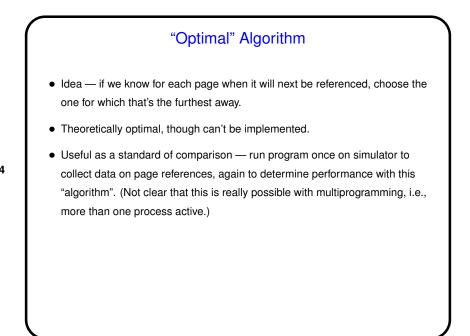


# Minute Essay From Last Lecture Only one person mentioned noticing anything strange. Could it be that most of you always exited the program with control-C?? More than one person mentioned not even testing what happens on bad commands (!?). Surely you do better at testing "corner cases" in other courses requiring programming?

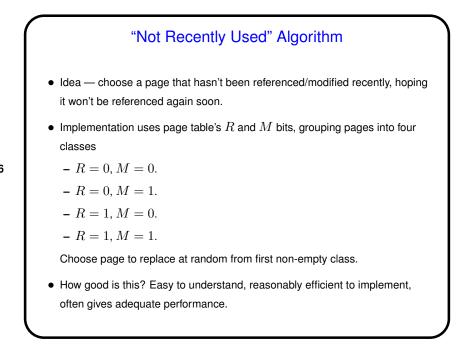


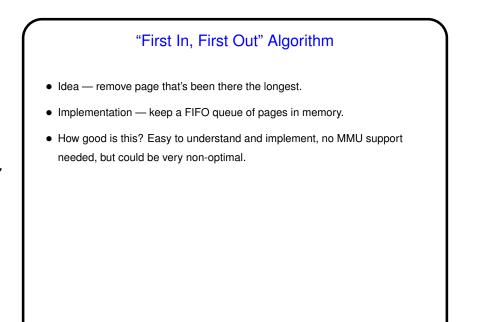
- Processing a page fault can involve finding a free page frame. Would be easy if the current set of processes aren't taking up all of main memory, but what if they are? Must steal a page frame from someone. How to choose one?
- Several ways to make choice (as with CPU scheduling) "page replacement algorithms".
- "Good" algorithms are those that result in few page faults. (What happens if there are many page faults?)
- Choice usually constrained by what MMU provides (though that is influenced by what would help O/S designers).
- Many choices (no surprise, right?) ....





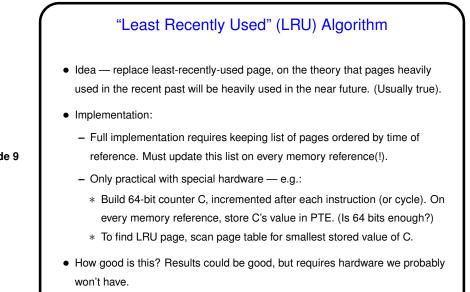
- Recall many architectures' page table entries contain bits called "R (referenced) bit" and "M (modified) bit". Idea is that these bits are set (to 1) by hardware and cleared by software (O/S) in some way that's useful.
- *R* bit set on any memory reference into page. Typically cleared by O/S periodically (on "clock ticks"). Allows tracking which pages have been used recently.
- *M* bit set on any write/store into page, cleared when page is written out to disk. If off, means that if we need this page's page frame, no need to write contents out to disk (since presumably we have a copy from a previous write).





## "Second Chance" Algorithm

- Idea modify FIFO algorithm so it only removes the oldest page if it looks inactive.
- Implementation use page table's R and M bits, also keep FIFO queue. Choose page from head of FIFO queue, *but* if its R bit is set, just clear R bit and put page back on queue.
- Variant "clock" algorithm (same idea, but keep pages in a circular queue).
- How good is this? Easy to understand and implement, probably better than FIFO.

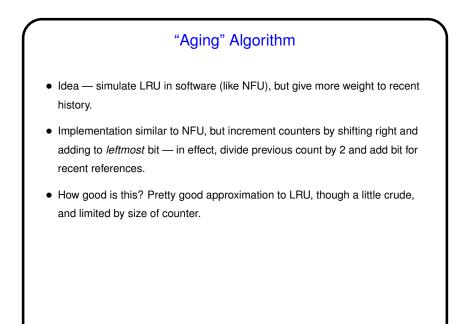


## "Not Frequently Used" (NFU) Algorithm

- Idea simulate LRU in software.
- Implementation:
  - Define a counter for each PTE. Periodically ("every clock-tick interrupt") update counter for every PTE with  ${\cal R}$  bit set.

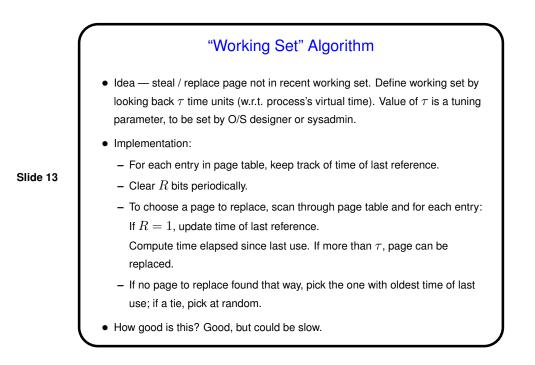
Slide 10

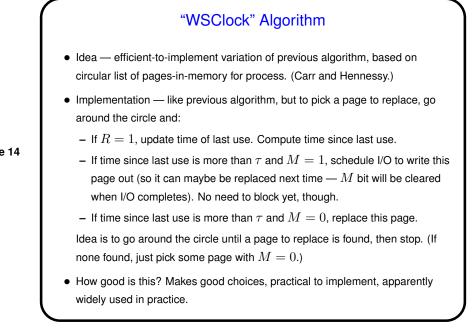
- Choose page with smallest counter.
- How good is this? Reasonable to implement, could be good, but counters track full history, which might not be a good predictor.

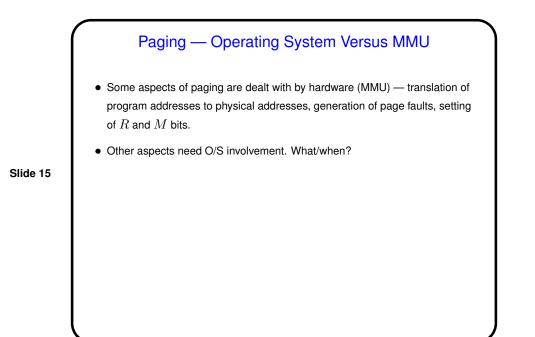


Slide 12

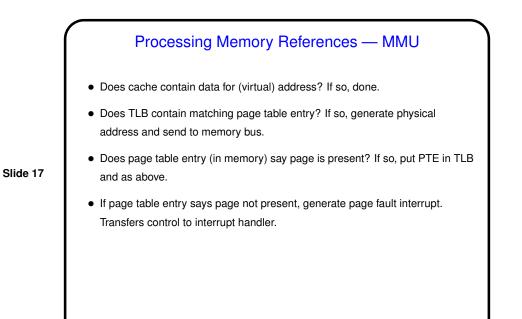
# Sidebar: Working Sets Most programs exhibit "locality of reference", so a process usually isn't using all its pages. A process's "working set" is the pages it's using. Changes over time, with size a function of time and also of how far back we look.

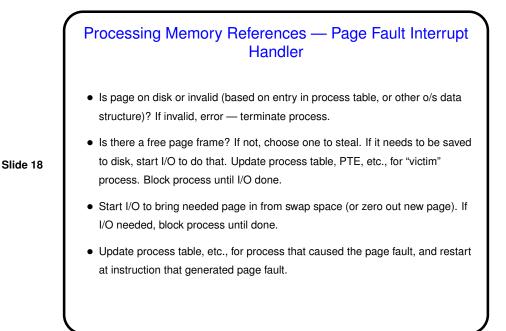


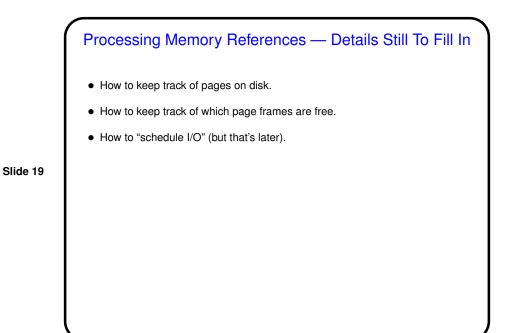


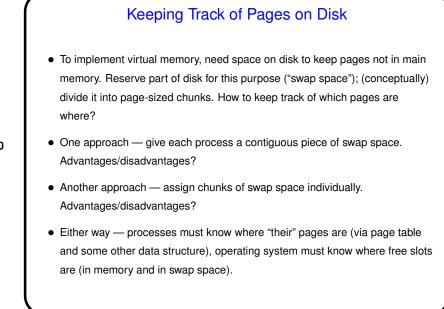


Paging — Operating System Involvement
Process creation requires setting up page tables and other data structures. Process termination requires freeing them.
Context switches require changing whatever the MMU uses to find the current page table.
And of course it's the operating system that handles page faults!
Some details ...









## Minute Essay

• Another story from long ago: Once upon a time, a mainframe computer was running very slowly. The sysadmins were puzzled, until one of them noticed that one of the disk drives seemed to be very busy and asked "which disk are you using for paging?" The answer made everyone say "aha!" What was wrong (to make the system so slow)?

Slide 21

• How did the midterm compare to your expectations (topics, level of difficulty, ...)?

### Minute Essay Answer

- The disk being used for paging was the one that was very busy. So, mostly likely the system was spending so much time paging ("thrashing") that it wasn't able to get anything else done. Usually this means that the system isn't able to keep up with active processes' demand for memory.
- Slide 22
- Memory sizes have increased to a point where the odds aren't as good as they were. But a few years ago we did run into problems with the machines in one of the classrooms trying to run both Eclipse and a Lewis simulation, and then more recently with some of them attempting to run a background program that asked for more memory than its author intended.