CS 008: COMPUTER ARCHITECTURE AND ORGANIZATION

Originator
mflora

Co-Contributor(s)

Name(s)
Kelbley, Robert
MacIntire, Doug

Justification / Rationale
Periodic update, add online modalities

Effective Term
Fall 2022

Credit Status
Credit - Degree Applicable

Subject
CS - Computer Science

Course Number
008

Full Course Title
Computer Architecture and Organization

Short Title
COMPUTER ARCH

Discipline

Disciplines List
Computer Science

Modality
Face-to-Face
100% Online
Hybrid

Catalog Description
The organization and behavior of computer systems at the assembly-language level. The translation of statements and constructs in a high-level language into sequences of machine instructions is studied, as well as the internal representation of simple data types and structures. Numerical computation is examined, noting the various data representation errors and potential procedural errors. Digital electronics with the Boolean algebra of logic gates is studied.

Schedule Description
The organization and behavior of real computer systems and data structures at the assembly-language level. Digital electronics and numerical computation is examined. Prerequisite: CS-007A null

Lecture Units
2
Lecture Semester Hours
36

Lab Units
1
Lab Semester Hours
54

In-class Hours
90

Out-of-class Hours
72

Total Course Units
3

Total Semester Hours
162

Prerequisite Course(s)
CS 007A

Required Text and Other Instructional Materials

Resource Type
Book

Open Educational Resource
No

Author
Vahid, Frank

Title
Introduction to Computer Systems and Assembly Programming

Year
2021

College Level
Yes

ISBN #
unique to each professor

Resource Type
Book

Open Educational Resource
No

Author
Stallings, W.

Title
Computer Organization and Architecture

Edition
11th

Publisher
Pearson

Year
2019

ISBN #
978-0-13-516093-0

Resource Type
Book
Open Educational Resource
No

Author
Smith, Bruce

Title
Raspberry Pi Operating System Assembly Language

Edition
4th

Publisher
CreateSpace Independent Publishing Platform

Year
2021

ISBN #
978-0648098737

Resource Type
Book
Open Educational Resource
Yes

Author
Plantz, R.

Title
Introduction to Computer Organization: ARM Assembly Language Using the Raspberry Pi

Publisher
http://bob.cs.sonoma.edu/IntroCompOrg-RPi/intro-co-rpi.html

Year
2021

ISBN #
none, available at http://bob.cs.sonoma.edu/IntroCompOrg-RPi/intro-co-rpi.html

---

Class Size Maximum
28

Entrance Skills
Demonstrate proficiency with programming development environment.

Requisite Course Objectives
CS 007A-Demonstrate effective use of a program development environment by entering/editing and executing programs.

Entrance Skills
Solve problems by designing and implementing algorithms.

Requisite Course Objectives

CS 007A-Synthesize problem-solving strategies to design algorithms implementing step-wise refinement for improving algorithmic efficiency and correctness by debugging.

Entrance Skills

Discuss basics of decimal and binary numbers.

Requisite Course Objectives

CS 007A-Convert decimal numbers to binary form and binary numbers to a decimal.

Entrance Skills

Discuss basic concepts of logic.

Requisite Course Objectives

CS 007A-Determine validity of a Boolean Logic expression.

Course Content

1. Bits, bytes, and words
2. Numeric data representation and number bases
3. Fixed- and floating-point systems
4. Signed and twos-complement representations
5. Representation of nonnumeric data (character codes, graphical data)
6. Representation of records and arrays
7. Basic organization of the von Neumann machine
8. Logic Gates and Logic Circuits
9. Control unit; instruction fetch, decode, and execution
10. Instruction sets and types (data manipulation, control, I/O)
11. Assembly/machine language programming
12. Instruction formats
13. Addressing modes
14. Subroutine call and return mechanisms
15. I/O and interrupts
16. History of computers and computer software

Lab Content

1. Arithmetic expression - add/sub
2. Memory load/store
3. Branch and less than instructions
4. Loops
5. Multiplications and divisions using shift operations
6. Multiplication instructions
7. Procedure calls
8. Nested procedures

Course Objectives

	Objectives
Objective 1	Describe the low-level storage of data, including the concepts of bits, bytes, and words.
Objective 2	Implement numeric data representation and calculate in different number bases.
Objective 3	Explain fixed- and floating-point systems.
Objective 4	Express numbers in signed and twos-complement representations.
Objective 5	Explain representation of nonnumeric data such as character codes and graphical data.
Objective 6	Describe and employ the representation of records and arrays in memory.

Objective 7	Explain the low-level architecture of computers, including the von-Neumann machine; components and data flow within the CPU; data flow to memory, registers and I/O devices; and Instruction Set Architecture.
Objective 8	Compare and contrast the different logic gates, and analyze logic circuits.
Objective 9	Describe the control unit; instruction fetch, decode, and execution.
Objective 10	Implement assembly instruction sets and types.
Objective 11	Build and debug assembly/machine language programming.
Objective 12	Compare and contrast different instruction formats and addressing modes.
Objective 13	Implement assembly code using subroutine calls and return mechanisms, I/O, and interrupts.
Objective 14	Explain the basic history of computers and computer software.

Student Learning Outcomes

	Upon satisfactory completion of this course, students will be able to:
Outcome 1	Explain and encode the low-level storage of data, including characters and other nonnumeric data, fixed and floating point numbers, signed and twos-complement representations of numbers, and representations of arrays.
Outcome 2	Identify and describe the low-level architecture of computers, including the von-Neumann machine; Boolean logic gates and basic logic circuits; processor components, architecture, behavior and control flow; and Instruction Set Architectures.
Outcome 3	Write simple assembly language programs to perform Input/Output (I/O), numerical, and logical calculations. Programs will include subroutines, branching, and loops.

Methods of Instruction

Method	Please provide a description or examples of how each instructional method will be used in this course.
Collaborative/Team	Take turns role-playing as designer/tester/developer in solving programming challenges to produce software meeting prescribed input/output specification.
Lecture	Computer architecture design and assembly programming practices and principles are introduced in concept and by example.
Laboratory	Students will practice developing the design principles introduced in lecture by writing assembly programs that solve problems of varying difficulty. Typically, students may be assigned to work either individually or in small groups to address the problem of writing code to accept input in a specific format and analyze that input produce a desired output.

Methods of Evaluation

Method	Please provide a description or examples of how each evaluation method will be used in this course.	Type of Assignment
Written homework	Pre-lab reading and question assignments that prepare students for lab projects.	Out of Class Only
Mid-term and final evaluations	There will be a midterm and a final exam, generally in written format, but this may be combined with some computer work. (4 hrs)	In Class Only
Laboratory projects	These will require students to solve problems from the their lab manuals while using object-oriented programming concepts introduced in lecture. (4 hrs/wk)	In Class Only

Assignments

Other In-class Assignments

1. Labs that require students to all use the same equipment/technology (such as Raspberry Pi kits), using college-provided equipment if possible for equity purposes

Other Out-of-class Assignments

1. Homework assignments that require students to all use the same equipment/technology (such as Raspberry Pi kits), using college-provided equipment if possible for equity purposes
2. Assigned reading and videos

Grade Methods

Letter Grade Only

Distance Education Checklist

Include the percentage of online and on-campus instruction you anticipate.

Online %

100

On-campus %

0

What will you be doing in the face-to-face sections of your course that necessitates a hybrid delivery vs a fully online delivery?

Although the course can be offered entirely online, it may also be offered hybrid to take advantage of collaboration activities that are more suited to in-person interaction.
Examinations can be given in a controlled location.

Lab Courses

How will the lab component of your course be differentiated from the lecture component of the course?

Lab assignments involve more interaction. For example, they may require students collaborate with a classmate, utilize a tutoring resource, or interview someone who is not part of the course.

From the COR list, what activities are specified as lab, and how will those be monitored by the instructor?

Lab activities are discussions and assignments that involve solving problems or exploring concepts with other students, with people not part of the course, or under the guidance of the professor or instructional support assistant. Discussions and other assignments that are completed in Canvas are monitored and evaluated by the professor. Assignments that do not take place in Canvas are evaluated by the professor based on write-ups (which may include summaries and feedback from the participants). Anonymous and non-anonymous feedback opportunities will be available to students to allow the professor further monitor effectiveness and appropriateness of activities that take place somewhere other than on the course LMS.

How will you assess the online delivery of lab activities?

Reports and other forms of write-ups will be submitted on the course LMS for evaluation and feedback.

Instructional Materials and Resources

If you use any other technologies in addition to the college LMS, what other technologies will you use and how are you ensuring student data security?

Depending on the textbook used, the professor may choose to use Pearson MyLab and Mastering, zyBooks, WebAssign, or GitHub. All of these are considered to be safe for use in education for both faculty and students. All can also be integrated with the college LMS (Canvas), which decreases the amount of times students will need to sign-in-and-out of accounts and open them up to data breaches.

If used, explain how specific materials and resources outside the LMS will be used to enhance student learning.

Professors who choose to use Pearson MyLab and Mastering, zyBooks, WebAssign, or GitHub do so in order to assign pre-written or instructor-created problems that are more complicated than those that can be created in Canvas while still receiving instantaneous feedback.

Effective Student/Faculty Contact

Which of the following methods of regular, timely, and effective student/faculty contact will be used in this course?

Within Course Management System:

Chat room/instant messaging
Discussion forums with substantive instructor participation
Online quizzes and examinations
Private messages
Regular virtual office hours

Timely feedback and return of student work as specified in the syllabus
Weekly announcements

External to Course Management System:

Direct e-mail
Posted audio/video (including YouTube, 3cmediasolutions, etc.)
Synchronous audio/video
Telephone contact/voicemail

For hybrid courses:

Scheduled Face-to-Face group or individual meetings

Briefly discuss how the selected strategies above will be used to maintain Regular Effective Contact in the course.

Faculty will regularly contact students individually and as a group through Canvas messages and/or COD email. Students will also receive regular announcements with information about the course, COD as a whole, or other relevant information.
In discussions and through other lab assignments, students will communicate with each other and their professor regularly and frequently.

If interacting with students outside the LMS, explain how additional interactions with students outside the LMS will enhance student learning.

Students may prefer to contact their professor via email or on the phone, which allows for an improved experience for those who communicate better in those contexts. The professor may direct students to access free supplemental resources as well.

Other Information

Comparable Transfer Course Information

University System

CSU

Campus

CSU Fullerton

Course Number

CPSC 240

Course Title

Computer Organization and Assembly Language

Catalog Year

2021-2022

Rationale

Course description from catalog: "Digital logic and architecture of a computer system, machine level representation of data, memory system organization, structure of low-level computer languages. Machine, assembly, and macro language programming. Principles of assembler operation, input-output programming, interrupt/exception handling. Laboratory programming assignments."

University System

CSU

Campus

CSU San Bernardino

Course Number

CSE 2130

Course Title

Machine Organization

Catalog Year

2021-2022

Rationale

The topics covered are similar. From the course description: "Number systems and data encoding, von Neumann computer architecture, instruction set architecture, addressing modes, arrays and records, subroutines and interrupts, I/O, assembly programming."

University System

UC

Campus

UC Riverside

Course Number

CS 061

Course Title

Machine Organization and Assembly Language

Catalog Year

2021-2022

Rationale

Course description: "An introduction to computer organization. Topics include number representation, combinational and sequential logic, computer instructions, memory organization, addressing modes, interrupt, input/output (I/O), assembly language programming, assemblers, and linkers."

MIS Course Data

CIP Code

11.0701 - Computer Science.

TOP Code

070600 - Computer Science (transfer)

SAM Code

E - Non-Occupational

Basic Skills Status

Not Basic Skills

Prior College Level

Not applicable

Cooperative Work Experience

Not a Coop Course

Course Classification Status

Credit Course

Approved Special Class

Not special class

Noncredit Category

Not Applicable, Credit Course

Program Status

Program Applicable

Transfer Status

Transferable to both UC and CSU

C-ID
COMP 142

Allow Audit
No

Repeatability
No

Materials Fee
No

Additional Fees?
No

Approvals

Curriculum Committee Approval Date
11/18/2021

Academic Senate Approval Date
12/09/2021

Board of Trustees Approval Date
01/21/2022

Chancellor's Office Approval Date
01/04/2017

Course Control Number
CCC000578623

Programs referencing this course
Liberal Arts: Business and Technology AA Degree (http://catalog.collegeofthedesert.eduundefined/?key=27)
Liberal Arts: Math and Science AA Degree (http://catalog.collegeofthedesert.eduundefined/?key=29)
Computer Science AS-T Degree (http://catalog.collegeofthedesert.eduundefined/?key=35)