

### **Catalogue Description**

Computer architecture and design, RISC assembly language programming, internal data representation, and digital logic. Prerequisite: COSC 182. Course web site is [home.messiah.edu/~chase/csc/281](http://home.messiah.edu/~chase/csc/281)

### **Personnel**

*Instructor.* Dr. Gene B. Chase. Email [chase@messiah.edu](mailto:chase@messiah.edu) . Telephone 766-7904 (home, before 10 pm), or 796-1800 extension 2770 (office). My office is Frey 123. My home page and my daily schedule are linked from the course home page. A signup sheet on my office door has blank spaces where you can sign up in advance for help.

*Student Helper.* My teaching assistant is Chuck Craig. Especially before each exam we will have him available to help you. Frey 349 becomes a help room on Monday through Thursday evenings from 7:00 to 9:00 p.m. Students there can help you one of those evenings. Chuck has a couple of those hours. See schedule posted on bulletin board in Frey 349.

### **Goal**

The goal of this course is to learn about computer system internals, not to learn an assembly language that you will program in for a future career. Learning the details of how computers work at lower levels will help you to understand higher level languages and operating systems, and will develop your interest in computer architecture. Those few (probably engineers) who are interested in writing in low-level languages as part of your career might consider a semester project that introduces you to Intel's PC assembly language.

### **Objectives**

1. To understand principles of design for computers and assembly languages.
2. To understand and to program using a specific RISC language. We choose a modern and simple one: MIPS.
3. To compare the RISC design to a more complex but popular CISC alternative, the Intel 80x86.
4. To understand computer arithmetic and internal storage formats.
5. To understand the basic logical design of the CPU and datapath.
6. To understand the role of system programs, such as compilers, assemblers, linkers, and loaders.

### **Outcomes**

Numbers in brackets correspond to the numbered objectives above. Note that the Fall semester version of this course contains an extra outcome which cannot be done in 11 weeks of the Spring.

1. You will write four programs of increasing difficulty in the language MIPS that demonstrate principles of reduced instruction set architecture. [2]
2. You will pass a written 50-minute exam on the process of compiling a high level language to assembly language and assembling the result to machine level instructions, including issues of performance. [1, 2, 6]
3. You will pass a written 50-minute exam on addressing modes, a complex instruction-set architecture, and data representation for negative numbers, strings, arrays, and functions. [3, 4]
4. You will pass a written two-hour exam on integer multiplication and division and all floating point operations of the central processing unit; and on single and multiple clock-cycle data paths. [4, 5]

## Textbook and Materials

*Computer Organization and Design: The Hardware/Software Interface, 3rd Edition.* David A. Patterson, John L. Hennessy. San Francisco: Morgan Kaufmann Publishers. 2005. This is the most widely used book for this course in the world. As you can see from page xvi, I had a hand in its revision. Our course is drawn from the professional recommendation CS 220c as described by “Computing Curricula 2001” of August 2001 < [www.acm.org/sigcse/cc2001/steelman/index.html](http://www.acm.org/sigcse/cc2001/steelman/index.html) >.

## Grading

- 32% **4 programming assignments.** Acknowledge any additional help you receive. Assignments are due at the beginning of class on the due date. **Late assignments are not accepted.** Work ahead. Programmers often encounter unexpected delays.
- 45% **3 non-comprehensive examinations,** based on text, lectures, and homeworks. Answers to end-of-chapter questions in the textbook are available on loan from me at my office.
- 19% **Project,** either individual or team
- 4% **6 quizzes, dropping the lowest two** random two-minute to encourage you to read the text before attending the lectures.

Unexcused lateness and absences count against you and active participation counts in your favor to adjust your final grade from the above raw calculation. If you know the material too well to need to attend the class time, then you know it too well to deprive the class of your insights.

To accomplish this goal of your being able actively to participate, I will check on your presence by giving a very short, easy question as a **quiz at the start of some random hours during the semester.** Anyone who even skimmed the reading for that day will have no trouble answering it.

Cutoff percentages needed for a final passing letter grade: A 93%, A- 90%, B+ 87%, B 83%, B- 80%, C+ 77%, C 73%, C- 70%, D+ 65%, D 60%. Percentages assigned for letter grades on homeworks and presentation: A 95%, A- 90%, B+ 87%, B 85%, B- 80%, C+ 77%, C 75%, C- 70%, D+ 67%, D 65%, D- 60%, F 50%, not turned in 0%. The course web site has a statement of what grades mean at Messiah College.

## Chapter-by-chapter Outline

Our schedule is roughly as follows. Chapters 5 and 6 will be covered quite superficially to allow plenty of time for group presentations.

1. Overview of the parts of a computer system. (1 week)
2. Overview of design principles and techniques for evaluation. (1 week)
3. Assembly language basics, including the role of assemblers and other system software. (3 weeks). See also Appendix A: The SPIM simulator, in your text.
4. Computer arithmetic, and basic digital logic. (4 weeks). See also Appendix B: Basic Logic Design on the CD that accompanies your text.
5. The Central Processing Unit: design of datapath and control. (2 weeks)
6. Pipelining to improve the efficiency of the processor. (1 week, time permitting)
7. Project presentations. (1 week)

A tentative day-by-day schedule with reading assignments begins on a page below. The course website contains the evolving current version of the schedule. **Read assigned pages in advance.** If there is anything you don't understand, please bring it to class for discussion. Copy the lab dates, assignment due dates, and exam dates to your daily calendar now before you forget it!

## Americans with Disabilities Act (ADA)

Messiah College welcomes students with disabilities. Any student whose disability falls within ADA guidelines should inform me within the first two weeks of class about any special accommodations or equipment needs that are necessary to complete the requirements for this course. Students must register documentation of any disability with the Office of Disability Services (Hoffman 101). If you have questions, call Dr. Keith Drahn at ext. 5387, or see me at my office.

## Academic Integrity

Plagiarism—representing another's work as your own—will surely result in a lower grade in this course, and may result in failing the course depending on its severity. You must document any sources that you use, whether from the Internet, another person, or printed materials. This includes especially the work of other students who are currently taking this course or who have taken this course before. Academic integrity is broader than plagiarism. It includes such things as not keeping library materials that you are not using so as to keep another student from completing his or her work.

All students at Messiah College must read and abide by the College's policy on academic integrity, which is found in your Student Handbook on pages 38-40. The current copy of that handbook is on the Internet at [www.messiah.edu/offices/student\\_affairs/student\\_handbook/resources/0607/academic.pdf](http://www.messiah.edu/offices/student_affairs/student_handbook/resources/0607/academic.pdf) and is linked from our course home page as well. You are not properly enrolled in this course unless you have read and understood these guidelines.

## Tentative Daily Schedule

This schedule is updated regularly at the course website.

Date (sequence, day of week, day)	Topic and Text Readings	
August		
1 Wed 30	From high level to hardware	
September		
2 Fri 1	Hardware intro	1.1, 2, pp 44–45
3 Mon 4	History	1.3, 4, history handout
4 Wed 6	MIPS Assembly	1.5-6; 2.1-3; A.3-5, parts of A.9, handouts [Appendices are on the CD ROM]
5 Fri 8	SPIM emulator	2.4
6 Mon 11	MIPS Assembly	2.5
7 Wed 13	MIPS Assembly	2.6; A.10 for reference only
8 Fri 15	Logic control	Catch up
9 Mon 18	Procedure calls	No reading; <b>Lab 1 due</b>
10 Wed 20	Nested proc calls	2.7
11 Fri 22	Addressing modes	2.8, A.6
12 Mon 25	Jump Tables	2.9; pp. 106–107; 2.13
13 Wed 27	Two's compl negs	Jump Tables, pp. IMD 2.20–3(In More Depth on CD) Exercise 2.10; 2.15; 2.16 (only two figures); 2.17–18; <b>Lab 2 due</b>
14 Fri 29	Exception handling	3.1–2, Review

## October

15 Mon 2	<b>Exam 1</b> , on Chapters 1–2; 3.1–2; Appendix A.1–5, and parts of A.9
16 Wed 4	Signed integer arith. no readings
17 Fri 6	Algorithms for arith. 3.3, 4
18 Mon 9	Hardware mult, div 3.5; optionally B.6
19 Wed 11	Floating pt operations 3.6

---

 Oct 12–15 Mid-Fall Recess
 

---

20 Mon 16	Floating pt ops	Optional: Booth's algorithm for integer division, IMD on CD <b>Lab 3 due</b>
21 Wed 18	Boolean alg proof	3.8,9
22 Fri 20	Boolean algebra	no new reading: catch up
23 Mon 23	Elementary gates	Appendix B.2
24 Wed 25	Combinational logic	Appendix B.3
25 Fri 27	Truth tables to circuits	Review for Exam 2
26 Mon 30	<b>Exam 2</b> , Chapters 3–4	
November		
27 Wed 1	Creating 1-bit ALU	Appendix B.5
28 Fri 3	Creating 1-bit ALU	re-read B.5
29 Mon 6	Creating 32-bit ALU	<b>Lab 4 due</b>
30 Wed 8	Creating 32-bit ALU	5.1
31 Fri 10	Single cycle data path	5.1–5.3
32 Mon 13	Single cycle data path	5.1–5.3 again
33 Wed 15	Performance	5.1-4 through p. 314; <b>Lab 5 proposal is due</b>
34 Fri 17	Performance	two diagrams from 5.5: Fig. 5.24 and Fig. 5.28.
35 Mon 20	Multi-cycle data path	4.1–5, handout

---

 Nov 22–26 Thanksgiving Recess
 

---

36 Mon 27	Multi-cycle data path	Chapters 4; 5; Appendix B
37 Wed 29	Pipelining	6.1
December		
38 Fri 1	Pipelining hazards	6.2
	Maybe input-output	Appendix A.38–40
39 Mon 4	<b>Lab 5 presentations</b>	
40 Wed 6	<b>Lab 5 presentations; and all written project reports due</b>	
41 Fri 8	Review for (non-comprehensive) final exam	

**Final exam week, Monday, December 11, Exam 3, 10:30 a.m. – 12:30 p.m.**