CS 261

- C and Unix/Linux
  - Terminal interface
    - Shells (bash and scp)
    - Basic file utilities (cd, ls, cat, cp, etc.)
    - Command line parameters
    - Text editors (nano, vim, emacs)
    - Build systems (make)
  - Compilation
    - Preprocessing
    - Compiling
    - Assembling
    - Linking
  - Debugging
    - Breakpoints
    - Watchpoints
    - Step into
    - Step over
    - Backtracing
  - Memory model
    - Static code
    - Static data
    - Stack
    - Heap
  - C language
    - Pointers
    - Strings
    - File I/O
    - Getopt
    - Structs
    - Typedefs
    - Bitwise operations
- Binary representations
  - Positional representations
    - Binary
    - Octal
    - Decimal
    - Hexadecimal
  - Endianness
    - Big
    - Little
  - Integers
    - Unsigned
    - Sign magnitude
    - Ones' complement
    - Two's complement
    - Bias/excess/offset
  - Bitwise operations
    - AND
    - OR
    - XOR
    - Left shift
    - Right shift
      - Logical
      - Arithmetic
  - Floating point
    - Normalized
    - Denormalized
    - Infinity (+/-)
    - Not-a-Number (NaN)
  - Text
    - ASCII
    - Unicode
    - UTF-8 vs. UTF-16/32
  - Code
    - Machine code
      - RISC vs. CISC
      - Opcodes
      - Operands
      - Registers
    - Assembly code
- Assembly/machine code
  - Movement & arithmetic
    - Operands
      - Immediate
      - Register
      - Memory
    - Addressing modes
      - Direct
      - Indirect
      - Base + displacement
      - Indexed
      - Scaled indexed
    - Stack management
    - Integer arithmetic
    - Bitwise logical & shifts
    - Floating-point arithmetic
      - Vector (SIMD) instructions
      - SSE/AVX instruction sets
      - XMM/YMM registers
  - Control flow
    - Condition flags
      - CF
      - OF
      - SF
      - ZF
    - Comparisons and tests
    - C goto code
    - Jumps
      - Labels
      - Unconditional vs. conditional
      - Direct vs. indirect
    - Conditional moves
    - Loops
      - Do-while
      - Jump-to-middle
      - Guarded-do
    - Jump tables
  - Procedures
    - Application Binary Interface (ABI)
    - Calling conventions
    - Stack frames
      - Return address
      - Call and return
      - Base pointers
    - Parameters
    - Return values
    - Registers
      - Caller-saved
      - Callee-saved
    - Local variables
  - Data structures
    - Arrays
      - Homogeneous vs. heterogeneous
      - Pointer arithmetic
      - Indexing
      - Nested / multidimensional
    - Structs / records
      - Offsets
      - Alignment
    - Unions
  - Buffer overruns
    - Shellcode
    - Stack randomization
    - Canary values
    - Read-only code regions
- Architecture
  - von Neumann architecture
  - Combinational circuits
    - Logic gates
      - Relays
      - Transistors
      - Digital signals
      - Universal gates
    - Boolean formulas
    - HDL
    - Circuit equivalency
    - Basic
      - Equality checkers
      - Multiplexers
      - Half adders
      - Full adders
    - Advanced
      - Ripple-carry adders
      - Adder/subtractors
  - Sequential circuits
    - Circuit "memory" via feedback
    - Time sensitivity
    - Edge-triggers vs. master-slave
    - Clocks
    - Basic
      - SR latch
      - D latch
      - Flip-flops
    - Advanced
      - Registers
      - Register files
      - Memory
      - Shift registers
      - Counters
  - CPU design
    - Stages
      - Fetch
      - Decode
      - Execute
      - Memory
      - Write back
      - PC update
    - Components
      - Clocked PC register
      - ALU / register file / memory
      - Control logic
      - Wires & buses
    - "No reading back" principle
    - Throughput and latency
    - Pipelining
      - Instruction-level parallelism
      - Tradeoff: throughput vs. latency
      - Issue: non-uniform partitioning
      - Issue: data and control dependencies
      - Stalling and forwarding
      - Branch prediction
  - Memory
    - Technologies
      - SRAM
      - DRAM
      - HDD / SSD
      - Tape
      - Historical trends
    - I/O architecture
      - Caches
      - Main memory
      - I/O buses
      - Disk drives
      - DMA
      - Remote storage
    - Locality
      - Temporal
      - Spatial
      - Strides
      - Latency
      - Prefetching
  - Caching
    - Implementations
      - Direct-mapped
      - Set-associative
      - Fully-associative
      - Replacement policies
        - Random replacement
        - Least recently used
        - Least frequently used
      - Write-through vs. write-back
      - Write-allocate vs. no-write-allocate
    - Cache memory
      - Hit vs. miss
      - Hit/miss rate
      - Read throughput
      - Memory mountain
- Operating systems
  - Kernel
    - User vs. kernel mode
    - Monolithic vs. microkernel
    - Unix system design philosophy
  - Virtual memory
    - Virtual vs. physical memory
    - Address spaces
    - Memory pages
    - Address translation
    - Page faults / swapping
    - MMU and TLB
    - Allocation
      - Explicit
      - Implicit
  - Exceptions
    - Async vs. sync
    - Recovery options
    - Types
      - Interrupt
      - Trap
      - Fault
      - Abort
    - Exception table
    - System calls
      - SysV
      - POSIX
    - Signals
      - Common signals
        - SIGALRM
        - SIGINT
        - SIGSEGV
        - SIGKILL
        - SIGTERM
      - Signal handlers
      - signal() / raise()
      - Debuggers
  - Processes
    - Logical flows
    - Concurrency
    - Multitasking / time-sharing
    - Parallelism
    - Context switching
    - C language
      - fork
      - wait
      - execve
    - Shell
      - Jobs and process groups
      - Foreground vs. background
  - Files
    - Types
      - Regular
      - Directory
      - Socket
    - File systems
      - File hierarchy
      - Absolute vs. relative file name
      - Mount points
      - Networked file systems
      - Metadata (stat)
    - Implementation
      - Low-level syscalls
      - File sharing
      - High-level standard C functions
      - Robust network I/O functions
      - OS data structures
        - Descriptor tables
        - Open file table
        - v-node / inode table
    - Standard I/O
      - Input
      - Output
      - Error
      - Redirects
      - Pipes
    - Permissions
      - Unix permissions
      - ACLS
  - Threads
    - Processes vs. threads
    - Shared memory
    - POSIX threads (pthreads)
    - Master/worker parallel pattern
    - Data races
    - Mutual exclusion
    - Automated parallelism
    - History of parallelism
      - Uniprogramming
      - Multiprogramming
      - Multiprocessing
      - Distributed computing