

## CompSci 335 Schedule

#	Date	Day	Reading	Topic / Due Dates
1	Sep 4	Wed	Chapter 1.1 - 1.3.4	Syllabus, History - Observe progression in the development of automating computation.
2	Sep 9	Mon	Chapter 1.3.5 - 1.7	History - How and why has computing technology progressed to support its goals? Current challenges?
3	Sep 11	Wed	Chapter 2.1 - 2.4	Binary numbers, addition, subtraction. How do circuits become numbers? What's the simplest way to subtract?
4	Sep 16	Mon	Chapter 2.5 - 2.8	Multiplication, division, floating point. How do they work? What are the issues?
5	Sep 18	Wed	Chapter 2.9.1	Transistors and gates, How do electrons become bits? How do switches become Boolean operators? Logisim.
6	Sep 23	Mon	Chapter 2.9.2	Logisim and prototyping. How to use the prototyping board. Building an adder - How can we turn Boolean operators into arithmetic operators?
7	Sep 25	Wed	Chapter 2.9.3	<b>Logic Lab 1 Due.</b> Multibit ripple carry adders. adder-subtractor
8	Sep 30	Mon	Chapter 2.10.1	Carry lookahead. How do we speed up arithmetic? How do we store values with electrons?
9	Oct 2	Wed	Chapter 2.10.2 - 11	<b>Logic Lab 2 due.</b> Register bits. How do we make a practical data memory?
10	Oct 7	Mon	Chapter 3.1 - 3.2	Arduino, basic microcontroller output (lights), sensor capabilities
11	Oct 9	Wed	Chapter 3.3 - 3.4	<b>Logic Lab 3 due.</b> Using pins for external I/O, handling interrupts.
	Oct 14	Mon	Holiday	
12	Oct 15	Tue	Chapter 3.5 - 3.6	Analog I/O, modulation, term project examples.
13	Oct 16	Wed	Chapter 7.1 - 7.2	<b>Logic Lab 4 due.</b> Control units. How do we coordinate the activity in a processor?
14	Oct 21	Mon	Chapter 4.1	<b>Project proposal due.</b> Instruction set architecture. What are the basic options for designing an instruction set? Addressing modes.
15	Oct 23	Wed	Chapter 4.2 - 4.6	Shift instructions. Assembly language. Basics of syntax and the programming model. First assembly programs.
16	Oct 28	Mon	Chapter 3.5 - 3.7	<b>Assembly Lab 1 Due.</b> How do we write device drivers? Memory mapped control registers.
17	Oct 30	Wed	Chapter 3.8	Assembly branching, looping, nested loops

18	Nov 4	Mon	Chapter 3.9 -3.11	<b>Assembly Lab 2 Due.</b> Pointers, array access, nested calls. How to keep more complex machine code organized.
19	Nov 6	Wed	Chapter 6, 7.3	Performance and pipelines. Now that we have the basics, how can we make the work go faster?
	Nov 11	Mon	Holiday	
20	Nov 13	Wed	Chapter 7.4 - 7.6	<b>Assembly Lab 3 Due.</b> Branch prediction. Branches are bad news for pipes, so how can we overcome that?
21	Nov 18	Mon	Chapter 9.1 - 9.2	Memory hierarchy, direct mapped cache
22	Nov 20	Wed	Chapter 9.3 - 9.4	Locality, associative caches
23	Nov 25	Mon	Chapter 9.5, 12.5-12.9	Virtual memory, buses
	Nov 27	Wed	Thanksgiving break	
24	Dec 2	Mon	Chapter 11	Secondary storage. How are your files stored? Why do you need to back them up?
25	Dec 4	Wed	Chapter 13	Parallelism. Different ways of combining processors to get more performance
26	Dec 9	Wed		Security concerns (possibly makeup day if we fall behind)
	<b>Dec. 16</b>	<b>Mon</b>	<b>Final Project Presentation (final exam period 3:30 - 5:30 PM)</b>	