CSCI 150 Outline

Assembly Language/Machine Architecture

*Assembly Language for x86 Processors*

by K. Irvine, 8th Edition, Prentice Hall

**Approved: Fall 2020 Effective: Spring 2021**

|  |  |  |
| --- | --- | --- |
| **Topics** | **Sections** | **Time** |
| Basic Concepts: Assembly language, applications, virtual machine concept, data representation, binary, hexadecimal, two’s complement, Boolean operations | 1.1 – 1.4 | 2.5 Hours |
| Processor Architecture: basic design, instruction execution cycle, floating-point unit, memory management (real-address mode and protected mode), major components, input/output system, 32-bit x86 vs. 64-bit x86 | 2.1 – 2.5 | 2.5 Hours |
| Assembly Language Fundamentals: basic elements of Assembly language, assembler (assembling and linking programs), defining data, symbolic constants, 64-bit programming | 3.1 – 3.6 | 5 Hours |
| Data Transfers, Addressing, and Arithmetic: data transfer instructions, addition and subtraction, data-related operators and directives, indirect addressing, flags, jump and loop instructions | 4.1 – 4.6 | 4 Hours |
| Procedures: linking to an external library, stack operations, defining and using procedures, saving and restoring registers, program design using procedures | 5.1 – 5.5 | 2.5 Hours |
| Conditional Processing: Boolean and comparison operators, conditional jumps, conditional loops, conditional structures | 6.1 – 6.5 | 2.5 Hours |
| Integer Arithmetic: shift and rotate instructions, multiplication and division, extended addition and subtraction, ASCII and packed decimal arithmetic (optional) | 7.1 – 7.4 | 4 Hours |
| Advanced Procedures: local variables, parameters, pass by value and pass by reference, stack frames, memory models, recursion, creating multi-module programs, Java byte codes (optional) | 8.1 – 8.5 | 3.75 Hours |
| Strings and Arrays: string primitive instructions, string routines, two-dimensional arrays, searching and sorting (optional) | 9.1 – 9.5 | 2.5 Hours |
| Structures and Macros: structures and unions, macros (defining, invoking, and examples), macro functions | 10.1 – 10.2 | 2 Hours |
| Memory Management, 32-Bit Windows Programming (optional), floating-point processing | 11.1 – 11.4  12.1 – 12.2 | 2.5 Hours |
| High-Level Language Interface: general conventions, inline assembly code, linking to C++ programs | 13.1 – 13.4 | 2 Hours |
| Disk Fundamentals: tracks, cylinders, sectors, partitions, file systems (FAT and NTFS), disk directory, reading and writing disk sectors, system-level file functions, decoding a FAT table (optional) | 15.1 – 15.2 | 2 Hours |
| BIOS-Level Programming and DOS programming: BIOS-level keyboard input and VIDEO programming (optional), DOS function calls, DOS file I/O services, run-time program structure, interrupt handling, memory organization (cache and  virtual memory) | 14.1  16.1 – 16.3 | 2.25 Hours |

Submitted by: Vo

Notes:

* 1 hour = 1 hour of face time
* The above outline allows for 3 hours review and exams
* 16 Week Term: 1 week = 2.8333 hours (face time)
* 6 Week Term: 1 week = 7.5 hours (face time)
* Keep in mind that most holidays affect MW or MWF classes, so this timeline – NOT the topical outline – may need adjustment