CS 471 Operating Systems

Yue Cheng

George Mason University
Fall 2019

Final Exam

- Tuesday, Dec 17, 1:30pm – 4:15pm
  - 150 min, closed book, closed note

- Covering topics from lec-0 to lec-6b

- Overall topic distribution: Three pillars of OS
  - Synchronization: **~20%**
  - CPU scheduling, memory mgmt. & cache replacement policies: **~40%**
  - I/O, storage (HDD, flash), RAID, and FS: **~40%**

Solving Synchronization Problems w/ CV

- Condition variables (CV): an explicit queue that threads can put themselves when some condition is not as desired (by waiting on that condition)

- Good rules of thumb when using CV
  - Always do wait and signal while holding the lock
  - Lock is used to provide mutual exclusive access to the shared resource
  - `while()` is used to always guarantee to re-check (re-enter) if the condition is being updated by others

Classic Problems of Synchronization

- Producer-consumer problem (various CV-based implementations)
  - How/why a buggy implementation breaks
  - Why the correct implementation works
- Readers-writers problem
  - Should understand how it works
  - Constraints for the readers, and constraints for the writers

CPU Scheduling

- FIFO, SJF, RR, Priority, **MLFQ**
- How MLFQ jointly optimizes for multiple objectives

Memory Management: Addresses & PT

- Virtual addresses and physical addresses
  - VPN, PFN, page offset
  - Virtual address = VPN | offset
    - Assume a 18-bit virtual address space
    - 4KB pages: offset has 12 bits
    - `18–12=6`: VPN has 6 bits (2^6 pages)

- Virtual to physical address translation
  - (Basic) linear page table: index of array using VPN
    - Each PTE contains PFN and other status info

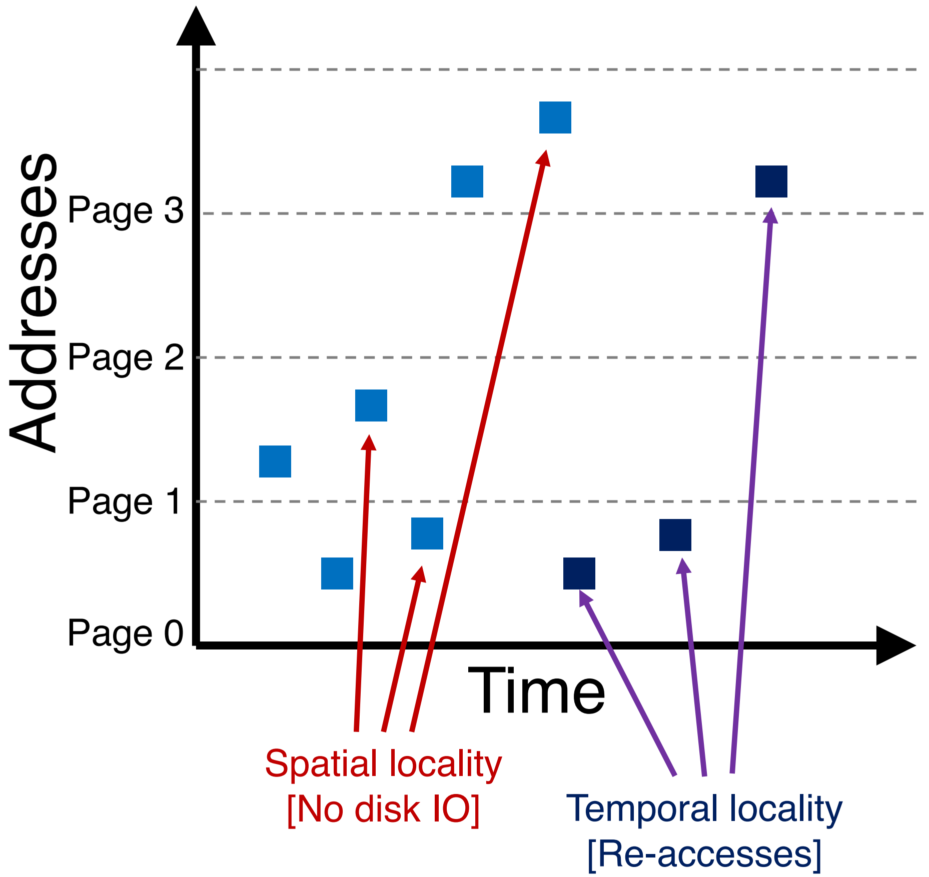
Memory Management: Demand Paging vs. Thrashing

- Demand paging
  - Paging-in: Access to a single byte on disk causes a page fault, which brings in the whole page
  - Swapping-out: System is suffering memory pressure
- Thrashing
  - High paging activities – The system is spending more time paging than executing

Memory Management: Locality & Cache Replacement Policies

- Cache exploits both
  - Spatial and temporal locality
- LRU/FIFO/OPT
- Spatial locality
  - Access to a single byte on disk brings in the whole page
- Temporal locality
  - Repetitive accesses to the same data

Assume all data are initially paged out to disk

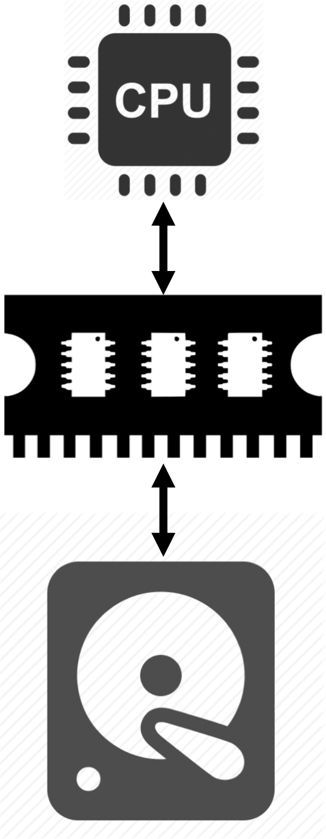


Spatial locality
[No disk IO]

Temporal locality
[Re-accesses]

Memory Management: AMAT

- Storage hierarchy:
  - CPU cache (e.g., L1, L2)
  - Main memory
  - Disk
- Data access flow
  - A CPU cache miss → A memory access
  - A page fault (memory miss) → A disk access



Storage: I/O and Storage Basics

- PIO vs. DMA
  - How they work
- Disk scheduling algorithms
  - FIFO, SPTF, SCAN, C-SCAN, C-LOOK
- Hardware storage mediums
  - HDD:
    - Internal mechanical pieces
    - Performance model: seek, rotate, data transfer
  - Flash:
    - Asymmetric read-write performance
    - Due to inherently different architecture

Storage: RAID and File Systems

- RAID
  - Tradeoffs of different RAID configurations
  - RAID-0: No redundancy, perf-capacity upper bound
  - RAID-1: Mirroring
  - RAID-4: A disk is solely used for storing parity
  - RAID-5: Rotating parity across disks

- File systems
  - File name abstractions: inode, path (files, directories), fd
  - Various FS syscalls: what they do
  - Disk-based FS implementation: on-disk structures

Question Types

- Multiple-choice questions
- True/false questions
- Problem solving

Good Luck!