CS 226

Computer Organization and Design

**Fall 2024**

Course Syllabus

Professor: Bill Pierce

E-mail: pierce@hood.edu

Office: Hodson 230 (inside Computer Science Suite 226)

Phone: 301-696-3983

Office Hours: Mon & Wed before and after class; Wed & Thu 5 – 6 PM; Zoom by

appointment

Class Times: Monday & Wednesday, 11:30 am – 12:55 pm, Hodson 237

**Course Catalog Description**

This course provides a comprehensive introduction to the general organization, architecture and functional characteristics of computer systems. Topics include machine level representation of data, assembly level machine organization, memory systems organization and architecture, alternative architectures and device interfaces.

**Course Goals and Objectives**

* Acquire the ability to use a hierarchical approach to analyze and understand complex systems from both hardware and software perspectives.
* Become sufficiently well acquainted with the basic principles of computer architecture to be able to recommend and make intelligent use of computers; further develop an understanding of

• computer programming

• device and system interfaces

• design and simulation of systems

• CPU and I/O performance

• computer system components and devices

* Acquire a working knowledge of assembly language - a tool for accomplishing the first two goals and understanding the real work performed by computer systems.
* Gain a better understanding of the terminology used to describe computer systems in terms of functionality and performance; and strengthen technical communication skills

Upon completion of this course, students will be able to

1. Describe the components of a computer system, their function and interconnections

2. Explain how information is represented and manipulated in computer hardware

3. Apply cost/performance criteria for evaluating computer systems

4. Write, debug and analyze assembly language programs

5. Explain the steps involved in the compilation and execution of computer programs

6. Discuss the technological changes that have driven the growth of computer systems

**Course Prerequisites**

Discrete Math (MATH 207) and Programming I (CS 201)

Introductory programming experience in a high-level programming language such as Java or C++ should be sufficient. You should be competent using a computer to edit files, create and navigate folders/directories, and run windowing software programs (Windows, Linux or OS X). Concepts from discrete mathematics are useful in studying digital computers which operate in discrete steps and store data in discrete bits.

**Where This Course Fits in Computer Science at Hood College**

This course is required for the undergraduate computer science major. It is usually the first class you encounter that deals with hardware internals. This class is a prerequisite for the following courses: Digital Logic and Switching Theory (CS 450), Data Communications and Network Security (CS 453), Computer Architecture (CS 461), Operating Systems (CS 464), and Programming Languages: Their Design and Compilation (CS 471).

On a professional level, possessing a degree in computer science assumes you have fundamental knowledge of the hardware and systems level software, and how software executes on hardware.

**Text Book**

**Computer Organization and Design: The Hardware/Software Interface**, RISC-V Edition, Patterson, David A., and Hennessy, John L., Morgan Kaufmann Publishers, Inc., 2nd edition, ISBN 978-0-12-820331-6, 2021.

Most of the content of the course is taken from this text book. The book is available in print and online versions for both purchase and rental. Used copies are also available It is recommended that you have access to a copy of the book but it is not required.

**Course Modality**

This course is offered as a traditional *in-person (face-to-face)* class. Class time will be devoted to lecture discussions of theory and concepts. When we begin working with assembly language programming, class time will be dedicated to professor-led hands-on lab work. All lecture and supplementary materials will be provided on Blackboard. Lectures will be posted at regular intervals throughout the semester and students are to view this material as promptly as possible to avoid getting behind.

**Assignments and Out-of-Class Time**:

There is a substantial programming component in this course. Homework will consist of small to medium programming assignments and non-programming exercises. Homework assignments are designed for you to apply concepts presented during class to demonstrate your understanding of the material. Assignments will specify a due date and it is expected that all students will submit assignments on Blackboard by the due date.

Due dates are determined by the level of effort required to complete each assignment. Generally, assignments are due one or two weeks from the date they are assigned. If you have a valid reason for not meeting the stated due date, you must contact the professor ***prior*** to the due date for consideration of a grace period. The terms of a grace period will be determined on a case-by-case basis. Generally, no late assignments will be accepted after one week past the due date. It is the student’s responsibility to verify a successful Blackboard submission.

Each assignment will have specific requirements. All programming assignments will include a checklist/rubric from which assessments are based. Please make sure you read and follow all directions on assignments. If you are not clear about an assignment, contact the professor before you do work that may not be required.

Assignments should be typewritten, grammar and spell checked if appropriate. Computer programs will have very a specific format that will be explained in class and with examples. In the event that an assignment requires a diagram or other object that is difficult to produce in software, a hand drawn version is acceptable. Grades are assigned based on correctness, completeness and presentation (organization, neatness and readability).

The amount of time you need to devote out of class to review concepts and complete assignments will vary depending on the relative difficulty of the assignment and your understanding of the concepts upon which assignments are based. On average, expect to spend 1.5 – 2 hours per week outside of class on course work.

**Quizzes and Exams**

There will be periodic quizzes (about one every 2 - 3 weeks) and a final exam. There will be no mid-term exam. Quizzes will be announced one week in advance and you will be provided a list of topics and terminology that will be covered on each quiz. Quizzes will be administered on Blackboard during the last 30 – 40 minutes of class. The final exam is to be scheduled by the Registrar during the final week of the semester (exam week). A list of topics covered on the final exam will be provided in advance.

**Course Level of Difficulty and Student Expectations**

This is a bits and bytes, detail-oriented kind of course. It has been a difficult course for some students. It is a transitional class where we expand beyond thinking about computer science as just programming to thinking about computer science as being guided by fundamental, universal, time invariant concepts and principles. It is, perhaps, your first exposure to the important analysis of design tradeoffs. In this class we emphasize both programming (assembly language) and concepts of organization and architecture (hardware).

At first glance, the number of topics and amount of material we cover in this class can be overwhelming. Much of the course will address basic concepts and principles of system design and performance, with references to specific technologies at various levels of abstraction. It would be in your best interest to keep up with the lecture topics by reading the material in the text and reviewing the lectures in a timely manner.

When possible and feasible, each class will be recorded which will consist of the material visually displayed on the projector and the lecture audio as the material is presented. Links to the recordings will be posted in Blackboard. It is very important to remember that the recordings are not to be used as a substitute for attending class. You are advised to take notes during class to supplement the material covered. You are encouraged to ask questions in class on any topic on which you need further clarification. Such questions benefit the entire class.

**Use of Resources Outside of Class Materials**

In the interest of promoting individual research, you are encouraged to research and investigate information from other published sources and the Internet. However, you are not allowed to directly copy any external material to submit as your own work. For assembly language programming assignments, **never** copy or use sample code from Internet sources because such code does not conform to the programming guidelines enforced for this class. Collaboration among students should be limited to discussion of clarification and concepts.

The use of generative AI tools or large language models (LLMs), such as ChatGPT and AlphaCode, to complete any aspect of assignments for this course are not permitted and will be treated as plagiarism. Extensive testing of these tools as related to the content of this class do not produce correct results and are easily identified as being generated by AI.

All the work you create and submit for this class must be your own. Strict adherence to the Hood College Honor Code and copyright regulations is required for all students by affirming on each class assignment that you “have neither given nor received any unauthorized aid.” Assignments submitted without quoting the honor code will not receive full credit. Refer to the Hood College Catalog and the Student Handbook for additional information regarding academic policies with examples of violations. Honor Code and/or copyright violations may result in a failing grade for the assignment on which the infraction occurred. Severe violations of the honor code will be reported to the Academic Judicial Committee.

**Contacting the Professor**

The best and most efficient way to contact the professor is by email (pierce@hood.edu). I check email frequently during the weekdays (less so on weekends) and will respond to your questions as promptly as possible.

**Class Attendance**

Attending class is critical to your success in completing course requirements. If you must be absent from a class, please notify the professor prior to the class you will miss. Valid reasons for missing class include illness, quarantine, observance of religious holidays or participation as a representative of the College in athletic contests or cultural performances. Attendance is recorded for each class. Excessive absences or erratic attendance will result in an academic alert, and may potentially result in dismissal from the course.

**Course Assessment & Grades**

Each assignment, quiz and the final exam will be graded on a point system; the total number of points per assignment may vary. Generally, more complex assignments carry a higher number of points. A grading checklist/rubric will accompany programming assignments. The following weights will be applied in determining final grades:

Homework and programming assignments 50%

Quizzes 20%

Final Exam 25%

Class participation\* 5%

\* Class participation includes punctuality/attendance, asking/answering questions, observed level of effort in completing course work.

Final letter grades will be based on the overall course average according to the following scale (as stated in the college catalog):

A avg>93 C 73<avg<77

A - 90<avg<93 C - 70<avg<73

B + 87<avg<90 D + 67<avg<70

B 83<avg<87 D 63<avg<67

B - 80<avg<83 D- 60<avg<63

C + 77<avg<80 F 0<avg<60

**Software and Use of Computers**

All assembly language programming assignments will be implemented using RARS (RISC-V Assembler and Runtime System). The program is available on lab systems in Hodson. However, it would be best and more convenient if you use the program on your personal systems. The software is distributed as an executable Java Archive file and is supported on any platform that runs a Java Virtual Machine. It has successfully run on Windows, Apple and Linux systems. A copy will be made available on Blackboard when we begin working with assembly language programming. Demonstration of the software and an introduction to assembly language programming will be conducted during class meetings to help familiarize you with the operation of the program, the settings to be used, and the process for coding and executing programs.

**Course Schedule**

The semester schedule on the next page is a very ***tentative*** attempt to outline the topics to be covered by week/class. Lecture topics as they are organized in the Powerpoint presentations are not designed to be covered in a single class meeting, so some topics may span multiple classes. Expect adjustments to the schedule of topics as the semester progresses. Major changes will be announced as necessary. The lecture slides present topics based primarily in chapter order. However, some topics in the lectures do not necessarily align with the textbook order or being covered within the time constraints of a given class. When the time for the end of class comes and there is still material to be covered within a slide set, the next class will resume at the point reached in the prior class. Many chapters are divided into multiple parts and we will cover them in the order listed in the schedule.



**Accessibility Services (formerly Disability Services)**

Accommodations for Disabilities: If you have any kind of disability, whether apparent or non-apparent, learning, emotional, physical, or cognitive, and you need some accommodations or alternatives to lectures, assignments, or exams, please feel free to contact me to discuss reasonable accommodations for your access needs. If you have a diagnosis or history of accommodations in high school or previous postsecondary institutions, Accessibility Services can help you document your needs and create an accommodation plan. By making a plan through Accessibility Services, you can ensure appropriate accommodations without disclosing your condition or diagnosis to course instructors. Accessibility Services are located in the Beneficial - Hodson Library & Learning Commons (Suite 1027), phone: 301-696-3569, email: accessibilityservices@hood.edu.

**Title IX Compliance**

Sexual misconduct is one of the most serious violations of Hood College's values of honor and respect and no form of sexual misconduct is acceptable within our community. Consistent with state and federal laws, including Title IX of the Education Amendments of 1972 as amended ("Title IX") and Title VII of the Civil Rights Act of 1964, the College has implemented measures to ensure that all allegations of sexual misconduct are investigated and resolved in a timely, confidential, fair, and impartial manner. The College has designated certain officials "responsible officials". A report to these individuals is an official report to the College which necessitates a response by the College. Faculty members are considered responsible officials and, as such, are not considered confidential resources. Therefore, incidents shared with the faculty must be reported to the Title IX Coordinator. Please be advised that if you wish to speak confidentially about an incident and/or to obtain an academic accommodation, you may contact the Wellness Center to speak with a counselor, Health Services Staff and/or the Dean of the Chapel.