

Course Syllabus

Lecture	Date	Topics	Book Sections
1	1/14	Review: problem solving using systematic decomposition, examples	Ch. 6
2	1/16	Review: assembly language programming and process	Ch. 7
3	1/18	I/O abstractions: input from the keyboard, output to the monitor	§ 8.1, 8.2, 8.3, 8.4
4	1/23	Repeated code: subroutines and TRAPs	Ch. 9
5	1/25	Repeated code: interrupts and libraries	Ch. 9
6	1/28	Stacks: abstraction and use in assembly code	Ch. 10
7	1/30	Stacks: conventions and use in building subroutines	Ch. 10
8	2/1	Re-introduction to C programming: data types, operators, scope, and storage class	Ch. 11, Ch. 12
	2/4	OPTIONAL REVIEW SESSION FOR MIDTERM #1	
9	2/6	C examples using variables and operators	Ch. 11, Ch. 12
10	2/8	C control and iteration constructs	Ch. 13
11	2/11	Comprehensive C examples	(Ch. 11 – 13)
12	2/13	Problem solving with C	(Ch. 11 – 13)
13	2/15	Functions: introduction, syntax, stack frames/activation records	§ 14.1, 14.2, 14.3
14	2/18	C examples using functions	§ 14.3, 14.4
15	2/20	Pointers and Arrays: introduction	§ 16.1, 16.2
16	2/22	Pointers and Arrays: problem solving and examples	§ 16.3
17	2/25	Testing and Debugging: introduction, error taxonomy	§ 15.1, 15.2, 15.3
18	2/27	Testing and Debugging: strategy and tactics	§ 15.4, 15.5, 15.6
19	3/1	Testing and Debugging: examples	(Ch. 15)
	3/4	OPTIONAL REVIEW SESSION FOR MIDTERM #2	
20	3/6	Recursion: introduction and basic examples	§ 17.1, 17.2, 17.3
21	3/8	Recursion: example showing run-time stack	§ 17.4, 17.5, 17.6
22	3/11	Input and Output in C: introduction and underpinnings	Ch 18
23	3/13	C library I/O functions	notes
24	3/15	Extensive I/O example	notes
25	3/25	Data Structures: introduction	§ 19.1, 19.2
26	3/27	Data Structures: defining new types, enumerations	§ 19.3
27	3/29	Dynamic memory allocation and management	§ 19.4

28	4/1	Dynamic arrays and linked lists	notes, § 19.5
29	4/3	Dynamic memory implementation	notes
30	4/5	Comprehensive example	notes
	4/8	OPTIONAL REVIEW SESSION FOR MIDTERM #3	
31	4/10	Pointer-based data structures: introduction	notes
32	4/12	Pointer-based data structures: problem solving	notes
33	4/15	Structure hierarchies and subtyping	notes
34	4/17	Inheritance, information hiding, and object-oriented design	notes
35	4/19	C++ concepts: classes, inheritance, and type management	notes
36	4/22	C++ concepts: access control and namespaces	notes
37	4/24	C++ concepts: constructors and destructors	notes
38	4/26	C++ concepts: bug avoidance	notes
39	4/29	C++ concepts: dynamic memory management	notes
40	5/1	Wrap-up and advice	notes