MADE EASY
India's Best Institute for IES, GATE & PSUs

Harsh Kasyap (Enroll User) | Log Out

HOME
MY TEST
BOOKMARKS
MY PROFILE
REPORTS
BUY PACKAGE
NEWS
Ask an Expert

SCORE CARD | TIME MANAGEMENT | QUESTIONS REPORT | SOLUTION | COMPARE YOUR SELF

Solution Report For Operating System-1

Represent whole test solution with correct and incorrect answers.

View Your Test Analysis

Q. No	Question Status
Q.1	**Assume a page reference string for a process with 3 frames (initially all are empty). The page reference string has length 52 with 4 distinct page number occurring in it. For any page replacement algorithm, what is a lower bound and upper bound on the number of page faults?**

a. 2, 26
b. 3, 4
c. 4, 26
d. 4, 52

Not Attempt | **Correct Ans.** | d | FAQ? | Have any doubt? | bookmark

Solution. 1

(d)

Since 4 distinct page numbers are only to be accessed. Hence the best condition i.e., the condition with minimum number of page faults will be accessing all those elements repeatedly that are in the frame already, which will give maximum 4 page faults. If, considered the worst case, it will be on every iteration, we are accessing the same element that has been removed from the frame, which will give 52 page faults.

Q.2

A single processor system has five resource types A, B, C, D and E which are shared by four processes. The current allocation and maximum needs are as follows:

Process	Allocated					Maximum					Available				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
P_0	1	0	2	1	1	1	1	2	1	3	0	0	1	2	3
P_1	2	0	1	1	0	2	2	2	1	0					
P_2	1	1	0	1	0	2	1	3	1	0					
P_3	1	1	1	1	0	1	1	2	2	1					

Which of these processes will finish LAST?

a. P_0

b. P_1

c. P_2

d. None because system is in deadlock

Attempt Correct **Correct Ans.** b FAQ? Have any doubt? bookmark

Solution. 2

(b)

Calculating the need matrix

	A	B	C	D	E
P_0	0	1	0	0	2
P_1	0	2	1	0	0
P_2	1	0	3	0	0
P_3	0	0	1	1	1

Since, available = 00123, hence only P_3 can be satisfied.
Remaining = (00123) – (00111) = (00012) + (11221) = (11233)
Now P_0 can be executed,
Remaining = (11233) – (01002) = (10231) + (11213) = (21444)
Now P_2 can be executed,
Remaining = (21444) – (10300) = (11144) + (21310) = (32454)
Now P_1 can be executed.

Q.3

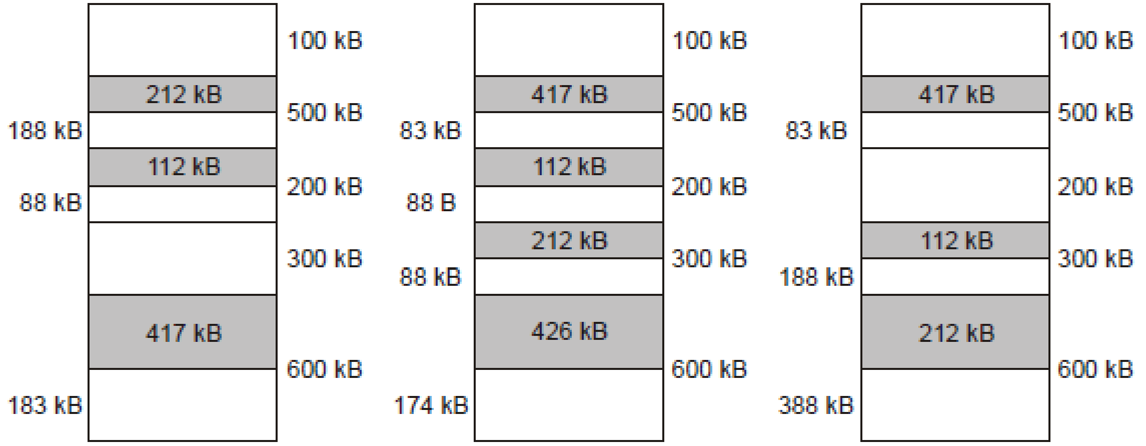
Consider a scenario, having fixed partition allocation scheme where each partition is of size 100 kB, 500 kB, 200 kB, 300 kB and 600 kB. In these partitions, the processes that are needed to be placed are 212 kB, 417 kB, 112 kB and 426 kB (in order). How many partitions will remain unallocatted after placing the processes in each of first fit, best fit and worst fit algorithm respectively. Also, which policy(s) will be successful in placing all these processes.

a. 4, 1, 2 ; Best fit and First fit
b. 2, 1, 2 ; Best fit
c. 4, 2, 3 ; Best fit
d. 3, 2, 3 ; Best fit and First fit

Attempt | Correct | **Correct Ans.** | b | FAQ? | Have any doubt? | bookmark

Solution. 3

(b)



First fit
Unallocated partition = 2
{1 process left for placed}

Best fit
Unallocated partition = 1
{all process placed}

Worst fit
Unallocated partition = 2
{1 process left for placed}

Q.4

Consider the virtual page reference string:

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

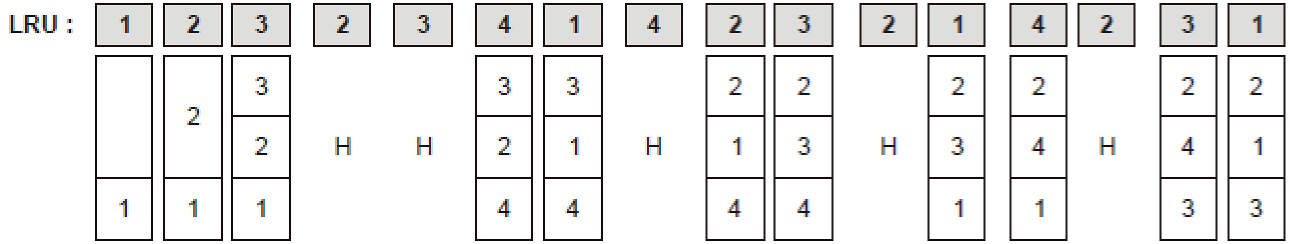
on a demand paged virtual memory system running on computer system that has main memory size of 3 page frames which are initially empty. Let LRU, FIFO and OPTIMAL denote the number of page faults under the corresponding page replacement policy. Then

a. OPTIMAL < LRU < FIFO
b. OPTIMAL < FIFO < LRU
c. OPTIMAL = LRU
d. OPTIMAL = FIFO

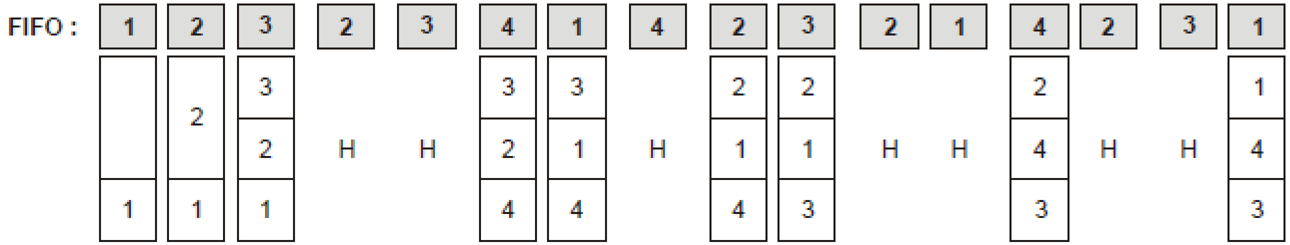
Attempt | Correct | **Correct Ans.** | b | FAQ? | Have any doubt? | bookmark

Solution. 4

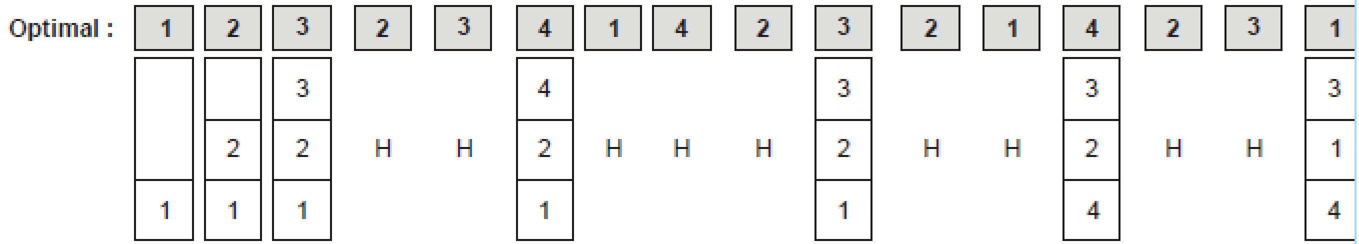
(b)



Total page faults = 11



Total page faults = 9



Total page faults = 7

Q.5

A CPU generates 64 bits virtual address. Page size is of 8 KB. The processor has a translation-look a side-Buffer (TLB) which can hold a total of 256 page table entries and is 4-way set associative. The minimum size of TLB tag is __________ bits.

Attempt Correct **Correct Ans.** 45 FAQ? Have any doubt? bookmark

Solution. 5

45

$$\begin{aligned}
\text{Page Size} &= 8\text{ K} \\
\text{Offset bits} &= 13 \\
\text{Virtual Address} &= 64 \text{ bits} \\
\text{Remaining bits} &= 64 - 13 = 51 \text{ bits} \\
\text{Number of sets} &= \frac{256}{4} = 64 = 6 \text{ bits} \\
\text{Tag bits} &= 51 - 6 = 45 \text{ bits}
\end{aligned}$$

Q.6 **Suppose 10 processes P_1 to P_{10} share 7 identical resource units which can be reserved and release 1 at are time the maximum resource requirement of a processs P_i is S_P where S_P is greater then 0. The maximum value of** $S_i\left(\sum_{i=1}^{10} S_P\right)$ **that ensures deadlock does not occurs is ________ .**

Attempt | Incorrect | **Your Ans.** 64 | **Correct Ans.** 16 | FAQ? | Have any doubt? | bookmark

Solution. 6

16

In order to ensure a deadlock free system,

Sum of resource needs < [Number of resources + Number of processes] < [7 + 10] < 17

Maximum value that can be used is 16.

Q.7

A demand paging system has page fault service time as 125 time units if page is not dirty and 400 times units of page fault service time if it is a dirty page. Memory access time is 10 time units. The probability of a page fault is 0.3. In case of page fault, the probability of page being dirty is P. It is observed that average access time is 50 time units. Then, the value of P is ______? [upto four decimal places]

Not Attempt | **Correct Ans.** | **0.0667** | FAQ? | Have any doubt? | bookmark

Solution. 7

0.0667

$$
\begin{aligned}
50 &= 0.3\,(P * 400 + (1-P) * 125) + 0.7 * 10 \\
\Rightarrow \quad 50 &= 0.3(400\,P + 125 - 125\,P) + 7 \\
\Rightarrow \quad 43 &= 0.3\,(275\,P + 125) \\
\Rightarrow \quad 43 &= 82.5\,P + 37.5 \\
\Rightarrow \quad 43 - 37.5 &= 82.5\,P \\
\Rightarrow \quad 5.5 &= 82.5\,P \\
\Rightarrow \quad P &= \frac{5.5}{82.5} = 0.0667
\end{aligned}
$$

Q.8

Assume 2 processes computer ( ) and science ( ) that are concurrent and that the three semaphore mutex, Q and R initialized to 1 are shared between the two processes. Q is a semaphore on file 1 and R on file 2.

```
Computer ( )                Science ( )
{                           {
    P(mutex);                   P(Q);
    P(Q);                       /*read from file1*/
    /*write to file1 */     P(R);
    P(R);                       /*write to file2*/
    /*write to file2*/      V(Q);
    V(Q);                       V(R);
    V(mutex);                   P(mutex);
    /*do something              P(Q);
    P(Q);                       /*write to file1 */
    /*read fro file1 */     V(Q);
    /*write to file2 */         V(mutex);
    V(R);
    V(Q);
}
```

Which of the following holds by above process?

a. Deadlock and no starvation
b. No deadlock but starvation
c. Both deadlock and starvation
d. No deadlock or starvation

Not Attempt | **Correct Ans.** | c

FAQ? | Have any doubt? | bookmark

Solution. 8

(c)

1. Computer ( ) → p (mutex) → mutex = 0
   p (Q) → Q = 0
2. Science ( ) → p (Q) → process sleep
3. Computer ( ) → p (R) → R = 0
   v(Q) → Q = 1, science ( ) awake
4. Science → p(Q); Q = 0; p(R) → process sleep
5. Computer → v(mutex) → mutex = 1
   p(Q) → process sleep

Hence a deadlock.

Q.9

A system has five processes and four allocatable resources. The current allocation and maximum needs are as follows:

Process	Allocated				Maximum				Available			
	X	*Y*	*Z*	*W*	*X*	*Y*	*Z*	*W*	*X*	*Y*	*Z*	*W*
P_0	1	0	2	0	3	2	4	2	a	0	0	b
P_1	0	3	1	2	3	5	1	2				
P_2	2	4	5	1	2	7	7	5				
P_3	3	0	0	6	5	5	0	8				
P_4	4	2	1	3	6	2	1	4				

What is the smallest value of a, b for which the system is in a safe state?

a. a = 2, b = 2
b. a = 4, b = 5
c. a = 3, b = 4
d. a = 2, b = 1

Attempt | Correct | Correct Ans. | d | FAQ? | Have any doubt? | bookmark

Solution. 9

(d)

	X	Y	Z	W
P_0	2	2	2	2
P_1	3	2	0	0
P_2	0	3	2	4
P_3	2	5	0	2
P_4	2	0	0	1

Since available is a 0 0 b, let's suppose a takes value 2 and b takes the value 1.

Available = 2 0 0 1

$P_4 \rightarrow$ Complete $\rightarrow$ Avail = (0000 + 6214) = 6214

$P_1 \rightarrow$ Complete $\rightarrow$ Avail = (6214) – (3200) = (3014) + (3512) = (6526)

$P_0 \rightarrow$ Complete $\rightarrow$ Avail = (6526) – (2222) = (4304) + (3242) = (7546)

$P_2 \rightarrow$ Complete $\rightarrow$ Avail = (7546) – (0324) = (7222) + (2775) = (9, 9, 9, 7)

$P_3 \rightarrow$ Complete $\rightarrow$ Avail = (9997) – (2502) = 7495

Hence, the system is in a safe state will value of a as 2 and value of b as 1.

Q.10

Consider the following statements w.r.t. deadlock,

1. An OS implements a policy that requires a process to release all resources before making a request for another resource can lead to starvation but not deadlock.

2. In deadlock prevention, the request for resource is always guaranteed if the resulting state is safe.

3. Deadlock can be prevented, if the resources are numbered uniquely, and processes are allowed to request for resources only in decreasing resource number.

4. Deadlock avoidance has less restrictions than deadlock prevention.

Which of the above statement is/are false?

a. Only 1 and 2
b. Only 1 and 3
c. Only 3 and 4
d. Only 2

Attempt | Incorrect | Your Ans. c | Correct Ans. d | FAQ? | Have any doubt? | bookmark

Solution. 10

(d)
1. The policy is a deadlock prevention policy, but can lead to starvation.
2. In deadlock prevention, one of the four condition for deadlock must not be satisfied. So, state even being safe can't led to successful request.
3. It will help in violating circular wait condition for deadlock.
4. Under deadlock avoidance, just the safe state need to be checked and hence is less restrictive deadlock prevention scheme.

Q.11

Consider the following statements:

S_1 : The total size of address space in a virtual memory system is limited by the available main memory.

S_2 : The best fit techniques for memory allocation ensures the memory will never be fragmented.

S_3 : Locality of reference implies that the page reference being made by a process will always be the page that is being used in the previous page reference

S_4 : Virtual memory reduces the context switching overhead.

How many of the above statements are false?

a. Only S_1 and S_2
b. Only S_1 and S_3
c. Only S_1, S_2 and S_3
d. S_1, S_2, S_3 and S_4

Attempt | Correct **Correct Ans.** | d | FAQ? | Have any doubt? | bookmark

Solution. 11

(d)

• The total size of address space in a virtual memory system is limited by the available secondary storage.

• Best fit technique can also suffer from fragmentation.

• Locality of reference implies that the page reference being made by a process is likely to be the page used in the previous page reference.

• In a system with virtual memory context switch includes extra overhead in switching of address space.

Q.12

Suppose that you wish to design a virtual memory system with the following characteristics:

• Size of page table entry is 4 bytes.

• Each page table must fit into a single page frame.

• System must be able to support virtual address space of 4 GB.

You decided to use a multi-level paging scheme with no more than 2 levels of page tables. What is the minimum page size that the system must have? (Assume last level page table must be fit into single page frame)

a. 2^{10} B
b. 2^{11} B
c. 2^{12} B
d. 2^{13} B

Not Attempt | **Correct Ans.** | c | FAQ? | Have any doubt? | bookmark

Solution. 12

(c)

Let 2^P be the page size.
Since page table entries are 4 bytes in size.

$$1^{st} \text{ Page table size} = \text{Page table entry} \times \text{Page table entry size}$$
$$= \frac{2^{32}}{P} \times 2^2$$
$$= 2^{34-P}$$
$$2^{nd} \text{ Page table size} = \text{Page table entry} \times \text{PTE size}$$
$$= \frac{2^{34-P}}{2P} \times 2^2$$
$$= 2^{36-P-P}$$

Last level page table must be fit into page size
So,

$$2^P = 2^{36-P-P}$$
$$P = 36 - P - P$$
$$3P = 36$$
$$P = 12$$

So page size will be 2^{12} bytes.

Q.13

A system uses optimal policy for a page replacement. It has 4 page frames with no pages loaded to begin with. Consider the following scenario

Case-1: System first accesses 200 distinct pages in sequential order and then access same 200 distinct pages in same order.

Case-2: System first accesses 200 distinct pages in sequential order and then access same 200 distinct page in reverse order.

The difference in the number of faults occurred in both case are _________.

Attempt | **Incorrect** | **Your Ans.** **1** | **Correct Ans.** **0** | FAQ? | Have any doubt? | bookmark

Solution. 13

0

Case-1: System accesses 200 distinct pages. So, all these 200 pages are the page fault, next the pages are accessed again, at that time page number 1, 2, 3 and 200 are in the frame. Now, when 4 will accessed, it will be replaced by 1. Next when 5 will be accessed, it will also be replaced by 2 and so till 199. So, total page faults = 200 + 196 = 396

Case-2: Again after the first access is over 197, 198, 199 and 200 are in the page frame. From 196 to 1 be fault. So, total page faults = 200 + 196 = 396

Difference = 396 – 396 = 0

Q.14 **An operating system uses the Banker's Algorithm for deadlock avoidance when managing the allocation of three resources types *X*, *Y* and *Z* to five processes. Consider the following scenario**

Process	Allocated			Maximum			Available		
	X	Y	Z	X	Y	Z	X	Y	Z
P_0	0	1	0	7	5	3	3	3	2
P_1	2	0	0	3	2	2			
P_2	3	0	2	9	0	2			
P_3	2	1	1	2	2	2			
P_4	0	0	2	4	3	3			

Total possible safe sequences are _________.

Attempt | Incorrect | Your Ans. 6 | Correct Ans. 16 | FAQ? | Have any doubt? | bookmark

Solution. 14

16

Calculating the need matrix

Process	Need		
	X	Y	Z
P_0	7	4	3
P_1	1	2	2
P_2	6	0	0
P_3	0	1	1
P_4	4	3	1

Since, the available resources are $\langle 3, 3, 2\rangle$.

Hence the request can only be satisfied for P_1 or P_3 at initial stage.

Considering P_3 first,

Available after $P_3 \rightarrow \langle 3, 3, 2\rangle - \langle 0, 1, 1\rangle = \langle 3, 2, 1\rangle + \langle 2, 2, 2\rangle = \langle 5, 4, 3\rangle$

After P_3 only, P_1 or P_4 can be executed.

Considering P_1 first, rest all three processes can be scheduled in any way hence 6 possible ways.

After P_3, consider P_4, Next P_1 can only be scheduled, then P_0 and P_2 can be scheduled in any way he
2 possible ways.

Considering P_1 now: Need after $P_1 \rightarrow \langle 3, 3, 2\rangle - \langle 1, 2, 2\rangle = \langle 2, 1, 0\rangle + \langle 3, 2, 2\rangle = \langle 5, 3, 2\rangle$

Now, condition can be satisfied either for P_3 or P_4.

Considering P_3 first, any possible combination on P_0, P_4 and P_2 possible hence 6 sequence.

Considering P_4 first, followed by P_3, then any combination of P_0 and P_1 hence sequence.

$$\text{Total} = 16 \text{ sequences.}$$

Q.15

Consider a demand paging environment where the page fault service time is 14 ms if the free frame is available or the page to be replaced is NOT modified. It takes 18 ms if the page to be replaced is modified. The main memory access time is 180 ns. If the page fault rate is 10%, and the page to be replaced is modified 40% of the time then the effective access time is (in µsec) ________ (upto two decimal places).

Not Attempt | **Correct Ans.** | 1560.16 (1560.15 - 1560.18) | FAQ? | Have any doubt? | bookmark

Solution. 15

1560.16 (1560.15 - 1560.18)

Effective memory access time (EMAT) = $P \times S + (1 - P) \times m$

$$= 0.1\,(0.6 \times 14 \text{ ms} + 0.4 \times 18 \text{ ms}) + 0.90 \times 180 \text{ ns}$$
$$= 0.1\,(0.6 \times 14 \times 10^6 \text{ ns} + 0.4 \times 18 \times 10^6 \text{ ns}) + 0.90 \times 180 \text{ ns}$$
$$= 1560162 \text{ ns}$$
$$= 1560.162\ \mu\text{sec.}$$

Q.16 **Consider the main memory with five page frames and the following sequence of page references: 9, 8, 7, 9, 3, 0, 2, 9, 8, 3, 9, 2, 0, 9. What will be result of (*X* + *Y*) [where *X* is number of page faults using LRU policy and *Y* is number of page faults using FIFO policy]?**

Attempt Correct **Correct Ans.** 15 FAQ? Have any doubt? bookmark

Solution. 16

15

By using LRU policy:

9	8	7	9	3	0	2	9	8	3	9	2	0	9
					0	0	0	0	0	0	0	0	0
				3	3	3	3	3	3	3	3	3	3
		7	7	7	7	7	7	8	8	8	8	8	8
	8	8	8	8	8	2	2	2	2	2	2	2	2
9	9	9	9	9	9	9	9	9	9	9	9	9	9
F	F	F	H	F	F	F	H	F	H	H	H	H	H

7 faults

By using FIFO policy:

9	8	7	9	3	0	2	9	8	3	9	2	0	9
					0	0	0	0	0	0	0	0	0
				3	3	3	3	3	3	3	3	3	3
		7	7	7	7	7	7	8	8	8	8	8	8
	8	8	8	8	8	8	9	9	9	9	9	9	9
9	9	9	9	9	9	2	2	2	2	2	2	2	2
F	F	F	H	F	F	F	F	F	H	H	H	H	H

8 faults

So, [LRU – FIFO] page fault: 7 + 8 = 15