上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Shanghai Jiao Tong University

CS365 Systems Programming (Online)

Instructor Information	Chentao Wu Home Institution: Shanghai Jiao Tong University Email: wuct@cs.sjtu.edu.cn		
Term	December 17, 2020 - January 8, 2021	**Credits**	4 units
Course Delivery	The class will be delivered in the format of online. Other than recorded lecture videos, the instructor will arrange 4 hours' real-time interactions with students per week (via discussion forum, zoom meeting, and WeChat). The workload students are expected to complete to properly pass this course is about 20 hours per week.		
Required Texts (with ISBN)	Randal E. Bryant and David R. O'Hallaron, Computer Systems: A Programmer's Perspective, Third Edition, Pearson, 2016. ISBN: 9787111561279.		
Prerequisite	Data Structure, C/C++ Programming, Operating Systems		

Course Overview

This course provides a programmer's view of how computer systems execute programs, store information, and communicate. It enables students to become more effective programmers, especially in dealing with issues of performance, portability and robustness. It also serves as a foundation for courses on compilers, networks, operating systems, and computer architecture, where a deeper understanding of systems-level issues is required. Topics covered include: machine-level code and its generation by optimizing compilers, performance evaluation and optimization, computer arithmetic, memory organization and management, networking technology and protocols, and supporting concurrent computation.

Learning Outcomes

The following “realities” are some of the major areas where the abstractions you’ve learned in previous classes break down:

1. Int’s are not integers, Float’s are not reals. Our finite representations of numbers have significant limitations, and because of these limitations we sometimes have to think in terms of bit-level representations.
2. You’ve got to know assembly language. Even if you never write programs in assembly, The behavior of a program cannot be understood sometimes purely based on the abstraction of a high-level language. Further, understanding the effects of bugs requires familiarity with the machine-level model.
3. Memory matters. Computer memory is not unbounded. It must be allocated and managed. Memory referencing errors are especially pernicious. An erroneous updating of one object can cause a change in some logically unrelated object. Also, the combination of caching and virtual memory provides the functionality of a uniform unbounded address space, but not the performance.
4. There is more to performance than asymptotic complexity. Constant factors also matter. There are systematic ways to evaluate and improve program performance.
5. Computers do more than execute instructions. They also need to get data in and out and they interact with other systems over networks.

By the end of the course, you will understand these “realities” in some detail.

Course Structure

This course is a bridge for connecting different programming on computer science courses as below,

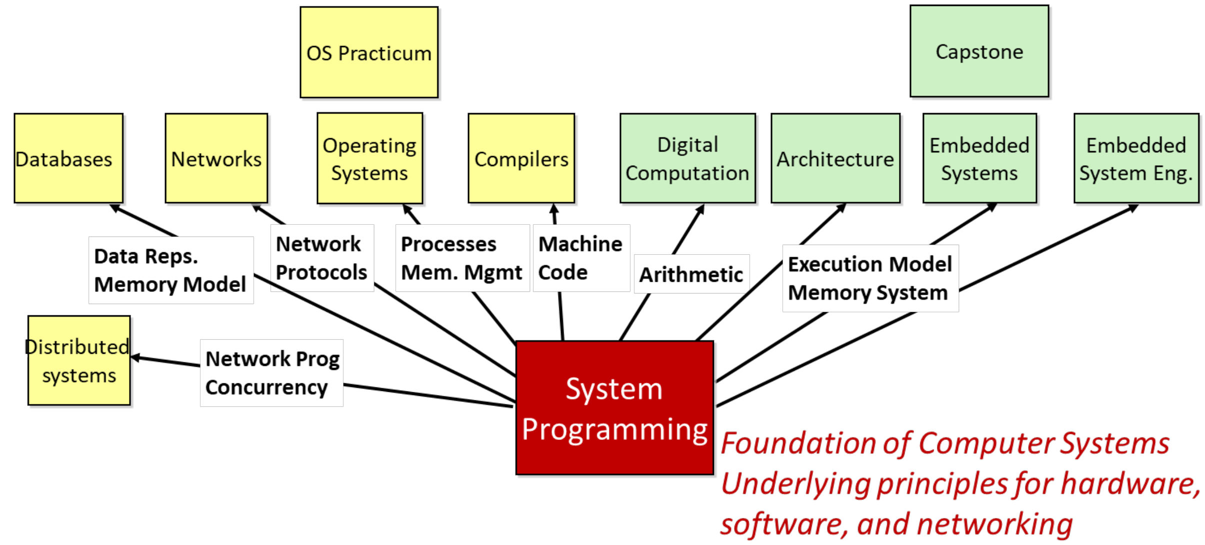
- Computer Architecture
  - Design pipelined processor in Verilog
- Operating Systems
  - Implement sample portions of operating system
- Compilers
  - Write compiler for simple language
- Networking
  - Implement and simulate network protocols

This course is a Programmer-Centric

- Purpose
  - to show that by knowing more about the underlying system, one can be more effective as a programmer

- Enable you to
  - Write programs that are more reliable and efficient
  - Incorporate features that require hooks into OS
  - E.g., concurrency, signal handlers
- Cover material in this course that you won't see elsewhere
  - Not just a course for dedicated hackers
  - We bring out the hidden hacker in everyone!

In the following is the related course on programming,



Foundation of Computer Systems
Underlying principles for hardware, software, and networking

Grading Policy

Attendance	10%
Middle Exam	20%
Projects/Homework	30%
Final Exam	40%

Grading Scale is as follows

Number grade	Letter grade	GPA
90-100	A	4.0
85-89	A-	3.7
80-84	B+	3.3
75-79	B	3.0
70-74	B-	2.7
67-69	C+	2.3
65-66	C	2.0
62-64	C-	1.7
60-61	D	1.0
≤59	F (Failure)	0

Class Schedule

Date	Lecture	Readings
Day 1	A Tour of Computer Systems	Chapter 1
Day 2	Representing and Manipulating Information	Chapter 2
Day 3	Machine-Level Representation of Programs	Chapter 3
Day 4	Online meeting, review & discussion	
Day 5	Processor Architecture	Chapter 4
Day 6	Optimizing Program Performance	Chapter 5
Day 7	The Memory Hierarchy	Chapter 6
Day 8	Online meeting, review & discussion	
Day 9	Middle Exam	
Day 10	Linking	Chapter 7
Day 11	Exceptional Control Flow	Chapter 8
Day 12	Virtual Memory	Chapter 9
Day 13	Online meeting, review & discussion	
Day 14	System-Level I/O	Chapter 10
Day 15	Network Programming	Chapter 11
Day 16	Concurrent Programming	Chapter 12
Day 17	Online meeting, review & discussion	
Day 18	Final Exam	