Operating Systems(CS-301)

u19cs012@coed.svnit.ac.in Switch account

Saving...

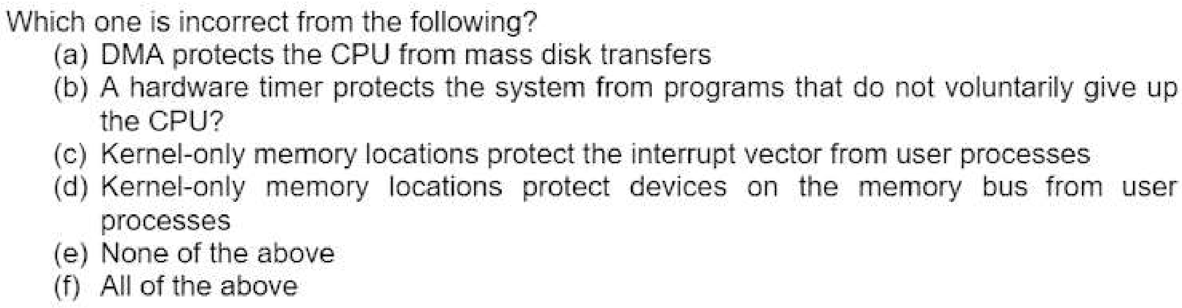
Your email will be recorded when you submit this form

* Required

OS(CS-301) END SEM. EXAM.-07-DEC.-2021

Total 80 MCQs, each of 01 mark. 90 Min.

*



Which one is incorrect from the following?

(a) DMA protects the CPU from mass disk transfers
(b) A hardware timer protects the system from programs that do not voluntarily give up the CPU?
(c) Kernel-only memory locations protect the interrupt vector from user processes
(d) Kernel-only memory locations protect devices on the memory bus from user processes
(e) None of the above
(f) All of the above

● A

○ B

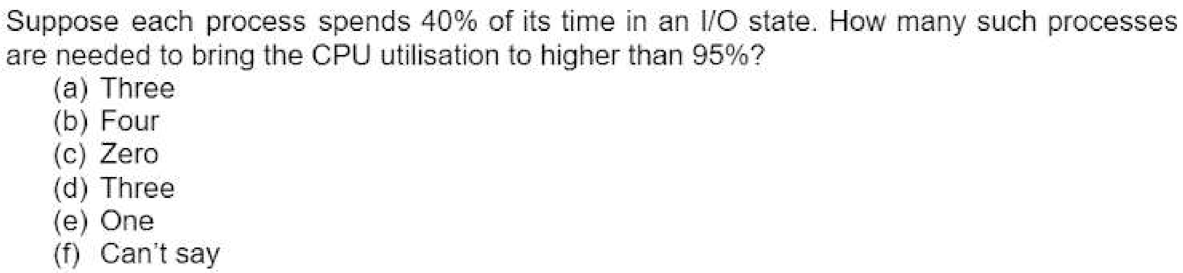
○ C

○ D

○ E

○ F

*



Suppose each process spends 40% of its time in an I/O state. How many such processes are needed to bring the CPU utilisation to higher than 95%?

(a) Three
(b) Four
(c) Zero
(d) Three
(e) One
(f) Can't say

○ A

● B

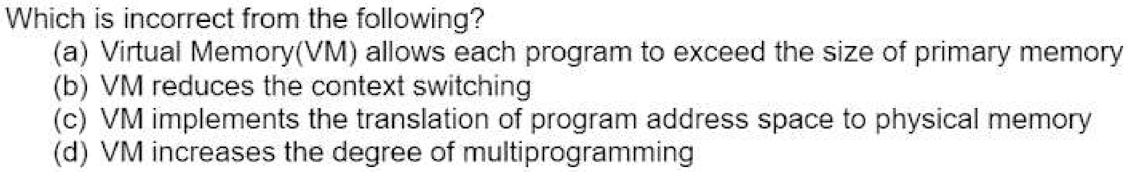
○ C

○ D

○ E

○ F

*



Which is incorrect from the following?

(a) Virtual Memory(VM) allows each program to exceed the size of primary memory
(b) VM reduces the context switching
(c) VM implements the translation of program address space to physical memory
(d) VM increases the degree of multiprogramming

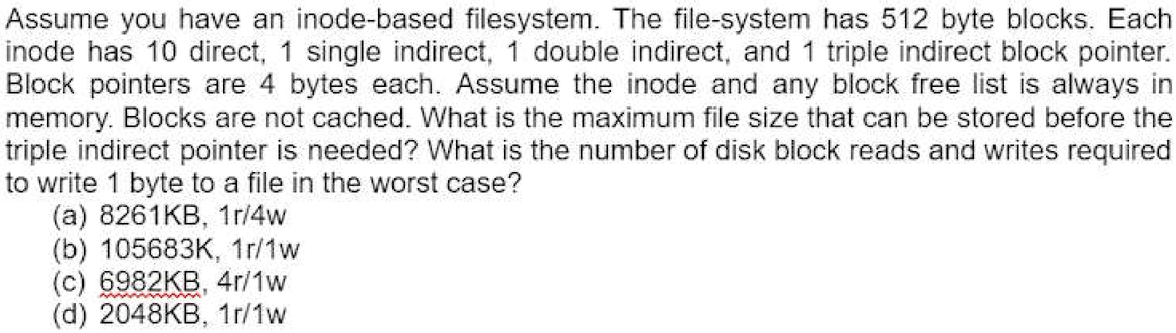
○ A

● B

○ C

○ D

*



Assume you have an inode-based filesystem. The file-system has 512 byte blocks. Each inode has 10 direct, 1 single indirect, 1 double indirect, and 1 triple indirect block pointer. Block pointers are 4 bytes each. Assume the inode and any block free list is always in memory. Blocks are not cached. What is the maximum file size that can be stored before the triple indirect pointer is needed? What is the number of disk block reads and writes required to write 1 byte to a file in the worst case?

(a) 8261KB, 1r/4w
(b) 105683K, 1r/1w
(c) 6982KB, 4r/1w
(d) 2048KB, 1r/1w

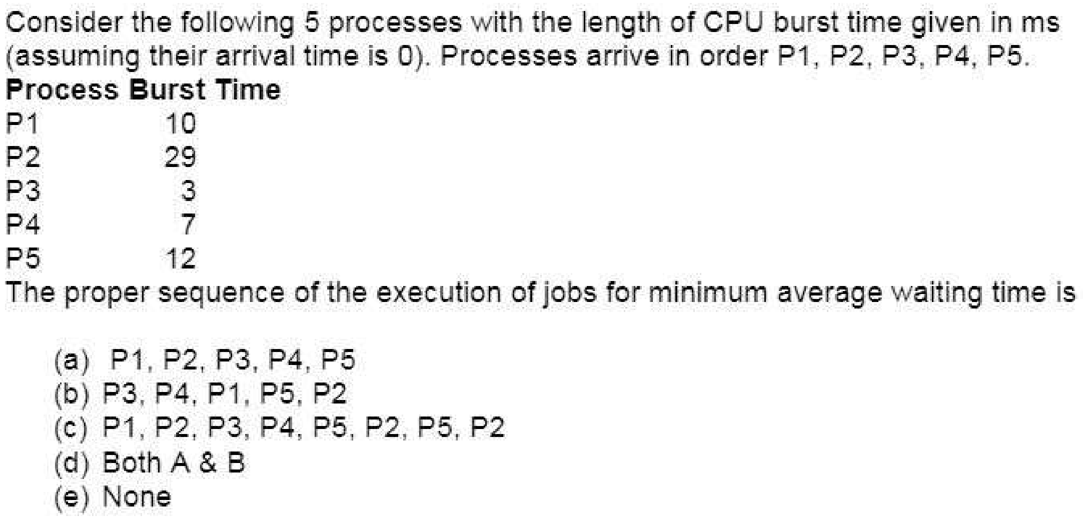
○ A

○ B

○ C

● D

*



Consider the following 5 processes with the length of CPU burst time given in ms (assuming their arrival time is 0). Processes arrive in order P1, P2, P3, P4, P5.

Process	**Burst Time**
P1	10
P2	29
P3	3
P4	7
P5	12

The proper sequence of the execution of jobs for minimum average waiting time is

(a) P1, P2, P3, P4, P5
(b) P3, P4, P1, P5, P2
(c) P1, P2, P3, P4, P5, P2, P5, P2
(d) Both A & B
(e) None

○ A

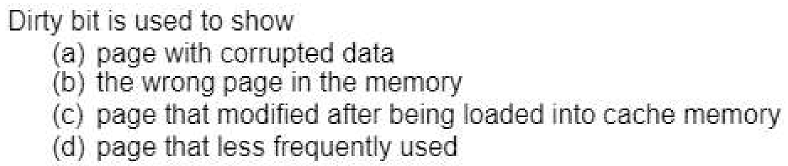
● B

○ C

○ D

○ E

*



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

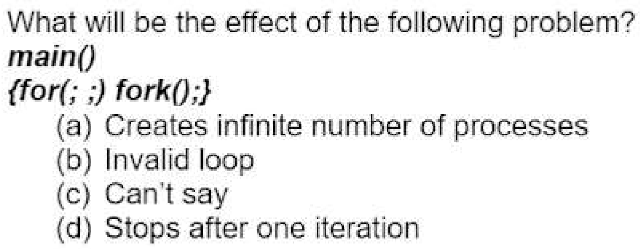
○ A

○ B

● C

○ D

*



What will be the effect of the following problem?
main()
{for(; ;) fork();}

(a) Creates infinite number of processes
(b) Invalid loop
(c) Can't say
(d) Stops after one iteration

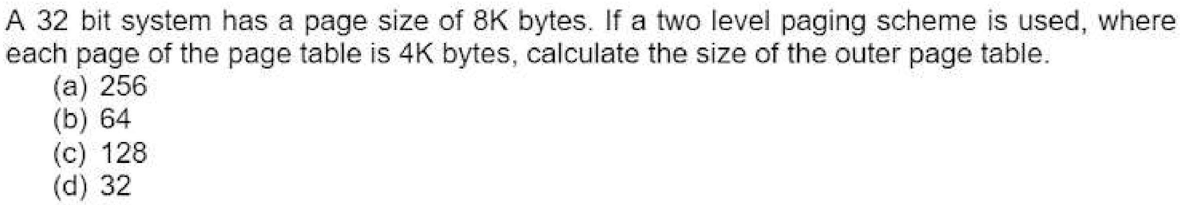
● A

○ B

○ C

○ D

*



A 32 bit system has a page size of 8K bytes. If a two level paging scheme is used, where each page of the page table is 4K bytes, calculate the size of the outer page table.

(a) 256
(b) 64
(c) 128
(d) 32

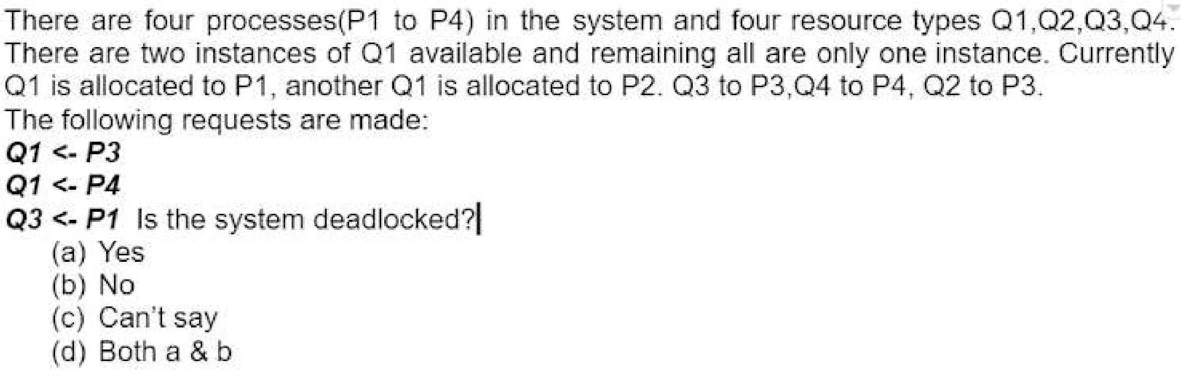
○ A

○ B

● C

○ D

*



There are four processes(P1 to P4) in the system and four resource types Q1,Q2,Q3,Q4. There are two instances of Q1 available and remaining all are only one instance. Currently Q1 is allocated to P1, another Q1 is allocated to P2. Q3 to P3,Q4 to P4, Q2 to P3.

The following requests are made:

Q1 <- P3

Q1 <- P4

Q3 <- P1 Is the system deadlocked?

(a) Yes
(b) No
(c) Can't say
(d) Both a & b

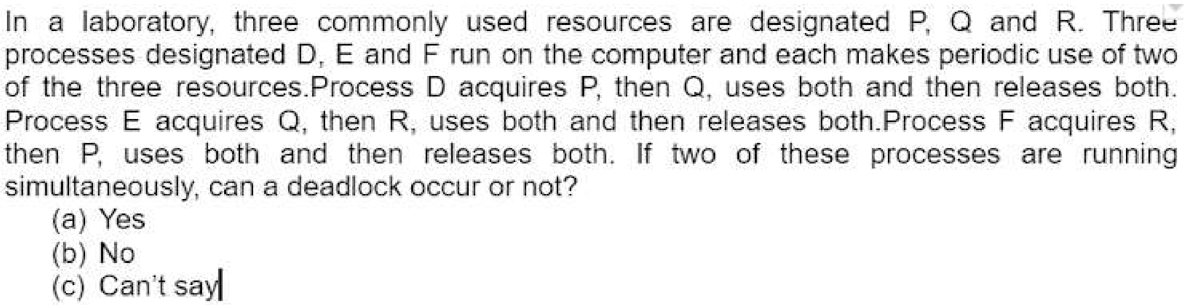
● A

○ B

○ C

○ D

*



In a laboratory, three commonly used resources are designated P, Q and R. Three processes designated D, E and F run on the computer and each makes periodic use of two of the three resources.Process D acquires P, then Q, uses both and then releases both. Process E acquires Q, then R, uses both and then releases both.Process F acquires R, then P, uses both and then releases both. If two of these processes are running simultaneously, can a deadlock occur or not?

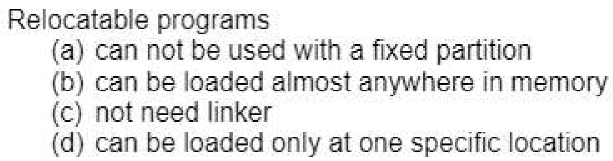
(a) Yes
(b) No
(c) Can't say

○ A

● B

○ C

*



Relocatable programs
(a) can not be used with a fixed partition
(b) can be loaded almost anywhere in memory
(c) not need linker
(d) can be loaded only at one specific location

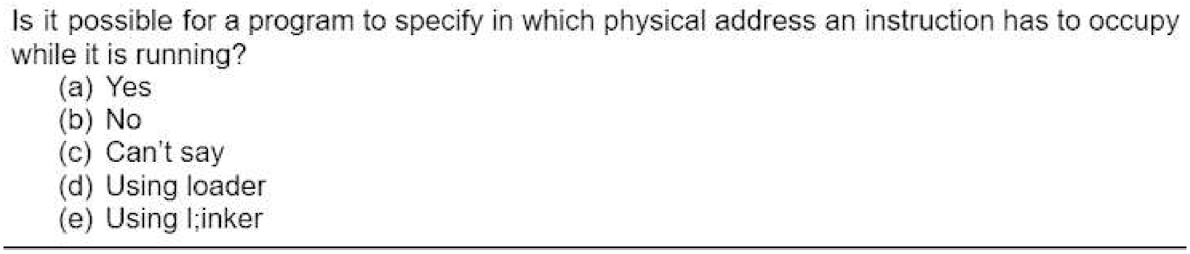
○ A

● B

○ C

○ D

*



Is it possible for a program to specify in which physical address an instruction has to occupy while it is running?
(a) Yes
(b) No
(c) Can't say
(d) Using loader
(e) Using l;inker

○ A

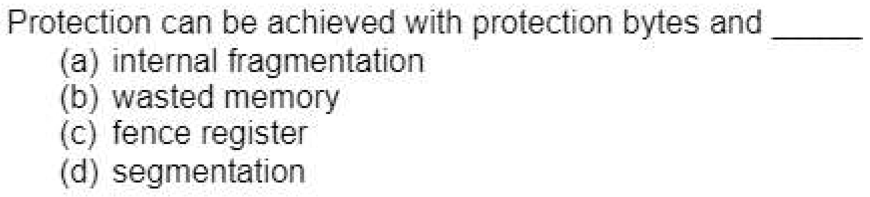
○ B

○ C

● D

○ E

*



Protection can be achieved with protection bytes and ______

(a) internal fragmentation
(b) wasted memory
(c) fence register
(d) segmentation

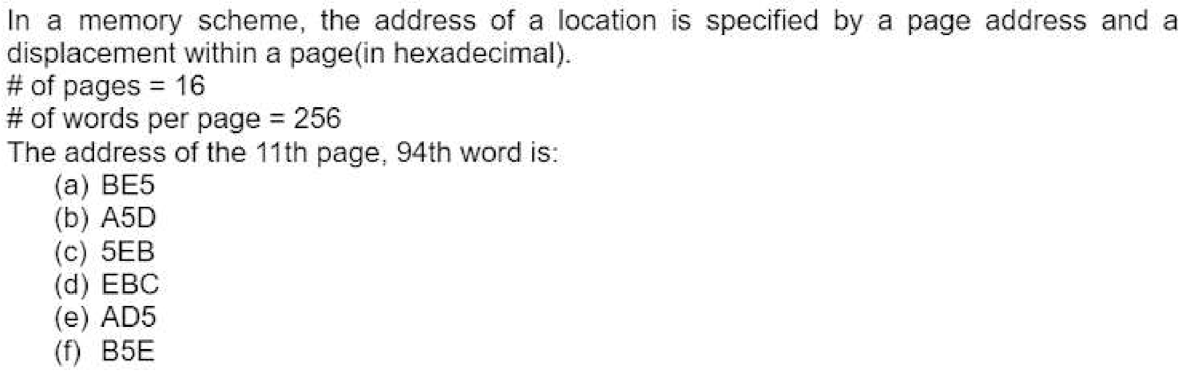
○ A

○ B

● C

○ D

*



In a memory scheme, the address of a location is specified by a page address and a displacement within a page(in hexadecimal).
# of pages = 16
# of words per page = 256
The address of the 11th page, 94th word is:

(a) BE5
(b) A5D
(c) 5EB
(d) EBC
(e) AD5
(f) B5E

○ A

○ B

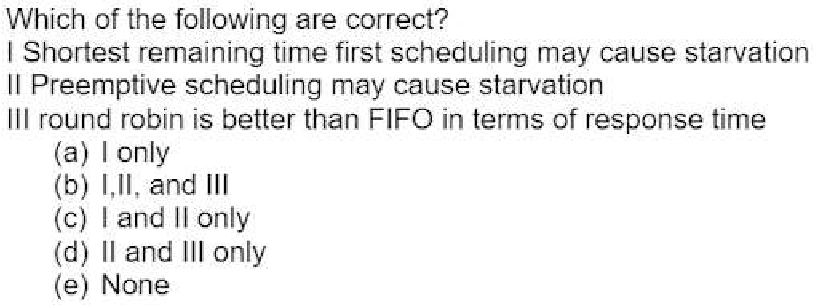
○ C

○ D

○ E

● F

*



Which of the following are correct?
I Shortest remaining time first scheduling may cause starvation
II Preemptive scheduling may cause starvation
III round robin is better than FIFO in terms of response time

(a) I only
(b) I,II, and III
(c) I and II only
(d) II and III only
(e) None

○ A

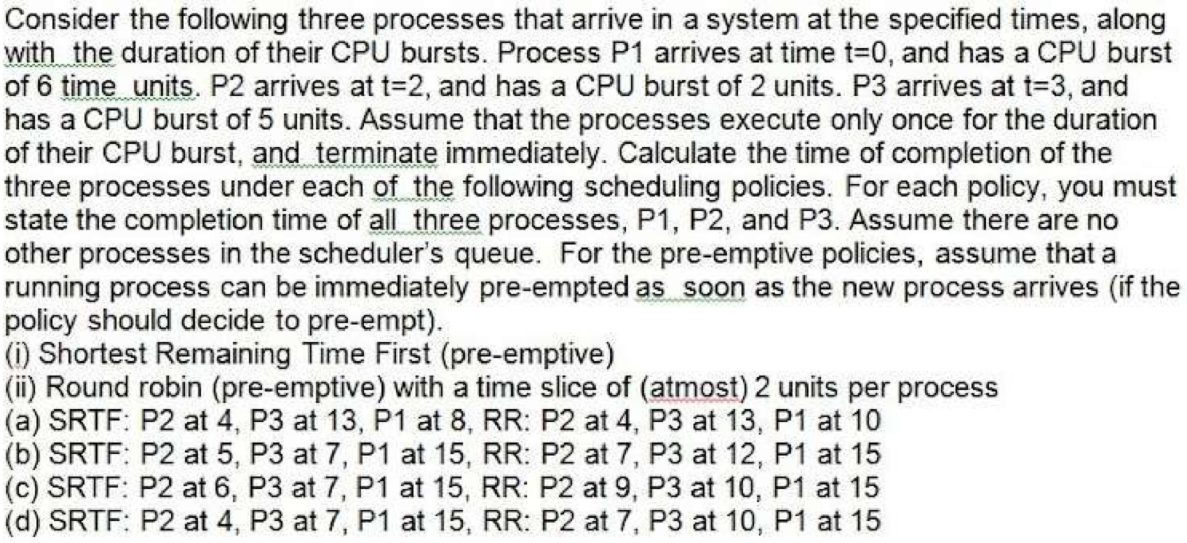
● B

○ C

○ D

○ E

*



Consider the following three processes that arrive in a system at the specified times, along with the duration of their CPU bursts. Process P1 arrives at time t=0, and has a CPU burst of 6 time units. P2 arrives at t=2, and has a CPU burst of 2 units. P3 arrives at t=3, and has a CPU burst of 5 units. Assume that the processes execute only once for the duration of their CPU burst, and terminate immediately. Calculate the time of completion of the three processes under each of the following scheduling policies. For each policy, you must state the completion time of all three processes, P1, P2, and P3. Assume there are no other processes in the scheduler's queue. For the pre-emptive policies, assume that a running process can be immediately pre-empted as soon as the new process arrives (if the policy should decide to pre-empt).
(i) Shortest Remaining Time First (pre-emptive)
(ii) Round robin (pre-emptive) with a time slice of (atmost) 2 units per process
(a) SRTF: P2 at 4, P3 at 13, P1 at 8, RR: P2 at 4, P3 at 13, P1 at 10
(b) SRTF: P2 at 5, P3 at 7, P1 at 15, RR: P2 at 7, P3 at 12, P1 at 15
(c) SRTF: P2 at 6, P3 at 7, P1 at 15, RR: P2 at 9, P3 at 10, P1 at 15
(d) SRTF: P2 at 4, P3 at 7, P1 at 15, RR: P2 at 7, P3 at 10, P1 at 15

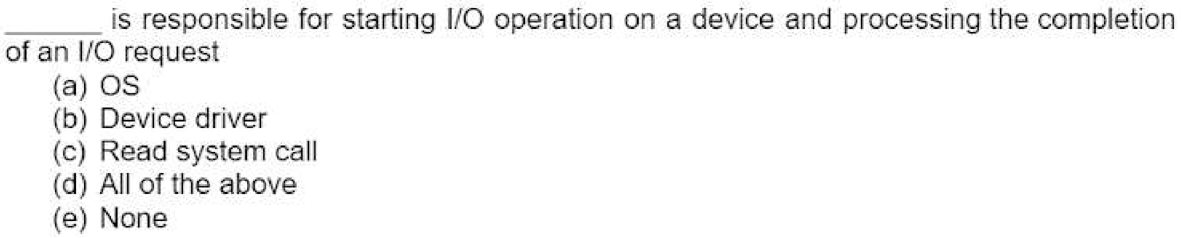
● A

○ B

○ C

○ D

*



_______ is responsible for starting I/O operation on a device and processing the completion of an I/O request

(a) OS
(b) Device driver
(c) Read system call
(d) All of the above
(e) None

○ A

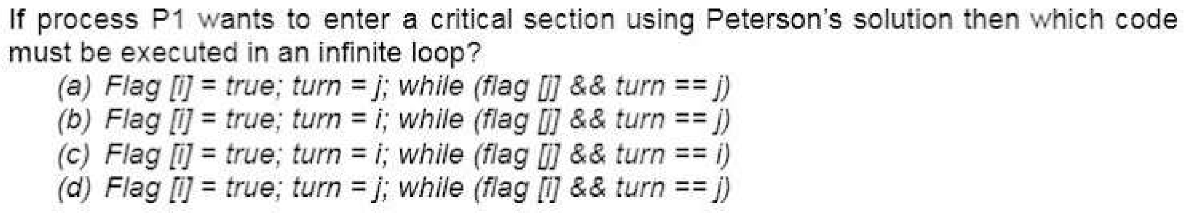
● B

○ C

○ D

○ E

*



If process P1 wants to enter a critical section using Peterson's solution then which code must be executed in an infinite loop?

(a) *Flag [i] = true; turn = j; while (flag [j] && turn == j)*
(b) *Flag [i] = true; turn = i; while (flag [j] && turn == j)*
(c) *Flag [i] = true; turn = i; while (flag [j] && turn == i)*
(d) *Flag [i] = true; turn = j; while (flag [i] && turn == j)*

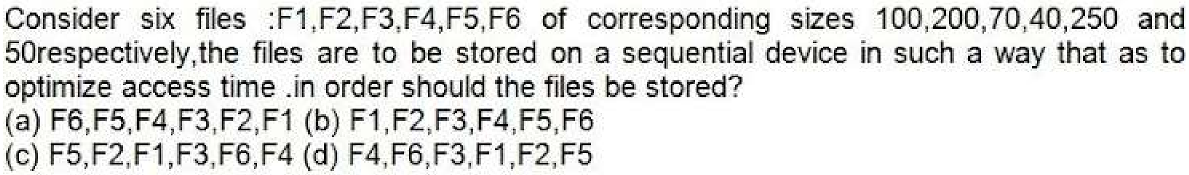
● A

○ B

○ C

○ D

*



Consider six files :F1,F2,F3,F4,F5,F6 of corresponding sizes 100,200,70,40,250 and 50respectively,the files are to be stored on a sequential device in such a way that as to optimize access time .in order should the files be stored?
(a) F6,F5,F4,F3,F2,F1 (b) F1,F2,F3,F4,F5,F6
(c) F5,F2,F1,F3,F6,F4 (d) F4,F6,F3,F1,F2,F5

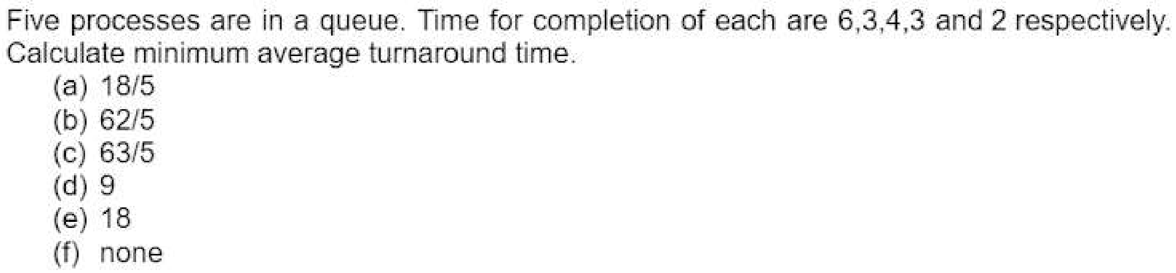
○ A

○ B

○ C

● D

*



Five processes are in a queue. Time for completion of each are 6,3,4,3 and 2 respectively. Calculate minimum average turnaround time.

(a) 18/5
(b) 62/5
(c) 63/5
(d) 9
(e) 18
(f) none

● A

○ B

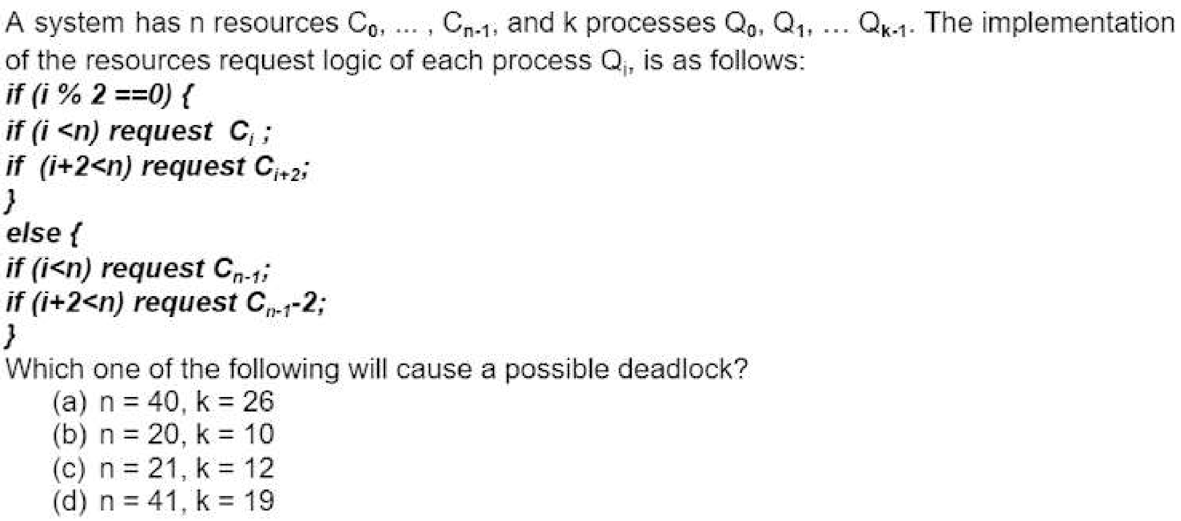
○ C

○ D

○ E

○ F

*



A system has n resources C_0, ... , C_{n-1}, and k processes Q_0, Q_1, … Q_{k-1}. The implementation of the resources request logic of each process Q_i, is as follows:

```
if (i % 2 ==0) {
if (i <n) request  $C_i$ ;
if  (i+2<n) request $C_{i+2}$;
}
else {
if (i<n) request $C_{n-1}$;
if (i+2<n) request $C_{n-1}$-2;
}
```

Which one of the following will cause a possible deadlock?

(a) n = 40, k = 26
(b) n = 20, k = 10
(c) n = 21, k = 12
(d) n = 41, k = 19

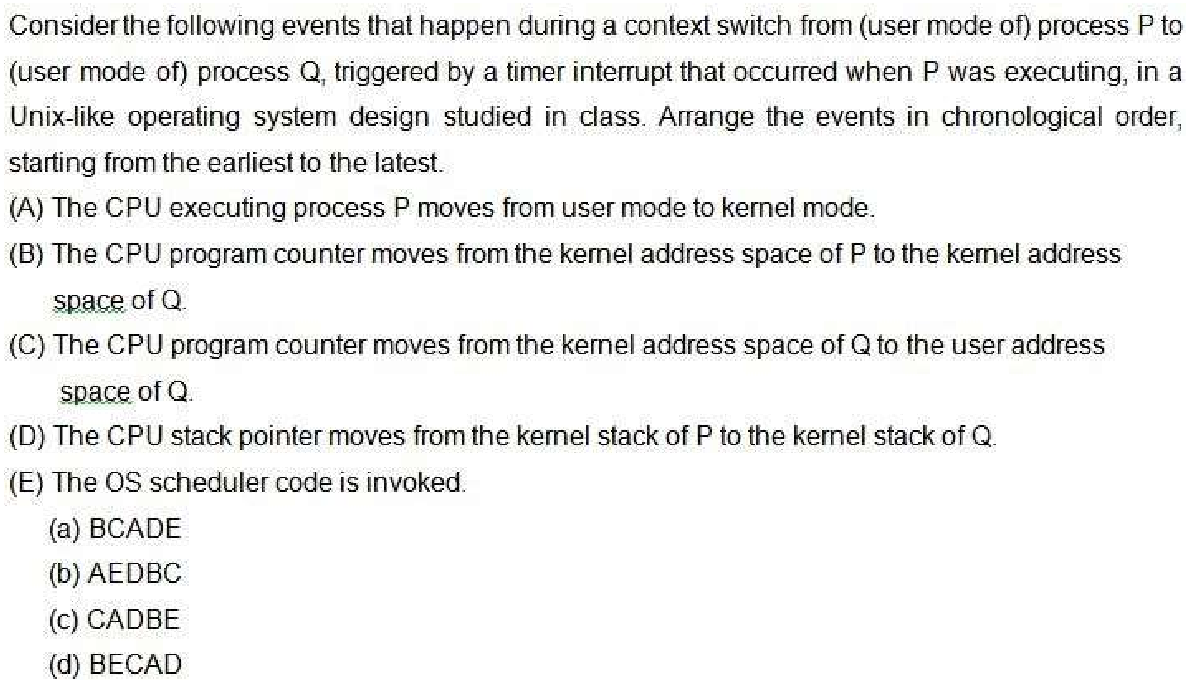
○ A

○ B

● C

○ D

*



Consider the following events that happen during a context switch from (user mode of) process P to (user mode of) process Q, triggered by a timer interrupt that occurred when P was executing, in a Unix-like operating system design studied in class. Arrange the events in chronological order, starting from the earliest to the latest.

(A) The CPU executing process P moves from user mode to kernel mode.

(B) The CPU program counter moves from the kernel address space of P to the kernel address space of Q.

(C) The CPU program counter moves from the kernel address space of Q to the user address space of Q.

(D) The CPU stack pointer moves from the kernel stack of P to the kernel stack of Q.

(E) The OS scheduler code is invoked.

(a) BCADE

(b) AEDBC

(c) CADBE

(d) BECAD

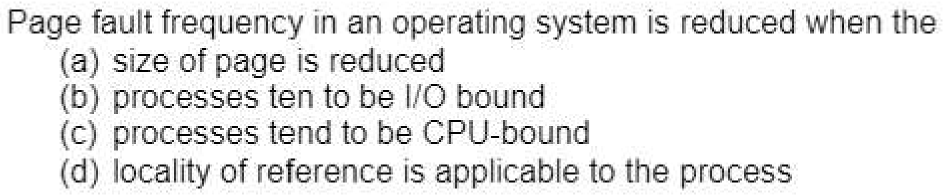
○ A

● B

○ C

○ D

*



Page fault frequency in an operating system is reduced when the

(a) size of page is reduced
(b) processes ten to be I/O bound
(c) processes tend to be CPU-bound
(d) locality of reference is applicable to the process

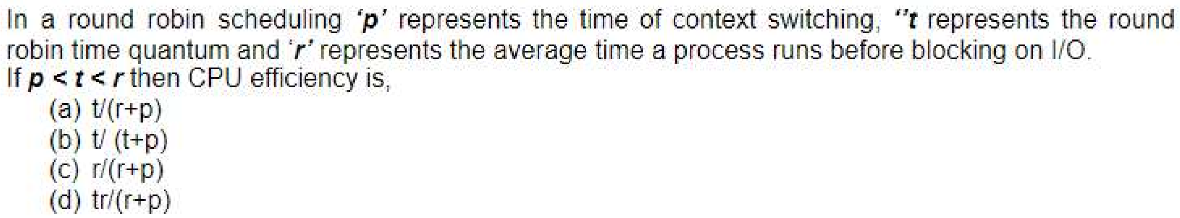
○ A

○ B

○ C

● D

*



In a round robin scheduling ***'p'*** represents the time of context switching, ***''t*** represents the round robin time quantum and ***'r'*** represents the average time a process runs before blocking on I/O.
If ***p < t < r*** then CPU efficiency is,

(a) t/(r+p)
(b) t/ (t+p)
(c) r/(r+p)
(d) tr/(r+p)

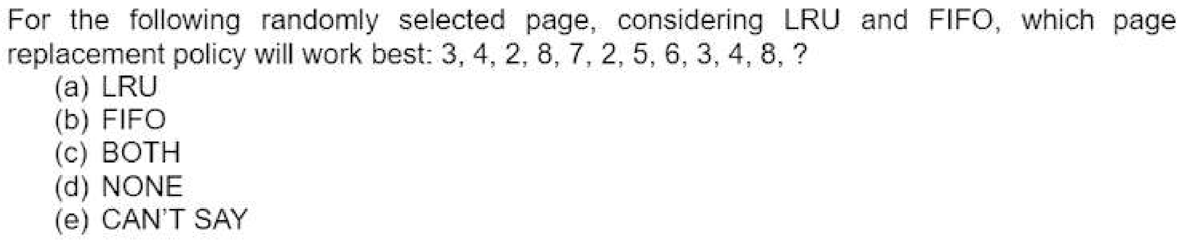
○ A

● B

○ C

○ D

*



For the following randomly selected page, considering LRU and FIFO, which page replacement policy will work best: 3, 4, 2, 8, 7, 2, 5, 6, 3, 4, 8, ?

(a) LRU
(b) FIFO
(c) BOTH
(d) NONE
(e) CAN'T SAY

○ A

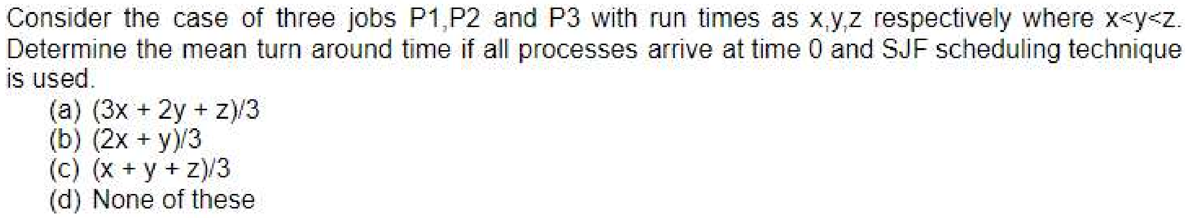
○ B

○ C

● D

○ E

*



Consider the case of three jobs P1,P2 and P3 with run times as x,y,z respectively where x<y<z. Determine the mean turn around time if all processes arrive at time 0 and SJF scheduling technique is used.

(a) (3x + 2y + z)/3
(b) (2x + y)/3
(c) (x + y + z)/3
(d) None of these

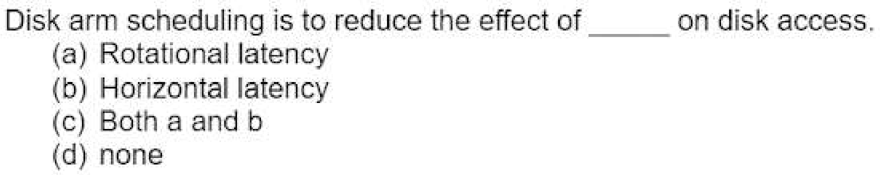
● A

○ B

○ C

○ D

*



Disk arm scheduling is to reduce the effect of _____ on disk access.

(a) Rotational latency
(b) Horizontal latency
(c) Both a and b
(d) none

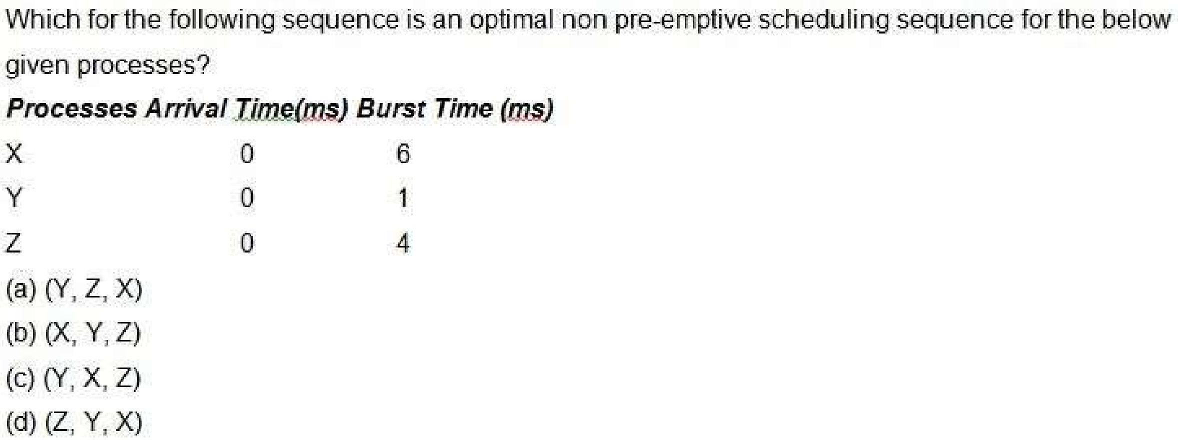
● A

○ B

○ C

○ D

*



Which for the following sequence is an optimal non pre-emptive scheduling sequence for the below given processes?

Processes	***Arrival Time(ms)***	***Burst Time (ms)***
X	0	6
Y	0	1
Z	0	4

(a) (Y, Z, X)
(b) (X, Y, Z)
(c) (Y, X, Z)
(d) (Z, Y, X)

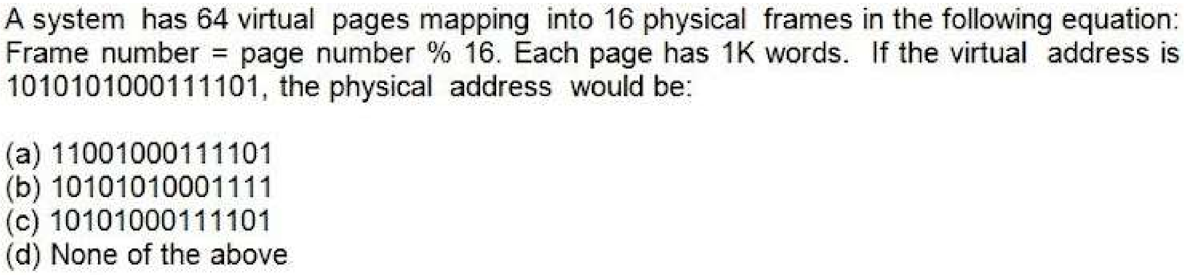
● A

○ B

○ C

○ D

*



A system has 64 virtual pages mapping into 16 physical frames in the following equation: Frame number = page number % 16. Each page has 1K words. If the virtual address is 1010101000111101, the physical address would be:

(a) 11001000111101
(b) 10101010001111
(c) 10101000111101
(d) None of the above

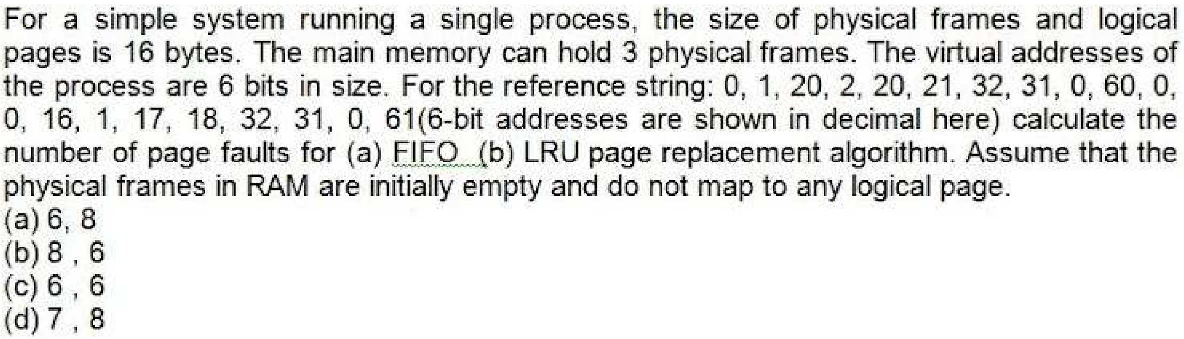
○ A

○ B

● C

○ D

*



For a simple system running a single process, the size of physical frames and logical pages is 16 bytes. The main memory can hold 3 physical frames. The virtual addresses of the process are 6 bits in size. For the reference string: 0, 1, 20, 2, 20, 21, 32, 31, 0, 60, 0, 0, 16, 1, 17, 18, 32, 31, 0, 61(6-bit addresses are shown in decimal here) calculate the number of page faults for (a) FIFO (b) LRU page replacement algorithm. Assume that the physical frames in RAM are initially empty and do not map to any logical page.
(a) 6, 8
(b) 8 , 6
(c) 6 , 6
(d) 7 , 8

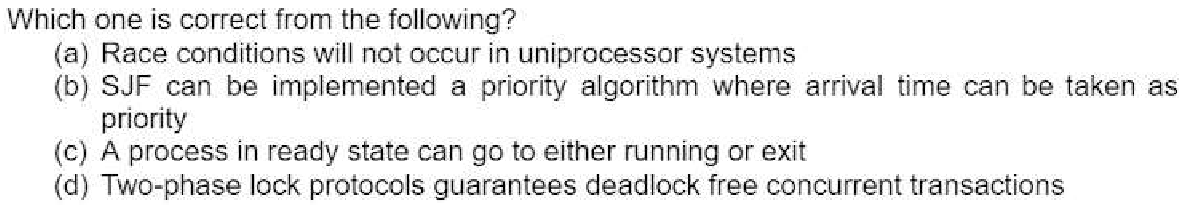
● A

○ B

○ C

○ D

*



Which one is correct from the following?

(a) Race conditions will not occur in uniprocessor systems
(b) SJF can be implemented a priority algorithm where arrival time can be taken as priority
(c) A process in ready state can go to either running or exit
(d) Two-phase lock protocols guarantees deadlock free concurrent transactions

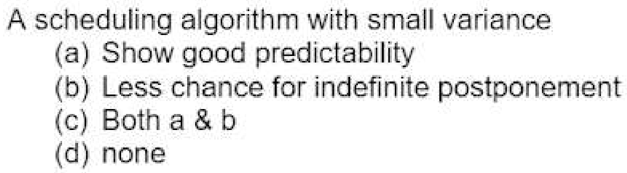
● A

○ B

○ C

○ D

*



A scheduling algorithm with small variance

(a) Show good predictability
(b) Less chance for indefinite postponement
(c) Both a & b
(d) none

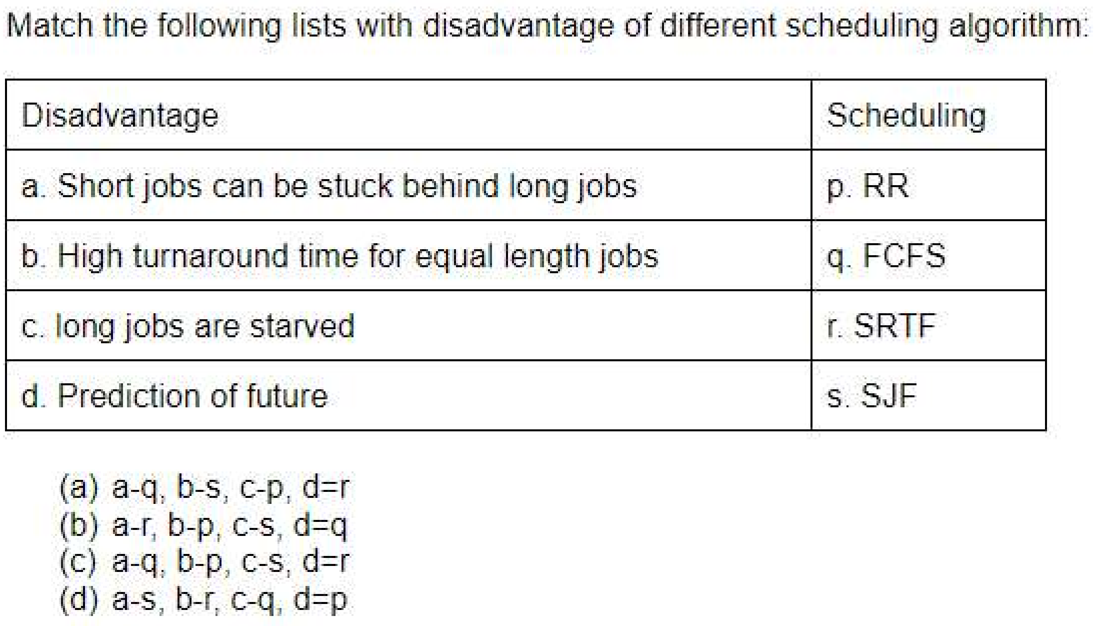
○ A

○ B

● C

○ D

*



Match the following lists with disadvantage of different scheduling algorithm:

Disadvantage	Scheduling
a. Short jobs can be stuck behind long jobs	p. RR
b. High turnaround time for equal length jobs	q. FCFS
c. long jobs are starved	r. SRTF
d. Prediction of future	s. SJF

(a) a-q, b-s, c-p, d=r
(b) a-r, b-p, c-s, d=q
(c) a-q, b-p, c-s, d=r
(d) a-s, b-r, c-q, d=p

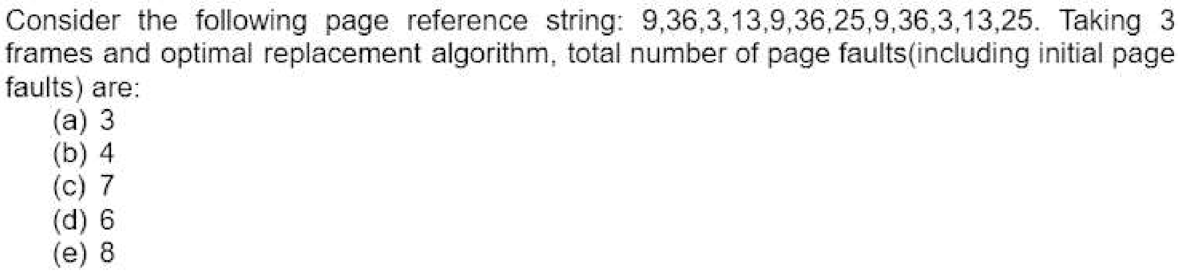
○ A

○ B

● C

○ D

*



Consider the following page reference string: 9,36,3,13,9,36,25,9,36,3,13,25. Taking 3 frames and optimal replacement algorithm, total number of page faults(including initial page faults) are:

(a) 3
(b) 4
(c) 7
(d) 6
(e) 8

○ A

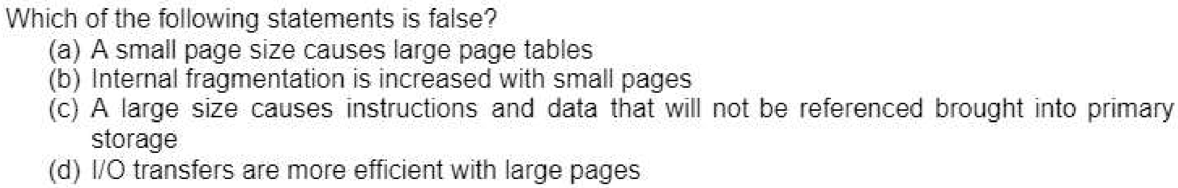
○ B

● C

○ D

○ E

*



Which of the following statements is false?

(a) A small page size causes large page tables
(b) Internal fragmentation is increased with small pages
(c) A large size causes instructions and data that will not be referenced brought into primary storage
(d) I/O transfers are more efficient with large pages

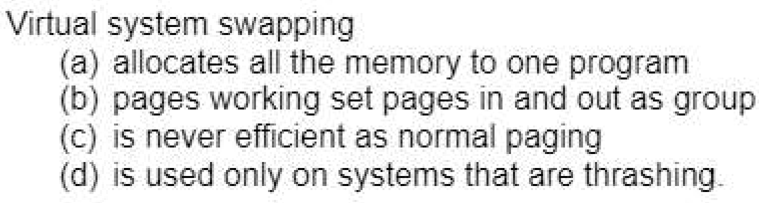
○ A

● B

○ C

○ D

*



Virtual system swapping

(a) allocates all the memory to one program
(b) pages working set pages in and out as group
(c) is never efficient as normal paging
(d) is used only on systems that are thrashing.

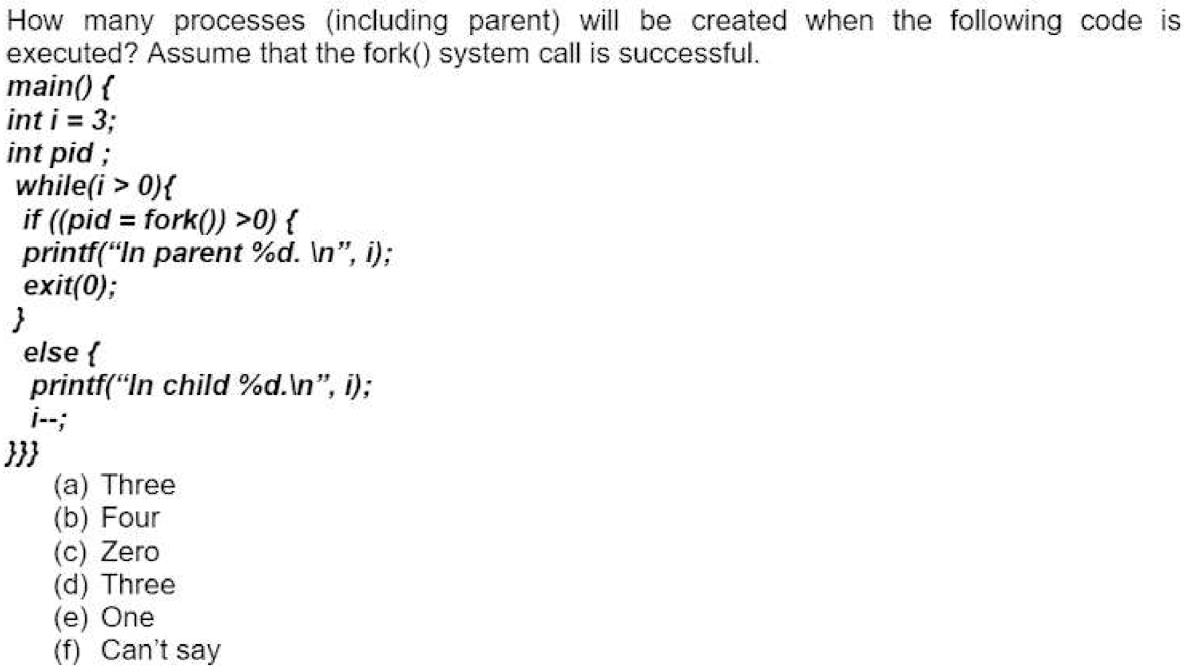
○ A

○ B

○ C

● D

*



How many processes (including parent) will be created when the following code is executed? Assume that the fork() system call is successful.

```
main() {
int i = 3;
int pid ;
 while(i > 0){
  if ((pid = fork()) >0) {
  printf("In parent %d. \n", i);
  exit(0);
 }
  else {
   printf("In child %d.\n", i);
   i--;
}}}
```

(a) Three
(b) Four
(c) Zero
(d) Three
(e) One
(f) Can't say

○ A

○ B

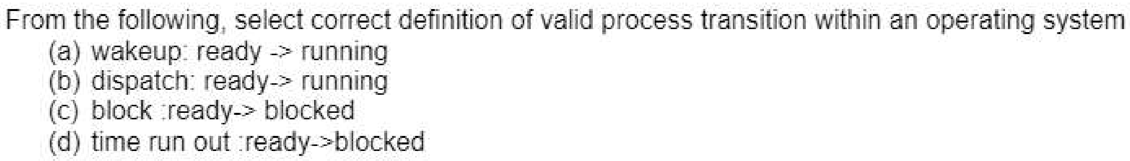
○ C

● D

○ E

○ F

*


From the following, select correct definition of valid process transition within an operating system

(a) wakeup: ready -> running
(b) dispatch: ready-> running
(c) block :ready-> blocked
(d) time run out :ready->blocked

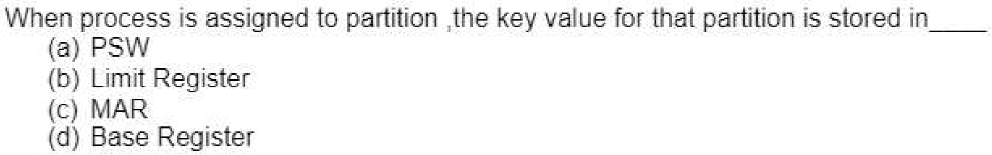
○ A

● B

○ C

○ D

*


When process is assigned to partition ,the key value for that partition is stored in____

(a) PSW
(b) Limit Register
(c) MAR
(d) Base Register

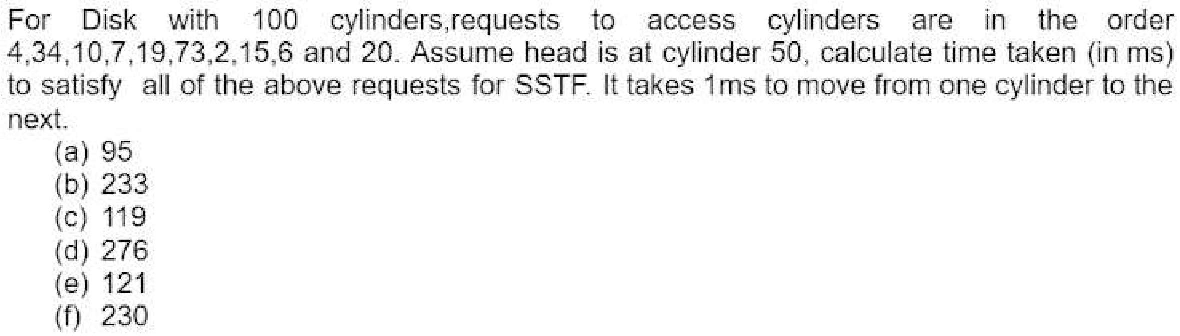
● A

○ B

○ C

○ D

*



For Disk with 100 cylinders,requests to access cylinders are in the order 4,34,10,7,19,73,2,15,6 and 20. Assume head is at cylinder 50, calculate time taken (in ms) to satisfy all of the above requests for SSTF. It takes 1ms to move from one cylinder to the next.

(a) 95
(b) 233
(c) 119
(d) 276
(e) 121
(f) 230

○ A

○ B

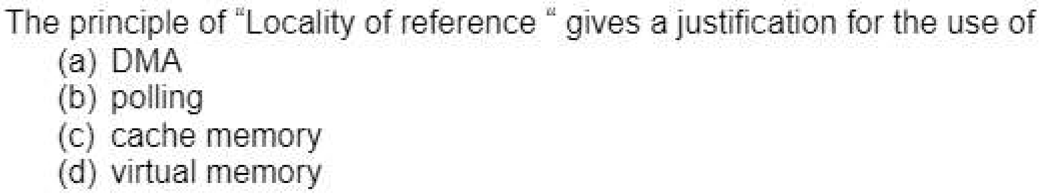
● C

○ D

○ E

○ F

*



The principle of "Locality of reference " gives a justification for the use of

(a) DMA
(b) polling
(c) cache memory
(d) virtual memory

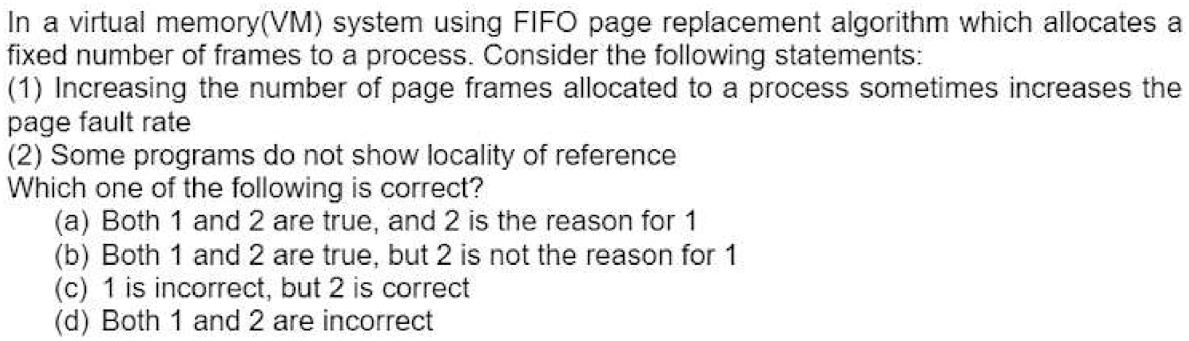
○ A

○ B

● C

○ D

*



In a virtual memory(VM) system using FIFO page replacement algorithm which allocates a fixed number of frames to a process. Consider the following statements:
(1) Increasing the number of page frames allocated to a process sometimes increases the page fault rate
(2) Some programs do not show locality of reference
Which one of the following is correct?

(a) Both 1 and 2 are true, and 2 is the reason for 1
(b) Both 1 and 2 are true, but 2 is not the reason for 1
(c) 1 is incorrect, but 2 is correct
(d) Both 1 and 2 are incorrect

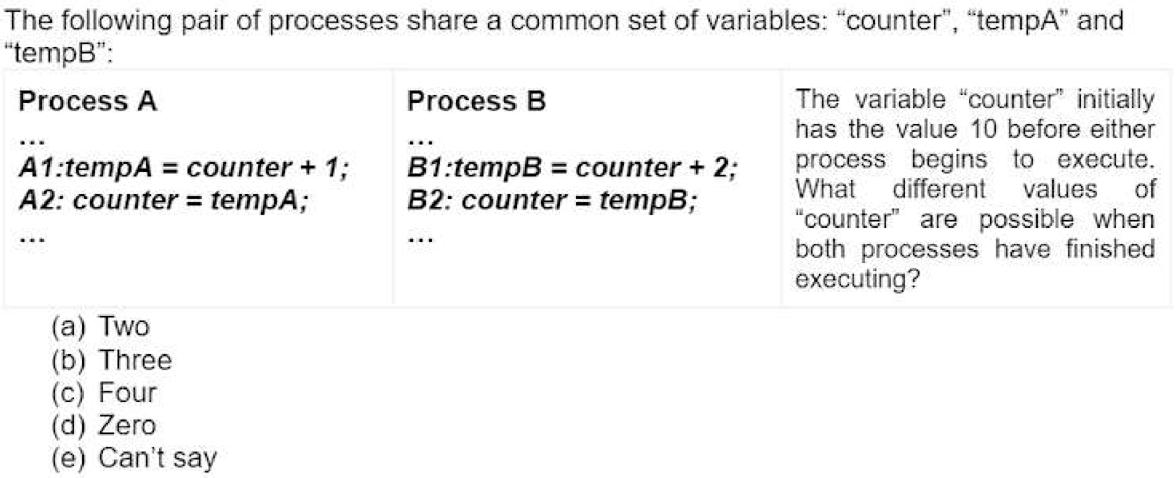
○ A

● B

○ C

○ D

*



The following pair of processes share a common set of variables: "counter", "tempA" and "tempB":

Process A … ***A1:tempA = counter + 1;*** ***A2: counter = tempA;*** …	**Process B** … ***B1:tempB = counter + 2;*** ***B2: counter = tempB;*** …	The variable "counter" initially has the value 10 before either process begins to execute. What different values of "counter" are possible when both processes have finished executing?

(a) Two
(b) Three
(c) Four
(d) Zero
(e) Can't say

○ A

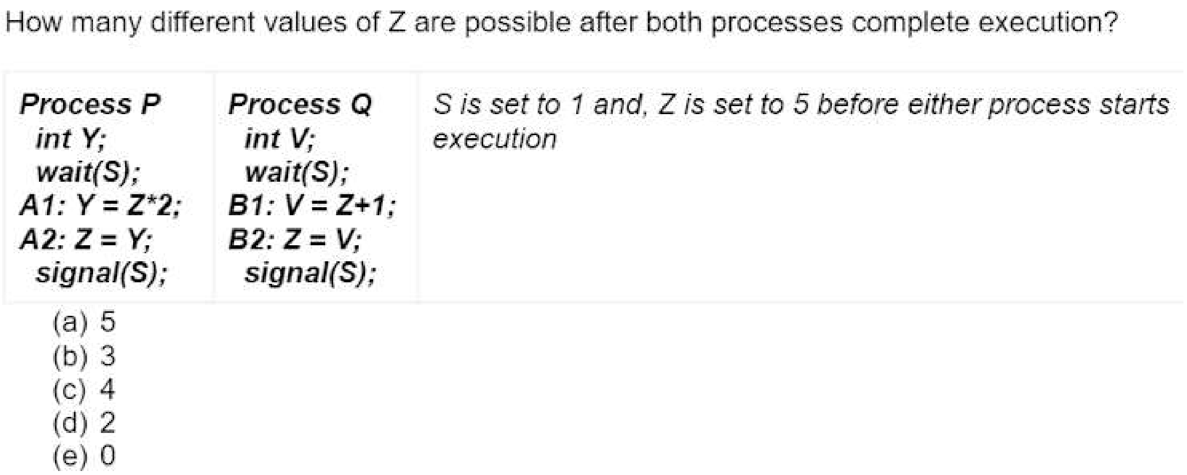
● B

○ C

○ D

○ E

*



How many different values of Z are possible after both processes complete execution?

Process P
int Y;
wait(S);
A1: Y = Z*2;
A2: Z = Y;
signal(S);
	Process Q
	int V;
	wait(S);
	B1: V = Z+1;
	B2: Z = V;
	signal(S);
		S is set to 1 and, Z is set to 5 before either process starts execution

(a) 5
(b) 3
(c) 4
(d) 2
(e) 0

○ A

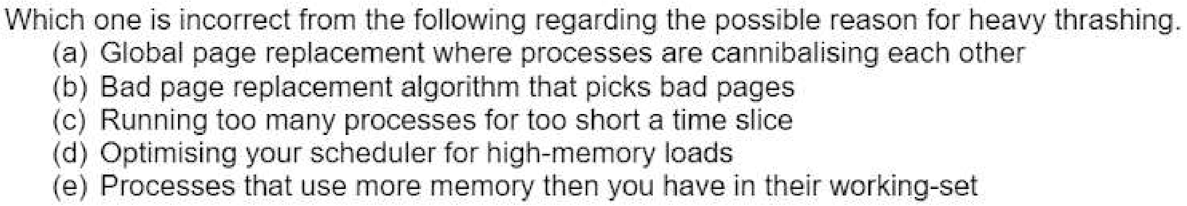
○ B

○ C

● D

○ E

*



Which one is incorrect from the following regarding the possible reason for heavy thrashing.

(a) Global page replacement where processes are cannibalising each other
(b) Bad page replacement algorithm that picks bad pages
(c) Running too many processes for too short a time slice
(d) Optimising your scheduler for high-memory loads
(e) Processes that use more memory then you have in their working-set

○ A

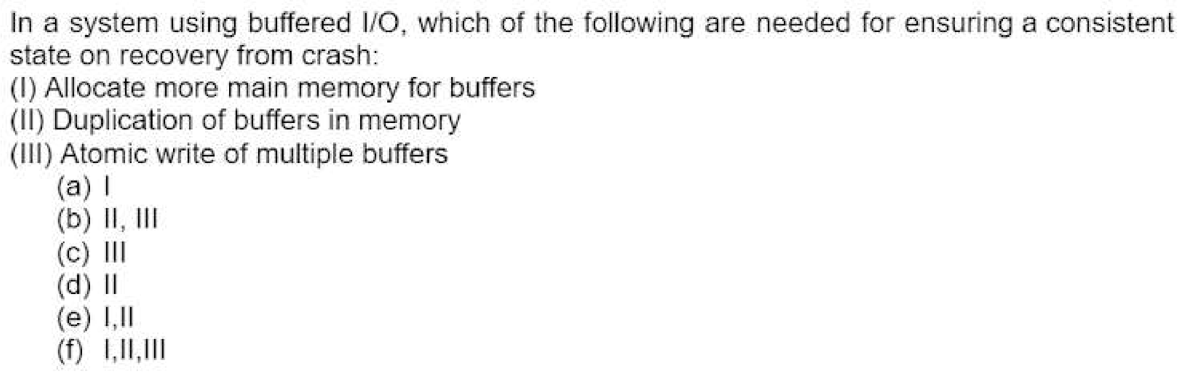
○ B

○ C

● D

○ E

*



In a system using buffered I/O, which of the following are needed for ensuring a consistent state on recovery from crash:

(I) Allocate more main memory for buffers
(II) Duplication of buffers in memory
(III) Atomic write of multiple buffers

(a) I
(b) II, III
(c) III
(d) II
(e) I,II
(f) I,II,III

○ A

● B

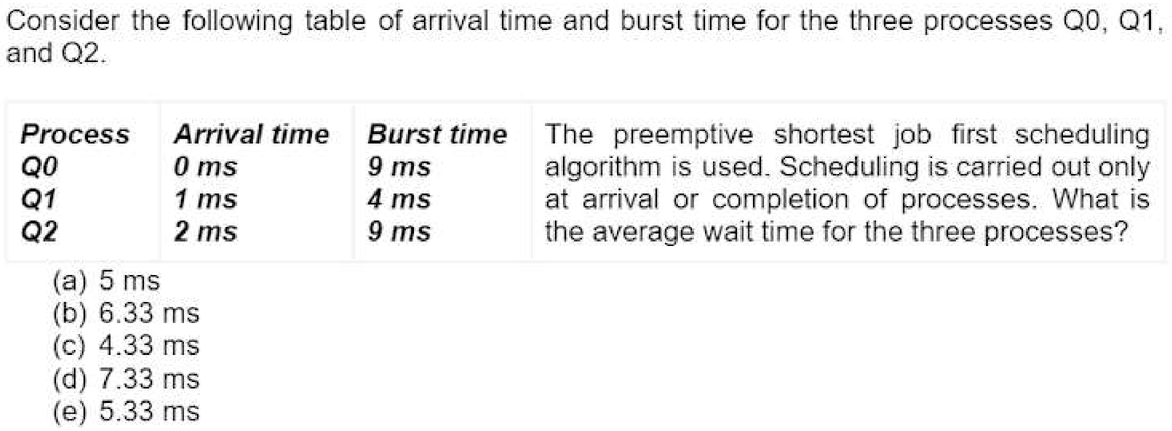
○ C

○ D

○ E

○ F

*



Consider the following table of arrival time and burst time for the three processes Q0, Q1, and Q2.

Process	***Arrival time***	***Burst time***
Q0	***0 ms***	***9 ms***
Q1	***1 ms***	***4 ms***
Q2	***2 ms***	***9 ms***

The preemptive shortest job first scheduling algorithm is used. Scheduling is carried out only at arrival or completion of processes. What is the average wait time for the three processes?

(a) 5 ms
(b) 6.33 ms
(c) 4.33 ms
(d) 7.33 ms
(e) 5.33 ms

○ A

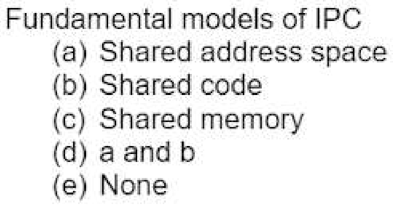
● B

○ C

○ D

○ E

*



Fundamental models of IPC

(a) Shared address space
(b) Shared code
(c) Shared memory
(d) a and b
(e) None

○ A

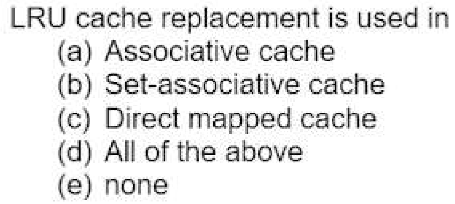
○ B

● C

○ D

○ E

*



LRU cache replacement is used in

(a) Associative cache
(b) Set-associative cache
(c) Direct mapped cache
(d) All of the above
(e) none

○ A

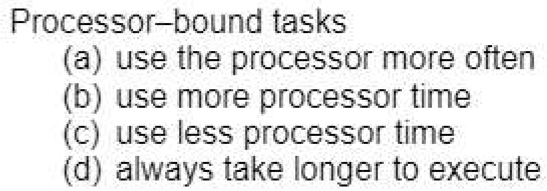
○ B

○ C

● D

○ E

*



Processor–bound tasks

(a) use the processor more often
(b) use more processor time
(c) use less processor time
(d) always take longer to execute

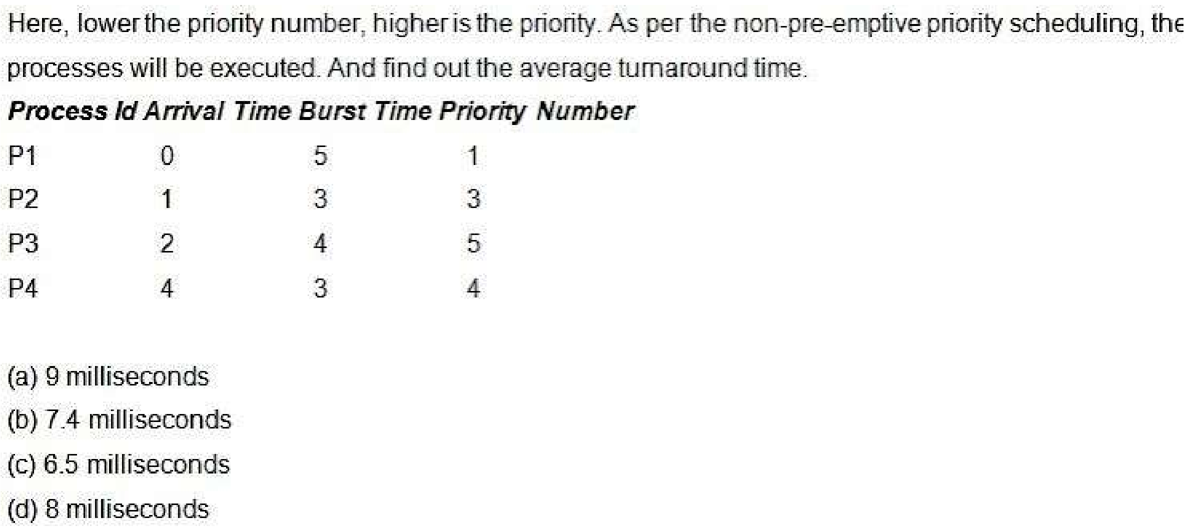
○ A

● B

○ C

○ D

*



Here, lower the priority number, higher is the priority. As per the non-pre-emptive priority scheduling, the processes will be executed. And find out the average turnaround time.

Process Id	***Arrival Time***	***Burst Time***	***Priority Number***
P1	0	5	1
P2	1	3	3
P3	2	4	5
P4	4	3	4

(a) 9 milliseconds

(b) 7.4 milliseconds

(c) 6.5 milliseconds

(d) 8 milliseconds

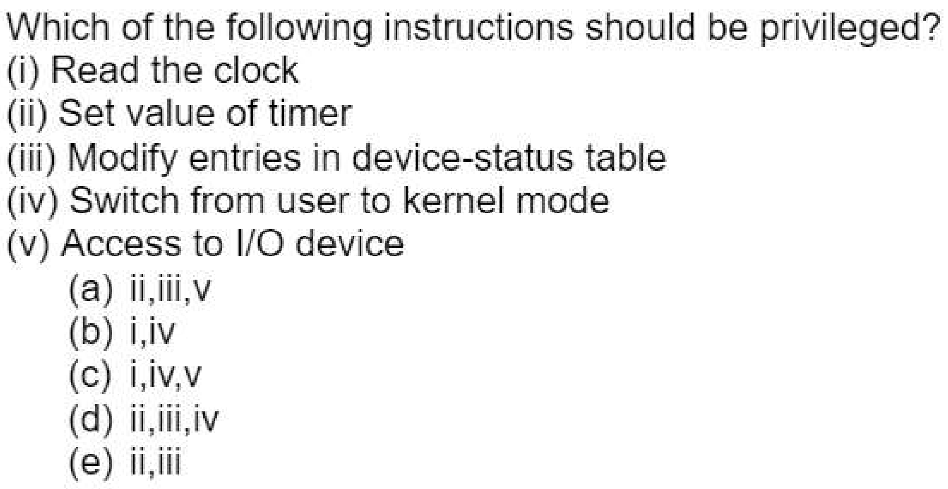
○ A

○ B

○ C

● D

*



Which of the following instructions should be privileged?
(i) Read the clock
(ii) Set value of timer
(iii) Modify entries in device-status table
(iv) Switch from user to kernel mode
(v) Access to I/O device

(a) ii,iii,v
(b) i,iv
(c) i,iv,v
(d) ii,iii,iv
(e) ii,iii

● A

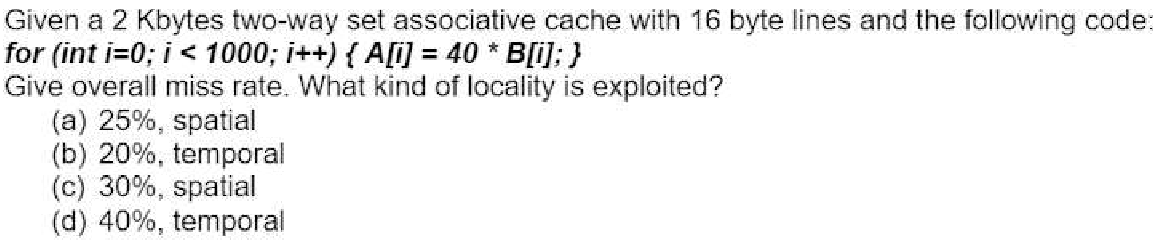
○ B

○ C

○ D

○ E

*



Given a 2 Kbytes two-way set associative cache with 16 byte lines and the following code:
for (int i=0; i < 1000; i++) { A[i] = 40 * B[i]; }
Give overall miss rate. What kind of locality is exploited?

(a) 25%, spatial
(b) 20%, temporal
(c) 30%, spatial
(d) 40%, temporal

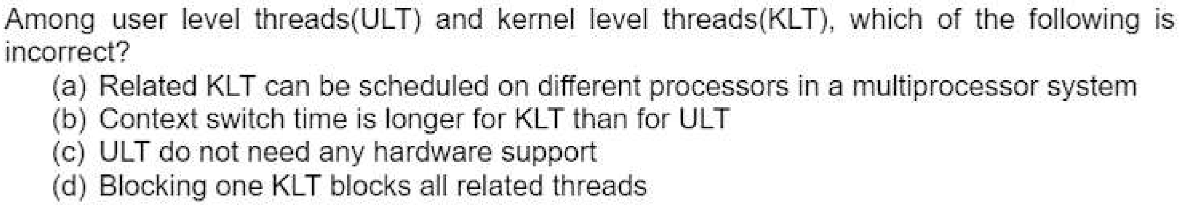
● A

○ B

○ C

○ D

*



Among user level threads(ULT) and kernel level threads(KLT), which of the following is incorrect?

(a) Related KLT can be scheduled on different processors in a multiprocessor system
(b) Context switch time is longer for KLT than for ULT
(c) ULT do not need any hardware support
(d) Blocking one KLT blocks all related threads

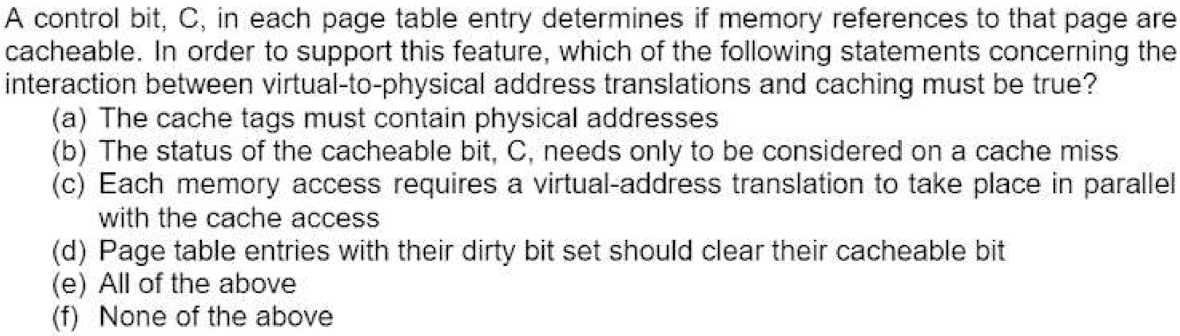
○ A

○ B

○ C

● D

*



A control bit, C, in each page table entry determines if memory references to that page are cacheable. In order to support this feature, which of the following statements concerning the interaction between virtual-to-physical address translations and caching must be true?

(a) The cache tags must contain physical addresses
(b) The status of the cacheable bit, C, needs only to be considered on a cache miss
(c) Each memory access requires a virtual-address translation to take place in parallel with the cache access
(d) Page table entries with their dirty bit set should clear their cacheable bit
(e) All of the above
(f) None of the above

○ A

○ B

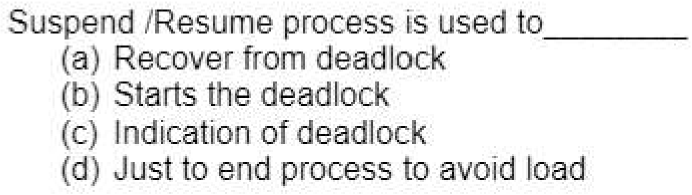
● C

○ D

○ E

○ F

*



Suspend /Resume process is used to________

(a) Recover from deadlock
(b) Starts the deadlock
(c) Indication of deadlock
(d) Just to end process to avoid load

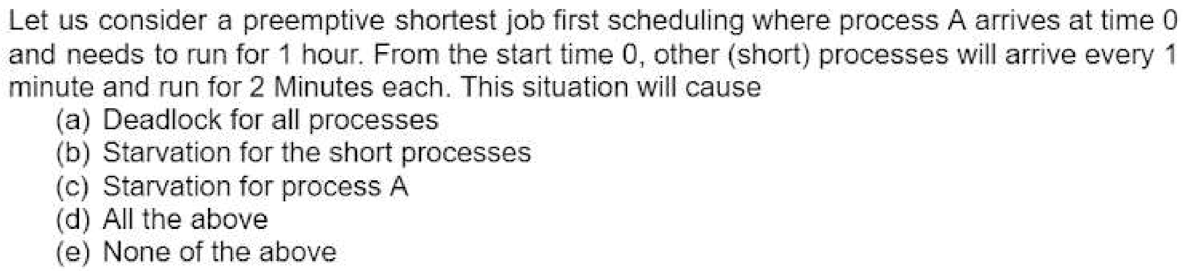
● A

○ B

○ C

○ D

*



Let us consider a preemptive shortest job first scheduling where process A arrives at time 0 and needs to run for 1 hour. From the start time 0, other (short) processes will arrive every 1 minute and run for 2 Minutes each. This situation will cause

(a) Deadlock for all processes
(b) Starvation for the short processes
(c) Starvation for process A
(d) All the above
(e) None of the above

○ A

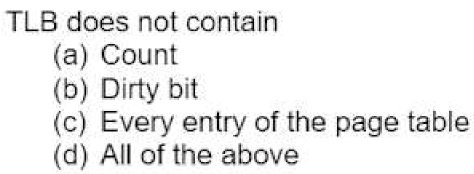
○ B

● C

○ D

○ E

*



TLB does not contain

(a) Count
(b) Dirty bit
(c) Every entry of the page table
(d) All of the above

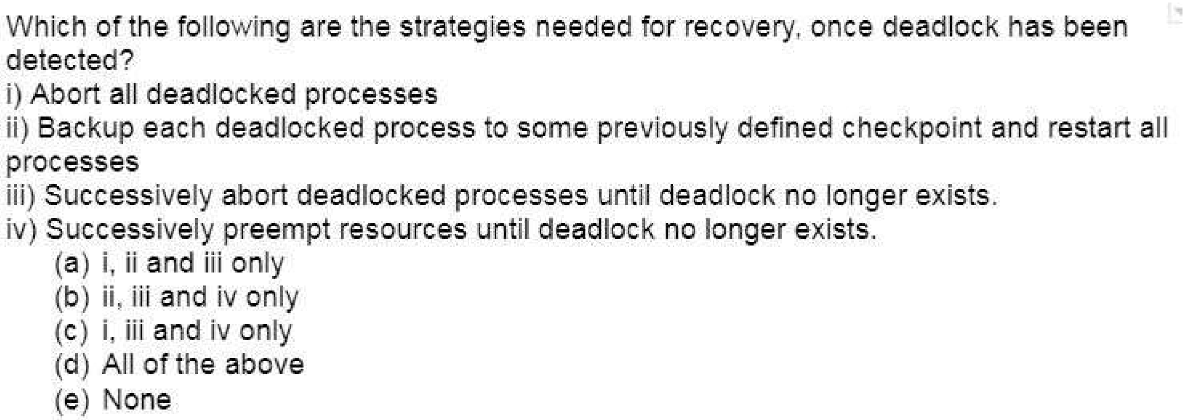
○ A

● B

○ C

○ D

*



Which of the following are the strategies needed for recovery, once deadlock has been detected?

i) Abort all deadlocked processes
ii) Backup each deadlocked process to some previously defined checkpoint and restart all processes
iii) Successively abort deadlocked processes until deadlock no longer exists.
iv) Successively preempt resources until deadlock no longer exists.

(a) i, ii and iii only
(b) ii, iii and iv only
(c) i, iii and iv only
(d) All of the above
(e) None

○ A

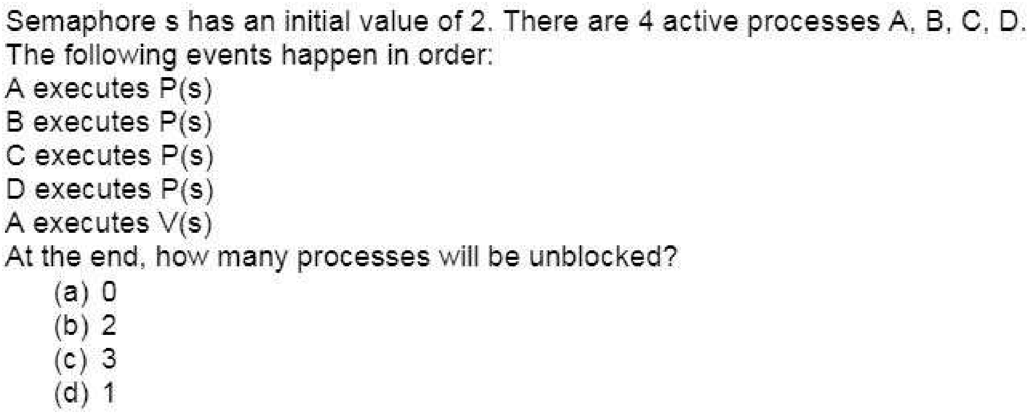
● B

○ C

○ D

○ E

*



Semaphore s has an initial value of 2. There are 4 active processes A, B, C, D.
The following events happen in order:
A executes P(s)
B executes P(s)
C executes P(s)
D executes P(s)
A executes V(s)
At the end, how many processes will be unblocked?

(a) 0
(b) 2
(c) 3
(d) 1

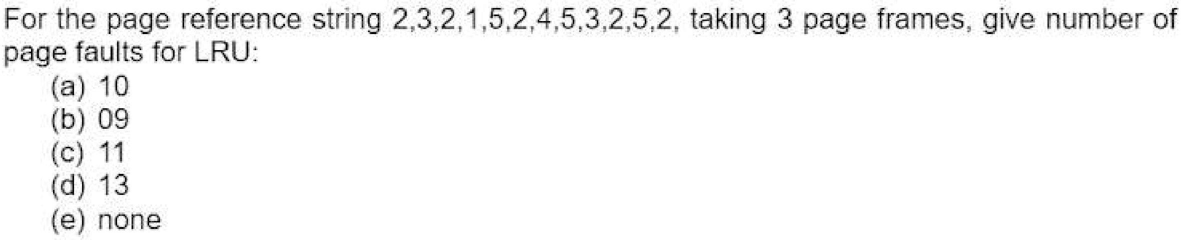
○ A

○ B

○ C

● D

*



For the page reference string 2,3,2,1,5,2,4,5,3,2,5,2, taking 3 page frames, give number of page faults for LRU:

(a) 10
(b) 09
(c) 11
(d) 13
(e) none

○ A

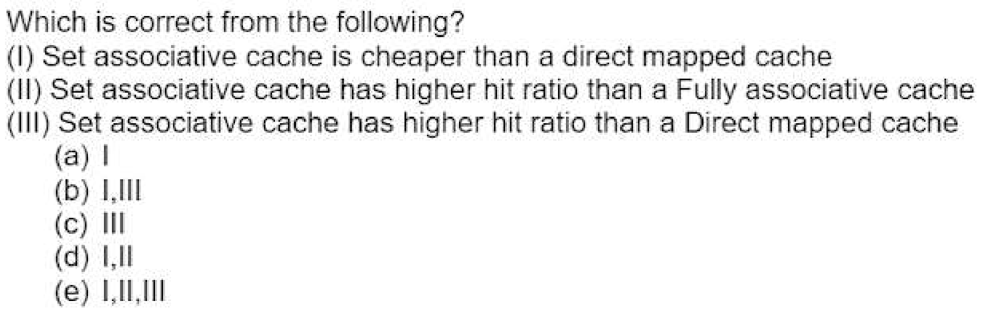
○ B

○ C

○ D

● E

*



Which is correct from the following?
(I) Set associative cache is cheaper than a direct mapped cache
(II) Set associative cache has higher hit ratio than a Fully associative cache
(III) Set associative cache has higher hit ratio than a Direct mapped cache

(a) I
(b) I,III
(c) III
(d) I,II
(e) I,II,III

○ A

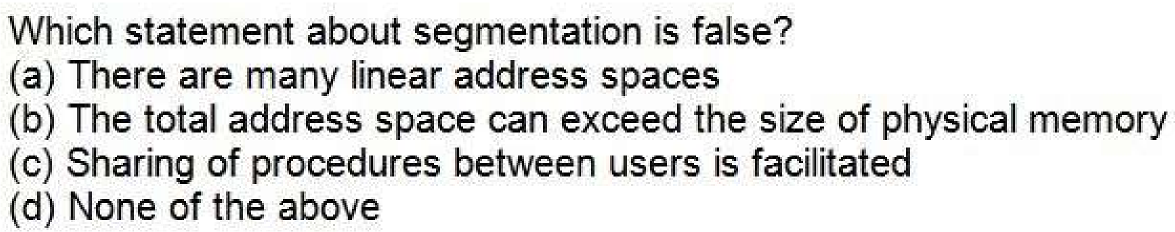
○ B

● C

○ D

○ E

*



Which statement about segmentation is false?
(a) There are many linear address spaces
(b) The total address space can exceed the size of physical memory
(c) Sharing of procedures between users is facilitated
(d) None of the above

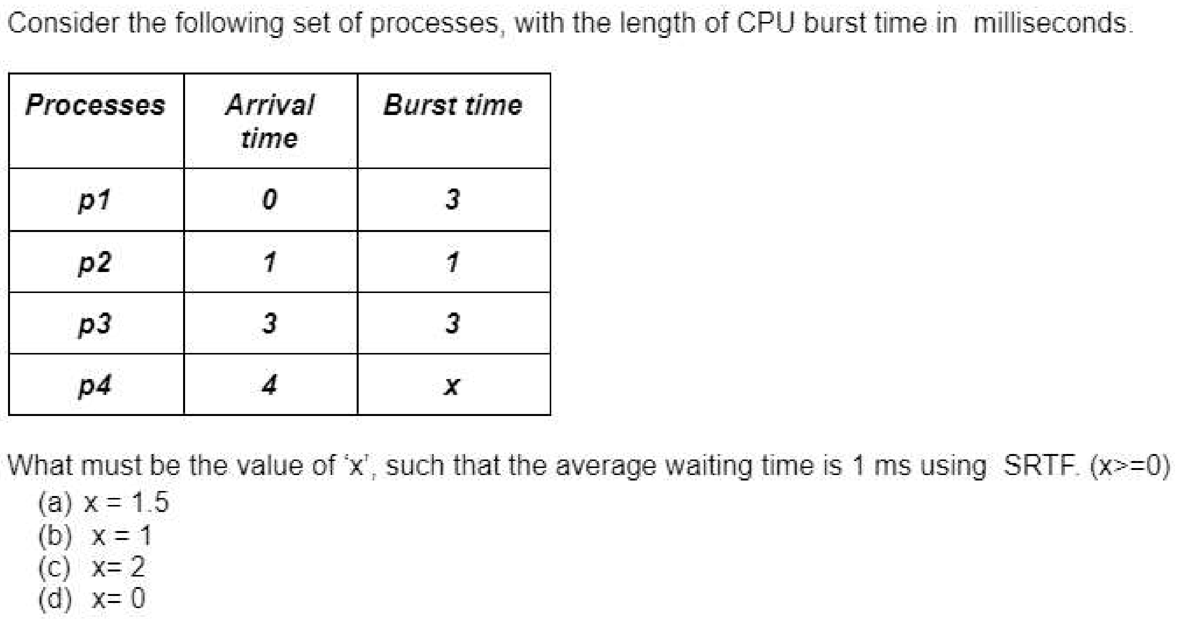
○ A

○ B

○ C

● D

*



Consider the following set of processes, with the length of CPU burst time in milliseconds.

Processes	*Arrival time*	*Burst time*
p1	*0*	*3*
p2	*1*	*1*
p3	*3*	*3*
p4	*4*	*x*

What must be the value of 'x', such that the average waiting time is 1 ms using SRTF. (x>=0)

(a) x = 1.5
(b) x = 1
(c) x= 2
(d) x= 0

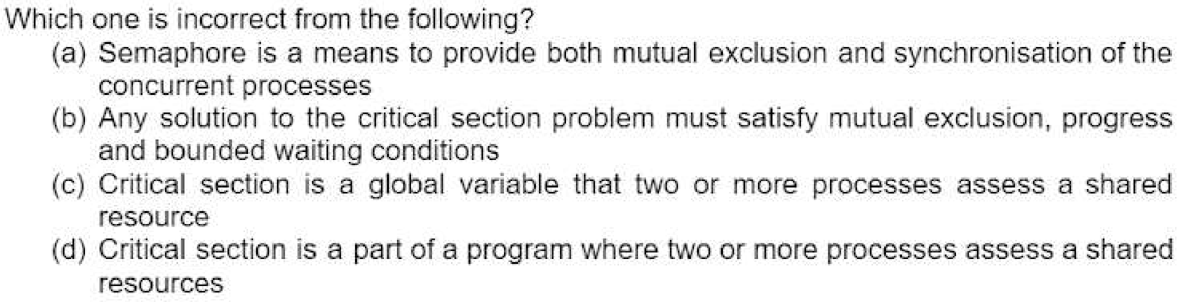
○ A

○ B

● C

○ D

*



Which one is incorrect from the following?

(a) Semaphore is a means to provide both mutual exclusion and synchronisation of the concurrent processes

(b) Any solution to the critical section problem must satisfy mutual exclusion, progress and bounded waiting conditions

(c) Critical section is a global variable that two or more processes assess a shared resource

(d) Critical section is a part of a program where two or more processes assess a shared resources

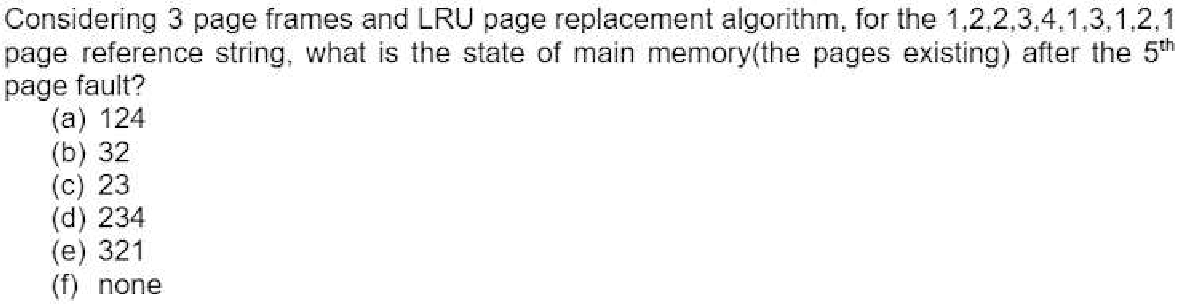
○ A

○ B

● C

○ D

*



Considering 3 page frames and LRU page replacement algorithm, for the 1,2,2,3,4,1,3,1,2,1 page reference string, what is the state of main memory(the pages existing) after the 5th page fault?

(a) 124

(b) 32

(c) 23

(d) 234

(e) 321

(f) none

○ A

○ B

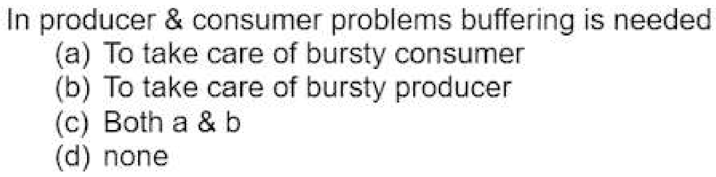
○ C

○ D

○ E

● F

*



In producer & consumer problems buffering is needed

(a) To take care of bursty consumer
(b) To take care of bursty producer
(c) Both a & b
(d) none

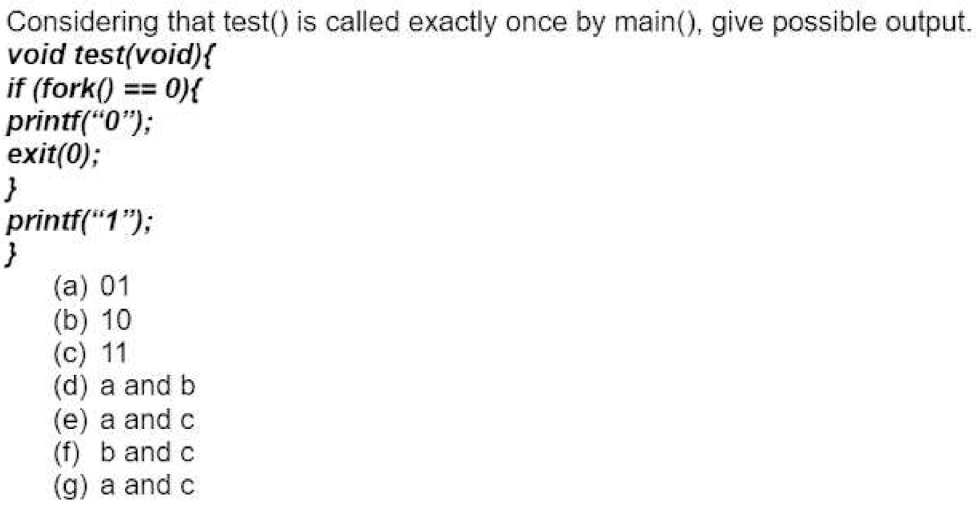
○ A

○ B

● C

○ D

*



Considering that test() is called exactly once by main(), give possible output.

```
void test(void){
if (fork() == 0){
printf(“0”);
exit(0);
}
printf(“1”);
}
```

(a) 01
(b) 10
(c) 11
(d) a and b
(e) a and c
(f) b and c
(g) a and c

○ A

● B

○ C

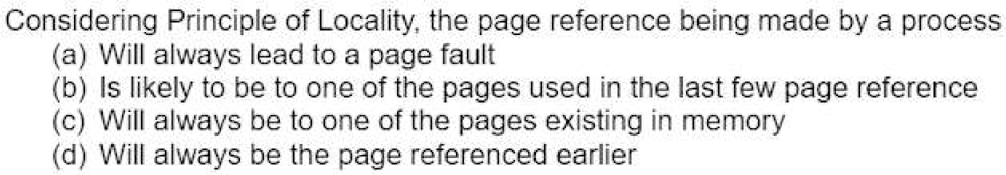
○ D

○ E

○ F

○ G

*



Considering Principle of Locality, the page reference being made by a process

(a) Will always lead to a page fault
(b) Is likely to be to one of the pages used in the last few page reference
(c) Will always be to one of the pages existing in memory
(d) Will always be the page referenced earlier

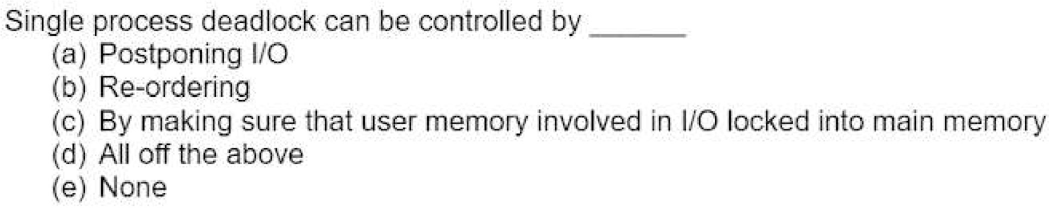
○ A

● B

○ C

○ D

*



Single process deadlock can be controlled by ______

(a) Postponing I/O
(b) Re-ordering
(c) By making sure that user memory involved in I/O locked into main memory
(d) All off the above
(e) None

○ A

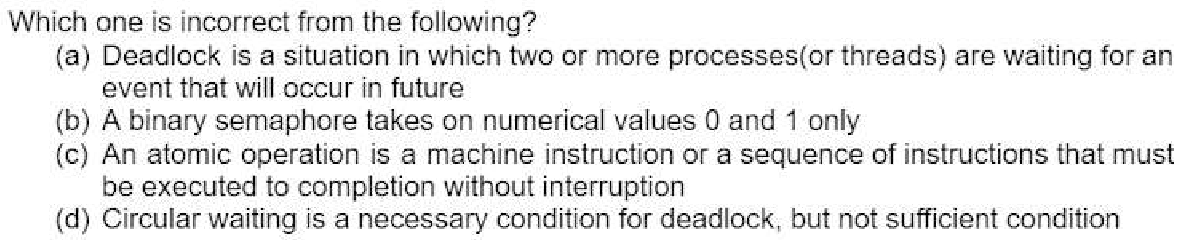
○ B

● C

○ D

○ E

*



Which one is incorrect from the following?

(a) Deadlock is a situation in which two or more processes(or threads) are waiting for an event that will occur in future

(b) A binary semaphore takes on numerical values 0 and 1 only

(c) An atomic operation is a machine instruction or a sequence of instructions that must be executed to completion without interruption

(d) Circular waiting is a necessary condition for deadlock, but not sufficient condition

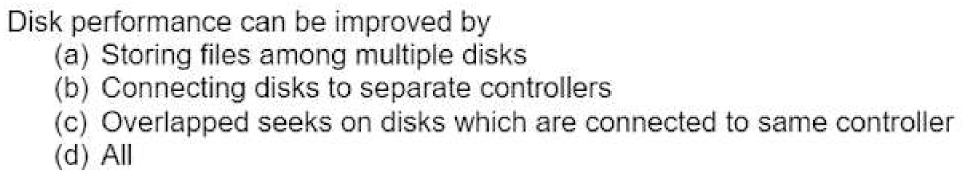
● A

○ B

○ C

○ D

*



Disk performance can be improved by

(a) Storing files among multiple disks

(b) Connecting disks to separate controllers

(c) Overlapped seeks on disks which are connected to same controller

(d) All

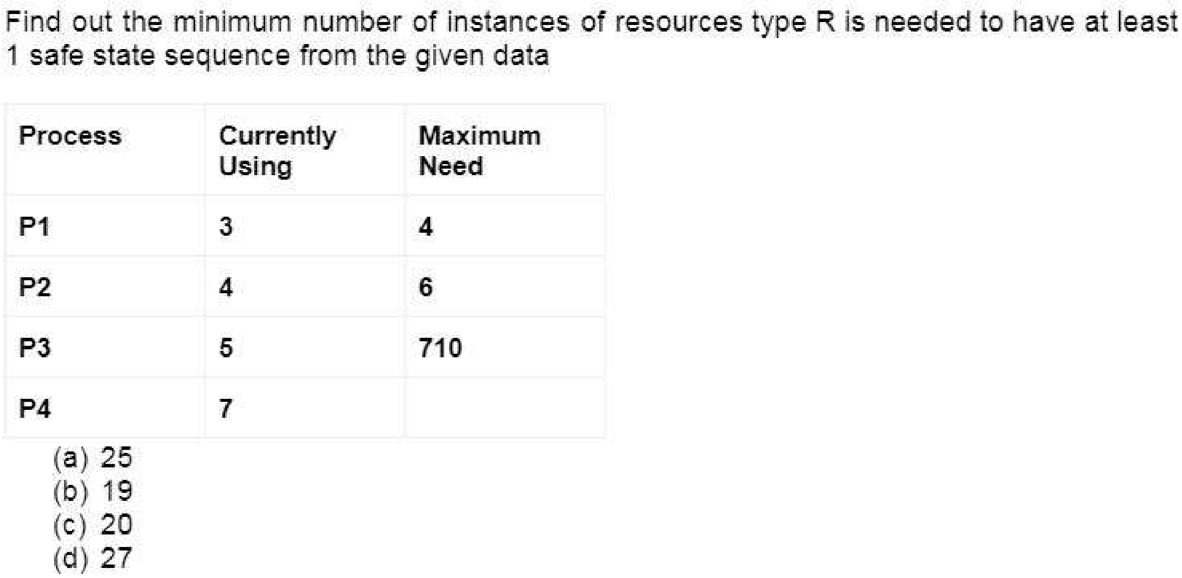
○ A

○ B

○ C

● D

*



Find out the minimum number of instances of resources type R is needed to have at least 1 safe state sequence from the given data

Process	Currently Using	Maximum Need
P1	3	4
P2	4	6
P3	5	710
P4	7	

(a) 25
(b) 19
(c) 20
(d) 27

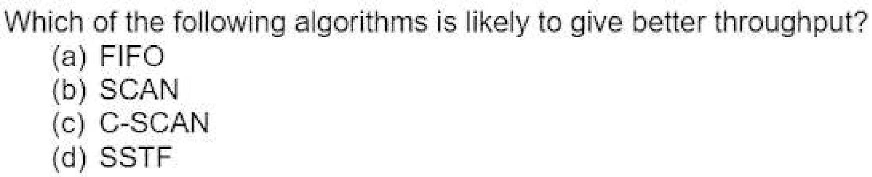
○ A

● B

○ C

○ D

*



Which of the following algorithms is likely to give better throughput?

(a) FIFO
(b) SCAN
(c) C-SCAN
(d) SSTF

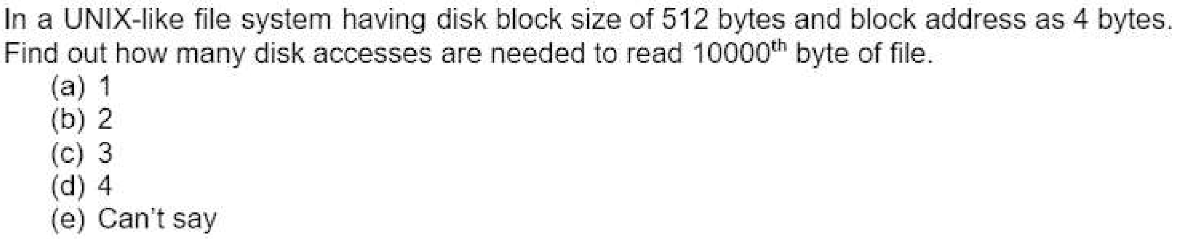
○ A

○ B

○ C

● D

*



In a UNIX-like file system having disk block size of 512 bytes and block address as 4 bytes. Find out how many disk accesses are needed to read 10000th byte of file.

(a) 1
(b) 2
(c) 3
(d) 4
(e) Can't say

○ A

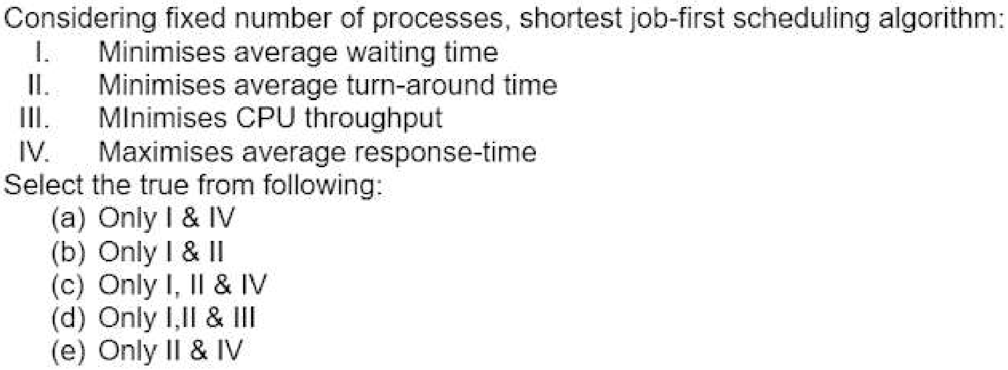
○ B

● C

○ D

○ E

*



Considering fixed number of processes, shortest job-first scheduling algorithm:

I. Minimises average waiting time
II. Minimises average turn-around time
III. MInimises CPU throughput
IV. Maximises average response-time

Select the true from following:

(a) Only I & IV
(b) Only I & II
(c) Only I, II & IV
(d) Only I,II & III
(e) Only II & IV

● A

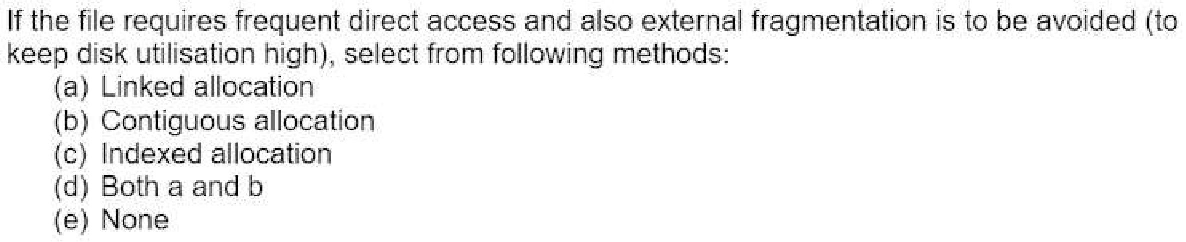
○ B

○ C

○ D

○ E

*



If the file requires frequent direct access and also external fragmentation is to be avoided (to keep disk utilisation high), select from following methods:

(a) Linked allocation
(b) Contiguous allocation
(c) Indexed allocation
(d) Both a and b
(e) None

○ A

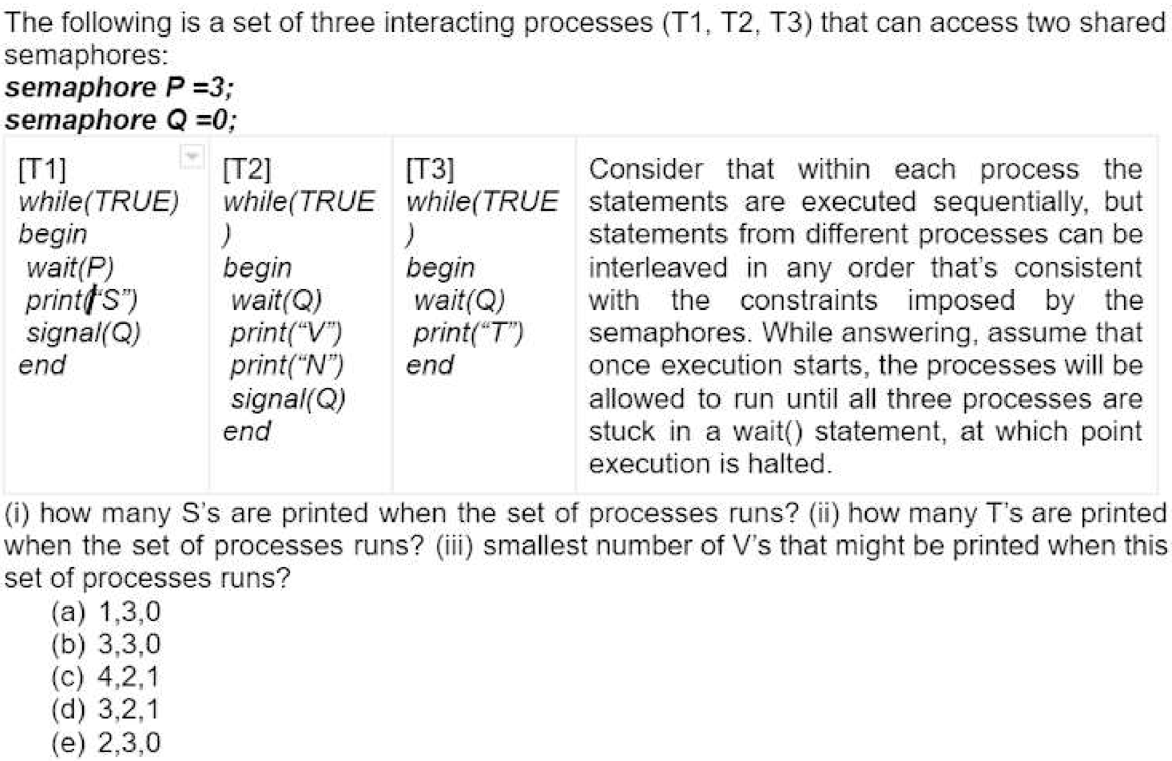
○ B

○ C

○ D

● E

*



The following is a set of three interacting processes (T1, T2, T3) that can access two shared semaphores:

semaphore P =3;

semaphore Q =0;

[T1]
while(TRUE)
begin
wait(P)
print(“S”)
signal(Q)
end
	[T2]
	while(TRUE
	)
	begin
	wait(Q)
	print(“V”)
	print(“N”)
	signal(Q)
	end
		[T3]
		while(TRUE
		)
		begin
		wait(Q)
		print(“T”)
		end
			Consider that within each process the statements are executed sequentially, but statements from different processes can be interleaved in any order that’s consistent with the constraints imposed by the semaphores. While answering, assume that once execution starts, the processes will be allowed to run until all three processes are stuck in a wait() statement, at which point execution is halted.

(i) how many S’s are printed when the set of processes runs? (ii) how many T’s are printed when the set of processes runs? (iii) smallest number of V’s that might be printed when this set of processes runs?

(a) 1,3,0
(b) 3,3,0
(c) 4,2,1
(d) 3,2,1
(e) 2,3,0

○ A

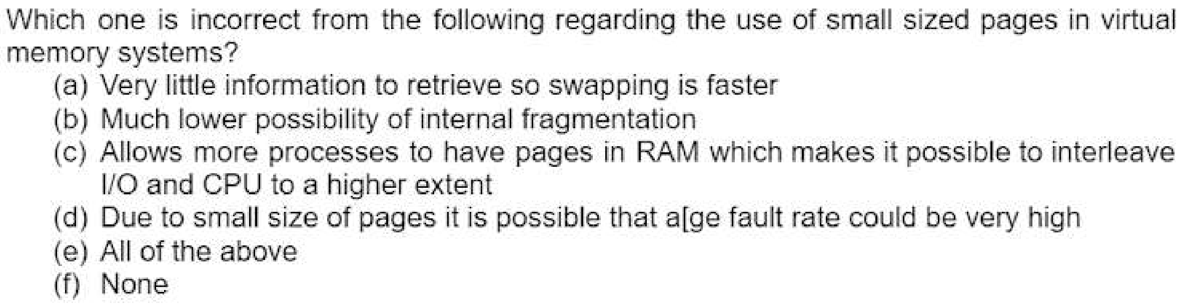
● B

○ C

○ D

○ E

*



Which one is incorrect from the following regarding the use of small sized pages in virtual memory systems?

(a) Very little information to retrieve so swapping is faster
(b) Much lower possibility of internal fragmentation
(c) Allows more processes to have pages in RAM which makes it possible to interleave I/O and CPU to a higher extent
(d) Due to small size of pages it is possible that a[ge fault rate could be very high
(e) All of the above
(f) None

○ A

● B

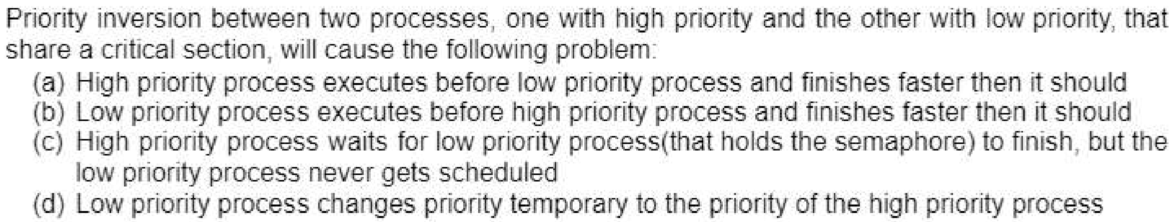
○ C

○ D

○ E

○ F

*



Priority inversion between two processes, one with high priority and the other with low priority, that share a critical section, will cause the following problem:

(a) High priority process executes before low priority process and finishes faster then it should
(b) Low priority process executes before high priority process and finishes faster then it should
(c) High priority process waits for low priority process(that holds the semaphore) to finish, but the low priority process never gets scheduled
(d) Low priority process changes priority temporary to the priority of the high priority process

○ A

○ B

● C

○ D

Back Submit Clear form

Never submit passwords through Google Forms.