Syllabus

Below is a preview of the week-by-week plan for the quarter. There may be adjustments and rearrangements as we go. Exam dates are set at quarter start and will not change.

In the readings listed below, B&O is Computer Systems (Bryant and O'Hallaron), K&R is The C Programming Language (Kernighan and Ritchie), and EssentialC is a PDF available at http://cslibrary.stanford.edu/101 (http://cslibrary.stanford.edu/101). Do the readings before lecture for best effect!

Any code examples worked in class will be posted after lecture into `/afs/ir/class/cs107/samples/lectN` where N is the lecture number.

Slides will be linked to below the morning of a given lecture.

Assignment due dates listed below do not include grace periods, if any.

Topics	Readings	Assignments
Week 1		
Lecture 1 (Mon 1/7): Welcome to CS107! We'll go through course logistics, learning goals, and a tour of C programs.	Handouts: 1 - General Information (/class/archive/cs/cs107/cs107.1194/handouts/1-General-Information.pdf) Slides: Lecture 1 Slides (lectures/1/Lecture1.pdf) B&O Ch 1 - skim this chapter for a quick overview of what is meant by systems, and for a preview of topics to come.	**Out (Tues. 1/8)**: assign0
No labs this week.		
Lecture 2 (Fri 1/11): Integers We'll learn more about the representation of the integer types: `char`, `short`, `int`, and `long`, in both unsigned and two's complement signed. We'll also discuss integer arithmetic, overflow, truncation and sign extension, and how mixed signed and unsigned comparison operations work.	Slides: Lecture 2 Slides (lectures/2/Lecture2.pdf) B&O Ch 2.2-2.3 - skim the formal proofs, but it's important to take away a solid working knowledge of two's complement and behaviors of integer operations.	
Week 2		
Lecture 3 (Mon 1/14): Bits and Bitwise Operators We'll dive further into bits and bytes, and how to manipulate them using bitwise operators.	Slides: Lecture 3 Slides (lectures/3/Lecture3.pdf) B&O Ch 2.1	**In**: assign0 **Out**: assign1
Lab 1: Bits and `ints`	Be sure to check out our guide to gdb (/class/archive/cs/cs107/cs107.1194/resources/gdb).	

Topics	Readings	Assignments
Lecture 4 (Fri 1/18): Chars and C-Strings We'll explore how strings and characters are manipulated in C using the `char` and `char *` types, discuss null termination, and become familiar with the `string.h` functions.	Slides: Lecture 4 Slides (lectures/4/Lecture4.pdf) K&R (1.9, 5.5, Appendix B3) or Essential C section 3 for C-strings and `string.h` library functions. C-strings are primitive compared to Java/C++ strings, so take note of the manual efforts required for correct use and pitfalls to avoid.	
Week 3		
Lecture 5 (Mon 1/21 - MLK HOLIDAY): VIDEO ONLY - More C-Strings Now it's time to start digging into the use of `*` and `&`, pointer operations/arithmetic, and memory diagrams. In lecture, we will trace through code, draw lots of pictures, and poke around in gdb.	Lecture 5 Video (https://youtu.be/6zacgQipflY) Lecture 5 Slides (lectures/5/Lecture5.pdf) K&R Ch 1.6, 5.5 or Essential C section 3 on the mechanics of pointers and arrays. Pay special attention to the relationship between arrays and pointers and how pointers/arrays are passed as parameters.	**In**: assign1 **Out**: assign2
Lab 2: C-Strings	Be sure to check out our guide to Valgrind (/class/archive/cs/cs107/cs107.1194/resources/valgrind).	
Lecture 6 (Fri 1/25): Arrays and Pointers We'll answer questions like: how are arrays and pointers the same? How are they different? How is an array/pointer passed/returned in a function call? After this lecture, you'll understand how arrays and pointers allow two syntaxes for accessing sequential memory locations, but the underlying reality of the memory is the same.	Lecture 6 Slides (lectures/6/Lecture6.pdf) K&R 5.2-5.5 or Essential C section 6 on advanced pointers	
Week 4		

Topics	Readings	Assignments
Lecture 7 (Mon 1/28): Stack and Heap We'll learn about stack allocation, stack frames, and parameter passing. Then, we'll introduce dynamic allocation on the heap (`malloc` / `realloc` / `free`), heap contractual guarantees and undefined behavior.	Lecture 7 Slides (lectures/7/Lecture7.pdf) K&R 5.6-5.9 or Essential C section 6 on the heap. The key concept is comparing and contrasting stack and heap allocation.	**In**: assign2 **Out**: assign3
Lab 3: Arrays/Pointers		
Lecture 8 (Fri 2/1): `void *` , Generics We'll continue comparing and contrasting stack and heap allocation. Then, we'll move on to untyped `void *` pointers and motivate C generics.	Lecture 8 Slides (lectures/8/Lecture8.pdf) still working through K&R 5.6-5.9 or Essential C section 6 on the heap.	
Week 5		
Lecture 9 (Mon 2/4): More Generics We'll talk about function pointers, which allow us to implement generic operations using client callbacks.	Lecture 9 Slides (lectures/9/Lecture9.pdf) K&R 5.11, review man pages or your C reference to be introduced to generic functions in the C library (`qsort` , `lfind` , `bsearch`)	**In**: assign3 **Out**: assign4
Lab 4: `void *` /Function Pointers		
Lecture 10 (Fri 2/8): Floating Point We'll learn about manipulating real-value data types (`float` / `double`) using the IEEE floating point representation. We'll also explore features and limitations of the floating point number line.	Lecture 10 Slides (lectures/10/Lecture10.pdf) B&O 2.4 on floats - it's ok to skim the details on exactly which bits go where, instead focusing on which values are representable and why) Cool visualization (https://twitter.com/D_M_Gregory/status/1044008750162604032)	
Week 6		

Topics	Readings	Assignments
Lecture 11 (Mon 2/11): Intro to x86-64, Data Movement We'll introduce assembly/machine language and find out what's happening underneath the hood of the C compiler, including a discussion of the x86-64 instruction set architecture and its almighty `mov` instruction. Then, we'll talk about addressing modes, data layout, and access to variables of various types.	Lecture 11 Slides (lectures/11/Lecture11.pdf) B&O 3.1-3.3 for background info on x86-64 assembly. Very carefully read B&O 3.4 on addressing modes and data transfer. The multitude of addressing modes is one of the things that puts the first "C" in CISC. Be sure to check out our x86-64 guide (/class/archive/cs/cs107/cs107.1194/guide/x86-64.html).	**In**: assign4 **Out**: assign5
Lab 5: Floats Hands-on float dissection		
MIDTERM EXAM IN-CLASS Friday, Feb. 15, 12:30-2:20pm (note: full class period)	See the midterm info page (/class/archive/cs/cs107/cs107.1194/exams/midterm.html) for more information.	
Week 7		
Lecture 12 (Mon 2/18 - PRESIDENTS' DAY HOLIDAY): VIDEO ONLY - x86-64 ALU and Condition Codes We'll talk about arithmetic and logical instructions, as well as condition codes and `cmp` / `test` and `jmp` / `jx` instructions.	Lecture 12 Video (https://youtu.be/4fOYdFmESgk) Lecture 12 Slides (lectures/12/Lecture12.pdf) B&O 3.5-3.6 Be sure to check out our x86-64 guide (/class/archive/cs/cs107/cs107.1194/guide/x86-64.html).	
Lab 6: Assembly x86-64 in all its glory	Be sure to check out our x86-64 guide (/class/archive/cs/cs107/cs107.1194/guide/x86-64.html).	
Lecture 13 (Fri 2/22): x86-64 Control Flow We'll see how to implement C `if` / `else` and loops in assembly, discuss other uses of condition codes (`setx` / `cmov`), and talk about procedures and the runtime stack.	Lecture 13 Slides (lectures/13/Lecture13.pdf) B&O 3.6 Be sure to check out our x86-64 guide (/class/archive/cs/cs107/cs107.1194/guide/x86-64.html).	**In**: assign5 **Out**: assign6
Week 8		

Topics	Readings	Assignments
Lecture 14 (Mon 2/25): x86-64 Runtime Stack We'll introduce instructions for call/return, parameter passing, local stack storage, and register use. We'll also explore optimizing compiler transformations to reduce instruction/cycle count.	Lecture 14 Slides (lectures/14/Lecture14.pdf) B&O 3.7, 5.1-5.6	
Lab 7: Runtime Stack Fun explorations with the stack!		
Fri 3/1: No Class, Nick out of town		
Week 9		
Lecture 15 (Mon 3/4): Managing the Heap We'll see how the heap fits into the address space. We'll introduce design decisions for implementing `malloc` / `realloc` / `free` , as well as performance tradeoffs (throughput, utilization). Then, we'll compare and contrast stack and heap allocation.	Lecture 15 Slides (lectures/15/Lecture15.pdf) B&O Ch. 9.9 and 9.11 cover heap allocation implementation and memory misuses. There's lots of very useful detail in 9.9 for the heap allocator assignment!	**In**: assign6 **Out**: assign7
Lab 8: Code and Memory Optimization Experiments in optimization and profiling	Be sure to check out our CS107 guide to callgrind (/class/archive/cs/cs107/cs107.1194/resources/callgrind).	
Lecture 16: (Fri 3/8) More Heap Management, Optimization We'll continue discussing heap management and take a look at the compilation process and optimization done by GCC. This could be helpful for final tuning of your heap allocator!	Lecture 16 Slides (lectures/16/Lecture16.pdf)	
Week 10		

Topics	Readings	Assignments
Lecture 17 (Mon 3/11): Wrap-up We'll wrap up the course themes, preview courses and opportunities post-107, and say some final words. Bring questions if you have them!	Lecture 17 Slides (lectures/17/Lecture17.pdf)	
No labs this week.		**In (Thurs. 3/14)**: assign7
Fri 3/15: No Class, study day		
Week 11		
FINAL EXAM Friday 3/22 12:15pm-3:15pm	See the final exam info page (/class/archive/cs/cs107/cs107.1194/exams/final.html) for more information.	