

EECS 381 Winter 2009 Schedule Version 4

Readings Source: K&R:= Kernighan & Ritchie, S:= Stroustrup, H := Handout on course website (*assigned handouts must be covered in your paper*)

Your paper must discuss each chapter, section, or handout listed; cover the entire chapter or section unless part of it is listed as “skip.”

A section assigned as “skim” requires only a brief (e.g. one sentence) mention in your paper.

Date			<u>Lecture topics and due dates for readings and projects</u>
1	Jan	8 Th	Organizational and Introduction.
2	Jan	13 T	C concepts: prototypes, headers, linkage. <ul style="list-style-type: none"> • K&R 1-4. Much will be familiar but watch for new information, especially in Ch. 4. • H: Header File Guidelines for C Programs (see above)
3	Jan	15 Th	Pointers, Arrays, Function pointers, structures. <ul style="list-style-type: none"> • K&R 5. Read carefully about pointers, arrays, function pointers; skim 5.12 about complex declarations; K&R 6-6.4; skim 6.5-6.9
4	Jan	20 T	Pointers, etc, continued.
5	Jan	22 Th	I/O, Type safety, memory allocation. <ul style="list-style-type: none"> • Read K&R 7 on I/O and other functions.
6	Jan	27 T	C++ preview, review, filling gaps. <ul style="list-style-type: none"> • Stroustrup: All three prefaces, Ch. 1, skim C++ tour (Ch 2), skim Std Lib tour(Ch 3). • S 4 Types and Declarations: declaration terminology (4.9.1), scope (4.9.4), initialization (4.9.5), objects and lvalues (4.9.6), 4.10. • S 5 Pointers, Arrays, Structures: pointers and zero as a pointer value (5.1), const (5.4), references (5.5), void * (5.6), structs and incomplete (forward) declarations (5.7), 5.8, • H: Incomplete Declarations. <p><i>Throughout Stroustrup, take time to think about his "advice" sections at the end of each chapter - extremely valuable.</i></p> <p><i>For Chs 1-7 and 9, if you don't recognize something in the advice, I recommend that you read that part of the chapter even if unassigned.</i></p>
7	Jan	29 Th	C++ review, filling gaps (continued). <ul style="list-style-type: none"> • S 6 Expressions and Statements: Skim the extended example in 6.1, because he refers to it many places later. Read evaluation order (6.2.2), new and delete(6.2.6), casts and constructor notation(6.2.7, 6.2.8), declaration positions (6.3.1, 6.3.2.1, 6.3.3.1), 6.4, 6.5. • S 7 Functions: Introduction to 7.2, Overloaded functions (7.4), default arguments (7.5), 7.9. • S 8 Namespaces and exceptions: Skim S on namespaces, but read • H: "Using using." <p>Read 8.3 on error handling concepts and exceptions.</p> <ul style="list-style-type: none"> • S 9 Source Files and Programs: The one-definition rule (9.2.3), program startup and termination (9.4), 9.5. • H: C++ Header File Guidelines,
*** Jan	30 F		Project 1 Due
8	Feb	3 T	Classes, objects with dynamic memory contents, Operator Overloading, Templates. <ul style="list-style-type: none"> • S10 Classes: All of chapter. Some review, but study carefully. E.g. 10.2.7.1 &.2 are probably new to you, and 10.4.4.1 is vital, and 10.4.10 explains many otherwise weird error messages. Skip 10.4.11 on "placement new". • H: Static Members. • S 11 Operator Overloading. All of chapter, though it may be familiar. Don't overlook 11.4, 11.9, 11.11 <p>If this material is unfamiliar, read the Handout on Operator Overloading.</p> <ul style="list-style-type: none"> • S13 Templates: Routine use of templates(13.1-13.3).
9	Feb	5 Th	Library Organization & Standard Containers; Strings & Streams <ul style="list-style-type: none"> • S 16. Skip 16.2.1, 16.2.2 for now - we'll come back. <p>Rest of chapter presents <vector> as an example of how container classes work. Skip 16.3.11. Read 16.4 Advice.</p> <ul style="list-style-type: none"> • S 20 Strings. Skim this complete presentation of an extremely important and elaborate class; plan to look up as needed. • S 21 Streams. Skim through 21.5. Skip 21.6-21.8, read Advice 21.9. If confused, re-read the streams Handouts. • H: Basic I/O. • H: File Streams.

10 Feb	10 T	Standard Containers. <ul style="list-style-type: none"> • S 17. A survey of the containers. Learn to read this stuff without getting bogged down in the details. Focus on overviews and examples of use. Notice how he describes a container by presenting its public interface. Skip 17.5.3. Skim 17.6 about defining a new container - you might want to do it sometime, but not for a while! Read Advice (17.7). • H: Fill'er Up: Winners and Losers for Filling an Ordered Container.
11 Feb	12 Th	Algorithms, Iterators, and Function Objects. <ul style="list-style-type: none"> • S 18. There are lots of goodies in the Library. Read the overviews in each section, focus on the specific things called out in the following list, and skim the rest. Specifically: Read 18.1, 18.2, 18.3 (skip 18.3.1), 18.4 (skim 18.4.4). Read 18.5 intro, focus on 18.5.1, .2; read 18.6 intro, focus on 18.6.1 copy, 18.6.2 transform, 18.6.4, .5, 18.6.8. read 18.7 intro, focus on 18.7.1, .2, 18.9, 18.10, 18.12 • S 19 Iterators & Allocators. Skip 19.2.2, 19.2.6.1, skim 19.2.3, skip 19.3, 19.4, read 19.4.5, .6. • H: Why std::binary_search of std::list works, sorta ...
*** Feb	13 F	Project 2 Due
12 Feb	17 T	TBA
13 Feb	19 Th	Basic Class Design. No reading assignment, but bring H: Basic Class Design to lecture to mark up
	Feb 24 T	<i>Winter Recess - no classes</i>
	Feb 26 Th	<i>Winter Recess - no classes</i>
14 Mar	3 T	TBA
15 Mar	5 Th	Midterm Exam, room TBA <i>Notice: When "no reading assignment" is listed below, attending lecture is absolutely essential.</i>
*** Mar	6 F	Project 3 Due
16 Mar	10 T	Simple forms of inheritance and polymorphism: Inheritance & Virtual Functions <ul style="list-style-type: none"> • S 12 Derived Classes. All of the chapter is important. • S 16.2 Now go back and see one of the ways inheritance can be a bad idea in class design.
17 Mar	12 Th	Introduction to OO Design Lecture: Project 4 design overview. No reading assignment is due, but bring to lecture: a hard copy of H: Introduction to UML, and H: Basic OOP Concepts (BasicOOPConcepts-HO.pdf) or the lecture notes on Basic OOP Concepts.
18 Mar	17 T	Multiple inheritance and run-time type identification. <ul style="list-style-type: none"> • S 15. Class Hierarchies. Skip 15.6 intro and 15.6.1, but read 15.6.2 carefully.
19 Mar	19 Th	Exceptions and memory management, RAII, "smart pointers" <ul style="list-style-type: none"> • S 14. Exception Handling. Skim 14.6.
*** Mar	20 F	Project 4 Due
20 Mar	24 T	Some Idioms and Design Patterns (no reading assignment - bring hardcopy of Lecture Notes: IdiomsDesPattsX.pdfs to mark up)
21 Mar	26 Th	More Idioms and Design Patterns (no reading assignment - bring hardcopy of Lecture Notes: IdiomsDesPattsX.pdfs to mark up)
22 Mar	31 T	More Idioms and Design Patterns (no reading assignment - bring hardcopy of Lecture Notes: IdiomsDesPattsX.pdfs to mark up)
23 Apr	2 Th	More Idioms and Design Patterns (no reading assignment - bring hardcopy of Lecture Notes: IdiomsDesPattsX.pdfs to mark up)
*** Apr	3 F	Project 5 Due
24 Apr	7 T	<i>No class - Kieras at ACM SIGCHI conference</i>
25 Apr	9 Th	<i>No class - Kieras at ACM SIGCHI conference</i>
26 Apr	14 T	Designing Software <ul style="list-style-type: none"> • Ch 23. Development and design. Read all of chapter. There is much more that could be said about this topic, but Stroustrups's overview is a good one.
27 Apr	16 Th	TBA
*** Apr	17 F	*** Project 6 Due - time and place for submission of hard copy materials to be announced.
28 Apr	21 T	TBA
Apr	23 Th	FINAL EXAM, 4:00 PM - 6:00 PM, rooms TBA