CS 0449–Intro to Systems Software
Spring 2016

	Class	Recitation	Recitation
Time:	1:00–2:15pm	11:00-11:50pm	3:00-3:50pm
Days:	TTH	F	F
Room:	Benedum G36	6110 SENSQ	6110 SENSQ
Webpage:	http://www.cs.pitt.edu/~jacklange/teaching/cs449-s16		

Contact Information

Instructor: John Lange	
Office:	5407 Sennott Square
Email:	jacklange@cs.pitt.edu
Office Hours:	TTH. 3:00 - 4:00PM

TA: Yuyu Zhou	
Office:	5412 Sennott Square
Email:	yuyuzhou@cs.pitt.edu
Office Hours:	TBA

Description

A *Computer System* is comprised of both hardware and software working in concert to accomplish useful work. In this course, we will explore the issues of programming a real computer system by examining the abstractions, interfaces, and design decisions that influence the way that software runs. This includes the role the Operating System has in communication and resource management.

The perspective we will take is one of the lifecycle of a program from implementation to execution. The simple act of compiling and running a program, a sequence of events we often take for granted, is a complex interaction of many different components that work together to manage the computer's resources and perform the desired task. Together, these components form a working computer system.

Prerequisites

Before enrolling in this course, you need to have completed *CS 0441 – Discrete Structures for Computer Science*, *CS 0445 – Data Structures*, and have completed or be currently enrolled in *CS 0447 – Computer Organization and Assembly Language Programming*.

If you have any questions about the prerequisite material for the course, please ask at the beginning of the term.

Course Purposes and Goals

This course begins with the creation of executable programs in the C programming language. We will then explore the resultant program as it as stored on disk and as it is loaded for execution. Next, we will examine the interactions between our code and the code provided via libraries or the operating system to facilitate common, low-level tasks. Finally, we will look at the abstractions and resource management undertaken by the OS and its drivers to facilitate communication and hardware interaction.

The goals of the course are:

- Learning C programming. C is the most common language used for systems software.
- Exploring the layout of an executable program's code and data both as stored on disk and loaded into memory.
- Interacting with the abstractions that libraries and the operating system provide.
- Implementing our own abstractions, and manage hardware resources through device drivers.

Textbooks

[REQUIRED TEXT]

Oualline, Steve. *Practical C Programming*. O'Reilly, Sebastopol, CA, 1997.

ISBN: 1-56592-306-5

You may substitute instead (but you're responsible for the appropriate readings):

Kernighan, Brian W. and Ritchie, Dennis M. *C Programming Language*. 2nd Ed. Prentice Hall PTR, 1988.

ISBN: 0-13110-362-8

[ONLINE REQUIRED TEXTS]

There are three additional textbooks that are available online in PDF form that we will refer to throughout the term. Links can be found on the course website.

[Highly Recommended Text (Consider required if you want to go into systems)]

Bryant, Randal and O'Hallaron, David. *Computer Systems, A Programmer's Perspective.* 2nd Ed. Addison Wesley, 2010

ISBN: 0-13-610804-0

Class Policies

Exams: There will be two midterms and a final. The exams will be closed book/notes. The final exam will be as listed on the official university exam schedule in the normal classroom. Cheating on exams will not

be tolerated. Anyone caught cheating will be given a zero for the test or for the course and reported to the department following University procedures.

Projects: There will be 5 out-of-class assignments given. These are to be completed in the given time (no extensions will be given without a valid excuse. **LATE WORK IS NOT ACCEPTED**. Contact me *before* the deadline for clarifications.) These are meant to be your own work; anyone found to be collaborating will be disciplined in accordance to University policy. Cheating means (but is not limited to): using code from previous terms, other universities, your friends, finding it on the Internet, getting help from unapproved forums, or outsourcing it.

Labs and Quizzes: Attending recitation is an important part of this course. In recitation you will be able to work in a structured setting while completing small tasks (Labs). Concepts from class will be expanded upon and tested with unannounced quizzes.

Participation: Attendance will not be taken, but in a small class, any absence will be noticed. Several unexcused missed classes will adversely affect your grade.

Grading

Your grade will be based upon 3 exams, 5 projects, labs and quizzes (the lowest one of which will be dropped), and participation:

First Midterm	15%
Second Midterm	15%
Final Exam	15%
5 Projects	40% (8% each)
Labs and Quizzes	10%
Participation	5%
Total	100%

Disability Resources and Services:

If you have a disability for which you are requesting an accommodation, you are encouraged to contact both your instructor and Disability Resources and Services, 216 William Pitt Union, (412) 648-7890, as early as possible in the term. DRS will verify your disability and determine reasonable accommodations for this course.

Academic Integrity

Students are expected to comply with the University of Pittsburgh's Policy on Academic Integrity. Any student suspected of violating this obligation for any reason during the semester will be required to participate in the procedural process as outlined in the University Guidelines on Academic Integrity. For further information see: http://www.pitt.edu/~provost/ai1.html

Term Schedule

The daily topics are subject to change depending on our pace. They are there to assist you in the readings so you can focus on those concepts prior to class. The course website will host the definitive schedule, this is merely tentative.

The textbooks are indicated as follows:

- *Practical C* – Course text
- *Misurda* – CS 0449: Introduction to Systems Software (online)
- *ALP* – Advanced Linux Programming (online)

Week 1
READINGS: *Practical C*, Chapters 4, 6-9 TOPICS: • Intro to the Course • Introduction to common Linux commands • Introduction to C programming
Week 2
READINGS: *Misurda*: Chapter 1; *Practical C*: Chapters 5, 12, 13, 15, 18 TOPICS: • Pointers and addressing in C
Week 3
READINGS: *Practical C*, Chapters 10, 14, 16 TOPICS: • Dynamic memory management in C
Week 4
READINGS: *Misurda,* Chapter 7; *Practical C*, Chapters 17 TOPICS: • Dynamic memory (cont'd) • Data structures   ○ Linked Lists
Week 5
READINGS: *Misurda*, Chapters 2, 4 TOPICS: • Void * and templates • Midterm Review

Week 6
First Midterm Exam
Week 7
READINGS: Practical C, Chapter 11, *Misurda*, Chapters 6, *ALP* Chapter 9 TOPICS: • Bit operations • Assembly programming
Week 8
READINGS: *Misurda*, Chapters 3, 8; *ALP* Chapters 8 TOPICS: • Compiling and Linking • Introduction to OS architecture
Week 9
READINGS: *ALP* Chapters 3; *Misurda* Chapter 5 TOPICS: • Processes • Address Spaces • Virtual Memory
Spring Break
Spring Break: No classes
Week 10
Second Midterm Exam
Week 11
READINGS: *Misurda* Chapter 9 TOPICS: • Midterm answers • Introduction to threads
Week 12
READINGS: *Misurda*, Chapter 10; *ALP,* Chapter 4 TOPICS: • Threads cont'd • Synchronization
Week 13
READINGS: *Misurda*, Chapter 11, *ALP* Chapter 5 TOPICS: • IPC and sockets • Device Drivers

Week 14
READINGS: TOPICS: • Overview of Hardware • Review for the final exam
Finals Week
READINGS: Prepare for the final exam **Final Exam: In the normal classroom, date and time according to official Pitt schedule**