

California State University, Long Beach
Department of Computer Engineering and Computer Science

Spring 18 CECS 424 – Organization of Programming Language section 07&08

Instructor: Minhthong Nguyen

Class Meetings: TuTh 12:30-1:20 am at PH1

Lab: TuTh 1:30-2:45 pm at ECS 403

Office: VEC 404

Office hours: Th 11:30 am – 12: 30 pm

Phone:

Email: minhthong.nguyen@csulb.edu

Prerequisite:

CECS 326 and CECS 328 all with a grade of "C" or better.

Course description:

Understanding the variety of programming languages and the design trade-offs between current programming language paradigms. Comparison of programming languages in their design, implementation, and run-time supports. Includes programming projects.

Course Objectives:

We will study the how's and why's of programming language design and implementation to a much greater level of detail than is possible in lower-level courses. We will cover essential programming language concepts like binding time, type systems, abstraction mechanisms, reflection, recursion, memory management, lambda calculus, and message passing. We will also contrast different language paradigms (procedural, object-oriented, functional, logic, concurrent) and complete programming assignments in each. Particular emphasis will be placed on the functional programming paradigm, and its concerns of higher-order functions and immutable state.

Course Policies

1. Homework: Late homework will not be accepted except for when there are serious and documented reasons. Collaboration on homework is strongly encouraged but everyone should write up his/her own work individually. Only some homework problems will be graded but it is recommended that you complete all problems since similar problems might appear on an exam. Copying other's work without giving credit is considered plagiarism and will receive no credit on that homework. All assignments contribute equally toward the homework grade.

2. Lab: There are roughly 6 labs throughout the semester. Lab should be completed and turned in by the due day.

3. Exam: There are three exams (including the final exam). NOTE: **You must take the final.**

4. Academic Integrity: Any individual caught cheating or plagiarizing on quizzes, homework, lab projects, or exams will receive an F in the course. Further penalties may include probation, suspension, or expulsion from the university. For more information, please refer to the University document at this web address:

http://www.csulb.edu/divisions/aa/catalog/current/academic_information/cheating_plagiarism.html

5. Attendance & Makeup Exam:

- Attendance is required for exams and will not be used in grade determination.
- Missing exam without a legitimate excuse will result 0 point for that exam.
- Only students with legitimate excuses (e.g. medical emergencies, etc.) will be allowed to make up missed exams.
- Please notify instructor as soon as possible regarding schedule conflict for the exam.

6. Need for assistance: Students, who need accommodations because of disability, should contact Disabled Student Service. More information can be found at:

<http://www.csulb.edu/divisions/students/dss/>

Text/Material:

Text: Programming Language Pragmatics Fourth Edition, Scott, M. Morgan Kaufmann Publishers, 2016.

Withdraw Deadlines

It is the student's responsibility to withdraw from classes. Please do not wait until the last day if you intend to drop the class. Further information regarding schedule adjustment, fees, refunds and deadlines can be found on:

<http://www.csulb.edu/enrollment-services/key-dates-and-deadlines>

Grade Assignment:

Grades will be determined using the following percentages:

Grade distribution

Homework	15%
Lab	15%
Exam 1	20% (~week 6)
Exam 2	20% (~week 12)
Final Exam	30%

<http://www.csulb.edu/enrollment-services/key-dates-and-deadlines>

Grading Scale

Grading Scale

≥ 90%	A	70%-79.99%	C	< 60%	F
80%-89.99%	B	60%-69.99%	D		

This syllabus is subject to change during the semester. Last updated: 01/23/2017

Week	Material	Chapter
1	Introduction	Read 1
1-2	Imperative Programming	
2-3	Types and Type Systems	Read 7, 8
3-5	Binding and Allocation	Read 3
6-7	Functional Programming	Read 11
8-9	Control Flow and Subroutines	Read 6, 9
10	Objects	Read 10
10-12	Concurrent Programming	Read 13
12-15	Logic Programming	Read 12