

CS 553:

Cloud Computing

Syllabus

Ioan Raicu
Computer Science Department
Illinois Institute of Technology

Presented by:
Alex Orhean and Poornima Nookala

CS 553: Cloud Computing
January 8th, 2018

Introductions

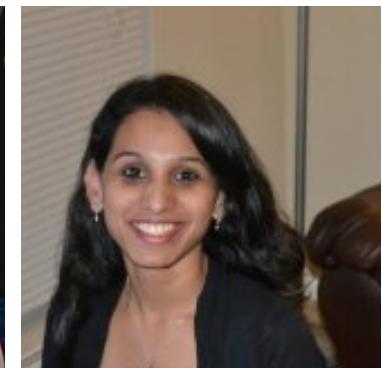
- **Professor:** **loan Raicu** <iraicu@cs.iit.edu>
 - Office Hours Time: Wednesday 12:45PM-1:45PM (SB226B)
 - More Information:
 - <http://www.cs.iit.edu/~iraicu/>
 - <http://datasys.cs.iit.edu/>
- **TAs**
 - **Alex Orhean**
 - aorhean@hawk.iit.edu
 - Mon 12:45PM-1:45PM (SB007)
 - Wed 10:15AM-11:15AM (SB007)
 - **Poornima Nookala**
 - pnookala@hawk.iit.edu
 - Mondays 10:15AM-11:15AM (SB007)
 - Thursday 12:45PM-1:45PM (SB007)



loan



Alex



Poornima

Course Overview

- This course is a tour through various topics and technologies related to Cloud Computing
- Explore solutions and learn design principles for building large network-based systems, to support compute and data intensive computing across geographically distributed infrastructures
- Discussions often grounded in real Cloud Computing systems:
 - Amazon AWS (EC2, S3, SQS), Microsoft Azure, Google AppEngine, OpenStack, Google's MapReduce, Yahoo's Hadoop, Spark, etc

Course Overview (cont)

- Understand methods and approaches to:
 - Design, implement, and evaluate cloud computing systems
- Course involves:
 - Lectures, outside invited speakers, programming assignments, projects, and exams
- Prerequisites:
 - Required: CS450 (Operating Systems) or CS455 (Data Communication)
 - Helpful: CS451, CS542, CS546, CS550, CS551, CS552, CS554, CS570, and CS595 (VMs)
- Required texts:
 - Distributed and Cloud Computing: Clusters, Grids, Clouds, and the Future Internet by Kai Hwang, Jack Dongarra & Geoffrey C. Fox.

Course Topics

- Distributed System Models
- Parallel Computing
- Cloud Platform Architectures
- Cloud Programming
- Grid Computing
- Supercomputing

Assignments

- Programming Assignments
 - 3 assignments
 - Will give hands on experience with cloud computing programming
 - Individual assignments
 - Expected to know (or learn quickly) some of these languages and systems:
 - Linux, Virtual Machines, Amazon AWS, Hadoop, Spark, multi-threading, sockets, C/C++, Java, Python, Bash
- Project
 - 1 assignment
 - Will enforce theoretical foundation of cloud computing technologies
 - Individual assignment

Cheating will not be tolerated

- MOSS: Measure Of Software Similarity
<https://theory.stanford.edu/~aiken/moss/>
- Automatic system for determining the similarity of programs
 - We will compare to past submissions starting from 2011
- Supports many languages:
 - 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 assignment; extremely serious offences will fail the course**

MOSS Plagiarism Detection

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/f1.py / (99%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/k1.py / (99%)	86
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/k2.py / (76%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/i1.py / (66%)	91
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/f2.py / (81%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/f3.py / (82%)	69
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/f4.py / (70%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/i2.py / (61%)	70
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/f5.py / (69%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/f6.py / (40%)	71
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/k3.py / (56%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/f7.py / (50%)	43
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/n1.py / (62%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/f8.py / (55%)	67
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/n2.py / (55%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/f9.py / (48%)	40
/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/k4.py / (54%)	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/6/raw/f10.py / (55%)	40

MOSS Plagiarism Detection

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

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

>>> file: LongJump.py
#S [REDACTED] a
[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
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+" : ", end="")
    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 = (at1+at2+at3).lower()
    d.append(at1)
    d.append(at2)
    d.append(at3)
    if x.find('oul') == -1:
        maxi.append(max(eval(at1),eval(at2),eval(at3)))

```

```
/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
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+":", end="")
    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:
        g = (attempt1+attempt2+attempt3).lower()
    d.append(attempt1)
    d.append(attempt2)
    d.append(attempt3)
    if g.find('oul') == -1:
        maximums.append(max(eval(attempt1),eval(attempt2),eval(attempt3)))
    else:
        d.remove("foul")
        if not "foul" in d:

```

Exams

- 1 Final Exam
- The exam will be individual
 - Closed book or notes
 - No electronic devices such as phones, eReaders, tables, or laptops will be allowed
 - The exam is worth 45% of the final grade
- Schedule:
 - Exam will take place during official exam week between April 30th and May 4th 2018
- **There will be no makeup exam.**

Late Policy

- Assignments will be due at 11:59PM on the date they are due; 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
- Exams
 - There will not be any makeup exams; do not miss any exam or you will get a 0

Grading

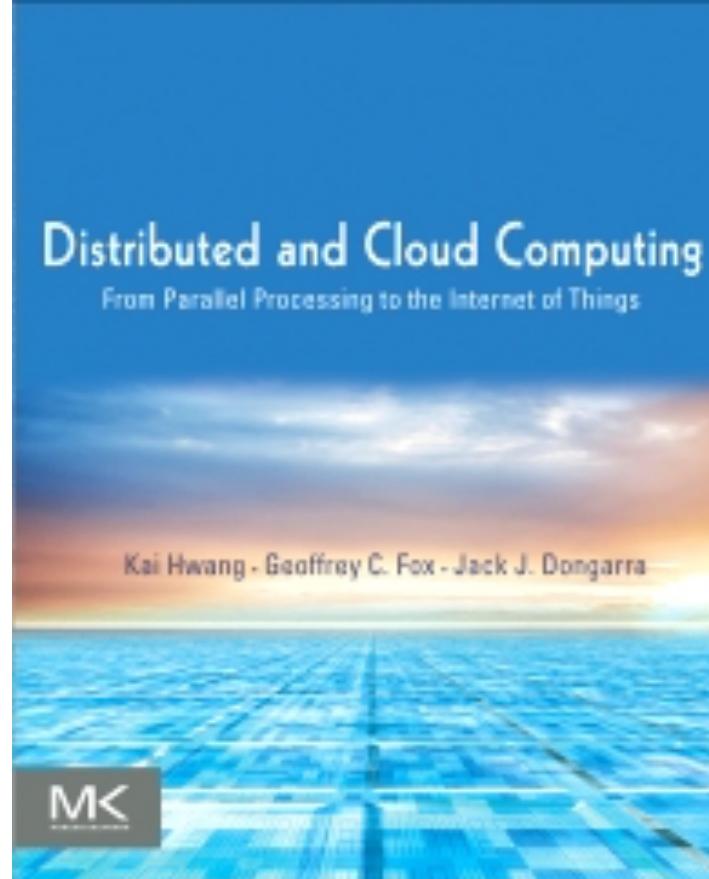
- Breakdown:
 - Programming Assignments (3): 45% -- can use late day passes (PA1=15%, PA2=15%, PA3=15%)
 - Project (1): 10% -- can use late day passes
 - Exam (1): 45% -- NO MAKEUPS
- Scale (graduate students):
 - **A:** 85% ~ 100%
 - **B:** 70% ~ 84% → class average
 - **C:** 50% ~ 69%
 - **E:** 0% ~ 49%

Grading (undergrads)

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

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.



Questions

- Write me:
 - iraicu@cs.iit.edu
- Call me:
 - 1-312-567-5704
- Write the TAs and me:
 - aorhean@hawk.iit.edu
 - pnookala@hawk.iit.edu
- Online discussion forum:
 - <http://piazza.com/iit/spring2018/cs553/home>