# Operating System

| SOURCE: 01 | Operating System (GATE EXAM) |
|---|---|
| 01 | Operating System Syllabus |
| 02 | Introduction to Operating System and Its Functions |
| 03 | Batch Operating System \| Types of Operating System |
| 04 | Multiprogramming and Multitasking Operating System |
| 05 | Types of OS (Real Time, Distributed, Clustered and Embedded) |
| 06 | Process States in Operating System |
| 07 | Important Linux Commands |
| 08 | System Calls in Operating System and Its Types |
| 09 | Fork System Call with Example |
| 10 | Fork System Call with Explanation |
| 11 | User Mode and Kernel Mode in Operating System |
| 12 | Process vs Threads in Operating System |
| 13 | User Level vs Kernle Level Thread in Operating System |
| 14 | Process Scheduling Algorithms (Preemption vs Non-Preemption) \| CPU Scheduling |
| 15 | What is Arrival, Burst, Completion, Turnaround, Waiting and Response Time in CPU |
| 16 | First Come First Serve (FCFS) CPU Scheduling Algorithm with Example |
| 17 | Shortest Job First (SJF) Scheduling Algorithm with Example |
| 18 | Shortest Remaining Time First (SJF with Preemption) Scheduling Algorithm |
| 19 | Shortest Job First (SJF with Preemption) Scheduling Algorithm |
| 20 | Round Robin (RR) CPU Scheduling Algorithm with Example |
| 21 | Pre-emptive Priority Scheduling Algorithm with Example |
| 22 | Example of Mix Burst Time (CPU and I/O both) in CPU Scheduling |
| 23 | Multi-Level Queue Scheduling |
| 24 | Multilevel Feedback Queue Scheduling |
| 25 | Process Synchronization Process Types \| Race Condition |
| 26 | Producer Consumer Problem \| Process Synchronization Problem |
| 27 | Printer-Spooler Problem \| Process Synchronization Problem |
| 28 | Critical Section Problem \| Mutual Exclusion, Progress and Bounded Waiting |
| 29 | LOCK Variable in OS \| Process Synchronization |
| 30 | Test and Set Instruction in OS \| Process Synchronization |
| 31 | Tum Variable \| Strict Alteration Method \| Process Synchronization |
| 32 | Semaphores \| Wait, Signal Operation \| Counting Semaphore Examples |
| 33 | What is Binary Semaphore \| Easiest Explanation |
| 34 | Practice Question on Binary Semaphore |
| 35 | Solution of Producer Consumer Problem Using Semaphore |
| 36 | Solution of Readers-Writers Problem Using Binary Semaphore |
| 37 | Dining Philosophers Problem and Solution Using Semaphore |
| 38 | DEADLOCK Concept Example \| Necessary Condition |
| 39 | Resource Allocation Graph in Deadlock \| Single Instance with Example |
| 40 | Multi-Instance Resource Allocation Graph with Example |
| 41 | Deadlock Handling Methods and Deadlock Prevention |
| 42 | Deadlock Avoidance Banker's Algorithm with Example |
| 43 | GATE Question on Banker's Algorithm \| Deadlock Avoidance |
| 44 | Question Explanation on Deadlock |
| 45 | GATE Question Explanation |

| | |
|---|---|
| 46 | Memory Management and Degree of Multiprogramming |
| 47 | Memory Management Techniques \| Contiguous and Non-Contiguous |
| 48 | Internal Fragmentation \| Fixed Size Partitioning \| Memory Management |
| 49 | Variable Size Partitioning \| Memory Management |
| 50 | First Fit, Next Fit, Best Fit, Worst Fit Memory Allocation |
| 51 | GATE Question Solved on First Fit, Best Fit and Worst Fit Memory Allocation |
| 52 | GATE Question Solved on First Fit, Best Fit and Worst Fit Memory Allocation with Timeline |
| 53 | Need of Paging \| Memory Management |
| 54 | What is Paging \| Memory Management |
| 55 | Question Explanation on Logical Address and Physical Address Space |
| 56 | Question Explanation on Paging \| Memory Management |
| 57 | Page Table Entries Format of Page Table |
| 58 | 2-Level Paging in Operating System \| Multilevel Paging |
| 59 | Inverted Paging \| Memory Management |
| 60 | Questions Paging in Operating System |
| 61 | What is Thrashing |
| 62 | Segmentation vs Paging \| Segmentation Working |
| 63 | Overlay \| Memory Management |
| 64 | Virtual Memory \| Page Fault \| Significance of Virtual Memory |
| 65 | Translation Lookaside Buffer (TLB) in Operating System |
| 66 | Numerical On Translation Lookaside Buffer (TLB) |
| 67 | Page Replacement Introduction \| FIFO Page Replacement Algorithm |
| 68 | Belady's Anomaly in FIFO Page Replacement with Example |
| 69 | Optimal Page Replacement Algorithm |
| 70 | Least Recently Used Page Replacement Algorithm |
| 71 | Most Recently Used Page Replacement Algorithm |
| 72 | Hard Disk Architecture in Operating System |
| 73 | Disk Access Time with Example \| Seek Time Rotational Time and Transfer Time |
| 74 | Disk Scheduling Algorithm |
| 75 | FCFS in Disk Scheduling with Example |
| 76 | SSTF in Disk Scheduling with Example |
| 77 | SCAN Algorithm in Disk Scheduling with Example |
| 78 | LOOK Algorithm in Disk Scheduling with Example |
| 79 | C-SCAN Algorithm in Disk Scheduling with Example |
| 80 | C-LOOK Algorithm in Disk Scheduling with Example |
| 81 | Question On Operating system |
| 82 | File System in Operating system \| Windows, Linux, Unix, Android, Etc |
| 83 | File Attributes and Operations in Operating System |
| 84 | Allocation Methods in Operating System \| Contiguous and Non-Contiguous |
| 85 | Contiguous Allocation in Operating System \| Advantages and Disadvantages |
| 86 | Linked List Allocation in File Allocation whit Example |
| 87 | Indexed File Allocation in Operating System |
| 88 | Unix Inode Structure with Numerical Example |
| 89 | Protection and Security in Operating System |
| 90 | Linker and Loader with Example |

| System Programming | |
|---|---|
| **SOURCE: 02** | **Operating System Lab** |
| 01 | Operating System Lab Introduction |
| 02 | Linux Installation |
| 03 | Read-Write System Call Program in Linux |
| 04 | Open() System Call Program in Linux |
| 05 | Iseek System Call Program in Linux |
| 06 | Dup System Call Program in Linux |
| 07 | How to Create Child Process Using fork() \| Duplicate Process |
| 08 | Wait System Call Program in C |
| 09 | Orphan Process Program in Linux |
| 10 | Zombie Process Program in Linux |
| 11 | How to Replace Process Image in Linux \| execl |
| 12 | Program to Create Threads in Linux \| pthread_create() |
| 13 | Program to Pass Parameters to a Thread |
| 14 | Cracking the Race Condition Program with C \| Undercover the Threads |
| 15 | Semaphore Program in C \| Process Synchronization |
| 16 | Mutex Locks Program to Avoid Race Condition \| Process Synchronization |
| 17 | Program for Inter-Process Communication Using popen \| pclose |
| 18 | Program for Inter-Process Communication Using pipe() function |
| 19 | Program for Inter-Process Communication Using Named pipes \| mkfifo |
| 20 | Program for Inter-Process Communication Using Shared Memory |
| 21 | Dining Philosopher Problem Program in C |
| 22 | Program for Inter-Process Communication Using Message Queues \| msgget \| msgsnd |
| 23 | Deadlock in OS \| Program to Simulate Deadlock \| C Program |
| 24 | System Calls Read, Write and Open Solved Programs \| OS Lab \| Operating System |
| 25 | System call Open-Iseek Solved Programs \| Operating System \| OS Lab |
| 26 | Fork System Call Programs \| Solved Programs \| fork |
| 27 | System Calls Viva Questions \| Read \| Write \| Open System Call \| OS |
| 28 | Fork System Call Programs \| Wait \| Solved Programs |
| 29 | Process Creation Viva Questions \| System Calls \| fork \| wait \| OS |
| 30 | Thread Creation Solved Programs \| Operating System |
| 31 | Semaphore Practice Programs in C \| Process Synchronization \| Operating System |
| 32 | Inter-Process Communication IPC Practice Programs \| Operating System |
| 33 | No Manual Entry for Man 2 Write \| Solved |
| 34 | RHCSA Success Stories \| Dextutor |
| 35 | Chown System Call Program in Linux |