**CACHING**

Write Policy

**Write back:** hit, standard, update mem on eviction

**Through:** update mem with cache

**Write-allocate:** miss, load and then write

**No-write allocate:** directly to memory

Placement

**Direct:** blocks map to one line, causes conflict misses

**Fully-associative:** blocks map anywhere, best hit rate, most complex

**Set:** a block can map to any line in a specific set

Lookup

**Valid bit:** being used, if 0, no assumptions!

**Dirty (write back):** modified

**Mem addr:** tag | index | offset

**Index:** set selection

**Tag:** line matching

**Offset:** within the block, pay attention to offset bit!

Replacement Policy

**Least recently used:** evict oldest used line in full set, recently used means use again soon

General Model

**:** # of set bits

**:** # of offset bits

**:** # of tag bits

**:** # of sets

**:** associativity, lines/set

**:** block size in bytes

Memory Hierarchy

Register, LX Cache SRAM, DRAM, SSD local net, HDD, tape Web

**PROCESSES**

**Program:** static instructions

Process: executing program

**Job:** used by shell process to manage children

**Process groups:** children under parent, direct shell children get own pgid

**Orphan:**

Exceptions

Event triggered

**Sync:** caused by program

**Async:** external, hardware, signals, system, causes context switch

**Handlers:** processes them

Context Switching

Caused by exceptions, managed by kernel, another process running

Process Model

**Fork:** creates new process exactly like parent

**Exec:** child switches state new process

**Tree:** remember shell exercises that you got all wrong

Signals

Async inter-process comms triggered by events

Process decides may decide response

**Signal blocking:** masks, pending signals… shell exercise print order question

Zombies

Process that terminated but still using resources

**Reaping:** using and handle termination status

Concurrency

Inter-process, w/ sig handlers

Threads

Concurrent, share data, not regs and stack

**Race conditions!**

**VIRTUAL MEMORY**

Virtual maps to physical, one to many

Translated by memory management unit

Organized in , like cache blocks

“Unlimited” virtual memory

Make’s compiler job easier, allows for reuse of programs