

# Syllabus

---

## CS 553: Cloud Computing

<http://www.cs.iit.edu/~iraicu/teaching/CS553-S18/>

**Semester:** Spring 2018

**Lecture Time:** Monday/Wednesday, 11:25AM - 12:40PM

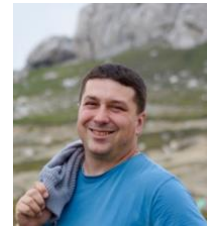
**Location:** Stuart Building 104

### Professor:

- **Dr. Ioan Raicu** ([iraicu@cs.iit.edu](mailto:iraicu@cs.iit.edu), 1-312-567-5704)
  - Office Hours: Wednesday 12:45PM-2:00PM (SB226B)

### Teaching Assistants

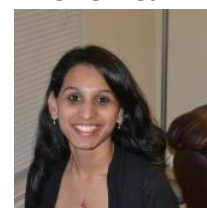
- **Alex Orhean** ([eorhean@hawk.iit.edu](mailto:eorhean@hawk.iit.edu))
  - Office Hours: Mondays 12:45PM-1:45PM / Wednesdays 10:15AM-11:15AM (SB007)
- **Poornima Nookala** ([pnookala@hawk.iit.edu](mailto:pnookala@hawk.iit.edu))
  - Office Hours: Mondays 10:15AM-11:15AM / Thursday 12:45PM-1:45PM (SB007)



Ioan Raicu



Alex Orhean



Poornima Nookala

## 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, Google App Engine, Amazon AWS, Microsoft Azure, and emerging cloud software stacks), 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. Prerequisites: [CS450](#) or [CS455](#).

## Required Texts

We will be using the textbook [Distributed and Cloud Computing: Clusters, Grids, Clouds, and the Future Internet](#) by [Kai Hwang](#), [Jack Dongarra](#) & [Geoffrey C. Fox](#). (Required)

## Prerequisites

[CS450](#) (OS) 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](#), CS552, [CS554](#), [CS570](#), and CS595 (VMs). 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
- Grid Computing
- Supercomputing Computing

## Programming Assignments

There will be 3 programming assignments throughout the semester, each worth 15% of the total grade, and each taking about 3~4 weeks to complete. These assignments must be completed individually. The projects will require knowledge of Java, C and/or C++. It is expected that students know the basics of these languages, but students are free to generally choose the languages they want to implement their assignments in. 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**

## Project

There will be 1 project which will be worth 10% of the grade. This assignment must be completed individually, and will require about 4 weeks to be completed. It will involve students to read online material, compute and generate graphs, and write a report.

## Late Policy

Assignments will be due at 11:59PM on the day of the due date. 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. Any late submissions beyond the grace period and beyond the 4-day late pass, will be penalized 20% 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
<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (99%)</a>	<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/ki██████████/ (99%)</a>	86
<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (76%)</a>	<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (66%)</a>	91
<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (81%)</a>	<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (82%)</a>	69
<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (70%)</a>	<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (61%)</a>	70
<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (69%)</a>	<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (40%)</a>	71
<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (56%)</a>	<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (50%)</a>	43
<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (62%)</a>	<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (55%)</a>	67
<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (55%)</a>	<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (48%)</a>	40
<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (54%)</a>	<a href="#">/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/██████████/ (55%)</a>	40

Line Range	Similarity	Line Range	Similarity
4-71	68%	2-60	73%
95-111		90-100	
74-91		69-86	
115-132		110-127	

```
>>> 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 = 1,0,0,0,0,0
maxi,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:
    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:
        x = "foul"
    d.append(at1)
    d.append(at2)
    d.append(at3)
    maxi.append(max(eval(at1),eval(at2),eval(at3)))

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
maximuns,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("Enter the distances for each competitor: ")
    at1 = input("Attempt 1:\n")
    at2 = input("Attempt 2:\n")
    at3 = input("Attempt 3:\n")
    g = (at1+at2+at3).lower()
    if (at1+at2+at3).find("oul") != -1:
        g = "foul"
    d.append(attempt1)
    d.append(attempt2)
    d.append(attempt3)
    maximuns.append(max(eval(attempt1),eval(attempt2),eval(attempt3)))
    d.remove("foul")
    if not "foul" in d:
```

# Exams

There will be one exam worth 45% 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 April 30<sup>th</sup> and May 4<sup>th</sup> 2018. **THERE WILL BE NO MAKEUP EXAMS.**

## Grades

Grading Policies:

- **Programming Assignments (3):** 45% -- can use late day passes (PA1=15%, PA2=15%, PA3=15%)
- **Project (1):** 10% -- can use late day passes
- **Exam (1):** 45% -- 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/spring2018/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".