Syllabus

Instructor

Stan Warford   Office: RAC 112

Email: Stan.Warford@Pepperdine.edu   Phone : 310-506-4332 (on campus, x4332)

Office hours

Monday, 4:00 – 4:50   Tuesday, 1:00 – 1:50

Thursday 4:00 – 4:50   Friday, 3:00 – 3:50

nd by appointment

Course Web page

The course Web page will be used to post the assignments. You are responsible for checking it regularly.

https://stanwarford.com/ComputerSystems/

Objective

The primary objective of this course is for you to obtain a working knowledge of the lower levels of abstraction of a computer system. In Programming Principles II, you learned the Python programming language. Python is at Level 6, the high-order languages level. In this course you will learn Level 5, the assembly level, and Level 3, the instruction set architecture level, as well as C at Level 6.

Each level of abstraction in a computer system has a language, so learning the lower levels of abstraction is equivalent to learning new languages. The language at Level 5 is assembly language and the language at Level 3 is machine language. Your knowledge will be a working knowledge because you will learn how to program in these languages. Most programming in the industry today is at Level 6 and not at these lower levels. So, why learn these lower level languages? Because an in-depth understanding of how computers work is possible only by considering the design of the levels of abstraction and the relationships between the levels.

One secondary objective of this course is to continue to develop your programming skill at Level 6. Later in the course you will be required to write a large programming project in Java using the object-oriented techniques from the Programming Principles II course. The project is a language translation program known as an assembler.

Another secondary objective is to learn the theory of languages – syntax and semantics. Syntax theory applies to both natural and artificial languages, and will be the basis of your language translation program.

Non-required text

J. Stanley Warford, *Computer Systems*, Fifth edition, Jones and Bartlett Publishers, 2017.

The above textbook is under revision, and so no purchase is necessary. Prepublication manuscripts of the relevant chapters for the course will be provided on the course webpage.

Learning outcomes
The program learning outcomes (PLO) for the computer science/mathematics major are the ability to:
> 1. Implement algorithms
> 2. Prove computational theorems
> 3. Analyze computational systems
> 4. Communicate technical results

The course student learning outcomes (SLO) for CoSc 330, Computer Systems are the ability to:
> Translate a program in an high-order language to assembly language. (PLO 3)
> Convert data between high-level representation and binary. (PLO 3)
> Analyze an artificial language specified by a regular expression, finite state machine, or grammar. (PLO 3)
> Implement a translator from assembly language to machine language. (PLO 1)

Class schedule
The course web page has the schedule for the homework assignments, which are due twice weekly. The exam schedule is as follows:

Test 1, Thursday, September 11
Test 2, Thursday, October 9
Test 3, Monday, November 10
Final, Wednesday, December 10, 1:30 p.m. – 4:00 p.m.

Final grade
22% Homework
54% Tests (18% each test)
24% Final - cumulative

Late Homework Policy
There are two types of homework assignments in this course — written assignments and programming assignments. Both are handed in electronically on Pepperdine Canvass and are due at 11:55 p.m. on the due date. Half credit for written homework one assignment late. No credit thereafter. Note that your total homework score is equivalent to more than one test.

Programming Homework Policy
If your program does not compile (assemble)
> Automatic 20% of total points for that assignment
> No resubmission
Late submission
> Accepted up to the time of the following assignment
> Hand in with the following assignment
> Automatic 50% of graded score
Late submission that does not compile
> Automatic 10% of total points for that assignment

Course evaluations
Course evaluations are required online near the end of the semester and count as a homework assignment. After you complete the evaluation, email the notice of completion for this course to me.

Attendance policy
Attendance is important and may affect your final grade. You are responsible for making sure that your attendance has been recorded. Please provide written documentation for excused absences. There will be no makeup exams. If you miss an exam due to illness or an unexpected major emergency, the final exam score will be substituted for your missed exam score. Doctor's note required for all missed exams.

Accessibility notice
Any student with a documented disability (physical, learning, or psychological) needing academic accommodations should contact the Office of Student Accessibility (SAC 105, x6500) as early in the semester as possible. All discussions will remain confidential. Please visit https://www.pepperdine.edu/student-accessibility/ for additional information.

Academic integrity
See http://seaver.pepperdine.edu/academicintegrity/ for the academic integrity standards at Seaver College

Mission support
See https://www.pepperdine.edu/about/our-story/mission-vision/ for the mission statement of the university and https://seaver.pepperdine.edu/about/our-story/seaver-mission/for the mission statement of Seaver College. This course supports these mission statements by investigating the truth of its discipline and by preparing students for lives of service to others in the field of computer science.