UCLA Computer Science 33, winter 2020. Introduction to Computer Organization

Course objective: Understand how computer systems work, so that you can write better software.

- Piazza
- Course material news
- Syllabus
- Assignments
- Grading

Instructor: Paul Eggert, Engineering VI 363. Office hours are Mondays 10:00–11:00 and Thursdays 09:30–10:30.

Teaching assistant, with office hours announced on CCLE: Yugo Watanabe <yugowatanabe@cs.ucla.edu>.

Lecture, 4 hours; discussion, 2 hours; outside study, 9 hours.

Prerequisite: Computer Science 32.

Introductory course on computer architecture, assembly language, and operating systems fundamentals. Number systems, machine language, and assembly language. Procedure calls, stacks, interrupts, and traps. Assemblers, linkers, and loaders. Operating systems concepts: processes and process management, input/output (I/O) programming, memory management, file systems. Letter grading.

Related Computer Science Curricula 2013 (CS2013) knowledge areas:

- AR/Machine Level Representation of Data
- AR/Assembly Level Machine Organization
- AR/Memory System Organization and Architecture
- AR/Interfacing and Communication
- AR/Functional Organization
- AR/Multiprocessing and Alternative Architectures
- AR/Performance Enhancements
- OS/Memory Management
- PD/Parallelism Fundamentals
- SF/Parallelism

Related IEEE Software Engineering Body of Knowledge, Version 3.0 (SWEBOK V3.0), 2014:

- CMP.cf.5. Computer organization
- CMP.ct.6. Error handling, exception handling, and fault tolerance

Related Computer Engineering Curricula 2016 (CE2016) knowledge units:

- CE-CAO-3. Instruction set architecture
- CE-CAO-4. Measuring performance
- CE-CAO-5. Computer arithmetic
- CE-CAO-6. Processor organization
- CE-CAO-7. Memory system organization and architectures
- CE-CAO-8. Input/Output interfacing and communication
- CE-CAO-10. Multi/Many-core architectures

$Id: index.html,v 1.21 2020/01/07 20:51:33 eggert Exp $