

Rules for Midterm, 150 points: Work alone and independently; open books, open internet, open notes. Place your name onto your midterm. Each question is worth 10 points max. Answer via single words or short phrases where feasible, minimize full sentences.

Mario Palacios

CSC 139 – Section 06

1. Outline briefly thread, hyper-thread, key difference between them.

- thread → runs concurrently, has own stack, register set, condition codes, and control over full static address space
- hyper-thread → simultaneous multi-threading, faster than a single core that is shared with multiple threads, has a least on real, complete core, with all ALU functions, registers, condition codes in silicon and can access memory

2. How does an OS become active upon **power-up**? Any typical OS includes a loader that moves object programs into memory; yet how does the OS itself get moved into memory?

- Bootstrap program is loaded up, which then loads the OS kernel and starts execution. It stores bootstrap in the firmware (ROM) and initializes all aspects of the system.

3. What does **Privileged Mode** execution mean in an OS?

- Privileged Mode allows the user to execute system calls, this mode also protects the kernel from being misused. Thus, Privilege Mode can run only privileged instructions, if one of these instructions ran outside privileged mode then it will trap the user to OS and not execute.

4. Processes run concurrently, sharing resources, such as memory. How can the OS prevent a process from accessing data of other processes?

- via virtual addresses and memory

5. Briefly sketch **Demand Paged Virtual Memory Management**. Explain “demand”, “paged”, “virtual”, “page-size”, “page alignment”. Contrast briefly with physical memory.

- virtual address space can be way larger than physically available memory
- VM via paging, only part of program needs to be in memory for execution

6. Outline the core message of Amdahl’s Law

- Is a formula that identifies potential performance gains from adding additional computing cores to an application that has both serial (*nonparallel*) and parallel components. [[*Governs parallel computing*](#)]

7. Explain **program** and OS **process**. What are key differences?

- **program**, contains set of instructions that can complete a specific task, and exists until it is either executed or deleted. OS **process**, is a program that is being executed and exists until the program is terminated.

8. Outline **relative speeds** of *storage devices* on a Computer System from fast to slow.

- Registers → 20,000 – 100,000 (MB/sec)
- Cache → 5,000 – 10,000 (MB/sec)

- Main Memory → 1,000 – 5,000 (MB/sec)
- Solid State Drive → 500 (MB/sec)
- Magnetic Disk → 20 – 150 (MB/sec)

9. Devices such as keyboard, printer, display etc. are part of a computer system. How do they receive, share, and communicate information with the OS?

- Devices receive and share by an IO transfer of information between the main memory and peripheral device. While interrupts are sent from the device to the CPU to inform that operations are completed.

10. List and explain in a word or short phrase various (≥ 5) physical and program-related parameters of **mass-storage**. What are some mass storage components of a computer?

- Some mass storage components of a computer are: speed, cost, capacity, access method, access time, and volatility

11. Argue, whether it is "fair" that some process **p** is not currently running, though **p** has all resources (except the CPU) needed to execute.

- It is considered "fair" if **p** has a higher priority program and needs to be executed once the CPU is available

12. List process-related responsibilities of a multi-user, time-sharing OS.

- creating and deleting processes, both user and system, suspending and resuming processes, providing mechanisms for process synchronization/communication/deadlock handling

13. Explain typical data **capacities** and general access times of different HW storage devices; list from smallest to largest. (Note: enumerated in earlier question 8.)

- registers
 - capacity: < 1KB
 - access time: 0.25 – 0.5 (ns)
- cache
 - capacity: < 16MB
 - access time: 0.5 – 25 (ns)
- main memory
 - capacity: < 64GB
 - access time: 80 – 250 (ns)
- solid-state disk
 - capacity: < 1TB
 - access time: 25,000 – 50,000 (ns)
- magnetic disk
 - capacity: < 10TB
 - access time: 5,000,000 (ns)

14. What does **PCB** stand for? What is it? What is its use? Which information is contained in a PCB?

- Process Control Block (**PCB**), per-process kernel data structure containing many pieces of information associated with a process. A repository that holds all the information needed to start, or restart a process, also contains some accounting data.

15. List the (conventional) names and types of three special, optional parameters of a C or C++ **main()** function. Explain types and plausible uses of these parameters.

- **argc**: first parameter, counts the number of command line arguments that argv[] is holding
- **argv**: second parameter, can be optional, is pointer for the list of constant string arguments
- **envp**: third parameter, can be optional, in UNIX/Linux holds environmental variables as array of string constants, plus the header (PATH=)