Comparative Programming Languages

ITP 499 (3 Units)

USC Viterbi School of Engineering

Fall 2018

Description

While exploring different programming languages, this class will serve as a platform for understanding how to approach problem solving with unfamiliar tools.

Objective

This course is as much a study of languages as it is a study of how to learn languages. By the end of the semester, students will have:

1. Experience with different programming techniques and paradigms
2. Experience comparing programming languages and evaluating them for suitability for a particular task

Concepts

Exploration of programming paradigms, debugging

Prerequisites

Three semesters of programming

Instructor	Nathan Greenfield
Office Hours	Listed on Blackboard under Contacts
Lab Assistants	Listed on Blackboard under Contacts
Lecture / Lab	110 minutes, twice weekly

Textbook

Seven Languages in Seven Weeks. Bruce A. Tate. O'Reilly Media Inc. ISBN: 9781934356593
Head First Ruby. Jay McGavren. O'Reilly Media Inc. ISBN: 9781449372651
Introducing Go. Caleb Doxsey. O'Reilly Media Inc. ISBN: 9781491941959
Learning Scala. Jason Swartz. O'Reilly Media Inc. ISBN: 9781449367930
Learn You a Haskell for Great Good!. Miran Lipovaca. No Starch Press. ISBN: 9781593272838
Many of these titles are available on Safari Books online, which is available through the USC Library (https://libraries.usc.edu/databases/safari-technical-books)

Course website

All course material will be on Blackboard (http://blackboard.usc.edu).
We will use Piazza (http://piazza.com/) as an online question and discussion forum.

Course Structure

The course will review 4 languages over the semester. Lectures will cover the languages broadly – you will be expected to learn the details of each language's syntax on your own. There will be in-class lab exercises to complete at almost every class meeting. Additionally, there will be weekly homework assignments. All homework assignments must be completed *individually* and outside of regularly scheduled class meetings.
Regular class meetings will feature a 60-minute lecture followed by an in-class lab exercises. These "labs" will immediately apply material from lecture and serve as an introduction to the other programming assignments.
There are two exams in this course.

Grading

The following percentage breakdown will be used in determining the grade for the course.

In class lab exercises	10%
Homework assignments	50%
Midterm exam	20%
Final exam	20%
Total	**100%**

Grading Scale

The following shows the grading scale to be used to determine the letter grade.

93% and above	A
90% - 92%	A-
87% - 89%	B+
83% - 86%	B
80% - 82%	B-
77% - 79%	C+
73% - 76%	C
70% - 72%	C-
67% - 69%	D+
64% - 66%	D
63% and below	F

Policies

Lab exercises

There will be lab exercises after most lectures. These exercises will be immediate application of the material presented in lecture. These exercises will be graded as pass/fail. For credit on each lab exercise you must complete the exercise before class time has ended. Each lab exercise will contribute to your overall grade. There is no way to make up a missed lab exercise.

Homework assignments

Each homework assignment must be completely individually. There are no group projects in this class.

It is your responsibility to submit your all homework assignments on or before the due date. Homework assignments turned in one day late will have 20% of the total points deducted from the graded score. Homework assignments turned in two days late will have 50% of the total points deducted from the graded score. After two days, submissions will not be accepted and you will receive a 0.

All homework assignments must be digitally submitted through Blackboard except when otherwise specified by the course staff. Do not email homework assignments to the instructor or lab assistant.

Homework assignment questions should be posted to the online question forum. Class time is for lecture and lab assignments only. Do not send any email to the instructor regarding homework assignments or ask specific homework questions during the lecture sessions. You are encouraged to attend the office hours for homework related questions.

Assignments

All course work must be completely *individually*. There are no group projects in this class.

The assignments will be posted on Blackboard in the "Assignments" section. Each assignment will include instructions, a due date, and a link for electronic submission. Assignments must be submitted using this link.

All course work must be digitally submitted through Blackboard except when otherwise specified by the course staff. Do not email assignments to the instructor or lab assistant.

Assignment questions should be posted to the online question forum. Class time is for lecture and labs only. Do not send any email to the instructor regarding assignments or ask specific assignment questions during the lecture sessions. You are encouraged to attend the instructor's office hours for assignment related questions.

Policies (continued)

Exams

Make-ups are only allowed under extraordinary circumstances. Students must provide a satisfactory reason (as determined by the instructor) along with proper documentation. There are two exams: a midterm and a final. These exams are comprehensive of all topics covered.

Lab facilities

You are encouraged to save your work using a USB flash drive or a website such as Dropbox. You must keep a copy of all coursework. You will not be able to save your work on the ITP lab computers. Any work saved to the computer will be erased after restarting the computer.
ITP is not responsible for any work lost.
ITP will have open lab hours starting the third week of the semester. The open labs may not have course staff there. These lab times are there in case you need extra time to complete your work.

Incomplete and Missing Grades

Excerpts for this section have been taken from the University Grading Handbook, located at http://www.usc.edu/dept/ARR/grades/gradinghandbook/index.html. Please see the link for more details on this and any other grading concerns.
A grade of Missing Grade (MG) "should only be assigned in unique or unusual situations… for those cases in which a student does not complete work for the course before the semester ends. All missing grades must be resolved by the instructor through the Correction of Grade Process. One calendar year is allowed to resolve a MG. If an MG is not resolved [within] one year, the grade is changed to [Unofficial Withdrawal] UW and will be calculated into the grade point average a zero grade points."
A grade of Incomplete (IN) "is assigned when work is no completed because of documented illness or other 'emergency' **occurring after the twelfth week** of the semester (or 12th week equivalency for any course scheduled for less than 15 weeks)."

Academic Conduct and Support Systems

Academic Conduct:

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Part B, Section 11, "Behavior Violating University Standards" https://policy.usc.edu/scampus-part-b/. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct.

Support Systems:

Student Counseling Services (SCS) - (213) 740-7711 – 24/7 on call

Free and confidential mental health treatment for students, including short-term psychotherapy, group counseling, stress fitness workshops, and crisis intervention. https://engemannshc.usc.edu/counseling/

National Suicide Prevention Lifeline - 1-800-273-8255

Provides free and confidential emotional support to people in suicidal crisis or emotional distress 24 hours a day, 7 days a week. http://www.suicidepreventionlifeline.org

Relationship & Sexual Violence Prevention Services (RSVP) - (213) 740-4900 - 24/7 on call

Free and confidential therapy services, workshops, and training for situations related to gender-based harm. https://engemannshc.usc.edu/rsvp/

Sexual Assault Resource Center

For more information about how to get help or help a survivor, rights, reporting options, and additional resources, visit the website:http://sarc.usc.edu/

Office of Equity and Diversity (OED)/Title IX compliance – (213) 740-5086

Works with faculty, staff, visitors, applicants, and students around issues of protected class. https://equity.usc.edu/

Bias Assessment Response and Support

Incidents of bias, hate crimes and microaggressions need to be reported allowing for appropriate investigation and response.https://studentaffairs.usc.edu/bias-assessment-response-support/

Student Support & Advocacy – (213) 821-4710

Assists students and families in resolving complex issues adversely affecting their success as a student EX: personal, financial, and academic.https://studentaffairs.usc.edu/ssa/

Diversity at USC – https://diversity.usc.edu/

Tabs for Events, Programs and Training, Task Force (including representatives for each school), Chronology, Participate, Resources for Students

A Further Note on Plagiarism

All submissions will be compared with current, previous, and future students' submissions using a code plagiarism identification program. If your code significantly matches another

student's submission, you will be reported to SJACS with the recommended penalty of an F in the course.

You may discuss solutions to specific problems with other students, but you should not look through another's code. The code can be from an online forum or another student, the source is immaterial – all code submitted in this course must be your own. Do not share your code with anyone else in this or future sections of the course, as allowing someone to copy your code carries the same penalty as copying the code yourself.

Comparative Programming Languages

ITP 499 (3 Units)

Course Outline

Note: Schedule subject to change

W	Summary	Topics	Assignments
1	Introduction	Course introduction Programming paradigms The languages	
	Measuring a language	Typing models Programming models Language interaction Decision constructs Core data structures	*Read*: Tate Ch 1
2	Ruby	Language introduction Interpreter	*Read*: Tate Ch 2 *Do*: Lab01
		Decisions, repetitions Typing model	*Read*: McGavren Ch 1 *Do*: Lab02 Ruby HW1 (due in Week 3)
3		Using classes Collections	*Read*: McGavren Chs 2 – 3 *Do*: Lab03
		Code blocks Running from a file	*Read*: McGavren Chs 4 – 5 *Do*: Lab04 Ruby HW2 (due in Week 4)
4		Writing classes Writing modules	*Read*: McGavren Chs 6 – 7 *Do*: Lab05
		Extending Ruby Wrap-up	*Do*: Ruby HW3 (due in Week 5)
5	Go	Language introduction Interpreter Typing model	*Read*: Doxsey Ch 1 *Do*: Lab06
		Data types Decisions & repetitions	*Read*: Doxsey Ch 2 – 4 *Do*: Lab07 Go HW1 (due in Week 6)
6		Collections Classes Interfaces	*Read*: Doxsey Ch 5 – 6 *Do*: Lab08

		Functions Built in features Packages	*Read*: Doxsey Ch 7 – 8 *Do*: Lab09 Go HW2 (due in Week 7)
7		Concurrency	*Read*: Doxsey Ch 9 – 10 *Do*: Lab10
		Shared resources Synchronization Wrap-up	*Do*: Go HW3 (due in Week 8)
8	Midterm	Midterm exam	
		Midterm review	
9	Scala	Language introduction JVM Concurrency Interpreter Object oriented vs functional programming	*Read*: Tate Ch 5, Swartz Ch 1 *Do*: Lab11
		Typing model Decisions, repetitions Classes Collections Scala class tree	*Read*: Swartz Ch 2 – 3 *Do*: Lab12 Scala HW1 (due in Week 10)
10		Functional programming introduced	*Read*: Swartz Ch 4 – 5 *Do*: Lab13
		Functional programming with classes Functional programming with collections	*Read*: Swartz Ch 6 – 7 *Do*: Lab14 Scala HW2 (due in Week 11)
11		Pattern matching	*Read*: Swartz Ch 8 – 10 *Do*: Lab15
		Concurrency Pros & Cons Industry use	*Do*: Scala HW3 (due in Week 12)
12	Haskell	Language introduction I/O Data types	*Read*: Tate Ch 8, Lipovaca Ch 1 *Do*: Lab16
		Collections	*Read*: Lipovaca Ch 2 *Do*: Lab17 Haskell HW1 (due in week 13)

13		Functions	*Read*: Lipovaca Ch 3 – 4 *Do*: Lab18
		Lazy evaluation	*Read*: Lipovaca Ch 5 – 6 *Do*: Lab19 Haskell HW2 (due in week 14)
14		Classes Types revisited	*Read*: Lipovaca Ch 7 – 9 *Do*: Lab20
		Recursion Monads	Haskell HW3 (due in week 15)
15	Wrap up	Summary Where to go from here	
		Final exam preparation	
	Final exam – as according to the final exam schedule		