

## MIDTERM EXAM GUIDE

The exam is open book and open note, and focuses on material covered in the lectures, labs, assignments, and additional readings. The exam questions will require you to demonstrate a good understanding of the key concepts and the ability to analyze a particular situation and apply your knowledge.

**Material Covered:** The first half the class concentrates on the following three modules:

1. Data Representations,
2. Introduction to C,
3. Introduction to x86-64 Assembly

Hence, the midterm exam will cover all materials contained in Lectures 2-17. Topics covered in the lectures are listed in detail below:

### Data Representations

- Lecture 2: Bits & Bytes, Representing and Operating on Integers  
*bits and bytes, hexadecimal, integer representations, unsigned integers, signed integers, overflow, casting and combining types*
- Lecture 3: Bits and Bitwise Operators  
*bitwise operators, bitmasks, bit shift operators*
- Lecture 4: Floating Point  
*representing real numbers, fixed point, floating point, tiny floating point, floating point arithmetic, floating point in C*

### Introduction to C

- Lecture 5: Chars and Strings in C  
*characters, string, common string operations (comparing, copying, concatenating, substrings)*
- Lecture 6: More Strings in C, Pointers  
*searching in strings, points, printing the value of a pointer, strings in memory*
- Lecture 7: Arrays and Pointers  
*pointers and parameters, double pointers, arrays in memory, arrays of pointers, pointer arithmetic*
- Lecture 8: The Stack and The Heap  
*the stack, the heap and dynamic memory, realloc, calloc, strdup, freeing the memory with free, stack vs. heap*
- Lecture 9: Realloc, Memory Bugs  
*realloc, memory leakage, and other errors common when working with pointers*
- Lecture 10: void\*, Generics  
*generic swap, generic pitfalls, generic array swap*
- Lecture 11: Function Pointers  
*generic bubble sort, function pointers, generic printing, counting matches, function pointers as variables, generic C standard library functions*

- Lecture 12: const, Structures  
*const, struct, generic stack*
- Lecture 13: Compiling C programs  
*what really happens in GCC, make and makefiles*

#### Introduction to x86-64 Assembly

- Lecture 14: Introduction to x86-64, Data Movement  
*gcc and assembly, looking at an executable, registers, the mov instruction, operand forms, data and register sizes, mov and data sizes*
- Lecture 15: Arithmetic and Logic Operations  
*the lea instruction, logical and arithmetic operations, reverse engineering assembly code*
- Lecture 16: x86-64 Control Flow  
*executing instructions, the program counter register (%rip), unconditional and conditional jump instructions, control mechanics (condition codes, cmp and test instructions), implementation of if statements in assembly*
- Lecture 17: More Control Flow  
*implementation of while and for loops in assembly, other instructions that depend on condition codes (the set and cmov instructions)*