UVU UTAH VALLEY UNIVERSITY™

Master Course Syllabus

For additional course information, including prerequisites, corequisites, and course fees, please refer to the Catalog: https://catalog.uvu.edu/

Semester: Spring
Course Prefix: cs
Course Title: Computer Organization and Architecture

Year: 2025
Course and Section #: 2810-002
Credits: *Click here to enter text.*

Course Description

This course uses assembly language to introduce basic concepts of computer organization. Includes number systems, CPU organization, instruction sets, programming in assembly, memory organization, debugging, program design, and documentation. Covers interrupts, vector tables, and disk I/O.

Course Attributes

This course has the following attributes:
☐ General Education Requirements
☐ Global/Intercultural Graduation Requirements
☐ Writing Enriched Graduation Requirements
☒ Discipline Core Requirements in Program
☐ Elective Core Requirements in Program
☐ Open Elective
Other: *Click here to enter text.*

Instructor Information
Instructor Name: Dr. L. Thackeray

Student Learning Outcomes

At the completion of the course, the student should have obtained the following knowledge and skills:

> Accomplish basic mathematical operations in binary and convert between binary, decimal, and hexadecimal.

> Understand data types used in computer systems at a bit level, including signed and unsigned integers, ASCII, and floating point.

> Understand how transistors are used to build basic gates.

> Use basic logic gates to build a decoder, multiplexer, a four-bit adder, an R-S latch, and a gated latch.

> Fully appreciate the Von Neumann Model through exercises with an emulated Von Neumann computer.

> Fully understand the LC-3 architecture, instruction set, data path, interrupts, and control circuitry.

> Fully appreciate how an instruction set enables a programming environment.

Course Materials and Texts

Required Text:

Introduction to Computing Systems 3rd Edition by Patt & Pate

ISBN10: 1260150534 | ISBN13: 9781260150537

Logisim (logic simulator) download and information page. This is an updated version that will run on Windows and Mac (both Intel and M1):

https://github.com/logisim-evolution/logisim-evolutionLinks to an external site.

Install Instructions: in the Download section in the above link (scroll down). Select "download in compiled form", then select your processor in the Assets section.

Users guide and additional information on logisim: http://www.cburch.com/logisim/docs/2.7/en/html/guide/index.htmlLinks to an external site. http://reds-data.heig-vd.ch/logisim-evolution/IntroToLogisimEnglish.pdfLinks to an external site.

LC3 Simulator. This is the Interactive Development Environment (IDE) that will be used to develop both ISA and Assembly programs. The following is the latest Windows and Mac-based LC-3 simulator, documentation and tools:

LC3Tools is a cross-platform set of tools to build code for and simulate the LC-3 system described in *Introduction to Computing*.

Course Requirements

Course Assignments, Assessments, and Grading Policy

Lessons are self-contained in that they will have an objective and overview section, a reading assignment, and a graded assignment.

The lesson schedule, discussion topics, and due dates along with assignment and exam due dates are listed in detail in the modules section of Canvas. The modules section is also the home page for this course. Each lesson and lesson assignments have the prior lesson as a prerequisite. This means that you cannot skip lessons. As an example, to do the assignments in lesson 5, you must have completed lessons 1,2, 3, and 4.

This is a course quiz (lesson 0) that you must pass (multiple attempts are permitted) before continuing in the course. This quiz covers basic course information, such as honesty, assignment due dates and late penalties, grading, etc. You are responsible for knowing this information and this is why I have this simple text. This test must be completed before the semester drop date (01/24) or you will be automatically dropped from this class.

I do hold office hours and encourage you to contact me if you have questions. It is expected that you will attend class and read all assigned text book chapters and other assigned material lesson materials. This material is part of the course, and you will be tested on it. Additionally, the classroom lectures are recorded and are available for review.

This course uses Canvas. You can use Canvas to:

- See the list of labs and programming projects
- See the grades that you have received on assignments
- Upload completed labs and programming projects.

The Canvas website is the only official site for course material. Be sure that you have set up Canvas to send you notifications.

If you have any difficulty using Canvas please call the Distance Education Service Center for immediate support. The Distance Education Service Center is a one-stop comprehensive support center. A service representative dedicated to your success is available to you as part of your course fee. Their Hours are Monday through Friday from 8:00 am to 7:00 pm.

 Live Help for Canvas is available at http://www.uvu.edu/de or

Email: dehelp@uvu.edu

Telephone: 801.863.8255 (Utah County) 888.425.4412 (Toll Free) 801.863.7298 Fax

 Access to Canvas requires that you log in and provide a password. This is done through myUVU, the student web portal.

How To Succeed

This is a self-directed/instructor-guided course. Which means you are responsible for your learning. I will present information in the form of lectures and in-class activities, but is is your responsibility to study this information along with the information that is contained in the text book and other course material.

It is also your responsibility to make sure you understand the course schedule and meet all due dates. Canvas enforces assignment submission dates and quiz dates. If you miss a due date, you do not receive credit. To succeed in this class, you must do the following:

1. Work on course material at least three times a week.
2. Required readings: To be successful, you must read and study this text book. The learning materials in this course are based on the chapters in this book.
3. All homework assignments must be completed and completed on time.
4. Learning is an active process requiring work and a positive attitude. Accept responsibility for learning.
5. You must remember what you learn from lesson to lesson. Just as in a Math class, successive lessons build upon the knowledge of previous lessons.

Your final grade is determined by points earned from assignments, discussions, main programming projects, quizzes, and exams. The final grade is calculated directly from these points as described in the syllabus.

Here comes my jerk professor's speech: Excuses for not completing work does not add points to your grade calculation. You have total control over your grade via the amount of work you complete. Grading is all point-based. All assignments, with the exception of in-class activities, can be turned in right up to the end of the course. All it takes to get a solid grade in this course is to stay on schedule, do all of your assignments, and do reasonably well on your quizzes.

Late Turn-in Policy

When you are late with your assignments you lose context, the support of your peers, and the focus of instruction. In this class being able to perform is of major importance. All professional programming is date driven. There are always release dates associated with any commercial programming effort. It is the same in this class. All assignments have a due date, all assignments have a prerequisite of the former assignment. This means you cannot skip around. All assignments must be complete in order.

The primary cause of failure in this class is missed due dates and failure to turn in work. Because of this, due dates will be rigidly enforced. Canvas' assessment and assignment due dates are enforced automatically. **Assignments are due at 11:59 p.m. on the date specified in the lesson modules.** Be sure that your assignments are correctly submitted to Canvas. Follow all of the course guidelines on naming your assignments, projects, and source code files, and submitting your assignments.

Note that all assignments are due at a specific time and date. I will allow late assignments to be submitted up to 5 days after the due date with a 10% penalty. After 5 days, the late penalty will hold at 50% and I will accept late assignments through the end of the course (April 22). I do this because I feel strongly that we best learn programming by practice. In other words, by doing. I will not accept any assignments or exams after the course closes. This is a UVU policy.

In the professional world, software is developed on a schedule with expected completion dates. It is important to be able to work within a set timeline. Start early working on your programming assignments so that you will have time to complete them on time.

Grading

Assignments must be **submitted through Canvas** for grading. The due dates on all assignments are displayed in canvas. Ample time is allotted for completion of completion of assignments and you can resubmit if it is done before the due date. Please do not ask for an exception to this policy.

Scheduled quizzes and exams are set in Canvas.

Grading scale:

Name:	Range:
A	100% to 94%
A-	< 94% to 90%
B+	< 90% to 87%
B	< 87% to 82%
B-	< 82% to 80%
C+	< 80% to 77%

Name:	Range:
C	< 77% to 72%
C-	< 72% to 70%
D+	< 70% to 67%
D	< 67% to 62%
D-	< 62% to 60%
E	< 60% to 0%

I grade on a **points scale** which includes Exams, Assignments, discussions, and final project.

Exams

- Each module will have an exam. All exams are online. Once the due date for the quiz or exam is pasted, it will not be reopened.

In-Class Graded Activities

- In-class participation will include programming activities, pop quizzes, and other activities that are to be completed in-class and are usually due at the end of the class. Many of these activities will not be announced. So please plan on attending class, as you will need to be present to participate and receive credit. These activities cannot be made up.

Assignments

- Lesson Assignments: There will be binary math assignments, digital logic projects, and assembly language assignments as well as exercises assigned from the textbook for each lesson. Please review the **Late Policy** and check each week to verify the due date.

Main Project

Each student needs to complete an assembly language project using the LC-3 assembler that includes at least 100 lines of assembly code. The project will consist of three milestone deliverable components:

-
  - Functional Specification
  - Project design
  - Final project (running code and project documentation)
  - Final classroom presentation

Assignment Submission

This class has a large number of students and it is imperative that assignments be turned in correctly to facilitate timely grading. Please adhere to the following:

- Canvas' assignment submission facility is set to only accept file types applicable to a given assignment.
- Keep a separate folder on your computer for each assignment that contains all relevant files for that assignment.
- Keep your assignment files until the end of the course in case a submission is lost.
- Submit all files that are requested by a given assignment.

Submit your assignment via Canvas's entry for this course as follows:

1. Go to Assignments menu item
2. Select a specific assignment.
3. Go over to the top right and select **Submit Assignment**.
4. Click on Add Attachment and upload your files as required.
5. When the upload completes and your submission file is listed, click on **Submit Assignment**. If you do not click on Submit, the grader will not see your file.

Required or Recommended Reading Assignments
Chapter 1 to Chapter 9 of the Textbook

General Description of the Subject Matter of Each Lecture or Discussion
Lesson 1 (Jan 7) - Introduction to Computer Architecture
Lesson 2 (Jan 9, 14, 16, 21) Binary Arithmetic, Data Types
Lesson 3 (Jan 23, 28, 30, Feb 4) Digital Logic, State Machines
Lesson 4 (Feb 6, 11, 13) von Neumann model
Lesson 5 (Feb 18, 20, 25, 27) LC-3 Machine Language
Lesson 6 (March 4, 6, 18) - LC-3 Assembly Language
Lesson 7 (March 20, 25, 27, Apr 1) - Subroutines, Data Structures
Lesson 8 (Apr 3, 8) Input/Output, Interrupts

Lesson 9 (Apr 10, 15) Useful Subroutines

Final Project (Apr 17, 22, 24, 29)

Required Course Syllabus Statements

Generative AI

Generative AI
Statement on using ChatGPT (and similar tools) for this class: Use ChatGPT as a Learning Assistant, not an Oracle. ChatGPT can significantly streamline the learning process for students in a programming class, but it's crucial to ensure that it doesn't inadvertently promote dependency or shortcut genuine understanding. The tool can provide immediate help on complex topics and code troubleshooting. Many companies are expecting students to graduate knowing how to use it to generate code, so using it can no longer be considered cheating or plagiarism. Nevertheless students are encouraged to independently problem solve and think critically. These most important skills in

programming are best developed through a balance of self-led exploration and learning with an assistant.

Using Remote Testing Software
☒ This course does not use remote testing software.

☐ This course uses remote testing software. Remote test-takers may choose their remote testing locations. Please note, however, that the testing software used for this may conduct a brief scan of remote test-takers' immediate surroundings, may require use of a webcam while taking an exam, may require the microphone be on while taking an exam, or may require other practices to confirm academic honesty. Test-takers therefore shall have no expectation of privacy in their test-taking location during, or immediately preceding, remote testing. If a student strongly objects to using test-taking software, the student should contact the instructor at the beginning of the semester to determine whether alternative testing arrangements are feasible. Alternatives are not guaranteed.

Required University Syllabus Statements

Accommodations/Students with Disabilities
Students needing accommodations due to a permanent or temporary disability, pregnancy or pregnancy-related conditions may contact UVU Accessibility Services at accessibilityservices@uvu.edu or 801-863-8747.

Accessibility Services is located on the Orem Campus in BA 110.

Deaf/Hard of Hearing students requesting ASL interpreters or transcribers can contact Accessibility Services to set up accommodations. Deaf/Hard of Hearing services can be contacted at DHHservices@uvu.edu

DHH is located on the Orem Campus in BA 112.

Academic Integrity
At Utah Valley University, faculty and students operate in an atmosphere of mutual trust. Maintaining an atmosphere of academic integrity allows for free exchange of ideas and enables all members of the community to achieve their highest potential. Our goal is to foster an intellectual atmosphere that produces scholars of integrity and imaginative thought. In all academic work, the ideas and contributions of others must be appropriately acknowledged and UVU students are expected to produce their own original academic work.

Faculty and students share the responsibility of ensuring the honesty and fairness of the intellectual environment at UVU. Students have a responsibility to promote academic integrity at the university by not participating in or facilitating others' participation in any act of academic dishonesty. As members of the academic community, students must become familiar with their rights and responsibilities. In each course, they are responsible for knowing the requirements and restrictions regarding research and writing, assessments, collaborative work, the use of study aids, the appropriateness of assistance, and other issues. Likewise, instructors are responsible to clearly state expectations and model best practices.

Further information on what constitutes academic dishonesty is detailed in [UVU Policy 541: *Student Code of Conduct*](#).

Equity and Title IX

Utah Valley University does not discriminate on the basis of race, color, religion, national origin, sex, sexual orientation, gender identity, gender expression, age (40 and over), disability, veteran status, pregnancy, childbirth, or pregnancy-related conditions, citizenship, genetic information, or other basis protected by applicable law, including Title IX and 34 C.F.R. Part 106, in employment, treatment, admission, access to educational programs and activities, or other University benefits or services. Inquiries about nondiscrimination at UVU may be directed to the U.S. Department of Education's Office for Civil Rights or UVU's Title IX Coordinator at 801-863-7999 – TitleIX@uvu.edu – 800 W University Pkwy, Orem, 84058, Suite BA 203.

Religious Accommodation

UVU values and acknowledges the array of worldviews, faiths, and religions represented in our student body, and as such provides supportive accommodations for students. Religious belief or conscience broadly includes religious, non-religious, theistic, or non-theistic moral or ethical beliefs as well as participation in religious holidays, observances, or activities. Accommodations may include scheduling or due-date modifications or make-up assignments for missed class work.

To seek a religious accommodation, a student must provide written notice to the instructor and the Director of Accessibility Services at accessibilityservices@uvu.edu. If the accommodation relates to a scheduling conflict, the notice should include the date, time, and brief description of the difficulty posed by the conflict. Such requests should be made as soon as the student is aware of the prospective scheduling conflict.

While religious expression is welcome throughout campus, UVU also has a [specially dedicated space](#) for meditation, prayer, reflection, or other forms of religious expression.