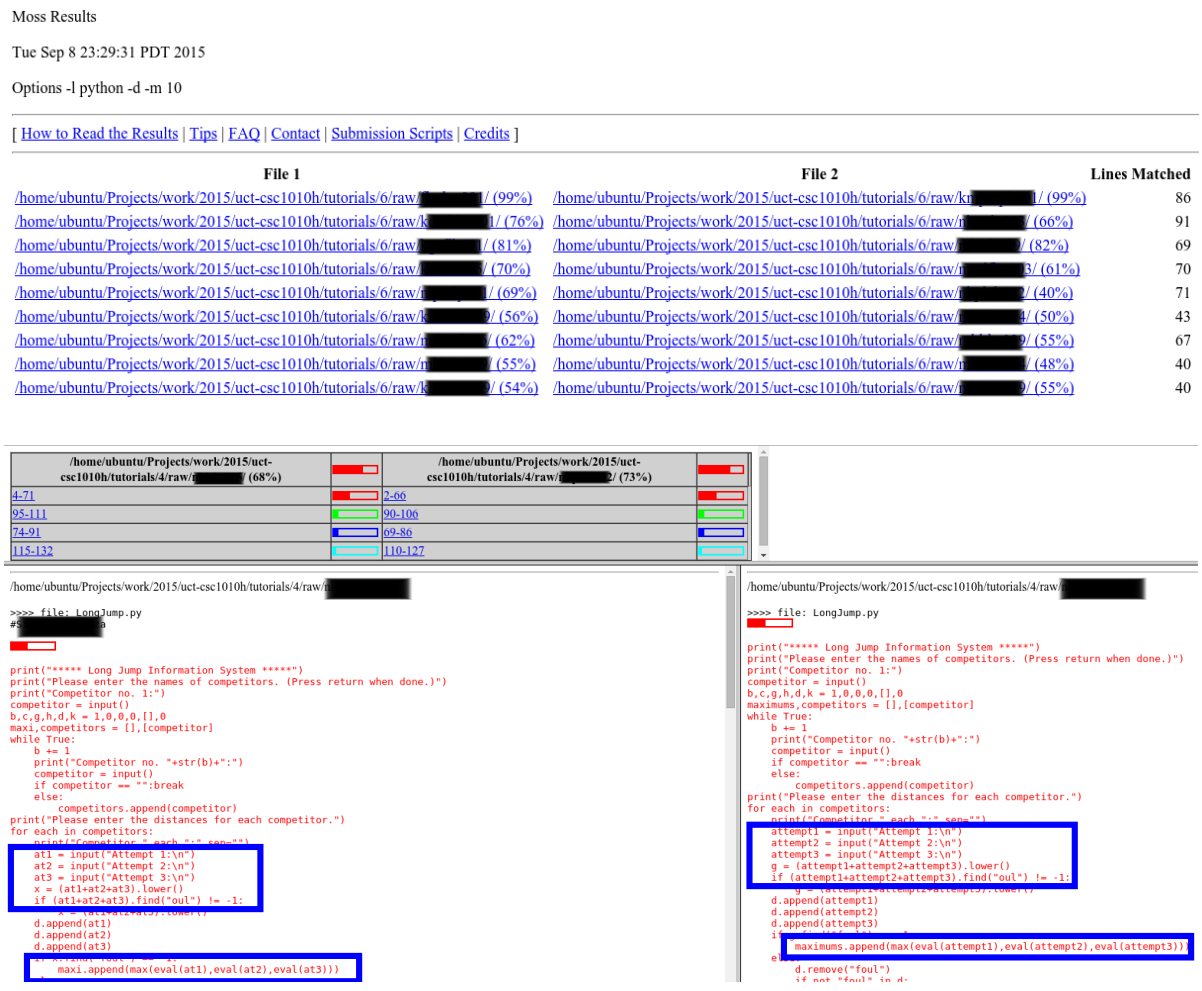
Assignments will be due at 11:59PM on the day of the due date. There will be a 15-minute grace period. Any assignments late beyond this grace period will be penalized 10% every day it is late.

Plagiarism Policy

Cheating will not be tolerated. We will use the MOSS: Measure Of Software Similarity system from Stanford (https://theory.stanford.edu/~aiken/moss/). It is used to automatically determine the similarity of programs (even if they are written in different programming languages). The supported languages are: C, C++, Java, C#, Python, Visual Basic, Javascript, FORTRAN, ML, Haskell, Lisp, Scheme, Pascal, Modula2, Ada, Perl, TCL, Matlab, VHDL, Verilog, Spice, MIPS assembly, a8086 assembly, a8086 assembly, MIPS assembly, HCL2.

You will receive a 0 on the assignment; extremely serious offences will fail the course and be reported to the university.

Some example screen shots from the MOSS system:



Moss Results

Tue Sep 8 23:29:31 PDT 2015

Options -l python -d -m 10

[How to Read the Results | Tips | FAQ | Contact | Submission Scripts | Credits]

File 1	File 2	Lines Matched
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (99%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (99%)	86
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (76%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (66%)	91
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (81%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (82%)	69
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (70%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (61%)	70
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (69%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (40%)	71
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (56%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (50%)	43
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (62%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (55%)	67
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (55%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (48%)	40
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (54%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/[redacted]/ (55%)	40

/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/4/raw/[redacted]/ (68%)		/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/4/raw/[redacted]/ (73%)	
4-71		2-66	
95-111		90-106	
74-91		69-86	
115-132		110-127	

/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/4/raw/[redacted]

```
>>>> file: LongJump.py
#[redacted]

print("***** Long Jump Information System *****")
print("Please enter the names of competitors. (Press return when done.)")
print("Competitor no. 1:")
competitor = input()
b,c,g,h,d,k = 1,0,0,0,0,[],0
max1,competitors = [],[competitor]
while True:
    b += 1
    print("Competitor no. "+str(b)+":")
    competitor = input()
    if competitor == "":break
    else:
        competitors.append(competitor)
print("Please enter the distances for each competitor.")
for each in competitors:
    print("Competitor " each ":" sep="")
    at1 = input("Attempt 1:\n")
    at2 = input("Attempt 2:\n")
    at3 = input("Attempt 3:\n")
    x = (at1+at2+at3).lower()
    if (at1+at2+at3).find("oul") != -1:
        [redacted]
    d.append(at1)
    d.append(at2)
    d.append(at3)
    [redacted]
        maxi.append(max(eval(at1),eval(at2),eval(at3)))
```

/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/4/raw/[redacted]

```
>>>> file: LongJump.py

print("***** Long Jump Information System *****")
print("Please enter the names of competitors. (Press return when done.)")
print("Competitor no. 1:")
competitor = input()
b,c,g,h,d,k = 1,0,0,0,[],0
maximums,competitors = [],[competitor]
while True:
    b += 1
    print("Competitor no. "+str(b)+":")
    competitor = input()
    if competitor == "":break
    else:
        competitors.append(competitor)
print("Please enter the distances for each competitor.")
for each in competitors:
    print("Competitor " each ":" sep="")
    attempt1 = input("Attempt 1:\n")
    attempt2 = input("Attempt 2:\n")
    attempt3 = input("Attempt 3:\n")
    g = (attempt1+attempt2+attempt3).lower()
    if (attempt1+attempt2+attempt3).find("oul") != -1:
        [redacted]
    d.append(attempt1)
    d.append(attempt2)
    d.append(attempt3)
    [redacted]
        maximums.append(max(eval(attempt1),eval(attempt2),eval(attempt3)))
    else:
        d.remove("foul")
```

Exams

There will be one exam worth 30% of the overall grade. If you are an in-class student, you must take the exams in class; for remote students, you can either come take it in class, or you can take the exam at an official testing center in a proctored environment. The exams will be individual. Students **WILL NOT BE ALLOWED** to use any material on the exam, such as textbooks, slides, notes, electronic devices (e.g. phones, eReaders, tablets, or laptops). The exam date, time, and location will be announced once the university has scheduled the final exams during the official final exam week between May 4th to May 9th 2020. **THERE WILL BE NO MAKEUP EXAMS.**

Grades

Grading Policies:

- **Assignments (10):** 70%
- **Exam (1):** 30% -- NO MAKEUPS

The following grading scale will be used. The scale will be adjusted downwards based on the overall performance of the entire class. Traditionally, in my classes, the class average score typically falls around 80% (a solid B-grade). There are two separate (but similar) grading scales, one for undergraduate students, and one for graduate students.

Undergraduate Students:

- **A: 85% ~ 100%**
- **B: 70% ~ 84%**
- **C: 60% ~ 69%**
- **D: 50%~59%**
- **E: 0% ~ 49%**

Graduate Students:

- **A: 85% ~ 100%**
- **B: 70% ~ 84%**
- **C: 50% ~ 69%**
- **E: 0% ~ 49%**

Discussion Forum

This course will use Piazza to facilitate discussions for assignments, at http://piazza.com/iit/spring2020/cs553/home (it has not been setup yet, more instructions will follow). Piazza should be the primary mechanism of communication between the students and the professor and the TAs. Direct email communication can be used, but make sure to clearly mark your emails in the subject with "CS553".