UNIT-IX

OPERATING SYSTEM

ENGINEERS ACADEMY®
Your GATEway to Professional Excellence
IES • GATE • PSUs • JTO • IAS • NET

www. eapublications.org

NOTES

INTRODUCTION TO OPERATING SYSTEMS

CHAPTER 1

OBJECTIVE QUESTIONS

1. Which of the following is an example of spooled device?
   (a) A line printer used to print the output of a number of jobs.
   (b) A terminal used to enter input data to a running program
   (c) A secondary storage device in a virtual memory system
   (d) A graphic display device

2. A multi-user, multi-processing operating system cannot be implemented on hardware that does not support.
   (a) Address translation
   (b) DMA for disk transfer
   (c) At least two modes of CPU execution (privileged and non-privileged)
   (d) Demand paging

3. CPU performance is measured through ________.
   (a) Throughput
   (b) MHz
   (c) Flops
   (d) None of the above

4. Which of the following is contained in Process Control Block (PCB)?
   (a) Process Number
   (b) List of Open files
   (c) Memory Limits
   (d) All of the Above

5. A processor
   (a) is a device that performs a sequence of operations specified by instructions in memory
   (b) is the device where information is stored
   (c) is a sequence of instructions
   (d) is typically characterized by interactive processing and time of the CPU's time to allow quick response to each other

6. The operating system of a computer serves as a software interface between the user and
   (a) hardware (b) peripheral
   (c) memory (d) screen

7. Multiprogramming was made possible by
   (a) input/output units that operate independently of the cpu
   (b) operating system
   (c) both (a) and (b)
   (d) None of the above

8. Real time systems are
   (a) primarily used on mainframe computers
   (b) used for larger response time
   (c) used for program analysis
   (d) used for real-time interactive

9. Unix Operating System is an __________.
   (a) Time Sharing Operating System
   (b) Multi-User Operating System
   (c) Multi-tasking Operating System
   (d) All the Above

10. Which of the following is not advantage of multiprogramming?

(a) Increased throughput

(b) Shorter response time

(c) Decreased operating system overhead

(d) Ability to assign priorities to jobs

11. In ______ OS, the response time is very critical.

(a) Multitasking

(b) Batch

(c) Online

(d) Real-time

12. Which is not an Operating System?

(a) Windows 95 (b) MS-DOS

(c) Windows 3.1 (d) Windows 2000

13. Multiprogramming systems ________.

(a) Are easier to develop than single programming systems

(b) Execute each job faster

(c) Execute more jobs in the same time

(d) Are used only on large main frame computers

14. The number of processes completed per unit time is known as __________.

(a) Output

(b) Throughput

(c) Efficiency

(d) Capacity

15. The operating system manages ________.

(a) Memory

(b) Processor

(c) Disk and I/O devices

(d) All of the above

16. Which technique was introduced because a single job could not keep both the CPU and the I/O devices busy?

(a) Time-sharing

(b) SPOOLing

(c) Preemptive scheduling

(d) Multiprogramming

❑❑❑

Answer Key

1. *Ans. (a)*

   Spooled devices are those which fatches the data from job waiting area one by one and spool means simultaneous peripheral operation on line, so printer is a spooling device.

2. *Ans. (d)*

   Multiple jobs have to be placed in small RAM so, demand paging concept should be there by that we can place more no. of jobs into RAM at a time.

3. *Ans.(a)*
4. *Ans .(d)*
5. *Ans: (a)*
6. *Ans: (a)*
7. *Ans: (c)*
8. *Ans: (d)*
9. *Ans. (d)*
10. *Ans. (c)*
11. *Ans. (d)*
12. *Ans. (c)*
13. *Ans. (c)*
14. *Ans. (b)*
15. *Ans. (d)*
16. *Ans. (d)*

❑❑❑

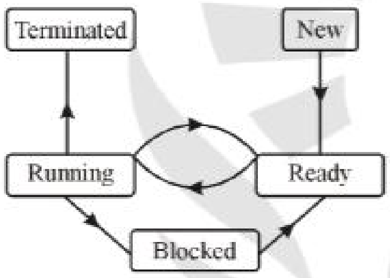
CHAPTER 2

PROCESS MANAGEMENT

OBJECTIVE QUESTIONS

1. In a time-sharing operating system, when the time slot given to a process is completed, the process goes from the RUNNING state to the
   (a) BLOCKED state
   (b) READY state
   (c) SUSPENDED state
   (d) TERMINATED state

2. The process state transition diagram in figure is representative of



   (a) a batch operating system
   (b) an operating system with a preemptive scheduler
   (c) an operating system with a non-preemptive scheduler
   (d) a uni-programmed operating system

3. PCB =
   (a) Program Control Block
   (b) Process Control Block
   (c) Process Communication Block
   (d) None of the above

4. Moving Process from main memory to disk is called
   (a) scheduling (b) caching
   (c) swapping (d) spooling

5. Information about a process is maintained in a _________.
   (a) Stack
   (b) Translation Lookaside Buffer
   (c) Process Control Block
   (d) Program Control Block

6. A scheduler which selects processes from secondary storage device is called
   (a) Short term scheduler.
   (b) Long term scheduler.
   (c) Medium term scheduler.
   (d) Process scheduler.

7. The Purpose of Co-operating Process is __________.
   (a) Information Sharing
   (b) Convenience
   (c) Computation Speed-Up
   (d) All of the above

8. The removal of process from active contention of CPU and reintroduce them into memory later is known as ____________.
   (a) Interrupt (b) Swapping
   (c) Signal (d) Thread

9. In the blocked state
   (a) the processes waiting for I/O are found
   (b) the process which is running is found
   (c) the processes waiting for the processor are found
   (d) none of the above

10. Which is not the state of the process ?

(a) Blocked (b) Running

(c) Ready (d) Privileged

11. The state of a process after it encounters an I/O instruction is __________.

(a) Ready (b) Blocked/Waiting

(c) Idle (d) Running

12. ________ scheduler selects the jobs from the pool of jobs and loads into the ready queue.

(a) Long term (b) Short term

(c) Medium term (d) None of the above

13. Saving the state of the old process and loading the saved state of the new process is called ________.

(a) Context Switch

(b) State

(c) Multi programming

(d) None of the above

14. A program in execution is called

(a) Process (b) Instruction

(c) Procedure (d) Function

15. Which of the following is contained in Process Control Block (PCB)?

(a) Process Number

(b) List of Open files

(c) Memory Limits

(d) All of the Above

16. Suppose that a process is in 'BLOCKED' state waiting for some I/O service. When the service is completed, it goes to the

(a) RUNNING state

(b) READY state

(c) SUSPENDED state

(d) TERMINATED state

❑❑❑

Answer Key

1. *Ans. (b)*
2. *Ans. (b)*

   Since there is an arrow from running to ready, so it is pre-emptive.
3. *Ans. (b)*
4. *Ans. (c)*
5. *Ans. (c)*
6. *Ans. (b)*
7. *Ans. (d)*
8. *Ans. (a)*
9. *Ans. (a)*
10. *Ans. (d)*
11. *Ans. (b)*
12. *Ans. (a)*
13. *Ans. (a)*
14. *Ans. (a)*
15. *Ans. (d)*
16. *Ans. (b)*

❑❑❑

CHAPTER 3

THREAD

OBJECTIVE QUESTIONS

1. Consider the following statements with respect to user-level threads and kernel supported threads

   (i) context switch is faster with kernel-supported threads

   (ii) for user-level threads, a system call can block the entire process

   (iii) Kernel supported threads can be scheduled independently

   (iv) User level threads implementation is easy.

   Which of the above statements are true?

   (a) (ii), (iii) and (iv) only

   (b) (ii) and (iii) only

   (c) (i) and (iii) only

   (d) (i) and (ii) only

2. A thread

   (a) is a lightweight process where the context switching is faster

   (b) is a lightweight process where the context switching is slower

   (c) is used to speed up paging

   (d) none of the above

3. Consider n processes sharing the CPU in a round-robin fashion. Assuming that each process switch takes a seconds, what must be the quantum size q such that the overhead resulting from process switching is minimized but at the same time each process is guaranteed to get its turn at the CPU at least every seconds ?

   (a) $q \le \frac{t-ns}{n-1}$ (b) $q \ge \frac{t-ns}{n-1}$

   (c) $q \le \frac{t-ns}{n+1}$ (d) $q \ge \frac{t-ns}{n+1}$

4. System calls are usually invoked by using

   (a) a software interrupt

   (b) polling

   (c) an indirect jump

   (d) a privileged instruction

5. A processor needs software interrupt to

   (a) Test the interrupt system of the processor

   (b) Implement co-routines

   (c) Obtain system services which need execution of privileged instructions

   (d) Return from subroutine

6. A CPU has two modes-privileged and non-privileged. In order to change the mode form privileged to non-privileged

   (a) a hardware interrupt is needed

   (b) a software interrupt is needed

   (c) a privileged instruction (which does not generate an interrupt) is needed

   (d) a non-privileged instruction (which does not generate an interrupt) is needed

7. Consider the following statements with respect to user-level threads and kernel-supported threads

   (i) Context switch is faster with kernel-supported threads

   (ii) For user-level threads, a system call can block the entire process

   (iii) Kernel-supported threads can be scheduled independently

   (iv) User-level threads are transparent to the kernel

   Which of the above statements are true ?

   (a) (ii), (iii) and (iv) only

   (b) (ii) and (iii) only

   (c) (i), and (iii) only

   (d) (i) and (ii) only

8. Which one of the following is NOT shared by the threads of the same process ?
   (a) Stack
   (b) Address Space
   (c) File Descriptor Table
   (d) Message Queue

9. A process executes the following segment of code:
   for (i = 1; i < = n; i++)
   fork ( );
   The number of new processes created is
   (a) n (b) ((n(n + 1))/2)
   (c) $2^n - 1$ (d) $3^n - 1$

10. A user level process in Unix traps the signal sent on a Ctrl-C input, and has a signal handling routine that saves appropriate files before terminating the process. When a Ctrl-C input is given to this process, what is the mode in which the signal handling routine executes?
   (a) kernel mode (b) super user mode
   (c) privileged mode (d) user mode

11. Consider the following code fragment:
   if (fork ( ) = = 0)
   {a = a + 5; printf("%d, %d\n", a, &a);}
   else{a = a – 5; printf("%d, %d\n", a, &a);}
   Let u, v be the values printed by the parent process, and x, y be the values printed by the child process. Which one of the following is TRUE?
   (a) u = x + 10 and v = y
   (b) u = x + 10 and v ≠ y
   (c) u + 10 = x and v = y
   (d) u + 10 = x and v ≠ y

12. Consider the following statements about user level threads and kernel level threads. Which one of the following statements is FALSE?
   (a) Context switch time is longer for kernel level threads than for user level threads
   (b) User level threads do not need any hardware support
   (c) Related kernel level threads can be scheduled on different processors in a multiprocessor system
   (d) Blocking one kernel level thread blocks all related threads

13. Which of the following statements about synchronous and asynchronous I/O is NOT true?
   (a) An ISR is invoked on completion of I/O in synchronous I/O but not in asynchronous I/O
   (b) In both synchronous and asynchronous I/O and ISR (Interrupt Service Routine) is invoked after completion of the I/O
   (c) A process making a synchronous I/O call waits until I/O is complete, but a process making an asynchronous I/O call does not wait for completion of the I/O
   (d) In the case of synchronous I/O, the process waiting for the completion of I/O is woken up by the ISR that is invoked after the completion of I/O

14. A process executes the following code for (i = 0; i < n; i ++) fork ( );
   The total number of child processes created is
   (a) n (b) $2^n - 1$
   (c) 2^n (d) $2^{n+1} - 1$

15. A computer handles several interrupt sources of which the following are relevant for this question.
   - Interrupt from CPU temperature sensor (raises interrupt if CPU temperature is too high)
   - Interrupt from Mouse (raises interrupt if the mouse is moved or a button is pressed)
   - Interrupt from Keyboard (raises interrupt when a key is pressed or released)
   - Interrupt from Hard Disk (raises interrupt when a disk read is completed)

   Which one of these will be handled at the **HIGHEST** priority?
   (a) Interrupt from Hard Disk
   (b) Interrupt from Mouse
   (c) Interrupt from Keyboard
   (d) Interrupt from CPU temperature sensor

16. A thread is usually defined as a "light weight process" because an Operating System (OS) maintains smaller data structures for a thread than for a process. In relation to this, which of the following is TRUE?
   (a) On per-thread basis, the OS maintains only CPU register state
   (b) The OS does not maintain a separate stack for each thread
   (c) On per-thread basis, the OS does not maintain virtual memory state
   (d) On per-thread basis, the OS maintains only scheduling and accounting information

17. A process executes the code

fork ( );

fork ( );

fork ( );

The total number of child processes created is

(a) 3 (b) 4

(c) 7 (d) 8

18. Which one of the following is FALSE?

(a) User level threads are not scheduled by the kernel.

(b) When a user level thread is blocked, all other threads of its process are blocked.

(c) Context switching between user level threads is faster than context switching between kernel level threads.

(d) Kernel level threads cannot share the code segment.

19. The maximum number of processes that can be in *Ready* state for a computer system with n CPUs is

(a) n (b) n^2

(c) 2^n (d) Independent of n

20. Threads of a process share

(a) global variables but not heap

(b) heap but not global variables

(c) neither global variables nor heap

(d) both heap and global variables

21. Which of the following is /are shared by all the threads in a process?

I. Program counter

II. Stack

III. Address space

IV. Registers

(a) I and II only (b) II only

(c) IV only (d) III and IV only

qqq

Answer Key

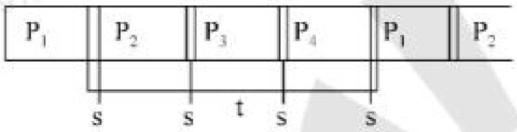
1. *Ans.(a)*

For user-level threads, a system call can block the entire process ,Kernel supported threads can be scheduled independently User level threads implementation is easy.

2. *Ans. (a)*

3. *Ans. (a)*

Let n = 4



$$t \geq n \times s + q \times (n - 1)$$

$$\text{So, } q \leq \frac{t - ns}{n - 1}$$

4. *Ans. (a)*

System calls are invoked by using software interrupt.

5. *Ans. (c)*

To execute privileged instructions, system services can be obtained using software interrupt.

6. *Ans. (d)*

Because we want to change the mode from privileged to non-privileged, to the next instruction to be executed should be non-privileged instruction.

7. *Ans. (a)*

Consider the each statement

(i) Statement is false because there is no connection between kernel-supported threads and context switch.

(ii) Statement is true because it is the drawback of user-level threads blocking system call can block the entire process.

(iii) Statement is true because kernel-supported threads having own memory area and scheduled independently by the operating system.

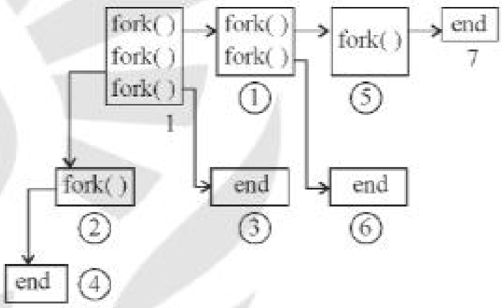
(iv) Statement is true because kernel is unware about user level threads and there is not kernel support to user-level threads. So user-level threads are transparent to the kernal.

8. *Ans. (a)*

Stack and registers are not shared by the threads of the same process while address space, message queue etc. are shared.

9. *Ans. (c)*

The number of process created will be $2^n - 1$.



$2^3 - 1 = 7$

10. *Ans. (a)*

When user level process trapping the Ctrl+C signal then the trap signal is going through system call and that's why mode changed to kernel mode from user mode and then the request is handling.

One more thing kernel mode and privilege mode are same, answer is kernel mode (privilege mode).

11. *Ans. (d)*

Fork ( ) returns 0 is child process and process ID of child process is parent process.

In child(x), a = a + 5

In parent (u), a = a – 5

Therefore, x = u + 10

The physical addresses of parent and child processes will be different, v ≠ y.

12. *Ans. (d)*

In Kernel level threads, the blocking system call causes, the kernel can schedule another thread in the application for execution. so statement (d) is false about kernel level threads.

13. *Ans. (b)*

Statement (b) is not true because an ISR (Interrupt Service Routine) is invoked after completion of I/O in synchronous but not in asynchronous I/O.

14. *Ans. (b)*

Fork ( ) system call creates the child process initially number of processes is 0. After first fork ( ), it creates a single process. After second fork ( ), it creates one parent and two child processes. After n + 1 fork ( ), the total number of processes is 2^n but we subtract the main process then total number of child processes is $2^n - 1$.

15. *Ans. (d)*

Interrupt from CPU temperature sensor is given top priority to protect system resources. When CPu temperature is too high, the BIOS initiate an interrupt and informs the Operating System. OS gives top priority to this interrupt and immediately shuts down the system.

16. *Ans. (c)*

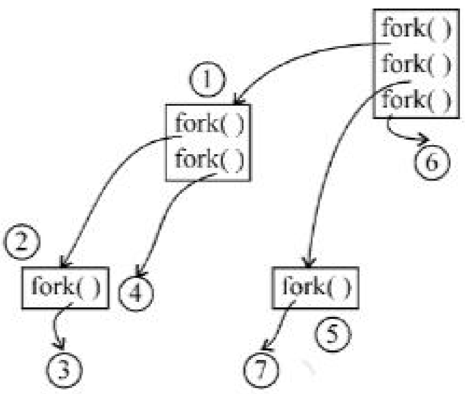
(a) False : On per thread basics, OS maintains both CPU register state and stack. Hence option (a) is false, because only CPU register state mentioned.

(b) False : OS maintains a separate stack for each thread.

(c) Ture : OS does not maintain virtual memory state.

(d) False : OS does not maintain scheduling and accounting information.

17. *Ans. (c)*



The number of child process created

$= 2^n - 1 = 2^3 - 1 = 7$

(where n is number of fork ( ) statements)

18. *Ans. (d)*

Kernel level threads shares the code segment.

19. *Ans. (d)*

Maximum number of processes that can be in ready state is independent of number of processes (n).

20. *Ans. (d)*

Generally every thread of a process have their on PC and stack. Both heap and global variable are shared by every thread of a process.

21. *Ans* **(b)**

All the threads share address space but other entities like,stack,PC, resisters are not shared and every thread will have its own.

□□□

CHAPTER 4

CPU SCHEDULING

OBJECTIVE QUESTIONS

1. Assume that the following jobs are to be executed on a single processor system.

Job Id	CPU Burst time
p	4
q	1
r	8
s	1
t	2

The jobs are assumed to have arrived at time 0^+ and in the order p, q, r, s, t. calculate the deperature time (completion time) for job p if scheduling is round robin with time slice 1.

(a) 4 (b) 10

(c) 11 (d) 12

2. Which scheduling policy is most suitable for a time-shared operating systems?

(a) Shortest job First

(b) Round Robin

(c) First come first serve

(d) Elevator

3. Which of the following is a criterion to evaluate a scheduling algorithm?

(a) CPU Utilization: Keep CPU utilization as high as possible.

(b) Throughput: number of processes completed per unit time.

(c) Waiting Time: Amount of time spent ready to run but not running.

(d) All of the above

4. Which of the following statements are true?

I. Shortest remaining time first scheduling may cause starvation

II. Preemptive scheduling may cause starvation

III. Round robin is better than FCFS in terms of response time

(a) I only

(b) I and III only

(c) II and III only

(d) I, II and III

5. An operating system uses Shortest Remaining Time first (SRTF) process scheduling algorithm. Consider the arrival times and execution times for the following processes:

Process	Execution time	Arrival time
P1	20	0
P2	25	15
P3	10	30
P4	15	45

What is the total waiting time for process P2?

(a) 5 (b) 15

(c) 40 (d) 55

6. 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.

(a) 1 (b) 2

(c) 3 (d) 4

7. An optimal scheduling algorithm in terms of minimizing the average waiting time of a given set of processes is ________.
   (a) FCFS scheduling algorithm
   (b) Round robin scheduling algorithm
   (c) Shorest job - first scheduling algorithm
   (d) None of the above

8. Consider the following set of processes, with the arrival times and the CPU-burst times given in milliseconds.

Process	Arrival time	Burst time
P1	0	5
P2	1	3
P3	2	3
P4	4	1

   What is the average turnaround time for these processes with the preemptive shortest remaining processing time first (SRPT) algorithm?
   (a) 5.50 (b) 5.75
   (c) 6.00 (d) 6.25

9. The scheduling in which CPU is allocated to the process according with least CPU-burst time is called
   (a) Priority Scheduling
   (b) Shortest job first Scheduling
   (c) Round Robin Scheduling
   (d) Multilevel Queue Scheduling

10. The "turn-around" time of a user job is the
   (a) time since its submission to the time its results become available.
   (b) time duration for which the CPU is allotted to the job.
   (c) total time taken to execute the job.
   (d) time taken for the job to move from assembly phase to completion phase.

11. The FIFO algorithm
   (a) executes first the job that last entered the queue
   (b) executes first the job that first entered the queue
   (c) execute first the job that has been in the queue the longest
   (d) executes first the job with the least processor needs

12. Four jobs to be executed on a single processor system arrive at time 0+ in the order A, 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
   (a) 10 (b) 4
   (c) 8 (d) 9

13. Round robin scheduling is essentially the preemptive version of ________.
   (a) FIFO (b) Shortest job first
   (c) Shortes remaining (d) Longest time first

14. FIFO scheduling is ________.
   (a) Preemptive Scheduling
   (b) Non Preemptive Scheduling
   (c) Deadline Scheduling
   (d) Fair share scheduling

15. In Priority Scheduling a priority number (integer) is associated with each process. The CPU is allocated to the process with the highest priority (smallest integer = highest priority). The problem of, Starvation ? low priority processes may never execute, is resolved by __________.
   (a) Terminating the process.
   (b) Aging
   (c) Mutual Exclusion
   (d) Semaphore

❑❑❑

ANSWER KEY

1. ***Ans. (c)***

Create the gantt Chart with Round Robin scheduling.

0	1	2	3	4	5	6	7	8	9	10	11
p	q	r	s	t	p	r	t	p	r	p...	

2. ***Ans. (b)***

Round Robin scheduling working on time quantum, after certain time every process get back the CPU units for it's completion and the same phenomena used in time sharing system. So Round Robin is best for time sharing system.

3. ***Ans.(d)***

4. ***Ans.(d)***

* Shortest remaining time first scheduling may cause starvation

* Preemptive scheduling may cause starvation

* Round robin is better than FCFS in terms of response time

5. ***Ans.(b)***

P2 waited 5 units for P1 to complete and 10 units for P3 to complete. So its total waiting time is 15, or answer is (B).

6. ***Ans.(d)***

Context switches will be needed only at the time of switching over to some other process .The following table shows the time remaining for the three processes at different instance of time.

P_1	P_2	P_3	P_1	P_4

0 20 30 40 55 70

4- context switched.

7. ***Ans. (c)***

8. ***Ans.(a)***

Average turnaround time=(P1+P2+P3+P4)/4=(12+3+6+1)/4=5.5

9. ***Ans. (b)***

10. ***Ans. (c)***

11. ***Ans. (b)***

12. ***Ans. (d)***

Jobs	burst time
A	4
B	1
C	8
D	1

1	2	3	4	5	6	7	8	9
A	B	C	D	A	C	A	C	A

13. ***Ans.(a)***

14. ***Ans. (b)***

15. ***Ans. (b)***

❑❑❑

CHAPTER 5

PROCESS SYNCHRONIZATION

OBJECTIVE QUESTIONS

1. A critical region is
   (a) One which is enclosed by a pair of P and V operations on semaphores.
   (b) A program segment that has not been proved bug-free
   (c) A program segment that often causes unexpected system crashes
   (d) A program segment where shared resources are accessed.

2. At a particular time of computation the value of a counting semaphore is 7. then 20 P operations and 15V operations were completed on this semaphore. The resulting value of the semaphore is:
   (a) 42 (b) 2
   (c) 7 (d) 12

3. A critical section is a program segment.
   (a) Which should run in certain specified amount of time.
   (b) Which avoids deadlocks
   (c) Where shared resources are accessed
   (d) Which must be enclosed by a pair of semaphore operations, P and V

4. Each process Pi, i = 1 ..... 9 is coded as follows

```
repeat
    P(mutex)
    {critical section}
    v(mutex)
forever
```

   The code for P_{10} is identical except that it uses v(mutex) in place of P(mutex). What is the largest number of processes that can be inside the critical section at any moment?
   (a) 1 (b) 2
   (c) 3 (d) None of the above

5. At a particular time, the value of a counting semaphore is 10. It will become 7 after
   (a) 3 V operations
   (b) 3 P operations
   (c) 5 V operations and 2 P operations
   (d) 11 P operations and 10 V operations

6. When the result of a computation depends on the speed of the processes involved there is said to be.
   (a) cycle stealing (b) race condition
   (c) a time lock (d) a deadlock

7. At a particular time of computation, the value of a counting semaphore is 7. Then 20 P operations and 'x' V operations were completed on this semaphore. If the final value of the semaphore is 5, x will be
   (a) 22 (b) 18
   (c) 15 (d) 13

8. Mutual exclusion problem occurs between
   (a) two disjoint processes that do not interact
   (b) processes that share resources
   (c) processes that do not use the same resource
   (d) none of these

9. Semaphore can be used for __________.
   (a) Wait & signal (b) Deadlock
   (c) Synchronization (d) Priority

10. A process said to be in __________ state if it was waiting for an event that will never occur.
   (a) Safe (b) Unsafe
   (c) Starvation (d) Dead lock

11. What are the requirements for the solution to critical section problem?

(a) Mutual Exclusion

(b) Progress

(c) Bounded Waiting

(d) All of Above

12. What are the two types of Semaphore?

(a) Digital Semaphores and Binary Semaphores

(b) Analog Semaphores and Octal Semaphores

(c) Counting Semaphores and Binary Semaphores

(d) Critical Semaphores and System Semaphores

13. Consider the methods used by processes P1 and P2 for accessing their critical Sections whenever needed, as given below. The initial values of shared boolean variables S1 and S2 are randomly assigned.

Method used by P1	Method used by P2
while (S1 = = S2) ; Critical Section S1 = S2;	while (S1 != S2) ; Critical Section S2 = not (S1);

Which one of the following statements describes the properties achieved?

(a) Mutual exclusion but not progress

(b) Progress but not mutual exclusion

(c) Neither mutual exclusion nor progress

(d) Both mutual exclusion and progress

14. Two processes, P1 and P2, need to access a critical section of code. Consider the following synchronization construct used by the processes:

```
/* P1 */
while (true) {
wants1 = true;
while (wants2 == true);
/* critical section */
wants1 = false;
}
/* Remainder section */
```

```
/* P2 */
while (true) {
wants2 = true;
while (wants1 == true); /* critical section */
wants2 = false;
}
/* Remainder section */
```

Here, wants1 and wants2 are shared variables, which are initialized to false. Which one of the following statements is TRUE about the above construct?

(a) It does not ensure mutual exclusion.

(b) It does not ensure bounded waiting.

(c) It requires that processes enter the critical section in strict alternation.

(d) It does not prevent deadlocks, but ensures mutual exclusion.

15. Let S and Q be two semaphores initialized to 1, where P0 and P1 processes the following statements wait(S);wait(Q); —; signal(S);signal(Q) and wait(Q); wait(S);—;signal(Q);signal(S); respectively. The above situation depicts a ________.

(a) Semaphore (b) Deadlock

(c) Signal (d) Interrupt

16. At a particular time, the value of counting semaphore is 10. it will become 7 after

(a) 3 V operations

(b) 5P operations

(c) 5V and 2P operations

(d) 13 P and 10 V operations.

17. Consider two processes P_1 and P_2 accessing the shared variables X and Y protected by two binary semaphores S_X and S_Y respectively, both initialized to 1. P and V denote the usual semaphore operators, where P decrements the semaphore value, and V increments the semaphore value. The pseudocode of P_1 and P_2 is as follows :

```
P1 :                    P2 :
while true do {         while true do {
   L1 : .........          L3 : .........
   L2 : .........          L4 : .........
   X = X + 1;              Y = Y + 1;
   Y = Y - 1;              X = X - 1;
   V(SX);                  V(SY);
   V(SY);  }               V(SX);  }
```

In order to avoid deadlock, the correct operators at L_1, L_2, L_3 and L_4 are respectively

(a) $P(S_Y)$, $P(S_X)$, $P(S_X)$, $P(S_Y)$

(b) $P(S_X)$, $P(S_Y)$, $P(S_Y)$, $P(S_X)$

(c) $P(S_X)$, $P(S_X)$, $P(S_Y)$, $P(S_Y)$

(d) $P(S_X)$, $P(S_Y)$, $P(S_X)$, $P(S_Y)$

18. Synchronization in the classical readers and writers problem can be achieved through use of semaphores. In the following incomplete code for readers-writers problem, two binary semaphores mutex and wrt are used to obtain synchronization

```
wait (wrt)
writing is performed
signal (wrt)
wait (mutex)
readcount = readcount + 1
if readcount = 1 then S1
S2
reading is performed
S3
readcount = readcount - 1
if readcount = 0 then S4
signal (mutex)
```

The values of S1, S2, S3, S4 (in that order) are

(a) signal (mutex), wait (wrt) signal (wrt), wait (mutex)

(b) signal (wrt), signal (mutex), wait (mutex), wait (wrt)

(c) wait (wrt), signal (mutex), wait (mutex), signal (wrt)

(d) signal (mutex), wait (mutex), signal (mutex), wait (mutex)

19. The following program consists of 3 concurrent processes and 3 binary semaphores. The semaphores the initialized as S0 = 1, S1 = 0, S2 = 0.

Process P0	Process P1	Process P2

while (true) {
wait (S0);
print '0'
release (S1);
release (S2);
}
	wait (S1);
	release (S0);
		wait (S2)
		release (S0);

How many times will process P0 print '0'?

(a) At least twice (b) Exactly twice

(c) Exactly thrice (d) Exactly once

20. A shared variable x, initialized to zero, is operated on by four concurrent process W, X, Y, Z as follows. Each of the processes W and X reads x from memory, increments by one, stores it to memory, and then terminates. Each of the processes Y and Z reads x from memory, decrements by two, stores it to memory, and then terminates. Each process before reading x invokes the P operations (i.e., wait) on a counting semaphore S and invokes the V operation (i.e., signal) on the semaphore S after storing x to memory. Semaphore S is initialized to two. What is the maximum possible value of x after all processes complete execution?

(a) – 2 (b) – 1

(c) 1 (d) 2

21. A multithreaded program P executes with x number of threads and uses y number of locks for ensuring mutual exclusion while operating on shared memory locations. All locks in the program are non-reentrant, i.e., if a thread holds a lock l, then it cannot re-acquire lock l without releasing it. If a thread is unable to acquire a lock, it blocks until the lock becomes available. The minimum value of x and the minimum value of y together for which execution of P can result in a deadlock are:

(a) x = 1, y = 2 (b) x = 2, y = 1

(c) x = 2, y = 2 (d) x = 1, y = 1

22. Consider the following solution to the producer-consumer synchronization problem. The shared buffer size is *N*. There semaphores *empty, full* and mutex are defined with respective initial values of 0,*N* and 1. Semaphore empty denotes the number of available slots in the in the buffer, for the producer to write to. The placeholder variables, denoted by P, Q, R, and S, in the code below can be assigned either *empty* or *full*. The valid semaphore operations are wait () and signal ().

Producer:	Consumer:

do{
Wait (P) ;
Wait (mutex);
//Add item ot buffer
signal (mutex);
signal (Q);
} while (1) ;
	do{
	wait (R);
	Wait (mutex) ;
	//Consume item from buffer
	signal (mutex);
	signal (S);
	}while (1);

Which one of the following assignments to P, Q, R and S will yield the correct solution?

(a) P : *full*, Q : *full*, R : *empty*, S : *empty*

(b) P : *empty*, Q : *empty* : R : *full*, S : *full*

(c) P : full, Q : *empty*, R : *empty* , S : *full*

(d) P :*empty*, Q : *full*, R : *full*, S : *empty*

❑❑❑

ANSWER KEY

1. ***Ans. (d)***

   A critical region is a program segment where shared resources are accessed.

2. ***Ans. (b)***

   Initially semaphore value is 7 then apply down and up operation. So first 20 down so –13, then 15 up so at last 2.

3. ***Ans. (c)***

   Critical section is an area in a progam where the shared resources are available.

4. ***Ans. (d)***

   Everytime when process P_{10} enters it performs the 'UP' operation so after that one more process can enter into it, so this phenomena continue fill end. So 10 processes at a time can enter into critical section.

5. ***Ans. (b)***

6. ***Ans. (b)***

   Race condition.

7. ***Ans. (b)***

   $5 = 7 - 20 + x$

   $5 - 7 + 20 = x$

   $x = 18$

8. ***Ans. (b)***

9. ***Ans. (c)***

10. ***Ans. (d)***

11. ***Ans. (d)***

12. ***Ans. (c)***

13. ***Ans. (a)***

    Mutual exclusion but not progress. As per condition mutual exclusion is guaranteed but no one will finish its critical section so no progress will be there.

14. ***Ans. (d)***

    P1 has control of the critical section provided wants1 is true and wants2 is false. P2 has control of the critical section provided wants2 is true and wants1 is false. So if P1 has control it excludes P2 till it completes and vice versa, so mutual exclusion is ensured. This eliminates choice (A). (B) is false as the time spent by P1 and P2 in their critical sections is not controlled or bounded. (C) is not correct for one can easily see that P1 can use the resource, release it, use it again, release it and so on without P2 ever demanding it. (D) A deadlock can arise as the assignment to wants1 and wants2 is not done as an indivisible operation. So when wants1 is set to true at the same time wants2 can be set to true. This results in endless waiting.

    So (D) is the answer.

15. ***Ans. (b)***

16. ***Ans. (d)***

17. ***Ans. (d)***

    S_x and S_y are two binary semaphores if we assume P means wait and V means signal then for two process P_1 and P_2 we take alternate then there is no chance of dead lock.

18. ***Ans. (c)***

    S1: If readcount is 1 i.e. some reader is reading, wait on wrt so that no writer can write.

    S2: After readcount has been updated, signal on mutex is applied.

    S3: Wait on mutex is applied to update readcount.

    S4: If readcount is zero (no reader is reading), signal on wrt is applied to allow writer to write.

19. ***Ans. (a)***

    When $S_0 = 1$, $S_1 = 0$

    So firstly the value of S_1 and $S_2 = 0$, P_1 and P_2 Not execute. The value of $S_0 = 1$, so it execute, process P_0 now value of $S_0 = 0$

    And it print zero one time when it release (S_1) call then value of S_1, S_2 increase by 1. So P_1 execute. It makes value of S_1 again zero and S_0

= 1, after it release (S_2) make value of $S_2 = 0$ by applying wait and $S_0 = 2$. So again P_0 process execute and wait (S_0) means it decrease by one so it print zero and this process continue. So P_0 print zero at least twice.

20. ***Ans. (d)***

	X	W	Y	Z
1.	P(s)	P(s)	P(s)	P(s)
2.	R(x)	R(x)	R(x)	R(x)
3.	x = x + 1	x = x + 1	x = x + 2	x = x + 2
4.	Store to MM	Store to MM	Store to MM	Store to MM
5.	V(s)	V(s)	V(s)	V(s)

1. Start with X. P(s) ⇒ s = 1 read x = 0, x = x + 1 Now before store into the memory process X preempt to process Y.
2. Y will perform P(s) ⇒ s = 0, read x = 0
   x = x − 2 then store x = − 2 and V(s) ⇒ s = 1
3. Z will perform P(s) ⇒ s = 0, read x = − 2
   x = x − 2
4. Now process x will store value of variable x into the MM so x = 1 and V(s) ⇒ s = 2
5. Now process W perform P(s) ⇒ s=1 read x=1
   x = x + 1, store x = 2, V(s) ⇒ s = 2.
   Option (d) is correct.

21. ***Ans. (d)***

A reentrant lock is a particular type of mutual execusion device that may be locked multiple time by same process or threads, without causing a deadlock i,e reentrant lock owned by last thread sucessfully done locking, but not unlock it till now. Since in question non-reentrant lock is given, so process cannot own same lock multiple time & leads to deadlock if it tries to acquire the same lock again, so only 1 thread & 1 lock needed to make deadlock.

22. ***Ans (c)***

Given,

Empty = 0

Full = N

Mutex = 1

Since value ofempty semaphore is 0,so you can not wait empty semaphore in first attempt.

❑❑❑

CHAPTER 6

DEADLOCK AND CONCURRENCY

OBJECTIVE QUESTIONS

1. A computer system has 6 tape drives, with n process completing for them. Each process may need 3 tape drives. The maximum value of n for which the system is guaranteed to be deadlock free is.
   (a) 2 (b) 3
   (c) 4 (d) 1

2. Consider a system having m resources of the same type. These resources are shared by 3 processes A, B and C, which have peak demands of 3, 4 and 6 respectively. For what value of m deadlock will not occur?
   (a) 7 (b) 9
   (c) 10 (d) 13

3. An operating system contains 3 user processes each requiring 2 units of resource R. the minimum number of units of r such that no deadlocks will ever arise is.
   (a) 3 (b) 5
   (c) 4 (d) 6

4. A computer has six tape drives, with n processes competing for them. Each process may need two drives. What is the maximum value of n for the system to be deadlock free?
   (a) 6 (b) 5
   (c) 4 (d) 3

5. A system has 3 processes sharing 4 resources. If each process needs a maximum of 2 units, then
   (a) deadlock can never occur
   (b) deadlock may occur
   (c) deadlock has to occur
   (d) none of these

6. Consider a system having 'm' resources of the same type. These resources are shared by 3 processes A, B, C, which have peak time demands of 3, 4, 6 respectively. The minimum value of 'm' that ensures that deadlock will never occur is
   (a) 11 (b) 12
   (c) 13 (d) 14

7. ________ is the situation in which a process is waiting on another process,which is also waiting on another process ... which is waiting on the first process. None of the processes involved in this circular wait are making progress.
   (a) Deadlock (b) Starvation
   (c) Dormant (d) None of the above

8. An operating system contains 3 resource classes. The number of resource units in these classes is 7, 7 and 10. The current resource allocation state is shown below:

Processes	Allocated resources			Maximum requirements		
	R1	R2	R3	R1	R2	R3
P1	2	2	3	3	6	8
P2	2	0	3	4	3	3
P3	1	2	4	3	4	4

   (i) Is the current allocation state safe?
   (ii) Can the request made by process P1 (1, 1, 0) be granted?
   (a) (i) safe state (ii) no
   (b) (i) not safe state (ii) no
   (c) (i) safe state (ii) yes
   (d) (i) not safe state (ii) yes

9. What is the method of handling deadlocks?

(a) Use a protocol to ensure that the system will never enter a deadlock state.

(b) Allow the system to enter the deadlock state and then recover.

(c) Pretend that deadlocks never occur in the system.

(d) All of the Above

10. Suppose n processes, $P_1, ..., P_n$ share m identical resource units, which can be reserved and released one at a time. The maximum resource requirement of process P_i is s_p where $s_i > 0$. Which one of the following is a sufficient condition for ensuring that deadlock does not occur ?

(a) $\forall i, s_i < m$

(b) $\forall i, s_i < n$

(c) $\sum_{i=1}^{n} s_i < (m+n)$

(d) $\sum_{i=1}^{n} s_i < (m*n)$

11. Consider the following snapshot of a system running n processes. Process i is holding x_i instances of a resource R, for $1 \le i \le n$. Currently, all instances of R are occupied. Further, for all i, process i has placed a request for an additional y_i instances while holding the x_i instances it already has. There are exactly two processes p and q such that $y_p = y_q = 0$. Which one of the following can serve as a necessary condition to guarantee that the system is not approaching a deadlock ?

(a) $\min(x_p, x_q) < \max_{k \neq p, q} y_k$

(b) $x_p + x_q \geq \max_{k \neq p, q} y_k$

(c) $\min(x_p, x_q) < 1$

(d) $\min(x_p, x_q) > 1$

12. A system shares 9 tape drives. The current allocation and maximum requirement of tape drives for three processes are shown below:

Process	Current Allocation	Maximum Requirement
P1	3	7
P2	1	6
P3	3	5

Which of the following best describes current state of the system?

(a) Safe, Deadlocked

(b) Safe, Not Deadlocked

(c) Not Safe, Deadlocked

(d) Not Safe, Not Deadlocked

13. Consider the following snapshot of a system running n concurrent processes. Process i is holding X_i instances of a resource R, $1 \le i \ge n$. Assume that all instances of R are currently in use. Further, for all i, process i can place a request for at most Y_i additional instances of R while holding the X_i instances it already has. Of the n processes, there are exactly two processes p and q such that $Y_p = Y_q = 0$. Which one of the following conditions guarantees that no other process apart from p and q can complete execution?

(a) $\text{Min}(X_p, X_q) \le \text{Max}\{Y_k \mid 1 \le k \le n, k \neq p, k \neq q\}$

(b) $X_p + X_q < \text{Min}\{Y_k \mid 1 \le k \le n, k \neq p, k \neq q\}$

(c) $\text{Min}(X_p, X_q) \ge \text{Min}\{Y_k \mid 1 \le k \le n, k \neq p, k \neq q\}$

(d) $X_p + X_q < \text{Max}\{Y_k \mid 1 \le k \le n, k \neq p, k \neq q\}$

❑❑❑

ANSWER KEY

1. ***Ans. (a)***

   If there are 2 processes then each process will hold 3 tape drives as there are 6 tape drives for which the system is guaranteed to be deadlock free.

2. ***Ans. (d)***

   If anyone peak demand satisfied then we can't get deadlock.

3. ***Ans. (c)***

   There are 3 processes, and demands '2' resource. So if any one gets equal resources as it's demand. So deadlock won't occur, and according to this question 4 resources are enough.

4. ***Ans. (b)***

   Every process needs 2 resource so by 5 at least one can get whole 2 and executes. So after execution it frees 2 resources which can be used by other processes.

   P_1 P_2 P_3 P_4 P_5

   1 1 1 1 1

5. ***Ans. (a)***

   Atleast one process will be holding 2 resources in case of a simultaneous demand from all the processes. That process will release the 2 resources, thereby avoiding any possible deadlock.

6. ***Ans. (a)***

   Having 11 resources ensures that atleast 1 process will have no pending requiest. This process after using will release the resources and so deadlock can never occur.

7. ***Ans.(a)***

8. ***Ans. (a)***

9. ***Ans. (d)***

10. ***Ans. (c)***

    There are n processes P_1, P_2, ..., P_n.

    Number of identical resources = m

    The maximum resource requirements for a process P_i is S_p, where $S_i > 0$.

    If all processes are in safe state then system can allocate resources to each process P_i in some order and still avoid a deadlock. We can avoid the deadlock if each P_i can still request can be satisfied by the current available resources plus the resources held by process P_j such that $j < i$. The maximum resource requirement of n processes

    is $\sum_{i=1}^{n} S_i$ and this amount must less than m + n.

    $$\therefore \sum_{i=1}^{n} S_i < (m+n)$$

11. ***Ans. (b)***

    Total number of processes = n

    Process i is holding x_i instances of a resource R for $1 \le i \le n$.

    Process i holding resource x_i and additional required y_i. There are two processes p and q such that $y_p = y_q = 0$.

    For no dead lock can occur

    $$x_p + x_q \ge (\max_{k \ne p,q} y_k)$$

12. ***Ans. (b)***

Process	Current Allocation	Maximum Requirement	Current Need	Current Available
P1	3	7	4	9 – 7 = 2
P2	1	6	5	=5
P3	3	5	2	=8
	Total = 7			=9

Safe Seq^n: P3 $\rightarrow$ P1 $\rightarrow$ P2

with 2 available tape drives, current need of P3 can be fulfilled, so P3 can execute completly then after that P3 will release its allocated resources. Which will make total available drives to 5. After that P1 and P2 processes can finish in any order. All processes can complete, hence safe state and no deadlock.

13. *Ans. (b)*

Two processes p & q are doing their work because they require zero more instances of resource.

After Finishing they will release whatever they have i.e. $(X_p + X_q)$ and then other process can continue their execution.

Holding Requesting

$P_1 - X_1 - Y_1 = 0 \leftarrow$ Assume its p

$P_2 - X_2 - Y_2 = 0 \leftarrow$ Assume its p

$$\left.\begin{array}{l} P_3 - X_3 - Y_3 = 8 \\ P_4 - X_4 - Y_4 = 7 \\ P_5 - X_5 - Y_5 = 15 \\ P_6 - X_6 - Y_6 = 12 \end{array}\right\} \text{Lets take random value}$$

Now, the guarantee that, any other process (O_3 to P_6) will not complete execution, we must have less resources than the minimum request i.e., 7

So, if $(X_1 + X_2)$ is less than 7, no process will execute after P_1 & P_2.

❑❑❑

CHAPTER 7

MEMORY MANAGEMENT

OBJECTIVE QUESTIONS

Common Data (Q1. and Q. 2) :

Consider a logical address space of 8 pages of 1024 words mapped into memory of 32 frames.

1. How many bits are there in the logical address?
   (a) 9 bits (b) 11 bits
   (c) 13 bits (d) 15 bits

2. How many bits are there in the physical address?
   (a) 9 bits (b) 11 bits
   (c) 13 bits (d) 15 bits

3. Dirty bit is used to show the
   (a) page with corrupted data
   (b) the wrong page in the memory
   (c) page that is modified after being loaded into cache memory
   (d) page that is less frequently accessed

4. In a paged memory, the page hit ratio is 0.35. The time required to access a page in secondary memory is equal to 100 ns. The time required to access a page in primary memory is 10 ns. The average time required to access a page is
   (a) 3.0 ns (b) 68.0 ns
   (c) 68.5 ns (d) 78.5 ns

5. The principle of locality justifies the use of
   (a) Interrupts (b) DMA
   (c) Polling (d) Cache Memory

6. In real-time opearting systems, which of the following is the most suitable scheduling scheme?
   (a) round-robin
   (b) first-come-first-served
   (c) preemptive
   (d) random scheduling

7. The capacity of a memory unit is defined by the number of words multiplied by the number of bits/word.
   How many separate address and data lines are needed for a memory of 4K × 16?
   (a) 10 address, 16 data lines
   (b) 11 address, 8 data lines
   (c) 12 address, 16 data lines
   (d) 12 address, 12 data lines

8. A 1000 Kbyte memory is managed using variable partitions but to compaction. It currently has two partitions of sizes 200 Kbytes and 260 Kbytes respectively. The smallest allocation request in Kbytes that could be denied is for.
   (a) 151 (b) 181
   (c) 231 (d) 541

9. If there are 32 segments, each of size 1 K byte, then the logical address should have
   (a) 13 bits (b) 14 bits
   (c) 15 bits (d) 16 bits

10. Thrashing
    (a) Reduces page I/O
    (b) Decreases the degree of multiprogramming
    (c) Implies excessive page I/O
    (d) Improve the system performance

11. Fixed partitions
    (a) are very common in current operating systems
    (b) are very efficient in memory utilization
    (c) are very inefficient in memory utilization
    (d) are most used on large mainframe operating systems

12. Which of the following statements is false?

(a) Segmentation suffers from external fragmentation.

(b) Paging suffers from internal fragmentation.

(c) Segmented memory can be paged.

(d) Virtual memory is used only in multi-user systems.

13. Fragmentation of the file system

(a) not occurs only if the file system is used improperly

(b) can always be prevented

(c) can be temporarily removed by compaction

(d) is a characteristic of all file systems

14. In memory management , a technique called as paging, physical memory is broken into fixed-sized blocks called ___________.

(a) Pages (b) Frames

(c) Blocks (d) Segments

15. What hole will allocates in "Worst-Fit" algorithm of memory management?

(a) It allocates the smaller hole than required memory hole

(b) It allocates the smallest hole from the available memory holes

(c) It allocates the largest hole from the available memory holes

(d) It allocates the exact same size memory hole

16. A page Faults when

(a) the deadlock happens

(b) the segmentation starts

(c) the page found in the memory

(d) the page is not found in the memory

17. Memory management is

(a) not used in modern operating system

(b) replaced with virtual memory on current system

(c) not used on multiprogramming systems

(d) critical for even the simplest operating system

18. Which of the following memory allocation scheme suffers from External fragmentation?

(a) Segmentation

(b) Pure demand paging

(c) Swapping

(d) Paging

19. MBR stands for

(a) Memory Buffer Bounded Register

(b) Memory Bounded Register

(c) Memory Buffered Register

(d) None of these

20. Paging _________.

(a) Solves the memory fragmentation problem

(b) Allows modular programming

(c) Allows structured programming

(d) Avoids deadlock

21. In a system with 32 bit virtual addresses and 1KB page size, use of one-level page tables for virtual to physical address translation is not practical because of

(a) the large amount of internal fragmentation

(b) the large amount of external fragmentation

(c) the large memory overhead in maintaining page tables

(d) the large computation overhead int the translation process

22. The memory allocation scheme subject to "external" fragmentation is

(a) segmentation

(b) swapping

(c) pure demand paging

(d) multiple fixed contiguous partitions

23. A page fault occurs

(a) when the page is not in the memory

(b) when the page is in the memory

(c) when the process enters the blocked state

(d) when the process is in the ready state

24. A page fault means that we referenced a page

(a) outside the memory boundaries

(b) with an incorrect I/O request

(c) that was not in secondary storage

(d) that was not in main memory

25. Consider a machine with 64 MB physical memory and a 32-bit virtual address space. If the page size of 4 KB, what is the approximate size of the page table?

(a) 16 MB (b) 8 MB

(c) 2 MB (d) 24 MB

26. Consider a system with a two-level paging scheme in which a regular memory access takes 150 nanoseconds and servicing a page fault takes 8 milliseconds. An average instruction takes 100 nanoseconds of CPU time and two memory accesses. The TLB hit ratio is 90% and the page fault rate is one in every 10,000 instructions. What is the effective average instruction execution time?

(a) 645 nanoseconds

(b) 1050 nanoseconds

(c) 1215 nanoseconds

(d) 1230 nanoseconds

27. A CPU generates 32-bit virtual addresses. The page size is 4KB. 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

(a) 11 bits (b) 13 bits

(c) 15 bits (d) 20 bits

28. A processor uses 36 bit physical addresses and 32 bit virtual addresses, with a page frame size of 4 Kbytes. Each page table entry is of size 4 bytes. A three level page table is used for virtual-to-physical address translation, where the virtual address is used as follows

- bits 30-31 are used to index into the first level page table,
- bits 21-29 are used to index into the second level page table
- bits 12-20 are used to index into the third level page table
- bits 0-11 are used as offset within the page

The number of bits required for addressing the next level page table (or page frame) in the page table entry of the first, second and third level page tables are respectively.

(a) 20, 20 and 20

(b) 24, 24 and 24

(c) 24, 24 and 20

(d) 25, 25 and 24

❑❑❑

ANSWER KEY

1. *Ans. (c)*

Logical address will have 3 bits to specify the page number (for 8 pages).

10 bits to specify the offset into each page (2^{10} = 1024 words)

= 13 bits.

2. *Ans. (d)*
3. *Ans. (c)*
4. *Ans. (c)*

$0.35 \times 10 + (1 - 0.35) \times 100 = 68.5$ ns

5. *Ans. (d)*
6. *Ans. (a)*
7. *Ans. (c)*
8. *Ans. (b)*
9. *Ans. (c)*

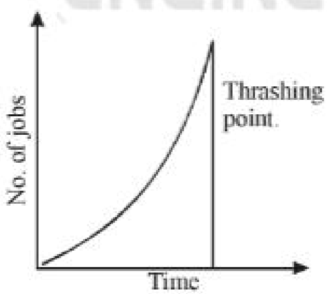
To specify a particular segment, 5 bits are required (since $2^5 = 32$).

Having selected a page, to select a particular byte one needs 10 bits (since 2^{10} = 1 k byte).

So, totally 5 + 10 = 15 bits are needed.

10. *Ans. (c)*

Thrashing : After reaching the maximum no. of jobs allowed in main memory. CPU efficiency drastically getting down this situation is called Thrashing.



Thrashing Implies excessive page I/O

11. *Ans. (c)*
12. *Ans. (d)*
13. *Ans. (c)*
14. *Ans. (b)*
15. *Ans. (c)*
16. *Ans. (d)*
17. *Ans: (b)*
18. *Ans.(a)*
19. *Ans. (c)*
20. *Ans. (a)*
21. *Ans. (c)*
22. *Ans. (a)*
23. *Ans. (a)*
24. *Ans. (d)*
25. *Ans. (c)*

Number of entries in page table $= \frac{2^{32}}{2^{12}} = 2^{20}$

Frame size $= \frac{2^{26}}{2^{12}} = 2^{14}$

$\therefore$ PT have to be stored in one frame so entry size must be 2 bytes, hence size of PT

$= 2^{20} \times 2 = 2^{21} = 2$ MB

26. *Ans. (d)*

Page fault rate $P = 1/10^4$

Effective memory access time (EMAT)

$= 0.9 \times 150$ ns $+ 0.1 \times [150 + 150] = 165$ ns

Average instruction execution time = CPU time

$+ P \times S + (1 - P)$ EMA

$= 100 + 10^{-4} \times 8$ ms $+ (1 - 10^{-4}) \times 2 \times 165$ ns

$= 1229.997 \simeq 1230$ nss

27. *Ans. (c)*

Size of virtual address = 32 bit

32 bit

No. of Page	Page Offset
$32 - 12 = 20$ bit	$\log_2$(Page Size) $= \log_2(4\text{ kB}) = \log_2(2^{12}\text{ Byte}) = 12$ bit

$\Downarrow$

Apply Set Associative TLB Mapping

$\Downarrow$

20bit

Tag	No set in TLB
$20 - 5 = 15$ bit	$\log_2(32) = 5$ bit

No. of set in TLB

$$= \frac{\text{No. of entry in TLB}}{\text{Associativity}}$$

$$= \frac{128}{4} = 32$$

Tag bit = 15 bit

28. ***Ans. (d)***

Virtual Address = 32 bit

Page size = 4 KByte

According to problem 3 level page table

P_1	P_2	P_3	Page Offset
32 bit	9 bit	9 bit	$\log_2$(Page Size) $\log_2$(4 KByte) $\log_2(2^{12}) = 12$ bit

$$\text{No of frame} = \frac{\text{Physical address space}}{\text{Frame size}}$$

$$= \frac{2^{36}\text{Byte}}{2^2 \times 2^{10}\text{ Byte}} = \frac{2^{36}}{2^{12}} = 2^{14}$$

Physical address = 36 bit

Page size = 4 KByte

or frame size

Third Level

0	14 bit
⋮	
$2^{20} - 1$	

No of Page

$\Rightarrow$ 18 bit other bit (Entry size = 4 Byte)

$\Rightarrow$ No. of bit [Frame] in First Level Page Table = 14 Bit

Second Level

4 Byte

0	
1	
2	
⋮	
$2^9 - 1$	

$$\text{No. of Frame} = \frac{\text{Physical Address Space}}{\text{Page Table Size Third Level}}$$

$$= \frac{2^{36}\text{ Byte}}{2^9 \times 2^2\text{ Byte}} = \frac{2^{36}\text{ Byte}}{2^{11}\text{ Byte}}$$

$= 2^{25}$ Byte = 25 bit

Second level no. of bit in (Frame) = 25 bit

First Level

Page table size (second level)

= No. of entries in page table * page table size

$= 2^9 * 2^2$ Byte $= 2^{11}$ Byte

$$\text{No. of Frame} = \frac{\text{Physical address space}}{\text{Page table size second level}}$$

$$= \frac{2^{36}\text{ Byte}}{2^{11}\text{ Byte}} = 2^{25}$$

First level no. of bit in (Frame) = 25 bit

= (25, 25, 24) bit

CHAPTER 8

VIRTUAL MEMORY

OBJECTIVE QUESTIONS

1. Determine the number of page faults when references to pages occur in the following order: 1, 2, 4, 5, 2, 1, 2, 4. Assume that the main memory can accommodate 3 pages and the main memory already has the pages 1 and 2, with page 1 having been brought earlier than page 2.

   (LRU algorithm is used)

   (a) 3 (b) 5

   (c) 4 (d) none of these

2. The first-fit, best-fit and the worst-fit algorithm can be used for

   (a) contiguous allocation of memory

   (b) linked allocation of memory

   (c) indexed allocation of memory

   (d) all of these

3. The total size of address space in a virtual memory systems is limited by

   (a) the length of MAR

   (b) the available secondary storage

   (c) the available main memory

   (d) All of the above

4. A memory page containing a heavily used variable that was initialized very early and is in constant use is removed when.

   (a) LRU page replacement algorithm is used

   (b) FIFO page replacement algorithm is used

   (c) LFU page replacement algorithm is used

   (d) None of these above

Common Data Q.5 and 6

A process refers to 5 pages, A, B, C, D and E in the following order A; B; C; D; A; B; E; A; B; C; D; E.

5. If the page replacement algorithm is FIFO, the number of page transfer with an empty internal store of 3 frames is

   (a) 10 (b) 9

   (c) 8 (d) 7

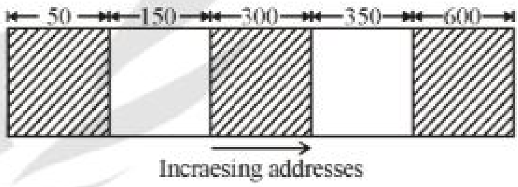
6. If the number of available page frames is increased to 4, then the number of page transfers

   (a) decreases (b) increases

   (c) remains same (d) none of these

7. Consider the following heap (figure) in which blank regions are not in use and hatched regions are in use.

   The sequence of requests for blocks of size 300, 25, 125, 50 can be satisfied if we use



   (a) either first fit or best fit policy (any one)

   (b) first fit but not best fit policy

   (c) best fit but not first fit policy

   (d) none of the above

8. The address sequence generated by tracing a particular program executing in a pure demand paging system with 100 records per page with 1 free main memory frame is recorded as follows. What is the number of page faults?

   0100, 0200, 0430, 0499, 0510, 0530, 0560, 0120, 0220, 0240, 0260, 0320, 0370.

   (a) 13 (b) 8

   (c) 7 (d) 10

9. In a virtual memory system the address space specified by the address lines of the CPU must be _____ than the physical memory size and ____ than the secondary storage size.

(a) smaller, smaller (b) smaller, larger

(c) larger, smaller (d) larger, larger

10. If an instruction takes i microseconds and a page fault takes an additional j microseconds, the effective instruction time if on the average a page fault occurs every k instruction is.

(a) $i + \frac{j}{k}$ (b) $i + j * k$

(c) $\frac{i+j}{k}$ (d) $(i + j) * k$

11. Which of the following is/are advantage of virtual memory?

(a) Faster access to memory on an average.

(b) Processes can be given protected address spaces.

(c) Linker can assign addresses independent of where the program will be loaded in physical memory.

(d) Progams larger than the physical memory size can be run.

12. A system uses FIFO policy for page replacement. It has 4 page frames with no pages loaded to begin with. The system first accesses 100 distinct pages in some order and then accesses the same 100 pages but noe in the reverse order. How many page faults will occur?

(a) 196 (b) 192

(c) 197 (d) 195

Common data 13 and 14

Consider following page trace :

4, 3, 2, 1, 4, 3, 5, 4, 3, 2, 1, 5

Number of page faults that would occur if FIFO page replacement algorithm is used with

13. Number of page faults for the Job M = 3, will be

(a) 8 (b) 9

(c) 10 (d) 12

14. Number of page fault for the Job M = 4, will be

(a) 8 (b) 9

(c) 10 (d) 12

15. What does Belady's Anomaly related to?

(a) Page Replacement Algorithm

(b) Memory Management Algorithm

(c) Deadlock Prevention Algorithm

(d) Disk Scheduling Algorithm

16. The mechanism that bring a page into memory only when it is needed is called _____

(a) Segmentation

(b) Fragmentation

(c) Demand Paging

(d) Page Replacement

17. Virtual Memory is commonly implemented by __________.

(a) Segmentation

(b) Swapping

(c) Demand Paging

(d) None of the above

18. Which amongst the following is not a valid page replacement policy?

(a) LRU policy (Least Recently Used)

(b) FIFO policy (First in first out)

(c) RU policy (Recurrently used)

(d) Optimal page replacement policy

19. Virtual memory is __________.

(a) An extremely large main memory

(b) An extremely large secondary memory

(c) An illusion of extremely large main memory

(d) A type of memory used in super computers.

❑❑❑

ANSWER KEY

1. ***Ans. (c)***
2. ***Ans. (a)***
3. ***Ans. (b)***

   Since virtual memory is implemented on secondary storage.
4. ***Ans. (b)***

   First in First out.
5. ***Ans. (b)***

   The first 3 references A, B, C fills the internal storage with A, B, C in 3 page transfers. Now next reference D results in a page fault. So, page A is downloaded and D takes its place, after a page transfer.

   So, a page transfer takes place and swaps B and A. Continuing this way, we find total 9 page transfers are necessary.
6. ***Ans. (b)***

   Applying the same logic, we find required number of page transfer is 10. So, increasing the number of pages need not necessarily reduce of the number of page faults. It is the actual sequences of references that decides.
7. ***Ans. (b)***
8. ***Ans. (c)***
9. ***Ans. (c)***
10. ***Ans. (a)***
11. ***Ans. (d)***
12. ***Ans. (a)***

    FIFO policy for page replacement used.

    Access 100 distinct pages by taking some example 2 3 4 5 6 7 8 9

    So by loading it get

    $$\begin{array}{|c|}\hline 5\\\hline 4\\\hline 3\\\hline 2\\\hline\end{array} = 4 \text{ page fault}, \quad \begin{array}{|c|}\hline 9\\\hline 8\\\hline 7\\\hline 6\\\hline\end{array} = 4 \text{ page fault}$$

    and now access these page in reverse so

    $$\underbrace{9\ 8\ 7\ 6}_{\text{So no page fault for these}} \quad \underbrace{5\ 4\ 3\ 2}_{\text{page fault for these}}$$

    $$\begin{array}{|c|}\hline 9\\\hline 8\\\hline 7\\\hline 6\\\hline\end{array} \quad \begin{array}{|c|}\hline 5\\\hline 4\\\hline 3\\\hline 2\\\hline\end{array} \ 4 \text{ pagefault}$$

    So total = 4 + 4 + 4 = 12 pagefault

    for 8 pages = 2 × 8 – 4 = 12

    so for n pages = 2n – 4

    so for 100 pages = 2 × (100) – 4 = 196
13. ***Ans. (b)***
14. ***Ans. (c)***
15. ***Ans. (a)***
16. ***Ans. (c)***
17. ***Ans. (c)***
18. ***Ans. (c)***
19. ***Ans. (c)***

❑❑❑

CHAPTER 9

FILE MANAGEMENT SYSTEM

OBJECTIVE QUESTIONS

1. Which of the following is the allocation method of a disk space?

(a) Contiguous allocation

(b) Linked allocation

(c) Indexed allocation

(d) All of the Above

2 Routine is not loaded until it is called. All routines are kept on disk in a relocatable load format. The main program is loaded into memory & is executed. This type of loading is called ________

(a) Static loading (b) Dynamic loading

(c) Dynamic linking (d) Overlays

3 Which file system does DOS typically use ?

(a) FAT16 (b) FAT32

(c) NTFS (d) WNFS

4. Which file system does Windows 95 typically use?

(a) FAT16 (b) FAT32

(c) NTFS (d) LMFS

5. __________ allocates the largest hole (free fragmant) available in the memory.

(a) Best Fit (b) Worst Fit

(c) First Fit (d) None of the above

6. Which directory implementation is used in most Operating System?

(a) Single level directory structure

(b) Two level directory structure

(c) Tree directory structure

(d) Acyclic directory structure

7. A 1000 Kbyte memory is managed using variable partitions but to compaction. It currently has two partitions of sizes 200 Kbytes and 260 Kbytes respectively. The smallest allocation request in Kbytes that could be denied is for

(a) 151 (b)181

(c) 231 (d)541

8. Determine the number of page faults when references to page occur in the following order: 1,2,4,5,2,1,2,4. assume that the main memory can accommodate 3 pages and the main memory already has the pages 1 and 2, with page 1 having been brought earlier than page 2. (LRU algorithm is used)

(a) 3 (b) 5

(c) 4 (d) none of these.

9. The page replacement policy that sometimes leads to more page faults when the size of the memory is increased is

(a) FIFO

(b) LRU

(c) no such policy exists

(d) none of the above.

10. I/O redirection

(a) implies changing the name of a file

(b) can be employed to use an existing file as input file for a program

(c) implies connecting 2 programs through a pipe

(d) none of the above

11. Which of the following disk scheduling strategies is likely to give the best throughput?

(a) Farthest cylinder next

(b) Nearest cylinder next

(c) First come first served

(d) Elevator algorithm

12. In the index allocation scheme of blocks to a file, the maximum possible size of the file depends on

(a) the size of the blocks, and the size of the address of the blocks.

(b) the number of blocks used for the index, and the size of the blocks

(c) the size of the blocks, the number of blocks used for the index, and the size of the address of the blocks

(d) none of the above

13. A Unix-style I-nod has 10 direct pointers and one single, one double and one triple indirect pointers. Disk block size is 1 Kbyte, disk lock address is 32 bits, and 48-bit integers are used. What is the maximum possible file size?

(a) 2^{24} bytes (b) 2^{32} bytes

(c) 2^{34} bytes (d) 2^{48} bytes

14. In the working-set strategy, which of the following is done by the operating system to prevent thrashing?

I. It initiates another process if there are enough extra frames.

II. It selects a process to suspend if the sum of the sizes of the working-sets.

III. Execeeds the total number of available frames.

(a) I only

(b) II only

(c) Neither I nor II

(d) Both I and II

15. A virtual memory system uses First In First Out (FIFO) page replacement policy and allocates a fixed number of frames to a process. Consider the following statements.

P : Increasing the number of page frames allocated to a process sometimes increases the page fault rate.

Q : Some programs do not exhibit locality of reference.

Which one of the following is TRUE?

(a) Both P and Q are true, and Q is the reason for P

(b) Both P and Q are true, but Q is not the reason for P

(c) P is false, but Q is true

(d) Both P and Q are false

16. Consider a disk system with 100 cylinders. The requests to access the cylinders occur in following sequence:

4, 34, 10, 7, 19, 73, 2, 15, 6, 20

Assuming that the head is currently at cylinder 50, what is the time taken to satisfy all requests if it takes 1 ms to move from one cylinder to adjacent one and shortest seek time first policy is used?

(a) 95 ms (b) 119 ms

(c) 233 ms (d) 276 ms

17. A multilevel page table is preferred in comparison to a single level page table for translating virtual address to physical address because

(a) it reduces the memory access time to read or write a memory location

(b) it helps to reduce the size of page table needed to implement the virtual address space of a process

(c) it is required by the translation look a side buffer

(d) it helps to reduce the number of page faults in page replacement algorithms.

18. A file system with 300 GByte disk uses a file descriptor with 8 direct block addresses, 1 indirect block address and 1 doubly indirect block address. The size of each disk block address is 128 Bytes and the size of each disk block address is 8 Bytes. The maximum possible file size in this file system is

(a) 3 KBytes

(b) 35 KBytes

(c) 280 KBytes

(d) dependent on the size of the disk

❑❑❑

Answer Key

1. *Ans. (d)*
2. *Ans. (b)*
3. *Ans. (a)*
4. *Ans. (b)*
5. *Ans. (b)*
6. *Ans. (c)*
7. *Ans.(b)*

Memory size of variable partition are 200 Kbytes and 260 Kbytes

So, total memory left = 1000(200+260) = 540.

So, 181 Kbytes smallest could be denied from request.

8. *Ans. (c)*
9. *Ans. (a)*
10. *Ans. (b)*

I/O redirection helps to use an existing file as input file for a program.

11. *Ans. (b)*

Shortest seek time next or nearest cylinder next gives the best throughout.

12. *Ans. (c)*

$\frac{\text{DB size}}{\text{DBA}}$ = Number of DBA's possible in one disk block.

13. *Ans. (c)*

The number of disk block pointers that will be fit in one block

$$= \frac{2^{10}\text{byte}}{32\text{bit}} = \frac{2^{10}\text{byte}}{4\,\text{byte}} = 256$$

Maximum file size due to single indirection pointer

$= 256 \times 1 = 256$ KB

Maximum file size due to direct pointer

$= 10 \times 1 = 10$ KB

Maximum file size due to double indirection pointer

$= 256 \times 256 \times 1$ KB

Maximum file size due to trip indirection pointer

$= 256 \times 256 \times 256 \times 1$ KB

So the maximum file size

$= 256 \times 256 \times 256 \times 1$ KB

$= 2^8 \times 2^8 \times 2^8 \times 2^{10}$ bytes

$= 2^{24} \times 10^{10}$ bytes

$= 2^{34}$ bytes

14. *Ans. (b)*

If there are enough extra frames, another process can be initiated. If the sum of the working-set sizes increases, exceeding the total number of available frames, the operating system selects a process to suspend. The process's pages are written out (swapped), and its frames are reallocated to other processes. The suspended process can be restarted later."

15. *Ans. (b)*

P and Q both are true but Q is not the reason for P increasing the number of Page frame allocated to a process sometimes increase the page fault rate or it is not concern with replacement policy.

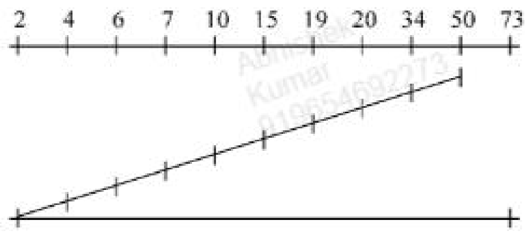
It is easy to write a program which jump around a lot and which donot Exhibit locality of reference.

Assume that array is stored in row major order and we are accessing column major order does not follow locality of reference.

P and Q statements boths are independent.

16. *Ans. (b)*

Head is currently at cylinder 50,



50 – 34 = 16

34 – 20 = 14

20 – 19 = 1

19 – 15 = 4

15 – 10 = 5

10 – 7 = 3

7 – 6 = 1

6 – 4 = 2

4 – 2 = 2

2 – 73 = 71

Total moves

= 16 + 14 + 1 + 4 + 5 + 3 + 1 + 2 + 2 + 71

= 119

17. ***Ans. (b)***

Multilevel page table in preferred to reduce the size of page table needed to implement the virtual address space of a process.

18. ***Ans. (b)***

Maximum possible size

$$= \left[\begin{pmatrix}\text{address pointed by}\\ \text{doubly indirect block}\end{pmatrix}^2\right.$$

$$+\begin{pmatrix}\text{address pointed by}\\ \text{single address}\end{pmatrix}$$

+ Address points by single direct address]

× [block size]

$$= \left[\left(\frac{128}{8}\right)^2 + \left(\frac{128}{8}\right) + 8\right] \times 128\text{B}$$

$= [2^8 + 2^4 + 2^3] \times 128\text{B}$

= 32KB + 2KB + 1KB

= 35 KB

❑❑❑

CHAPTER 10

DISK MANAGEMENT

OBJECTIVE QUESTIONS

1. Disk requests come to disk driver for cyliners 10, 22, 20, 2, 40, 56 and 38 in that order at a time when the disk drive is reading from cylinder 20. The seek time is 6 msec per cylinder. Compute the total seek time if the disk arm scheduling algorithm is.

(a) First come first served

(b) Closest cylinder next

(c) Both (a) and (b)

(d) None of these

2. The correct matching for the following pairs is.

A. Disk scheduling 1. Round robin

B. Batch processing 2. SCAN

C. Time sharing 3. LIFO

D. Interrupt processing 4. FIFO

(a) A-3, B-4, C-2, D-1

(b) A-4, B-3, C-2, D-1

(c) A-2, B-4, C-1, D-3

(d) A-3, B-4, C-1, D-2

3. When an interrupt occurs, an operating system.

(a) ignores the interrupt

(b) always changes state of interrupted process after processing the interrupt

(c) always resumes execution of interrupted process after processing the interrupt

(d) may change state of interrupted process to 'blocked' and schedule another process.

4. On receiving an interrupt from an I/O device, the CPU

(a) halts for a predetermined time

(b) branches off to the interrupt service routine after completion of the current instruction

(c) branches off to the interrupt service routine immediately

(d) hands over control of address bus and data bus to the interrupting device

5. Which of the following disk scheduling strategies is likely to give the best throughput ?

(a) Farthest cylinder next

(b) Nearest cylinder next

(c) First come first served

(d) Elevator algorithm

6. Disk scheduling involves deciding

(a) Which disk should be assessed next

(b) The order in which disk access requests must be serviced

(c) The physical location where files should be accessed in the disk

(d) None of these

Common Data 7 and 8

Disk requests come to a disk driver for cylinders 10, 22, 20, 2, 40, 6 and 38, in that order at a time when the disk drive is reading from cylinder 20. The seek time is 6 ms per cylinder.

7. The total seek time, if the disk arm scheduling algorithm is first-come-first-served is

(a) 360 ms (b) 876 ms

(c) 850 ms (d) 900 ms

8. If the scheduling algorithm is the closest cylinder next, then the total seek time will be
   (a) 360 ms (b) 876 ms
   (c) 850 ms (d) 900 ms

9. Consider a disk system with 100 cylinders. The requests to access the cylinders occur in following sequence:

   4, 34, 10, 7, 19, 73, 2, 15, 6, 20

   Assuming that the head is currently at cylinder 50, what is the time taken to satisfy all requests if it takes 1 ms to move from one cylinder to adjacent one and shortest seek time first policy is used?
   (a) 95 ms (b) 119 ms
   (c) 233 ms (d) 276 ms

10. Access time is the highest in the case of
    (a) Floppy disk
    (b) Cache
    (c) Swapping devices
    (d) Magnetic disks

11. SSTF stands for
    (a) Shortest-Seek-time-first scheduling
    (b) small – small-time-first
    (c) simple-seek-time-first
    (d) small-simple-time-first scheduling

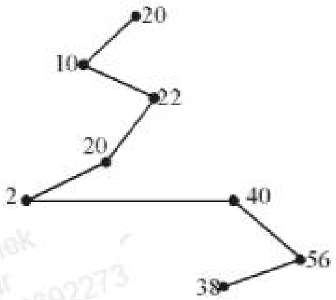
12. Consider the situation in which the disk read/write head is currently located at track 45 (of tracks 0-255) and moving in the positive direction. Assume that the following track requests have been made in this order: 40, 67, 11, 240, 87. What is the order in which optimized C-SCAN would service these requests and what is the total seek distance?
    (a) 505 (b) 510
    (c) 500 (d) 515

13. If the Disk head is located initially at 32, find the number of disk moves required with FCFS if the disk queue of I/O blocks requests are 98,37,14,124,65,67.
    (a) 310 (b) 324
    (c) 315 (d) 321

❑❑❑

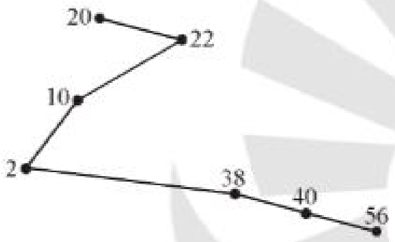
Answer Key

1. *(a)* FCFS



$6 \times (10 + 12 + 2 + 18 + 38 + 16 + 18) = \mathbf{684}$

(b) Closest cylinder next



$6 \times (2 + 12 + 8 + 36 + 2 + 16) = \mathbf{456}$

2. *Ans. (c)*
3. *Ans. (c)*
4. *Ans. (b)*
5. *Ans. (b)*
6. *Ans. (b)*
7. *Ans. (b)*
8. *Ans. (a)*
9. *Ans. (b)*
10. *Ans. (d)*
11. *Ans. (a)*
12. *Ans. (a)*
13. *Ans. (d)*

❑❑❑