Section 10: Last section!

Final review

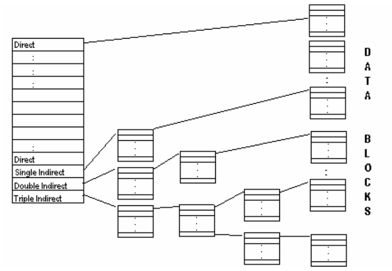
Review: file systems

- Two improvements BSD Fast File System implemented?
- What workload is LFS was based on?
- Calculate max UNIX file size for 1K blocks
- The sequence of actions that occurs when a user executes the following, in terms of inodes and directory entries:
  `touch file && ln file file2 && rm file && rm file2`

Review: file systems

- Two improvements BSD Fast File System implemented?
  - Cylinder groups (localize metadata and file data in the same or close cylinder groups)
  - Blocksize 1->4K, disk params
- What workloads/technical advances is LFS was based on?
  - Frequent writes of lots of small files; large main memories
- Max UNIX file size for 1K blocks?
  - Direct, single indirect, double indirect, triple indirect
  - (12+256+256^2+256^3)*1K = around 16 GB
- The sequence of actions that occurs in terms of inodes and directory entries: `touch file && ln file file2 && rm file && rm file2`
  - …
  - What about ln –s?

Unix File System



Source: wikipedia

File protection

- Compare ACLs and capabilities

File protection

- Compare ACLs and capabilities
- ACLs:
  - Object-based protection matrix
  - Support auditing
  - Provide tight access control
- Capabilities:
  - Principal-based protection matrix
  - Ease sharing
  - Support the least privilege principle better

Distributed File Systems

- What are some difficult issues in distributed FSs?
- Compare NFS, AFS and Sprite

Distributed File Systems

- What are some difficult issues in distributed FSs?
  - Naming: how to ensure uniqueness
  - Performance: how to hide network delay
    - Caching: how to make caches coherent
  - Replication: how to maintain replicas consistent
- Compare NFS, AFS and Sprite
  - NFS: for LANs; small in-mem caches; flush on close
  - AFS: for WANs; large, disk-based caches; single namespace; flush on close; refresh on open
  - Sprite: for diskless, large-memory clients; use cache invalidation protocol

Review:

- Consider a modern desktop computer on which the hard disk is spinning. The more significant delay in reading from a 4K byte file that has not been accessed in a long time is:
  a) context switch to enter the operating system
  b) time spent in the OS to determine what disk blocks to fetch
  c) latency awaiting disk rotation and arm movement
  d) transfer time

Review: Disks

- Consider a modern desktop computer on which the hard disk is spinning. The more significant delay in reading from a 4K byte file that has not been accessed in a long time is:
  a) context switch to enter the operating system
  b) time spent in the OS to determine what disk blocks to fetch
  c) latency awaiting disk rotation and arm movement
  d) transfer time

- Four out of the five of these instantiate a single basic principle. But the fifth one is essentially unrelated. Which is the unrelated one?
  a) processor (L2) cache
  b) disk buffer pool
  c) page table
  d) translation lookaside buffer
  e) DNS cache

Review: virtual memory

- Segmentation doesn’t have this problem of wasted space.
- How does copy-on-write work? What is it used for?
- What is Belady's anomaly?
- What is Belady’s algorithm?

Review: virtual memory

- Segmentation doesn’t have this problem of wasted space.
  - Internal fragmentation
- How does copy-on-write work? What is it used for?
  - …
- What is Belady's anomaly?
  - Bad property of FIFO – fault rate can increase with more allocated frames
- What is Belady’s (OPT) algorithm?
  - Optimal page replacement – evict page the won’t be needed longest into the future

- People do not program the Belady’s page replacement algorithm today because:

  a) it is too expensive to build the hardware support
  b) it is impossible to do so
  c) page replacement costs are not sufficiently high for optimality to be a significant concern
  d) there exist better-performing page replacement algorithms
  e) This policy are only applicable in computers that support page reference bits within the PTE. Not all processors do so.

Review: Security

- **T / F** A *microkernel* is a category of operating systems designed for sensors or other forms of micro-processors.
- **T / F** A *priority inversion* is said to occur if a higher priority task is waiting for a lower-priority task to perform some action.
- Why does Windows require ctrl+alt+del before you type your username/password?
- Very briefly, how would you implement a buffer overflow attack?

Review: Misc

Which of the following abstractions are implemented entirely in the kernel of an OS like Linux:

a) remote object invocation and remote procedure call
b) the process abstraction
c) dynamically linked libraries (DLLs)
d) the window manager
e) public key encryption/decryption

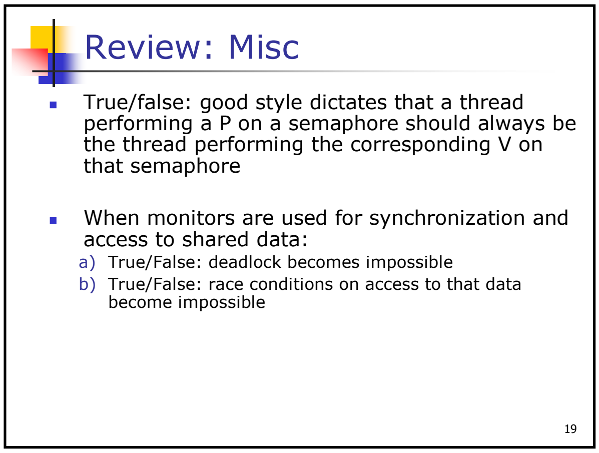
Review: Misc

- Which of the following are not typically saved when a process issues a system call that blocks, causing a context switch to some other process:
  - contents of registers
  - program counter value
  - interrupt vector table
  - name of the process that was executing
  - elapsed time since process last started executing

Review: Misc

- The best reason to use semaphores rather than busy-waiting in a user-level program is:

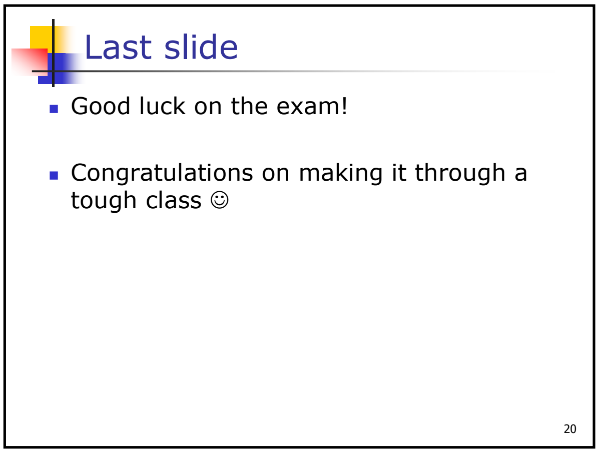
  a) decreases the chance of deadlock
  b) decreases the chance of livelock
  c) allows more different types of synchronization to be coded
  d) avoids wasted processor cycles
  e) avoids expensive context switches



Review: Misc

- True/false: good style dictates that a thread performing a P on a semaphore should always be the thread performing the corresponding V on that semaphore

- When monitors are used for synchronization and access to shared data:
  a) True/False: deadlock becomes impossible
  b) True/False: race conditions on access to that data become impossible



Last slide

- Good luck on the exam!

- Congratulations on making it through a tough class ☺