### **CS2210 – OPERATING SYSTEM TEST I –**

**SEMESTER: IV | TOTAL MARKS: 50 | DURATION: 2 HOURS**

### **PART A (7 x 2 = 14 Marks)**

**1. What is an operating system (OS)?**

An operating system is the **core software** that manages computer hardware and software resources, and provides services for computer programs. It acts as a **bridge between the user and hardware**.

* Examples: Windows, Linux, macOS

**2. Write one difference between process and thread?**

* **Process:** Independent execution unit with its own memory.
* **Thread:** Lightweight part of a process; shares memory with other threads.

**3. What is process management in an operating system?**

Refer to the process that involves **creation, scheduling, and termination** of processes. The OS allocates CPU time, memory, and I/O resources while ensuring concurrency and synchronization.

**4. What are distributed systems?**  
A distributed system is a network of independent computers that **work together as a single system**. Resources and processes are distributed but appear unified.

* Examples: Cloud computing, Hadoop clusters

**5. Define a virtual machine?**  
A virtual machine (VM) is an **simulated computer environment** that runs on a physical system. It provides the functionality of a physical computer but is isolated from the host machine.

* Examples: VMware, VirtualBox

**6. What is the system boot process?**  
Booting is the process of **starting up a computer and loading the operating system**. It includes:

1. Power-On Self-Test (POST)
2. Loading bootloader from BIOS
3. Bootloader loads OS kernel
4. OS initializes services and user interface

### **PART B (3 x 4 = 12 Marks)**

**8(a). Write the advantages and disadvantages of the Shortest Job First (SJF) scheduling algorithm. (** reference ya madam page 7 – unit 2 **)**

Shortest Job First refer to the process scheduling algorithm where by process with the shortest execution time is executed first.

**Advantages:**

* Minimizes average waiting time
* Highly efficient for batch processing

**Disadvantages:**

* Requires exact knowledge of job length
* Starvation of long jobs (not fair)
* Difficult to implement since it requires exact knowledge of length of next CPU burts.

**OR**

**8(b). Explain the types of thread scheduling.**

Thread scheduling is process done in OS to determine which thread to be runned next

1. **Preemptive Scheduling:** OS can interrupt a thread to assign CPU to another thread.
2. **Non-preemptive Scheduling:** Thread runs until it finishes or yields voluntarily.
3. **Priority Scheduling:** Threads with higher priority are scheduled first.
4. **Round Robin:** Threads are scheduled in time slices.

**9(a). What is the goal of a good scheduling algorithm?**

* Maximize CPU utilization
* Minimize waiting and turnaround time
* Ensure fairness among processes
* Reduce response time
* Meet system and user goals (deadlines, throughput)

**OR**

**9(b). What are the basic concepts of process scheduling?**

1. **Scheduling Queue:** Maintains all ready and waiting processes
2. **Dispatcher:** Component that assigns CPU
3. **Context Switching:** Switching CPU between processes
4. **Burst Time & Waiting Time:** Key scheduling metrics
5. **Throughput & Turnaround Time:** Efficiency indicators

Key Concepts in Process Scheduling:

• Context Switching: The act **of saving the state of a running process** and **loading the state of another process**. Frequent context switching can add overhead.

• CPU Bound vs I/O Bound:

o CPU-bound processes require heavy computation and consume a lot of CPU time.

o I/O-bound processes spend more time performing input/output operations than using the CPU.

• Preemptive vs Non-Preemptive Scheduling:

o Preemptive Scheduling: The OS can interrupt a running process to give CPU time to another.

o Non-preemptive Scheduling: The OS waits for a process to voluntarily release the CPU.

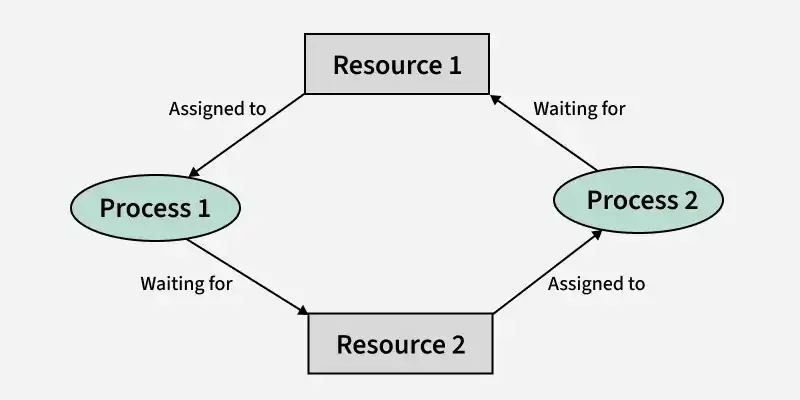
**10(a). What are the advantages of using threads?**

* Faster context switching than processes
* Shared memory within process = efficient communication
* Ideal for parallel processing and UI responsiveness

**OR**

**10(b). List the four necessary conditions for deadlock. (** reference ya mwalimu page 5 unit 2**)**

A deadlock is a situation where two or more processes are blocked indefinitely, each waiting for the other to release a resource that it needs to proceed.



1. **Mutual Exclusion** – One process at a time
2. **Hold and Wait** – Holding one resource while waiting for another
3. **No Preemption** – Resources can’t be forcibly taken
4. **Circular Wait** – Circular chain of processes each waiting on the next

### **PART C (2 x 12 = 24 Marks)**

**11(a). What are the challenges in multiple processor scheduling?**

Refer to the process or concept for managing systems with more than one processor or core.

* Load balancing among processors
* Processor affinity (binding process to CPU)
* Shared cache contention
* Synchronization between processors ( complexity of managing multiple processors )
* Deciding global vs local scheduling policies

Key Challenges in Multiple-Processor Scheduling

• Load Balancing: Ensuring that the workload is evenly distributed across all processors so that no processor is overburdened while others are underutilized.

• Processor Affinity: Some systems may prefer or require that certain processes or threads run on the same processor to take advantage of data locality (known as cache affinity).

• Synchronization: Managing processes and threads that need to communicate or synchronize, as multiple processors can lead to complexities in managing shared resources.

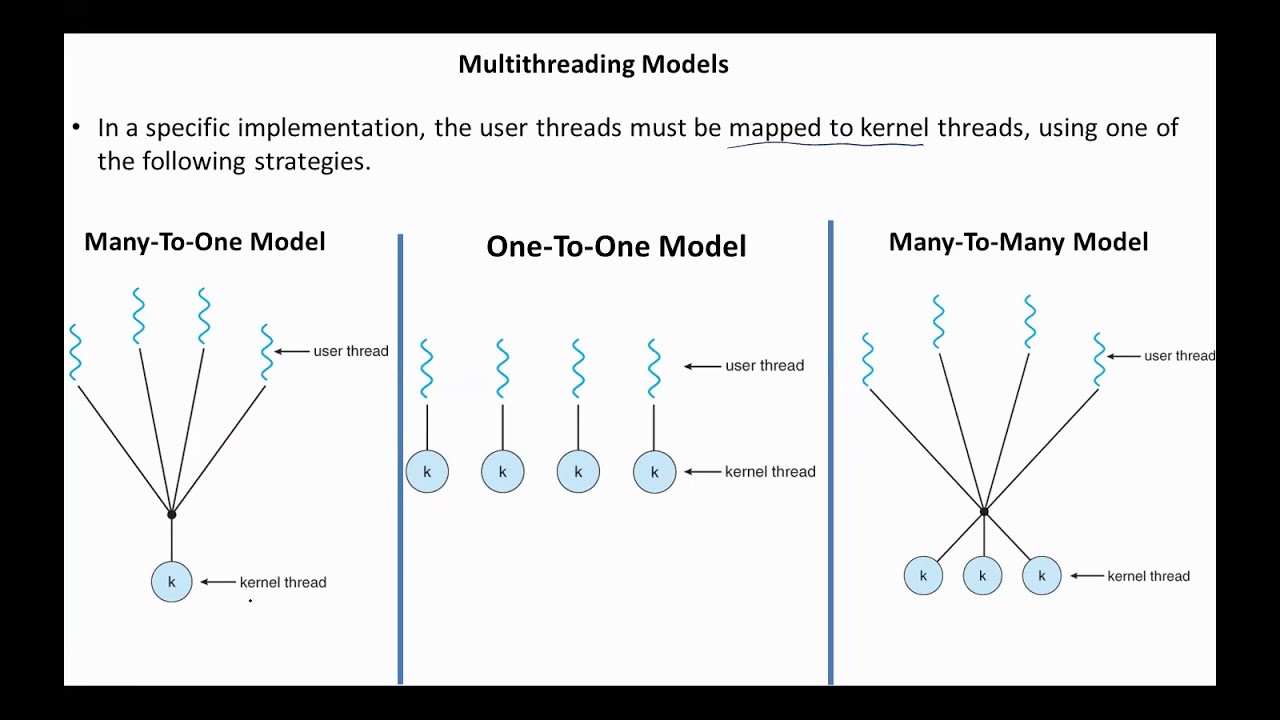
**OR**

**11(b). What are the common scheduling algorithms in operating systems? ( reference ya mwalimu page ya 7)**

1. **FCFS (First-Come-First-Served):** Simple, non-preemptive
2. **SJF (Shortest Job First):** Optimal but hard to predict job lengths
3. **Round Robin (RR):** Preemptive and fair
4. **Priority Scheduling:** Based on process priority
5. **Multilevel Queue Scheduling:** Different queues for different priority classes or Processes are grouped into multiple queues.

**12(a). What are the different multithreading models?**

**Multithreading models** describe how **user-level threads** are mapped to **kernel-level threads**. These models determine how efficiently the OS can manage concurrency, CPU utilization, and responsiveness.



1. **Many-to-One:** Many user threads mapped to one kernel thread (low concurrency)
2. **One-to-One:** Each user thread maps to a kernel thread (better concurrency)
3. **Many-to-Many:** User threads multiplexed to a set of kernel threads (flexible)
4. **Two-Level Model ( Hybrid Model ):** Combines many-to-many with one-to-one features

**OR**

**12(b). Describe the criteria used for process scheduling.**

The effectiveness of a scheduling algorithm can be judged based on several criteria:

1. **CPU Utilization**: Maximizing the use of the CPU.
2. **Throughput**: The number of processes completed per unit of time.
3. **Turnaround Time**: The total time taken for a process to complete (from arrival to completion).
4. **Waiting Time**: The total time a process spends waiting in the ready queue.
5. **Response Time**: The time taken from submitting a request to receiving the first response.
6. **Fairness:** Avoid starvation and ensure balanced use

### **CS2210 – OPERATING SYSTEM TEST II –**

**SEMESTER: IV | TOTAL MARKS: 50 | DURATION: 2 HOURS**

### **PART A (7 x 2 = 14 Marks)**

* 1. **What is the critical-section problem? ( reference ya Mwalimu, page 1 unit 3 )**

Refer to the issue in concurrent programming where multiple processes or threads attempt to access shared resources simultaneously.

The goal is to design a protocol so that **only one process executes in its critical section at a time**, ensuring:

* **Mutual exclusion**
* **Progress** (no unnecessary waiting)
* **Bounded waiting** (no process waits indefinitely)

**2. What are semaphores? (**Reference Mwalimu page 2 unit 3**)**  
Semaphores are **integer-based synchronization tools** used to control access to shared resources. Two main operations:

* wait() (P): Decreases semaphore; blocks if result < 0
* signal() (V): Increases semaphore and wakes waiting process

**Types:**

* **Binary Semaphore**: Acts as a simple lock (0 or 1)
* **Counting Semaphore**: Allows multiple processes

**3. What is demand paging?**

Demand paging is a memory management technique where **pages are loaded only when needed**.

**Paging** is a memory management scheme that eliminates the problems of contiguous memory allocation by dividing physical memory into fixed-size blocks called pages.

Advantages:

• Eliminates fragmentation issues.

• More flexible memory allocation.

• Processes can be allocated non-contiguous blocks of memory.

Disadvantages:

• Page table overhead:

• Page Faults:

**4. Page Replacement**

Page replacement refer to the process when memory is full, these algorithms decide which page to exit to load a new one.

Page replacement occurs when memory is full, and the OS must choose a page to remove to load a new one.

**Algorithms used:**

* FIFO (First-In-First-Out)
* LRU (Least Recently Used)
* Optimal (Theoretically best, but not implementable)

**5. What is mass storage in an operating system?**  
Mass storage refers to **non-volatile long-term storage devices** like hard disks, SSDs, and USB drives. The OS handles:

* File organization
* Block management
* Disk scheduling

**6. Name any two disk scheduling algorithms.**

Disk Scheduling algorithm refer to the process where by algorithm decide the order of I/O requests.

1. **FCFS (First-Come First-Served)**
2. **SCAN (Elevator Algorithm)**

Other common Algorithms:

• FCFS

• SSTF ( Shortest seek Time First )

• SCAN

• LOOK

**7. What is disk formatting?**  
Disk formatting refer to process to prepare a disk for use by the OS.

Steps:

1. **Low-Level Formatting:** Divides disk into sectors and tracks
2. **Partitioning:** Divides disk into logical parts
3. **High-Level Formatting:** Creates file system (e.g., NTFS, FAT32)

### **PART B (3 x 4 = 12 Marks)**

**8(a). Describe the basic structure of a file system.**  
A file system organizes data into files and directories.

Core components:

* **Boot Control Block:** Info about OS booting
* **Volume Control Block:** Details about size, file system type
* **Directory Structure:** Hierarchical list of files
* **File Control Block (FCB):** Metadata like file name, size, access rights

**OR**

**8(b). What are the main components of I/O hardware, and what are their functions?**

Devices and controllers that manage input/output operations.

1. **I/O Devices:** Hardware like keyboard, printer, disk
2. **Device Controller:** Manages I/O devices
3. **Device Driver:** Software bridge between OS and controller
4. **Buffers & Registers:** Store data during transmission
5. **Interrupts:** Signal CPU upon completion of tasks

**9(a). What are the different types of failures that require recovery?**

1. **System Crashes:** Power failure or OS malfunction
2. **Disk Failures:** Bad sectors, disk corruption
3. **Transaction Failures:** Invalid or incomplete transactions
4. **Software Errors:** Bugs causing data loss or inconsistency

**OR**

**9(b). Explain how I/O requests are transformed into hardware-level operations.**

1. **System call issued** by user-level program
2. **I/O scheduler** prioritizes requests
3. **Device driver** translates request
4. **Device controller** executes command
5. **Interrupt sent** when task completes

The OS manages all translations from high-level commands to low-level device signals

Example:

The OS converts an application’s request (e.g., read a file) into a series of steps: identifying file blocks, locating them on disk, sending commands to the device controller, and returning results. This abstraction allows software to use I/O efficiently without knowing hardware details.

**10(a). Explain the concept of demand paging and its advantages.**  
Demand paging loads memory pages only when needed. Pages not currently in memory cause a **page fault**, and the OS loads them from disk.

**Advantages:**

* Efficient use of memory
* Faster program loading
* Enables execution of large programs

**OR**

**10(b). What is copy-on-write? Describe its benefits.**

Copy-on-write (COW) is a technique where **multiple processes share the same memory pages** until one modifies them. Only then is a copy created.

**Benefits:**

* Saves memory
* Improves performance in fork() operations
* Efficient memory duplication in virtual memory systems

### **PART C (2 x 12 = 24 Marks)**

**11(a). Explain demand paging and its advantages and disadvantages?**  
**Advantages:**

* Reduces memory usage
* Allows more programs to run simultaneously
* Enables execution of large programs

**Disadvantages:**

* Page faults cause delays
* Thrashing can occur with excessive page faults
* Requires additional OS overhead (page table, fault handler)

**OR**

**11(b). Describe in detail the structure of I/O systems in an operating system.**  
I/O system consists of:

* **User-level I/O calls:** e.g., read(), write()
* **Kernel I/O subsystem:** Handles buffering, caching, spooling
* **Device Drivers:** Convert generic calls into hardware instructions
* **Interrupt Handlers:** Notify CPU of I/O completion
* **DMA (Direct Memory Access):** Moves data directly without CPU

Ensures efficient and safe data transfer between devices and memory

**12(a). Explain about semaphores?**  
Semaphores are variables used for **synchronization and mutual exclusion**.

**Operations:**

* wait(S) – Decrement and possibly block
* signal(S) – Increment and possibly wake up a blocked process

**Use cases:**

* Prevent race conditions
* Solve producer-consumer problem
* Implement critical section protection

**OR**

**12(b). Describe memory-mapped files and what are their uses?**  
A memory-mapped file maps a file directly into a process’s address space.

**Advantages and Uses:**

* Faster I/O than standard file handling
* Enables file sharing between processes
* Used in databases, multimedia applications
* Reduces number of system calls