CSC 4320/6320: Operating Systems

Guideline for Exam 1

Spring 2025

----

1 A4 size sheet, double-sided, handwritten by you is allowed in the exam.

Please print your name and date clearly on top of your cheat sheet.

MCQ: 40-50%

Short Answer type/Coding/Code output: 50-55%

Syllabus for Exam 1: Chapter 1 to Chapter 3 (Exclude C Review)

Go over slides from W1 lecture 1 to Week 4 lecture 2

Do your HW1 and HW2 very well.

Go over the slides from Chapter 1 to Chapter 3

Representative/Sample Questions

(Caution: Not a complete list of questions)

Chapter 1:
_____

1. What are the basic components of a computer system? Show their relative positions in a computer's logical organization.
2. Learn the functions of an operating system from different perspectives – users, workstations, mobile devices, embedded systems.
3. What is a system program and an application program?
4. Define Operating System.
5. You must learn concepts from the slides. Also read questions from slides.
6. How is a computer system generally organized? Show/describe.

7. What is an interrupt vector? Describe what happens when an interrupt happens.
8. The timing diagram of an interrupt.
9. Maskable and non-maskable interrupt.
10. Difference between synchronous and asynchronous I/O.
11. Learn the storage device hierarchy.
12. How does DMA work? When is DMA preferred over Interrupt? State some applications.
13. What is multiprocessing? Distinguish between symmetric and asymmetric multiprocessing.
14. Learn about CPU, Processor, Core, Multicore, and Multiprocessor
15. Multicore systems vs. Multiprocessor systems---Which one is better? Why?
16. What is NUMA System?
17. Does a dual-core system have its own cache memory? Why or why not?
18. How are clustered systems different from multicore systems?
19. Distinguish between multiprogramming and multitasking.
20. What are the benefits of multiprogramming over sequential processing?
21. How time-sharing systems run multiple processes currently even with a single processor?
22. Distinguish between user model and privileged (kernel) mode. Mention some operations that can be done in user-mode and in kernel-mode.
23. Process management and memory management activities
24. What is Caching? How does cache memory work?
25. Emulation vs. Virtualization. What are their use cases?
26. Name some kernel data structure and learn about them.

Chapter 2:

1. OS services – Mention some OS services for example.
2. What is a system call? Mention at least five system calls and tell their functions.
3. How would you access/use system calls in your program?
4. What are the different ways to pass parameters to system calls? Which method would you choose?
5. Learn the functions of major system calls like fork (), exit (), wait (), open (), read (), write (), close (), getpid(), sleep(), ioctl(), etc.
6. How do fork () and exec () system calls work together to run a new process?
7. Mention some system programs that provide some system services.
8. Linkers and loaders – what do they do?

9. Object code vs. Executable code
10. Why are applications OS specific?
11. Learn about different OS structure – simple/monolithic, layered, and microkernel. Benefits and challenges of each structure.
12. What are modules (loadable kernel modules)?
13. Have some ideas about the structures of macOS and iOS, Darwin, Android.
14. How to use fork () to create processes? Practice small programs.

Chapter 3

1. Program and process distinction.
2. Process layout in memory.
3. Process control blocks and process states. Define each state. Show state transition diagram.
4. What is context switching? Why is it a pure overhead?
5. Practice fork system calls.
6. Practice from HW2
7. Combine fork and execl to run an executable **hello**
8. Practice zombie and orphan process and (code).
9. How does a parent process wait for the child for its completion and the status?
10. Inter-process communication model – shared memory model and message passing model.
11. Race condition. Why does it happen? What are the ways you can think of to resolve the problem?
12. Problems with producer-consumer model.
13. Why there is no race condition in the circular array solution where one location is unused?
14. What is blocking/unblocking message passing?