

CS 0449–Intro to Systems Software

Fall Term: 2181

	Class	Recitation	Recitation
Time:	3:00 – 4:15pm	2:00 – 2:50pm	12:00 – 12:50pm
Days:	MW	W	T
Room:	LAWRN 205	SENSQ 5505	SENSQ 5505
Webpage:	http://www.cs.pitt.edu/~jmisurda/teaching/cs0449.htm		

Contact Information

Instructor: Jonathan Misurda	
Office:	6203 SENSQ
Email:	jmisurda@cs.pitt.edu
Office Hours:	

TA:	
Office:	
Email:	
Office Hours:	

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 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 is 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.

Class Policies

Exams: There will be two midterms and a final. The exams will be closed book/notes. The final exam will be Wednesday, December 13, 2017, from 4:00–5:50pm in the normal classroom. Cheating on exams will not be tolerated. Anyone caught cheating will be given a zero for the test 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.

We will be using Moss, a tool from Stanford for determining inappropriate collaboration.

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%

The scale for the term will be:

Percentage	100	95	90	89	85	80	79	75	70	69	65	60	< 60
Letter	A+	A	A-	B+	B	B-	C+	C	C-	D+	D	D-	F

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 textbooks are indicated as follows:

- *Practical C* – Course text
- *Misurda* – CS 0449: Introduction to Systems Software (online)
- *ALP* – Advanced Linux Programming (online)
- *LDD3* – Linux Device Drivers, 3rd edition (online)

Week 1: 8/28/2017 - 8/30/2017

READINGS: *Practical C*, Chapters 1-4, 6, 8, 11

TOPICS:

- Intro to the Course
- C Programming
 - Data Types and Representation
 - Operators & Bitwise Manipulation
 - Control Flow

Week 2: 9/4/2017 - 9/6/2017

MONDAY, SEPTEMBER 4, 2017: Labor Day – No class

FRIDAY, SEPTEMBER 8, 2017: Fall term add/drop period ends

READINGS: *Practical C*, Chapters 5, 7, 9

TOPICS:

- C Programming (continued)
 - Arrays
 - Strings
 - Functions

Week 3: 9/11/2017 - 9/13/2017

READINGS: *Practical C*, Chapters 10, 12-14, 17; *Misurda*, Chapter 1

TOPICS:

- C Programming (continued)
 - Scope vs. Lifetime
 - Pointers
 - I/O
 - Console
 - Files

Week 4: 9/18/2017 - 9/20/2017

READINGS: *Practical C*, Chapters 15-23

TOPICS:

- C Programming (continued)
 - Memory management
 - malloc/free
 - Structures, and Unions

Week 5: 9/25/2017 - 9/27/2017

WEDNESDAY, SEPTEMBER 27, 2017: First Midterm Exam

TOPICS:

- Review for first midterm exam

Week 6: 10/2/2017 - 10/4/2017

READINGS: *Misurda*, Chapters 2 - 3

TOPICS:

- Program Representation
- Linking
 - Static
 - Dynamic
- Libraries, archives, shared objects
- Executable file formats

Week 7: 10/9/2017 - 10/11/2017

MONDAY, OCTOBER 9, 2017: No class – we will meet Tuesday, October 10, 2017 instead.

READINGS: *Misurda*, Chapters 4 - 6

TOPICS:

- Processes & Address Spaces
- Data Representation
 - Globals, constants
 - Activation Records
 - Arrays and Structures

Week 8: 10/16/2017 - 10/18/2017

READINGS: *Misurda*, Chapter 7; *ALP* Chapters 3, 8

TOPICS:

- Interaction with Operating System
 - interrupts (int 0x80, int 3)
 - calling convention/ABI
- Syscalls
- Signal Handling

Week 9: 10/23/2017 - 10/25/2017

FRIDAY, OCTOBER 27, 2017: Withdrawal Deadline (For “W” grade)

READINGS: *LDD3*, Chapters 1-2

TOPICS:

- Linux Device Drivers

Week 10: 10/30/2017 - 11/1/2017

WEDNESDAY, NOVEMBER 1, 2017: Second Midterm Exam

TOPICS:

- Review for second midterm exam

Week 11: 11/6/2017 - 11/8/2017

READINGS: *Practical C*, Chapters 7,10,18

TOPICS:

- Multi-file Development

- Providing an interface/API
- Header files
- Makefiles

Week 12: 11/13/2017 - 11/15/2017

READINGS: *Misurda*, Chapter 8; *ALP*, Chapter 3

TOPICS:

- Threading
 - User vs. Kernel Threading
 - Scheduling/yield/sleep
 - pthreads

Week 13: 11/20/2017 - 11/22/2017

WEDNESDAY, NOVEMBER 22, 2017: **Thanksgiving Break, University Closed**

READINGS: *Misurda*, Chapter 9

TOPICS:

- Deadlocks/Synchronization

Week 14: 11/27/2017 - 11/29/2017

READINGS: *Misurda*, Chapter 10

TOPICS:

- Communication and Networking
 - Berkley Sockets
 - Pipes
 - Shared Memory

Week 15: 12/4/2017 - 12/6/2017

READINGS: Prepare for the final exam

TOPICS:

- Finish up Networking
- Review for the final exam

Finals Week

Final Exam: Wednesday, December 13, 2017, from 4:00–5:50pm in the normal classroom