



VIT[®]

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engg
FALL Semester (2018-19), CAT-I
CSE2005 - Operating Systems

Slot : F1

Marks : 50 Dur. : 90 Min.

1. (a) Early microcomputer or personal computer operating systems did not require user authentication and security was non-existent. Explain why and then describe the changes that occurred in the use of these systems which required security to be taken seriously. [3]
 (b) Enlist the advantages of multiprogramming. Further, List five services provided by an operating system, and explain how each creates convenience for users. In which cases would it be impossible for user-level programs to provide these services? Explain your answer. [7]
2. (a) There are two major design types for operating system kernels: monolithic kernels and microkernels. Which of these types better satisfies the following requirements? If both are equally as good write both. Justify your answers? [5]
 - i. Convenient access to operating system data structures.
 - ii. Addition of new operating system components.
 - iii. Modification of operating system components.
 - iv. Security and reliability.
 (b) Analyse the need of system calls? [3]
 (c) CPU Needs a way to find out which devices need attention. Justify which of the following methods should be used: polling or Interrupts. [2]
3. (a) The following events may occur to a process. Identify the starting process state it is in at the time of the event and the ending process state it transitions to. [6]

Event	Starting State	Ending State
I/O complete		
Process admitted to ready queue for first time		
Scheduler preempts (interrupts) process		
Process finishes execution (task is complete)		
Process initiates I/O		

- (b) This C program runs on Unix. [2]

- i. How many times would this program print "One" to the display?
- ii. How many times would this program print "Two" to the display?

```
int main(int argc, char** argv) {
    int i;
    for (i = 0; i < 2; i++) {
        fork();
        printf("One\n");
    }
    printf("Two\n");
}
```

- (c) What are two differences between user-level threads and kernel-level threads? [2]

- (d) Here are the arrival and burst times for a number of processes: From this table draw a Gantt Chart and calculate Average Waiting Time, CPU utilization for First Come First Serve and Round Robin Scheduling (Time Quantum=3).

Process	Arrival Time	Burst Time
A	0	5
B	2	3
C	3	4
D	12	3
E	14	2

5. (a) Which of the following is / are shared by all threads in a process (A) Program Counter (B) Stack (C) Address Space (D) Registers
 (i) A & B only (ii) C only (iii) D only (iv) C & D only.
- (b) A process control block (PCB) exists only for processes in:
 i. the ready state.
 ii. ready and running states.
 iii. ready and blocked states.
 iv. ready, running, and blocked states.
- (c) A processor switches execution from user mode to privileged mode via (more than one answer may be correct):
 i. a software interrupt.
 ii. a programmable I/O.
 iii. a hardware interrupt.
 iv. memory-mapped I/O.
- (d) The most important CPU scheduling criteria to be optimized in an interactive operating system is:
 i. Response time. ii. Waiting time. iii. Throughput. iv. Turnaround time.
- (e) Multiprogramming (vs. multitasking) allows the operating system to:
 i. interrupt a process to run another process.
 ii. switch execution to another process when the first process blocks on I/O or makes a system call.
 iii. allow a single process to take advantage of multiple processors.
 iv. distribute multiple processes across multiple processors in a system.
- (f) Process aging:
 i. helps estimate the length of the next compute period.
 ii. increases the runtime of a process.
iii. is a count of the total CPU time used by the process and is stored in the PCB.
 iv. improves the chances of a process getting scheduled to run.
- (g) Round robin is _____ but has poor _____.
 i. unfair, latency ii. unfair, throughput iii. fair, latency iv. fair, throughput
- (h) When a process is created using the classical fork() system call, which of the following is not inherited by the child process?
 i. Process address space ii. Process ID iii. User ID iv. Open files
- (i) The threading model supported by the Linux operating system is:
 i. many-to-one
 ii. one-to-one (windows)
 iii. one-to-many
 iv. many-to-many
- (j) The strategy of making processes that are logically runnable to be temporarily suspended is called
 i. Non preemptive scheduling
ii. Preemptive scheduling
 iii. Shortest job first
 iv. First come First served