OPERATING SYSTEM

IMP

Priority 1: Highly Important (Must Master)

These topics appear in almost every paper and often involve both theoretical explanations and numerical problems.

1. **CPU Scheduling Algorithms (Numerical & Theoretical)**
    - **Frequency:** Appeared in all 5 papers.
    - **Details:** FCFS, SJF (preemptive and non-preemptive), Priority (preemptive and non-preemptive), Round Robin (RR). Calculation of Average Waiting Time and Average Turnaround Time is consistently asked.
    - **Action:** Practice a wide range of problems for all these algorithms. Understand how to draw Gantt charts accurately.
2. **Deadlock (Banker's Algorithm - Numerical & Theoretical)**
    - **Frequency:** Appeared in 4 out of 5 papers.
    - **Details:** Banker's Algorithm (determining safe state, safe sequence, need matrix, resource requests). Also, necessary conditions for deadlock.
    - **Action:** Thoroughly understand and practice Banker's algorithm numerical problems. Be able to explain the conditions for deadlock.
3. **Page Replacement Algorithms (Numerical)**
    - **Frequency:** Appeared in 4 out of 5 papers.
    - **Details:** FIFO, LRU, Optimal page replacement algorithms. Determining the number of page faults.
    - **Action:** Practice many problems for these algorithms. Understand their working and be able to compare their efficiency.
4. **Operating System Services / Functions / Activities**
    - **Frequency:** Appeared in 4 out of 5 papers.
    - **Details:** Explaining services with respect to programs and users , listing and describing different services , major activities in Process Management, File Management, Main Memory Management, Secondary Storage Management.
    - **Action:** Prepare comprehensive notes on the various functions and services of an OS.

Priority 2: Very Important (Should Focus On)

These topics appear frequently and are important for a good score, often involving explanations or specific problem types.

1. **Disk Scheduling Algorithms (Numerical & Theoretical)**
    - **Frequency:** Appeared in 3 out of 5 papers.
    - **Details:** FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK. Calculating total head movement. Advantages of each.
    - **Action:** Understand the working of each algorithm and practice numerical problems to calculate total distance.
2. **File Allocation Methods**
    - **Frequency:** Appeared in 3 out of 5 papers.
    - **Details:** Contiguous, Linked, Indexed allocation methods. Explanation with diagrams, merits and demerits.
    - **Action:** Be able to explain and illustrate these methods clearly. There was also a numerical problem related to disk accesses for file insertion.
3. **Critical Section Problem & Solutions**
    - **Frequency:** Appeared in 3 out of 5 papers.
    - **Details:** Peterson's Solution , requirements for a solution. Semaphore-based solutions (e.g., Readers-Writers problem).
    - **Action:** Understand the problem, its requirements, and be able to explain Peterson's solution and semaphore usage for common synchronization problems.
4. **Paging (Mechanism & Concepts)**
    - **Frequency:** Appeared in 3 out of 5 papers.
    - **Details:** Demand Paging , handling page fault with diagrams , TLB's role , logical vs. physical address bits calculation.
    - **Action:** Understand the paging mechanism thoroughly, including how page faults are handled and the function of TLB. Practice logical/physical address bit calculations.
5. **Process Control Block (PCB)**
    - **Frequency:** Appeared in 2 out of 5 papers.
    - **Details:** Describing the contents of PCB.
    - **Action:** Know the structure and information stored in a PCB.
6. **Threads (User-level vs. Kernel-level)**
    - **Frequency:** Appeared in 2 out of 5 papers.

- **Details:** Definition of thread, differentiation between user-level and kernel-level threads. Context switching between kernel-level threads.
- **Action:** Understand the concept of threads and the distinctions between their types.

Priority 3: Moderately Important (Review and Understand)

These topics appear occasionally but are still part of the core syllabus.

1. **Types of Operating Systems:**
    - **Frequency:** Real-time OS appeared once. Microkernel OS appeared once. Virtual Machine appeared twice.
    - **Details:** Real-time OS (types, advantages, examples) , Microkernel structure , Virtual Machines (definition, concept, benefits, structure).
    - **Action:** Have a basic understanding of these different OS structures and their characteristics.
2. **System Calls:**
    - **Frequency:** Appeared once.
    - **Details:** Explanation of system calls and their categories with examples.
    - **Action:** Understand what system calls are and their role in OS.
3. **Inter-process Communication (IPC)**
    - **Frequency:** Appeared twice.
    - **Details:** Explanation of IPC and its types. Shared memory and message passing mechanisms.
    - **Action:** Understand the different mechanisms for processes to communicate.
4. **Schedulers (Long, Short, Medium Term)**
    - **Frequency:** Appeared once.
    - **Details:** Role of different types of schedulers.
    - **Action:** Know the function of each type of scheduler.
5. **Memory Management Concepts (Fragmentation, Working Set Model)**
    - **Frequency:** Fragmentation appeared once. Working set model appeared once. Memory layout of multiprogramming OS appeared once.
    - **Action:** Be familiar with these concepts, even if they aren't numerical.
6. **File Concepts and Operations:**
    - **Frequency:** Appeared once.
    - **Details:** Concept of a file, various file operations. File types.

- **Action:** Understand basic file system concepts.

7. **Free Space Management:**
    - **Frequency:** Linked allocation type disk free space management appeared once. Note on free space management appeared once. Bit map explanation asked once.
    - **Action:** Understand methods like linked allocation and bitmap for managing free disk space.
8. **Storage System Hierarchy:**
    - **Frequency:** Appeared once.
    - **Details:** Explanation with neat diagram.
    - **Action:** Be able to explain and draw the storage hierarchy.

Priority 4: Least Frequently Asked (Review if Time Permits)

These topics appeared only once or were very specific.

1. **Command Interpreter vs. Kernel:**
2. **Multiprocessor System Advantages:**
3. **Dining Philosopher's Problem:**
4. **Inodes and Maximum File Size Calculation:** (This is a specific numerical problem, could reappear, but less frequent than other numerical types).
5. **Boot-control block and Volume-control block terminology:**
6. **Resource Allocation Graph (RAG) in Deadlock Detection:**
7. **Swap instruction pseudocode:**