INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)
Dundigal, Hyderabad - 500 043

COMPUTER SCIENCE AND ENGINEERING

TUTORIAL QUESTION BANK

Course Name	**OPERATING SYSTEMS**
Course Code	ACS007
Class	IV Semester
Branch	Computer Science and Engineering
Year	2018 – 2019
Course Faculty	Dr. Chukka Santhaiah, Professor Mr. N V Krishna Rao, Associate Professor

COURSE OBJECTIVES (COs):
The course should enable the students to:

I.	Understand the fundamental principles of the operating system, its services and functionalities.
II.	Illustrate the concepts of processes, inter-process communication, synchronization and scheduling.
III.	Understand different types of memory management viz. virtual memory, paging and segmentation.
IV.	Identify the reasons for deadlock and understand the techniques for deadlock detection, prevention and recovery.
V.	Understand the need of protection and security mechanisms in computer systems.

COURSE LEARNING OUTCOMES (CLOs):
Students, who complete the course, will have demonstrated the asking to do the following:

CACS007.01	Describe the structure of operating system and basic architectural components involved in operating system design.
CACS007.02	Describe how the computing resources are managed by the operating system.
CACS007.03	Understand the objectives and functions of modern operating systems.
CACS007.04	Analyze and design the applications to run in parallel either using process or thread models of different operating system
CACS007.05	Understand and analyze implementation of virtual memory
CACS007.06	Understand the various resource management techniques for timesharing and distributed systems.
CACS007.07	Describe the mutual exclusion, deadlock detection in operating system
CACS007.08	Describe the common algorithms used for both pre-emptive and non-pre-emptive scheduling of tasks in operating systems, such a priority and performance comparison
CACS007.09	Understand the difference between a process and a thread
CACS007.10	Explain the state diagram that describes the states and state transitions during the whole lifetime of a process; likewise, interpret such a state transition diagram
CACS007.11	Identify the mapping between virtual memory address into a physical address
CACS007.12	Explain how a shared memory area can be implemented using virtual memory addresses in different processes
CACS007.13	Identify the need of memory management in operating systems and understand the limits of fixed memory allocation schemes
CACS007.14	Understand the fragmentation in dynamic memory allocation, and identify dynamic allocation approaches
CACS007.15	Understand how program memory addresses relate to physical memory addresses, memory management in base-limit machines, and swapping
CACS007.16	Understand the mechanisms adopted for file distribution in applications

CACS007.17	Describe different Mass storage structure and I/O systems
CACS007.18	Understand issues related to file system interface and implementation, disk management
CACS007.19	Identify the mechanisms adopted for file sharing in distributed applications
CACS007.20	Understand the concepts of Storage Management, disk management and disk scheduling
CACS007.21	Understand the concept of deadlock in operating systems and how they can be implemented in multiprogramming system
CACS007.22	Identify how deadlock can occur and know how it can be prevented or avoided
CACS007.23	Describe the protection and security aspects of operating systems
CACS007.24	Understand types of security risks in operating system and the role of operating system in establishing security
CACS007.25	Identify different protection and security mechanisms in operating system

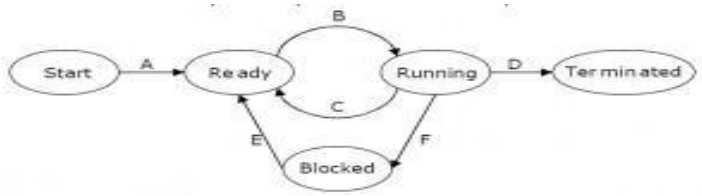
TUTORIAL QUESTION BANK

UNIT – I			
INTRODUCTION			
PART - A (Short Answer Questions)			
S. No.	**Question**	**Blooms Taxonomy Level**	**Course Outcomes**
1	Define distributed systems.	Remember	CACS007.02
2	Distinguish between user mode and kernel mode operations of the operating system.	Understand	CACS007.02
3	Define kernel.	Remember	CACS007.01
4	Describe the use of fork () and exec () system calls.	Remember	CACS007.02
5	List any four types of system calls.	Understand	CACS007.01
6	Define multiprocessor system.	Understand	CACS007.03
7	List the advantages of multiprogramming.	Understand	CACS007.02
8	Distinguish between multiprogramming and multitasking.	Remember	CACS007.02
9	Define interrupt.	Remember	CACS007.02
10	Define virtual machine.	Understand	CACS007.01
11	Define real-time operating system.	Understand	CACS007.02
12	Define operating system.	Understand	CACS007.02
13	List the memory hierarchy available in operating system.	Remember	CACS007.02
14	Define privileged instructions.	Remember	CACS007.01
15	Describe the different types of multiprocessing.	Remember	CACS007.02
16	Describe the different types of multiprocessor systems.	Understand	CACS007.02
17	List any four functions of operating system.	Understand	CACS007.03
18	Define time-sharing systems.	Understand	CACS007.02
19	Define system call.	Remember	CACS007.02
20	State the challenges in designing a distributed operating system.	Understand	CACS007.01
21	State the differences between system call and system program.	Understand	CACS007.02
22	State the five major activities of an operating system in regard to process management.	Understand	CACS007.01
23	State the main advantage of the layered approach to system design. What are the disadvantages of using the layered approac.	Remember	CACS007.01
24	List the contemporary operating systems that use the microkernel approach.	Remember	CACS007.01
25	List the various OS components.	Understand	CACS007.01
26	Discuss batch systems.	Remember	CACS007.02
PART-B (Long Answer Questions)			
1	State and explain various types of computer systems.	Understand	CACS007.01
2	a) Define an operating system. State and explain the basic functions or services of an operating system. b) Explain the differences between multiprogramming and time-sharing systems.	Understand	CACS007.02

3	Explain how protection is provided for the hardware resources by the operating system.	Understand	CACS007.01
4	Describe the system components of an operating system and explain them briefly.	Understand	CACS007.01
5	Describe the operating system structures.	Remember	CACS007.01
6	Define the essential properties of the operating systems.	Remember	CACS007.03
7	Explain briefly system calls with examples.	Understand	CACS007.02
8	Discuss the kernel structures of OS.	Understand	CACS007.01
9	a) Explain the architecture of an operating system. b) Draw and explain the architecture of windows 2000 and traditional UNIX.	Understand	CACS007.01
10	Explain how operating system services are provided by system calls.	Understand	CACS007.02
11	Does an operating system generally need to keep about running processes in order to execute them. Explain in detail.	Remember	CACS007.01
12	Discuss the view of an operating system as a resource manager.	Understand	CACS007.03
13	Distinguish between multiprogramming, multitasking and multiprocessing.	Understand	CACS007.02
14	Computer system architecture deals about how the component of a computer system may be organized. Discuss in detail about different architectures of a computer system.	Understand	CACS007.01
15	Describe the functionalities listed below. a) Batch programming b) Virtual Memory c) Timesharing	Remember	CACS007.02
16	Distinguish between the client-server and peer-to-peer models of distributed systems.	Understand	CACS007.02
PART-C (Problem Solving and Critical Thinking)			
1	How does the distinction between kernel mode and user mode function as a rudimentary form of protection (security) system. Justify.	Remember	CACS007.01
2	Explain using a simple system call as an example (e.g. getpid, or uptime), what is generally involved in providing the result, from the point of calling the function in the C library to the point where that function returns.	Understand	CACS007.02
3	In a multiprogramming and time-sharing environment, several users share the system simultaneously. This situation can result in various security problems. Explain two such problems. Can we ensure the same degree of security in a time-shared machine as we have in a dedicated machine. Explain your answer.	Remember	CACS007.02
4	Explain why must the operating system be more careful when accessing input to a system call (or producing the result) when the data is in memory instead of registers.	Understand	CACS007.02
5	Discuss how a multi-threaded application can be supported by a user- level threads package. It may be helpful to consider (and draw) the components of such a package, and the function they perform.	Understand	CACS007.01
6	Explain why do you think that idleness in CPU occurs.	Understand	CACS007.01
7	Explain Is OS is a resource manager. If so justify your answer	Knowledge	CACS007.03
8	Explain the difference between interrupt and exception.	Understand	CACS007.01
9	Differentiate between tightly coupled systems and loosely coupled systems.	Remember	CACS007.02
10	Explain If you run the same program twice, what section would be shared in the memory.	Remember	CACS007.01
UNIT – II			
Process and CPU Scheduling, Process Coordination			
PART - A (Short Answer Questions)			
S. No.	**Question**	**Blooms Taxonomy Level**	**Course Outcomes**
1	Define process. What is the information maintained in a PCB.	Understand	CACS007.10

2	Define process state and mention the various states of a process.	Remember	CACS007.10
3	Describe context switching.	Remember	CACS007.10
4	Distinguish between user threads and kernel threads.	Understand	CACS007.09
5	Distinguish between thread with process.	Understand	CACS007.09
6	Explain benefits of multithreaded programming.	Understand	CACS007.09
7	Distinguish between preemptive and non-preemptive scheduling techniques.	Understand	CACS007.08
8	State critical section problem.	Understand	CACS007.07
9	Define CPU scheduling.	Remember	CACS007.08
10	List the various scheduling criteria for CPU scheduling.	Understand	CACS007.08
11	State the assumption behind the bounded buffer producer consumer problem.	Remember	CACS007.07
12	Define turnaround time.	Remember	CACS007.08
13	List different types of scheduling algorithms.	Understand	CACS007.08
14	Explain different ways in which a thread can be cancelled.	Understand	CACS007.09
15	State the requirements that a solution to the critical section problem must satisfy.	Remember	CACS007.07
16	Define race condition.	Remember	CACS007.07
17	Define semaphores. Mention its importance in operating system.	Understand	CACS007.07
18	Explain the use of job queues, ready queues and device queues.	Understand	CACS007.10
19	Explain bounded waiting in critical region.	Understand	CACS007.07
20	Distinguish between semaphore and binary semaphore.	Understand	CACS007.07
21	State the factors on which the performance of the Round Robin CPU scheduling algorithm depends.	Understand	CACS007.08
22	Describe entry and exit sections of a critical section.	Understand	CACS007.07
23	State the real difficulty with the implementation of the SJF CPU scheduling algorithm.	Remember	CACS007.08
24	Define monitor.	Remember	CACS007.07
25	Name the algorithms used for foreground and background queue scheduling in a multilevel queue-scheduling algorithm.	Understand	CACS007.08
26	State two hardware instructions and their definitions which can be used for implementing mutual exclusion.	Understand	CACS007.07
PART-B (Long Answer Questions)			
1	Explain the reasons for process termination.	Understand	CACS007.10
2	Discuss the following process, program, process state, process control block, and process scheduling.	Understand	CACS007.10
3	Explain the infinite buffer producer/consumer problem for concurrent processing which uses binary semaphores.	Understand	CACS007.07
4	Discuss the attributes of the process. Describe the typical elements of process control block.	Understand	CACS007.10
5	Explain the principles of concurrency and the execution of concurrent processes with a simple example.	Remember	CACS007.10
6	Describe dining-philosophers problem. Device an algorithm to solve the problem using semaphores.	Understand	CACS007.07
7	Explain the Readers and Writers problem and its solution using the concept of semaphores.	Understand	CACS007.07
8	Define monitor. Distinguish between monitor and semaphore. Explain in detail a monitor with notify and broadcast functions using an example.	Remember	CACS007.07
9	List out the various process states and briefly explain the same with a state diagram.	Understand	CACS007.10
10	a) Describe process scheduling. Explain the various levels of scheduling. b) Distinguish pre-emptive and non-pre-emptive scheduling algorithms.	Understand	CACS007.08
11	Discuss about following. a) Process b) Components of process c) Program versus process d) Process states	Understand	CACS007.10

12	Discuss the following. a) CPU-I/O burst cycle b) CPU schedule c) Pre-emptive and non-preemptive scheduling d) Dispatcher	Understand	CACS007.08
13	Explain the concept of multi-threading. Discuss the following multi-threading models. a) Many-to-one b) One-to-one c) Many-to-many d) Two-level	Understand	CACS007.09
14	Explain the issues that may rise in multi-threading programming. Discuss about each in detail.	Understand	CACS007.09
15	Discuss the following CPU scheduling algorithms a) Round robin b) Multilevel- queue scheduling c) Multi-level feedback queue scheduling	Understand	CACS007.08
16	A scheduling mechanism should consider various scheduling criteria to realize the scheduling objectives. List out all the criteria.	Remember	CACS007.08
17	Define semaphore. Explain the method of application of semaphore for process synchronization.	Understand	CACS007.07
18	Explain the process state transition diagram with examples.	Understand	CACS007.10
19	Explain the uses of the following: a. Mutex object b. Semaphore object c. Waitable timer object	Understand	CACS007.07
20	Write short notes about the following: a. Binary Semaphores b. Bounded Waiting	Remember	CACS007.07
	PART-C (Problem Solving and Critical Thinking)		
1	Suppose we have a single processor system, and jobs arrive at a rate of 10 jobs a Seconds, suppose each job takes an average of 50 milli-seconds to complete. Assure that both distributions are exponential. **State** the expected number of jobs in the system and the average time in the system.	Remember	CACS007.10
2	Suppose the following jobs arrive for processing at the times indicated, each job will run the listed amount of time. Jobs Arrival Time Burst Time (in secs) 1 0.0 8 2 0.4 4 3 1.0 1 Give Gantt chart illustrating the execution of these jobs using the non- pre-emptive FCFS and SJF scheduling algorithms. **Compute** the average turnaround time and average waiting time of each job for above algorithms.	Remember	CACS007.08
3	Consider system with five processor P0 to P4 and 3 resources A, B and C, Resources type A has 10 instances, B has 5 instances and C has 7 instances. The snapshot at time T0 is ALLOTED MAX A B C A B C P0 0 1 0 7 5 3 P1 2 0 0 3 2 2 P2 3 0 2 9 0 2 P3 2 1 1 2 2 2 P4 0 0 2 4 3 3 Now the process P1 request one additional resource type A and two instances of C. Determine whether this new site is safe or not.	Remember	CACS007.08

4	Explain the advantage of using semaphores over Test And Set () and Swap () functions. Describe the use of wait () and signal () functions on semaphore and how these can provide the solution to the Critical section problem.	Understand	CACS007.07
5	Consider the following set of processes with the length of the CPU burst time given in milliseconds Process Burst Time Priority P1 10 3 P2 1 1 P3 2 3 P4 1 4 P5 5 2 The processes are assumed to have arrived in the order p1, p2, p3, p4, p5 all at time 0. a) Draw four Gantt charts illustrating the execution of these processes using FCFS, SJF, anon pre-emptive priority (a smaller priority number implies a higher priority) and RR (quantum=1) scheduling. b) What is the turnaround time of each process for each of the scheduling algorithms in part. c) What is the waiting time of each process for each of the scheduling algorithms in part. Which of the schedules in part a results in the minimal average waiting time.	Remember	CACS007.08
6	Explain Which scheduling algorithm allocates the CPU first to the process that requests the CPU first.	Understand	CACS007.08
7	Consider three CPU-intensive processes, which require 10, 20 and 30 time units and arrive at times 0, 2 and 6, respectively. How many context switches are needed if the operating system implements a shortest remaining time first scheduling algorithm. Do not count the context switches at time zero and at the end	Remember	CACS007.08
8	Explain the following process state transition diagram for a uniprocessor system, assume that there are always some processes in the ready state 	Understand	CACS007.10
9	Explain Four jobs to be executed on a single processor system arrive at time0intheorderA,B,C,D.their burst CPU time requirements are 4,1, 8, 1 time units respectively. The completion time of A under round robin scheduling with time slice of one time unit is.	Remember	CACS007.08

UNIT – III

Memory Management and Virtual Memory

PART - A (Short Answer Questions)

S. No.	Question	Blooms Taxonomy Level	Course Outcomes
1	Explain the main function of the memory-management unit.	Understand	CACS007.13
2	Distinguish between logical address and physical address.	Understand	CACS007.11
3	Describe dynamic loading and dynamic linking.	Remember	CACS007.14
4	Distinguish between compile time, load time and execution time address binding.	Understand	CACS007.13
5	Define swapping.	Remember	CACS007.15
6	List dynamic storage allocation strategies in contiguous memory allocation scheme.	Remember	CACS007.13
7	Distinguish between MFT and MVT.	Understand	CACS007.13

8	Distinguish between internal and external fragmentation.	Understand	CACS007.15
9	Define compaction.	Remember	CACS007.15
10	List and define non-contiguous memory allocation schemes.	Remember	CACS007.13
11	Distinguish between paging and segmentation.	Understand	CACS007.12
12	State the purpose of TLB.	Remember	CACS007.12
13	Explain the calculation of effective access time of a demand-paged memory system.	Understand	CACS007.11
14	Distinguish between page table and inverted page table.	Understand	CACS007.12
15	State the benefits of a virtual memory system.	Remember	CACS007.11
16	Distinguish between demand paging and pure demand paging.	Understand	CACS007.11
17	Distinguish between local and global page replacement strategies.	Understand	CACS007.13
18	Explain page fault and its effect on the performance of the demand paged memory system.	Understand	CACS007.11
19	Explain the need for page-replacement.	Understand	CACS007.13
20	List various page replacement algorithms.	Remember	CACS007.13
21	Explain the basic approach of page replacement.	Understand	CACS007.12
22	Distinguish between equal and proportional frame allocation strategies.	Understand	CACS007.13
23	Explain the concept of thrashing and why thrashing should be avoided in a system.	Understand	CACS007.15
PART-B (Long Answer Questions)			
1	Describe the following. a) Virtual Memory b) Cache Memory c) Auxiliary Memory	Understand	CACS007.11
2	Explain in detail the requirements that memory management technique needs to satisfy.	Understand	CACS007.13
3	Describe a) Paging b) Page table structure c) Translation look-aside buffer d) Segmentation	Remember	CACS007.12
4	Explain why the “principle of locality” is crucial to the use of virtual memory. What is accomplished by page buffering.	Understand	CACS007.12
5	Discuss briefly the swapping concept with necessary examples.	Understand	CACS007.15
6	Describe contiguous memory allocation concept with advantages and disadvantages.	Remember	CACS007.13
7	Differentiate the main memory organization schemes of contiguous- memory allocation, segmentation, and paging with respect to the following	Understand	CACS007.13
8	Differentiate between internal and external fragmentation and Which one occurs in paging scheme.	Understand	CACS007.13
9	Explain briefly about paging with neat diagram.	Understand	CACS007.12
10	Describe the following a) Hierarchical paging b) Inverted page Tables	Remember	CACS007.13
11	Draw and explain the working procedure of paging hardware in detail.	Understand	CACS007.12
12	Explain the basic concepts of segmentation with neat diagrams.	Understand	CACS007.13
13	Define page fault. When does a page fault occur. Describe the action taken by OS when page fault occurs.	Remember	CACS007.13
14	State and explain about virtual memory concept with neat diagram.	Remember	CACS007.12
15	Differentiate between paging and segmentation.	Understand	CACS007.13
16	Explain briefly the performance of demand paging with necessary examples.	Understand	CACS007.13
17	Explain the basic Scheme of page replacement and about the various page replacement strategies with examples.	Understand	CACS007.12

18	Define the Readers and Writers problem and its solution using the concept of semaphores.	Remember	CACS007.13
19	Explain the uses of the following: a. Mutex object b. Semaphore object c. Waitable timer object	Understand	CACS007.15
20	Write short notes about the following: a. Binary Semaphores b. Bounded Waiting	Remember	CACS007.15
21	Explain the Readers and Writers problem and its solution using the concept of semaphores.	Understand	CACS007.15
	PART-C (Problem Solving and Critical Thinking)		
1	Suppose you have 16M bytes of main memory. Using the list method there is an overhead of 8B per memory block. Using the bitmap method, the allocation granularity is of 128B. How many blocks are there when the space overhead of both methods is the same. Explain the average block size for this many blocks.	Remember	CACS007.13
2	Consider a computer system supports 32-bit virtual addresses as well as 32-bit physical addresses. Since the virtual address space is of the same size as the physical address space, the operating system designers decide to get rid of the virtual memory entirely.	Remember	CACS007.12
3	Consider a CPU generates 32-bit virtual addresses. The page size is 4 KB. The processor has a translation look-aside buffer (TLB) which can hold a total of 128 page table entries and is 4-way set associative. The minimum size of the TLB tag is:	Understand	CACS007.13
4	Consider there are 3 page frames which are initially empty. If the page reference string is 1, 2, 3, 4, 2, 1, 5, 3, 2, 4, 6, the number of page faults using the optimal replacement policy is	Remember	CACS007.13
5	Consider the following page reference string 7,0,2,0,3,0,4,2,3,0,3,2,1,2,0,1,7,0 Assuming three frames, how many page faults would occur in each of the following cases. a) LRU b) FIFO c) Optimal algorithms Note that initially all frames are empty.	Remember	CACS007.13
6	Analyze that we have a paging system with page table stored in memory A. If a memory reference takes 200 nanoseconds how long does a paged B. If we add associative registers and 75% of all page table references are memory reference take found in the associative registers, what is the effective memory reference time. Assume that finding a page table entry in the associative registers takes zero time, if the entry is there.	Remember	CACS007.13
7	In two level nested loops, the outer index (i) runs from 1 to 5 and the inner index (j) runs from 1 to 10.The page faults seem to occur for every 7^{th} inner most biterations. If it takes 0.02micro second to load a new page what is the extra time required because of occurrence of page faults.	Remember	CACS007.15
8	Given memory partitions of 100K, 500K, 200K, 300K, and 600K (in order), how would each of the First-fit, Best-fit, and Worst-fit algorithms place processes of 212K, 417K, 112K, and 426K (in order). Explain Which algorithm makes the most efficient use of memory.	Understand	CACS007.13
9	Suppose we have a demand paged memory. The page table is held in registers. It takes 8 milliseconds to service a page fault if an empty frame is available or the replaced page is not modified and 20 milliseconds if the replaced page is modified. Memory access time is 100 nanoseconds. Consider that the page to be replaced is modified 70 percent of the time. What is the maximum acceptable page-fault rate for an effective access time of no more than 200 nanoseconds.	Remember	CACS007.12

10	Consider a logical address space of eight pages of 1024 words each mapped onto a physical memory of 32 frames a) How many bits are in the logical address. b) How many bits are in the physical address.	Remember	CACS007.13

UNIT – IV			
File System Interface, Mass Storage Structure			
PART - A (Short Answer Questions)			
S. No.	**Question**	**Blooms Taxonomy Level**	**Course Outcomes**
1	Define the terms – file, file path, directory.	Remember	CACS007.16
2	Explain any four common file attributes.	Understand	CACS007.16
3	Explain any four file operations.	Understand	CACS007.16
4	Distinguish between shared and exclusive lock.	Understand	CACS007.16
5	Explain the allocation methods of a disk space.	Understand	CACS007.17
6	Explain the bit vector method free space management on disk.	Understand	CACS007.17
7	List the different file accessing methods.	Remember	CACS007.16
8	Explain the operations that can be performed on a directory.	Understand	CACS007.16
9	Discuss the most common schemes for defining the logical structure of a directory.	Understand	CACS007.16
10	Describe UFD and MFD..	Understand	CACS007.17
11	Describe file system mounting.	Understand	CACS007.16
12	Write the format of a typical file-control block.	Remember	CACS007.17
13	List the different disk-space allocation methods.	Understand	CACS007.17
14	List the various layers of a file system.	Remember	CACS007.16
15	Explain the functions of virtual file system (VFS).	Understand	CACS007.16
16	Describe about different types of disk scheduling.	Understand	CACS007.18
17	Define the terms with respect to disk I/O - seek time, latency time.	Understand	CACS007.17
18	List any four common file types and their extensions.	Remember	CACS007.16
19	Describe about logical formatting of the disk.	Remember	CACS007.18
20	State the advantages of indexed disk-space allocation strategy.	Remember	CACS007.17
21	List the different free disk-space management techniques.	Remember	CACS007.17
22	Explain the information associated with an open file.	Understand	CACS007.16
23	Discuss the advantages of contiguous memory allocation of disk space.	Understand	CACS007.18
24	Discuss the drawbacks of contiguous allocation of disk space.	Understand	CACS007.18
25	List any four secondary storage memory devices.	Remember	CACS007.18
26	State the advantages of linked disk-space allocation strategy.	Remember	CACS007.17
27	List various disk-scheduling algorithms.	Understand	CACS007.18
28	State the purpose of boot block.	Understand	CACS007.18
PART-B (Long Answer Questions)			
1	a) Discuss the criteria for choosing a file organization. b) Describe indexed file and indexed sequential file organization.	Understand	CACS007.16
2	Describe the file system of UNIX.	Understand	CACS007.16
3	List the common file types along with their extensions and describe each file type.	Remember	CACS007.16
4	Differentiate among the following disk scheduling algorithms. a) FCFS b) SSTF c) SCAN d) C-SCAN e) LOOK f) C-LOOK	Understand	CACS007.17

5	a) Define magnetic disk structure and its management. b) Exemplify swap space management.	Remember	CACS007.18
6	Explain the following in detail with respect to disk. a) Seek time b) Latency c) Access time d) Transfer time	Understand	CACS007.18
7	a) Explain in detail the interrupts and interrupt handling features. b) Explain with neat diagram the steps in DMA transfer.	Understand	CACS007.18
8	a) Define the N-step SCAN policy for disk scheduling. b) Explain how double buffering improves the performance than a single buffer for I/O.	Remember	CACS007.18
9	a) Explain the techniques used for performing I/O. b) Give an example of an application in which data in a file should be accessed in the following order: i. sequential ii. Random	Understand	CACS007.18
10	Explain the concept and techniques of free space management.	Understand	CACS007.20
11	Define how disk caching can improve disk performance.	Remember	CACS007.18
12	Explain low-level formatting or physical formatting.	Understand	CACS007.17
13	Define buffering, caching and spooling.	Remember	CACS007.16
14	Discuss the following a) File system mounting b)Thrashing	Understand	CACS007.18
15	Explain the following file concepts: a) File attributes b) File operations c) File types d) Internal file structure	Understand	CACS007.19
16	Explain the concept of file sharing. What are the criteria to be followed in systems which implement file sharing.	Remember	CACS007.18
17	Describe the following Directory Implementation methods. a)Linear List b) Hash Table	Knowledge	CACS007.19
18	Discuss in detail the performance issues of secondary storage management.	Understand	CACS007.18
19	Discuss about a) Disk space management b) Swap –space management	Understand	CACS007.20
PART-C (Problem Solving and Critical Thinking)			
1	Suppose we have files F1 to F4 in sizes of 7178, 572, 499 and 1195 bytes. Our disks have fixed physical block size of 512 bytes for allocation. Explain how many physical blocks would be needed to store these four files if we were to use a chained allocation strategy assuming that we need 5 bytes of information to determine the next block in the link. Which file results in the maximum internal fragmentation (measured as a percentage of the file size itself).	Understand	CACS007.18
2	Is there any way to implement truly stable storage. Explain your answer	Remember	CACS007.16
3	A hard disk has 63 sectors per tracks, 10 platters each with 2 recording surfaces and 1000 cylinders. The address of a sector is given as a triple < C, h, and s> where c is the cylinder number, h is the surface number and s is the sector number. Thus 0th sector is addressed as <0, 0, and 0>, the 1st sector is Addressed as <0, 0, and 1> and so on. Calculate the address of 1050thsector.	Understand	CACS007.18
4	Explain the maximum file size supported by a file system with 16 direct blocks, single, double, and triple indirection. The block size is 512 bytes. Disk block numbers can be stored in 4 bytes.	Understand	CACS007.18

5	Discuss the reasons why the operating system might require accurate information on how blocks are stored on disk. how could operating system improves file system performance with this knowledge	Understand	CACS007.18
6	Discuss how OS could maintain a free-space list for a tape-resident file system. Assume that the tape technology is append-only and that it uses EOT marks and locate, space and read position command	Understand	CACS007.16
7	Using a diagram, show how an indexed allocation of a file may be done for a disked based system with the following characteristics. The disc size is 30blocks each of 1024 bytes (may be modeled as 6 X 5 matrixes). File f1 is 11 logical records of 112 bytes, file f2 is 890 logical records of 13 bytes, file f3 is 510 bytes of binary data stream and file f4 is 4 logical blocks of 95 bytes.	Remember	CACS007.18
8	Could a RAID level 1 organization achieve better performance for read requests than RAID level 0 organization(with non redundant striping of data). If so, how.	Understand	CACS007.18
9	Compare the performance of write operations achieved by a RAID level 5 organization with that achieved by a RAID level 1 organization.	Understand	CACS007.20
10	Consider that a disk drive has 5,000 cylinders, numbered 0 to 4,999. The drive is currently serving request at cylinder 143, and the previous request was at cylinder 125. The queue of pending requests, in FIFO order, is: 86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130 Starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all pending requests for each of the following disk scheduling algorithms. A. FCFS B. SSTF C. SCAN D. C-SCAN E. LOOK F. C-LOOK	Remember	CACS007.18

UNIT – V

Deadlocks, Protection

PART - A (Short Answer Questions)

S. No.	Question	Blooms Taxonomy Level	Course Outcomes
1	What is Deadlock.	Remember	CACS007.21
2	Define resource. List some resources that a process might need for its execution.	Knowledge	CACS007.21
3	List the four data structures (matrices) that must be maintained to implement banker's algorithm.	Understand	CACS007.22
4	Describe the conditions under which a deadlock situation may arise.	Remember	CACS007.21
5	Explain safe state and unsafe state.	Understand	CACS007.21
6	Describe the representation of a resource-allocation graph.	Remember	CACS007.22
7	Distinguish between deadlock avoidance and prevention strategies.	Understand	CACS007.22
8	Define the purpose of banker's algorithm.	Remember	CACS007.22
9	Explain the sequence in which a process may utilize the resources in normal mode of operation.	Understand	CACS007.21
10	Describe the techniques for recovery from deadlock.	Remember	CACS007.22
11	List the goals of protection.	Remember	CACS007.23
12	Describe any one language-based protection schemes.	Understand	CACS007.24
13	Write the format of an access matrix.	Understand	CACS007.23
14	List the implementation techniques of access matrix.	Remember	CACS007.23
15	Describe role-based access control.	Understand	CACS007.23
16	List the schemes that implement revocation of capabilities.	Remember	CACS007.24
17	List any two example systems that implement capability-based protection.	Remember	CACS007.24
18	Define the terms – object, domain, access right.	Understand	CACS007.23
19	Write the main differences between capability lists and access lists.	Understand	CACS007.24
20	State the protection problems that may arise if a shared stack is used for parameter passing.	Understand	CACS007.24

<table>
<tr><td>21</td><td>State principle of least privilege.</td><td>Remember</td><td>CACS007.24</td></tr>
<tr><td colspan="4">PART-B (Long Answer Questions)</td></tr>
<tr><td>1</td><td>Define deadlock. What are the four conditions necessary for a deadlock situation to arise. How it can be prevented.</td><td>Remember</td><td>CACS007.21</td></tr>
<tr><td>2</td><td>Explain briefly resource allocation graph with examples.</td><td>Understand</td><td>CACS007.22</td></tr>
<tr><td>3</td><td>Differentiate the deadlock handling methods.</td><td>Understand</td><td>CACS007.22</td></tr>
<tr><td>4</td><td>Define in detail the technique of deadlock avoidance.</td><td>Remember</td><td>CACS007.22</td></tr>
<tr><td>5</td><td>Explain Banker’s algorithm for deadlock avoidance with an example.</td><td>Understand</td><td>CACS007.22</td></tr>
<tr><td>6</td><td>Discuss the various issues that need to be considered through the process of revocation of access rights.</td><td>Understand</td><td>CACS007.24</td></tr>
<tr><td>7</td><td>State and explain the methods involved in recovery from deadlocks.</td><td>Remember</td><td>CACS007.22</td></tr>
<tr><td>8</td><td>Describe resource-allocation graph. Explain how resource graph can be used for detecting deadlocks.</td><td>Understand</td><td>CACS007.22</td></tr>
<tr><td>9</td><td>Describe the terms.
a) Race condition
b) Atomic transaction
c) Critical section
d) Mutual exclusion</td><td>Remember</td><td>CACS007.22</td></tr>
<tr><td>10</td><td>Describe how the access matrix facility and role-based access control facility are similar. How do they differ.</td><td>Remember</td><td>CACS007.24</td></tr>
<tr><td>11</td><td>Explain why a capability based system such as Hydra provides greater flexibility than the ring- protection scheme in enforcing protection policies.</td><td>Understand</td><td>CACS007.23</td></tr>
<tr><td>12</td><td>Define the following.
a) Goals of protection
b) Principles of protection</td><td>Remember</td><td>CACS007.23</td></tr>
<tr><td>13</td><td>Discuss about domain of protection.</td><td>Understand</td><td>CACS007.23</td></tr>
<tr><td>14</td><td>Why do you need to provide protection to the system. Explain how access matrix can be used for the purpose.</td><td>Understand</td><td>CACS007.25</td></tr>
<tr><td>15</td><td>Discuss the access matrix implementation techniques.</td><td>Understand</td><td>CACS007.24</td></tr>
<tr><td>16</td><td>Compare the various access matrix implementation techniques.</td><td>Remember</td><td>CACS007.24</td></tr>
<tr><td>17</td><td>Discuss deadlock detection method in detail.</td><td>Understand</td><td>CACS007.22</td></tr>
<tr><td>18</td><td>Explain various schemes to implement revocation for capabilities.</td><td>Understand</td><td>CACS007.24</td></tr>
<tr><td>19</td><td>Describe how language-based protection scheme can be used for providing system protection at kernel level.</td><td>Remember</td><td>CACS007.24</td></tr>
<tr><td>20</td><td>Explain relative merits of compiler-based enforcement based solely on a kernel, as opposed to enforcement provided largely by a compiler.</td><td>Understand</td><td>CACS007.25</td></tr>
<tr><td colspan="4">PART-C (Problem Solving and Critical Thinking)</td></tr>
<tr><td>1</td><td>Consider the following snapshot of a system
<table>
<tr><td></td><td colspan="4">Allocation</td><td colspan="4">Max</td><td colspan="4">Available</td></tr>
<tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>A</td><td>B</td><td>C</td><td>D</td><td>A</td><td>B</td><td>C</td><td>D</td></tr>
<tr><td>P1</td><td>0</td><td>0</td><td>1</td><td>3</td><td>0</td><td>0</td><td>1</td><td>2</td><td>1</td><td>5</td><td>2</td><td>0</td></tr>
<tr><td>P2</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>7</td><td>5</td><td>0</td><td></td><td></td><td></td><td></td></tr>
<tr><td>P3</td><td>1</td><td>3</td><td>5</td><td>4</td><td>2</td><td>3</td><td>5</td><td>6</td><td></td><td></td><td></td><td></td></tr>
</table>
Answer the following questions using the banker’s algorithm:
a) What is the content of matrix “Need”.
b) Is the system in a safe state.
c) If a request from process P1 arrives for (0, 4, 2, 0) can the request</td><td>Remember</td><td>CACS007.22</td></tr>
<tr><td>2</td><td>Consider the version of the dining-philosophers problem in which the chopsticks are placed at the center of the table and any two of them can be used by a philosopher. Assume that requests for chopsticks are made one at a time. Describe a simple rule for determining whether a particular request can be satisfied without causing deadlock given the current allocation of chopsticks to philosophers.</td><td>Remember</td><td>CACS007.22</td></tr>
</table>

3	Consider a system consisting of *m* resources of the same type being shared by *n* processes. A process can request or release only one resource at a time. Show that the system is deadlock free if the following two conditions hold: a) The maximum need of each process is between one resource and *m* resources. *b)* The sum of all maximum needs is less than *m +n.*	Remember	CACS007.22
4	Explain How does the principle of least privilege aid in the creation of protection systems.	Understand	CACS007.24
5	Describe how the Java protection model would be compromised if a Java program were allowed to directly alter the annotations of its stack frame.	Understand	CACS007.25
6	Describe the Coffman's conditions that lead to a deadlock.	Remember	CACS007.21

7	A system has n resources R_0... R_{n-1},and k processes P_0,... P_{k-1}.The implementation of the resource request logic of each process P_i is as follows:
	if (i % 2 == 0)
	{
	if (i < n) request Ri
	if (i+2 < n) request Ri+2
	}
	else
	{
	if (i < n) request Rn-i
	if (i+2 < n) request Rn-i-2
	}
		Remember	CACS007.22

8	A system contains three programs and each requires three tape units for its operation. Explain the minimum number of tape units which the system must have such that deadlocks never arise is.	Remember	CACS007.21
9	A system has 6 identical resources and N processes competing for them. Each process can request at most 2 resources. Explain which one of the following values of N could lead to a deadlock.	Remember	CACS007.22
10	Two shared resources R_1 and R_2 are used by processes P_1 and P_2. Each process has a certain priority for accessing each resource. Let T_{ij} denote the priority of P_i for accessing R_j. A process P_i can snatch a resource R_h from process P_j if T_{ik} is greater than T_{jk}. Given the following: 1. $T_{11} > T_{21}$ 2. $T_{12} > T_{22}$ 3. $T_{11} < T_{21}$ 4. $T_{12} < T_{22}$ Explain which of the following conditions ensures that P_1 and P_2 can never deadlock.	Remember	CACS007.22

Prepared By:
Mr. N V Krishna Rao, Associate Professor

HOD, CSE