#1: How does “FCFS” process scheduling algorithm work?

#2: How does “RR” process scheduling algorithm work?

#3: How does “SJF” process scheduling algorithm work?

#4: How does “SRTF” process scheduling algorithm work?

#5: What is the “preemptive process scheduling”?

#6: What is the “non-preemptive process scheduling”?

#7: What is “throughput” (in the context of process scheduling)?

#8: What is “response time” (in the context of process scheduling)?

#9: What is “turnaround time” (in the context of process scheduling)?

#10: What is “process starvation”?

#11: Which process scheduling algorithms can cause “process starvation” (select all that apply)?

(1) FCFS

(2) RR

(3) SJF

(4) SRTF

#12: What is “race condition”?

#13: How can “race condition” happen? Show “how” using an example.

#14: What is “critical section”?

#15: What is “mutual exclusion”?
#16: What does “atomic” in “atomic operations” mean?

#17: What is “a binary (or mutex) semaphore”?

#18: What are the two primary system calls for manipulating a semaphore?

#19: Explain how a semaphore can prevent race condition.

#20: What are the two operations of a semaphore (just name them)?

#21: What “wait” system call to a semaphore exactly performs?

#22: What “signal” system call to a semaphore exactly performs?

#23: Why must the two system calls for semaphores (“wait” and “signal”) be atomic operations (explain the reason)?

#24: What are “counting semaphores” (how are “counting semaphores” different from “binary (mutex) semaphores”)?

#25: Who manages semaphores?

#26: Operating systems use “queue (FIFO data structure)” for managing processes blocked on a semaphore. Why is FIFO-queue used (the best reason for using FIFO structure)?