Operating Systems W15L2 - Final Review

Class	Operating Systems
Created	@Dec 9, 2020 6:26 PM
Materials	
Reviewed	☐
Type	Lecture

Exam Info

- Same as midterm
- Exam available from December 16th 10 am for 24 hours
- Keep your answers focused
- Submit as PDF → Handwritten is acceptable, just ensure it's legible
- Exam is cumulative Make sure any assumptions you make actually make sense
  - i.e. don't overthink when you might not need to
- View solutions as something that makes sense on the user's end

Problem 1

- The OS virtualizes I/O
- Many ways it does this, but main idea is that processes see something that is not "real"

- Virtualization allows a more efficient way of using devices, i.e. two processes waiting for the actual printer
- Also, it's necessary for security, which is a part of keeping proccesses separate from each other
- Virtualization is good becuase...
    1. More efficient device use
    2. Security
    3. Compiling and linking much easier

Problem 2

- Yes, in the event the process is removed from the core
    - Could be removed due to timeout, interrupt, a sleep call, exit, error, nad probably some others
- For timing out, it is specifically a process-based function

Problem 3

- No
- In the end, a process is just a bunch of pages
- Given `p` cores, then how many processes must be in memory at the same time?
    - At least 1 per core
    - Could be just 1 process with `p` threads
    - Answer: 1 to `p`, many different possibilities

Problem 4

- Max number of faults that can be causes is 2
    - For one instructino to cause two, that's pretty bad
    - Due to a misalignment of page tables
    - Page table being in memory is taken care of by OS, prior to loading

- See recording for an extra version of this problem modified

Problem 5

- Doesn't matter
- Multilevel page tables are to solve excessively sized page tables
  - i7 and i9 chips use 4-level page tables

Problem 6

👀

Problem 7

- A smaller page inplicates that we have more pages, and thus a larger page table
  - This larger page table is bad for performance
- A page is what moves between memory and disk, so the above is bad because more trips to the disk
- Waste of space, also big pages can cause poor fragmentation

Problem 8

👀

Problem 9

- The *type* of information will be the same
  - i-nodes contain the blocks in the file, the i-node just sees it like that
  - The difference only occurs in the device driver / at that level
  - i-node is independent from disk type (a good example of virtualization)