Please do not read the questions until your class starts, and do not share this document.

There will be no marks for answering these questions correctly. On the other hand, if we find that you knew the questions and/or the answers before your class you will receive zero participation marks for that tutorial.

In all questions below, you need to choose **one answer**.

Q1) What is the best Operating System in the world? :)

A. Linux  
B. Windows  
C. macOS  
D. Android  
E. There is no single best Operating System

Q2) A busy web server has many clients accessing it. The server is multi-threaded, so each request is serviced in a separate thread. Considering the servicing threads only, what type(s) of parallelism is this?

A. A combination of task and data parallelism  
B. Task parallelism  
C. Many-to-many parallelism  
D. Data parallelism  
E. Many-to-one parallelism

Q3) What resources are used when a thread is created? How do they differ from those used when a process is created?

A. Thread creation typically uses more resources than process creation  
B. Thread creation typically allocates a new process control block, containing a memory map, environment variables, CPU registers and more.  
C. Creating a thread involves allocating several program counters, several register sets, etc.  
D. Creating a thread involves allocating a small data structure to hold a register set, stack, and priority.  
E. Creating a new thread only requires the allocation of a program counter.

Q4) A CPU-scheduling algorithm determines an order for the execution of its scheduled processes. Given n processes to be scheduled on one processor nonpreemptively, how many different schedules are possible? Give a formula in terms of n.

A. n^2  
B. (n - 1)!  
C. n!  
D. n! / (n - 1)!  
E. n

Q5) What is the difference between preemptive and nonpreemptive scheduling?

A. Preemptive scheduling allows a process to be interrupted in the midst of its execution, taking the CPU away and allocating it to another process. Nonpreemptive scheduling ensures that a process relinquishes control of the CPU only when it finishes with its current CPU burst.  
B. Preemptive scheduling allows a process to be interrupted in the midst of its execution, taking the CPU away and allocating it to another process. Nonpreemptive scheduling ensures that a process relinquishes control of the CPU only when it terminates.  
C. Preemptive scheduling allows a process to be interrupted before execution has started, taking the CPU away and allocating it to another core. Nonpreemptive scheduling ensures that a process relinquishes control of the CPU only when when it terminates.  
D. Preemptive scheduling allows a process to be interrupted before execution has started, taking the CPU away and allocating it to another process. Nonpreemptive scheduling ensures that a process relinquishes control of the CPU only when it finishes with its current CPU burst.  
E. Preemptive scheduling allows a process to be interrupted in the midst of its execution, taking the CPU away and allocating it to another core. Nonpreemptive scheduling ensures that a process relinquishes control of the CPU only when it finishes with its current CPU burst.

Q6) Suppose that the following processes arrive for execution at the times indicated. Each process will run for the amount of time listed. Consider FCFS, SJF, SRTF and RR algorithms. Out of these, use a nonpreemptive algorithm in which decisions are based on the information you have at the time the decision must be made. What would be the average turnaround time using such an algorithm?

|  |  |  |
| --- | --- | --- |
| Process | Arrival time | Burst time |
| P1 | 0.0 | 8 |
| P2 | 0.4 | 4 |
| P3 | 1.0 | 1 |

A. 6.8  
B. 10.53  
C. 11  
D. 8.53  
E. 4.33