Note: in 2021 Spring, the midterm was given as an oral exam during which student will meet with Prof. Guo one-on-one to answer questions selected by Prof. Guo at exam time. **For 2022 Spring, the midterm will be an open-book, in-class written exam. As such, we will likely have fewer questions that can be realistically finished in 1 hours 20 mins.**

Midterm Exam: Questions

1. What is an operating system and why are they important to study?

2. You write a UNIX shell, but instead of calling `fork()` then `exec()` to launch a new job, you instead insert a subtle difference: the code first calls `exec()` and then calls `fork()`. What is the impact of this change to the shell, if any?

3. Assume the following schedule for a set of three jobs, A, B, and C:

   - A runs first (for 10 time units) but is not yet done
   - B runs next (for 10 time units) but is not yet done
   - C runs next (for 10 time units) and runs to completion
   - A runs to completion (for 10 time units)
   - B runs to completion (for 5 time units)

   Which scheduling disciplines could allow this schedule to occur? **(1)** FIFO; **(2)** Round Robin; **(3)** Multi-level Feedback Queue; **(4)** Lottery Scheduling

4. The lottery scheduler relies on random numbers in order to pick the winner of a lottery. This subtly-different lottery scheduler uses a simplified random number generator, which rotates through the following five pseudo-random numbers: 133, 12, 800, 442, 917. How does this change affect the behavior of the lottery scheduler?

5. The multi-level feedback queue (MLFQ) requires setting a number of parameters, e.g., how frequently to move all jobs back to the top-most queue. On a particular system, this is usually done every 10 seconds; if we shorten this value to 1 second, how does this subtle difference affect the MLFQ scheduler? In general, what is the effect of shortening this value?

6. The Multi-level Feedback Queue (MLFQ) is a fancy scheduler that does lots of things. Which of the following things could you say is possible for MLFQ approach? **(1)** MLFQ learns things about running jobs; **(2)** MLFQ starves long running jobs; **(2)** MLFQ uses different length time slices for jobs; **(3)** MLFQ uses round robin; **(4)** MLFQ forgets what it has learned about running jobs sometimes.

7. The simplest technique for virtualizing memory is known as dynamic relocation, or "base and bounds". Assuming the following system characteristics: **(1)** a 1KB virtual address space **(2)** a base register set to 10000 **(3)** a bounds register set to 100. Which of the following physical memory locations can be legally accessed by the running program? **PA1: 0**, **PA2: 1000**, **PA3: 10000**, **PA4: 10050**, and **PA5: 10100**.

8. Segmentation is a generalization of base-and-bounds. Which possible advantages does segmentation have as compared to base-and-bounds? **(1)** Faster translation **(2)** Less physical memory waste **(3)** Better sharing of code in memory **(4)** More hardware support needed to implement it **(5)** More OS issues to handle, such as compaction

9. Assume the following in a simple segmentation system that supports **two** segments: one (positive growing) for code and a heap, and one (negatively growing) for a stack: **(1)** virtual address space size is 128 bytes **(2)** physical memory size is 512 bytes **(3)** Segment 0's base register is 0, and its limit is 20 (decimal) **(4)** Segment 1's base register is 0x200 (decimal 512), and its limit is 20 (decimal) Which of the following are **valid** virtual memory accesses? **VA1: 0x1d (decimal: 29) VA2: 0x7b (decimal: 123) VA3: 0x10 (decimal: 16) VA4: 0x5a (decimal: 90) VA5: 0x0a (decimal: 10)**

10. In a simple page-based virtual memory, with a linear page table, assume the following: **(1)** virtual address space size is 128 bytes **(2)** physical memory size of 1024 bytes **(3)** page size of 16 bytes **(4)** The high-order (leftmost) bit of a page table entry is the VALID bit. If the bit is 1, the rest of the entry is the PFN. If the bit is 0, the page is not valid. Here are the contents of the page table (from entry 0 down to the max size).

VPN	PTE
0	0x80000034
1	0x00000000
2	0x00000000
3	0x00000000
4	0x8000001e
5	0x80000017
6	0x80000011
7	0x8000002e

Which of the following virtual addresses are valid? **VA1: 0x34 (decimal: 52)**, **VA2: 0x44 (decimal: 68)**, **VA3: 0x57 (decimal: 87)**, **VA4: 0x18 (decimal: 24)**

11. TLBs are a critical part of modern paging systems. Assume the following system: **(1)** page size is 64 bytes; **(2)** TLB contains 4 entries; **(3)** TLB replacement policy is LRU (least recently used). Each of the following represents a virtual memory address trace, i.e., a set of virtual memory addresses referenced by a program. In which of the following traces will the TLB possibly help speed up execution? **Trace A: 0, 100, 200, 1, 101, 201, ... (repeats in this pattern)**; **Trace B: 0, 1000, 2000, 3000, 4000, 0, 1000, 2000, 3000, 4000, ... (repeats)**; **Trace C: 0, 1, 2, 3, 4, 5, 6, 0, 1, 2, 3, 4, 5, 6, ... (repeats)**; **Trace D: 0, 100, 200, 300, 0, 100, 200, 300, ... (repeats)**

12. The multi-level page table is something that cannot be avoided. In this question, you will get your chance at a question about this foreboding structure. Fortunately, you don't have to perform address translations with the multi-level page table. Instead, just answer true/false these questions about multi-level page table. Be ready to defense your choices! **(1)** A multi-level page table may use more pages than a linear page table. **(2)** It is easier to allocate pages of the page table in a multi-level page table, as compared to a linear page table. **(3)** Multi-level page table looks up take longer than linear page table lookups. **(4)** TLBs are useful in making multi-level page tables even smaller.

13. OS is full of techniques that follow classic systems design principles such as hybrid, indirection, and caching. Pick a concrete technique that we have covered that follows at least one of the three design principles, and describe **(1)** what problem is this concrete technique solving; **(2)** how does this technique work.