Answers for Memory Management

Which of the following statements about the naive memory allocation approach(es) is(are) true? Select all that apply.

A. Correct. To manage memory by fixed partitions, we have to decide partition boundaries before booting the system.

Feedback: Fixed partitions requires to decide partition boundaries in advance. Therefore, this option is correct.

B. Incorrect. Managing memory by base and limit register need hardware support multiplication.

Feedback: Base and limit registers approach requires hardware support for add and compare to check whether an address is out of boundary. Multiplication is not necessary. Therefore, this option is incorrect.

Which of the following statements about free space management is true. Select the best answer.

A. Correct. A Bitmap is a vector of zeros and ones that specify availability. It is not efficient approach especially for a large memory system.

Feedback: Searching for k consecutive units in a large memory system essentially means searching for k consecutive zeros in a long array. It takes much memory space and time. So this option is correct.

B. Incorrect. The number of nodes in linked list is fixed in advance.

Feedback: The number of nodes in the linked list is not fixed and depends on the holds and blocks in the memory. So this option is not correct.

C. Correct. When a process terminates, the double directions of linked list make is easier to find the previous entry and to see if a merge is possible.

Feedback: The terminated process normally point to the link list entry for the process itself. A double-linked list helps to check and merge the adjacent blocks. So this option is correct.

Which of the following statements about memory allocation strategies is false? Select the best answer.

A. Incorrect. Best-Fit is the best strategy since it always find the smallest block that is big enough.
Feedback: Which algorithm works best is highly workload dependent. Best-Fit is not always be the best strategy. So this option is incorrect.

B. Correct. Worst-Fit is not necessary the worst strategy. Sometimes it helps to reduce the external fragmentation.
Feedback: Worst-Fit always take the largest available hole, so that the new hole will be big enough to be useful. This option is correct.

C. Correct. Best-Fit is slower than First-Fit because it must search the entire list every time it is called.

Feedback: First-Fit returns as long as it finds a hole that is large enough. Therefore this option is correct.

Suppose we have a consecutive chunk of memory of size 128 pages, and there is a request for 13 pages of memory.

What size of memory will the buddy algorithm allocate for the request? How many times should the memory be split to provide such an area? What the size of the internal fragmentation?
Please enter your answer in the space provided below. Your answer should be in the format x; y; z;, where x represents the size of memory, y represents the number of times the memory should be split, and z represents the size of the internal fragmentation.

Answer: 16; 3; 3
Alternate Correct Answer: 16;3;3

Solution:
Round up 13 to the power of 2, which is 16
We need to separate the memory 3 times to get a 16-page memory chunk. 128->64->32->16
The size of internal fragmentation is: 3 = 16 - 13

Is the following statement true or false?

Physical Memory Space is typically much larger than Virtual Memory Space.
A. Incorrect. True
B. Correct. False

Solution: Physical Memory is typically smaller. Physical memory is limited to the size of the RAM used in a system. Virtual memory is stored on disk and is typically larger.

You are interested in designing a new processor architecture with 16-bit addresses. What is the total size of the virtual address space?
A. Incorrect. 16384 Bytes
B. Incorrect. 16384 KB
C. Incorrect. 65536 KB
D. Correct. 65536 Bytes

Solution: The answer will be 2^16 Bytes because we can address 2^16 different byte addresses with 16 bits. (2^16 = 65536)

In a 32-bit computer with a page size of 4 KB, what is the size, in bits, of the virtual address?
Enter your answer as a number in the space provided below. Do not enter any spaces or punctuation.

Answer: 32

Solution:
The computer is 32-bit, which means the virtual memory size is 32-bit long.

In a 32-bit computer with a page size of 4 KB, what is the number of offset bits in the virtual address?
Enter your answer as a number in the space provided below. Do not enter any spaces or punctuation.
Answer: 12

Solution:

The page size is 4 KB, which is 2^12 Bytes. Therefore, the number of page offset bits is 12.

In a 32-bit computer with a page size of 4 KB, how many bits are there in the virtual address (x)? How many bits are the offset bits in the virtual address (y)? What is the size, in bits, of the page number? Enter your answer as a number in the space provided below. Do not enter any spaces or punctuation.

Answer: 20

Solution:
The computer is 32-bit, which means the virtual memory size is 32-bit long. The page size is 4 KB, which is 2^12 Bytes. Therefore, the number of page offset bits is 12. The rest of the bits in the virtual memory (32 - 12) is the page number bits, which means there are 20 page number bits.

Given the page table (first table) and virtual address (second table), what is the physical address after the page table translation? Enter your answer as number in the space provided below. Do not add any spaces or punctuation.

15	000	0
14	000	0
13	000	0
12	000	0
11	111	1
10	000	0
9	101	1
8	000	0
7	000	0
6	000	0
5	011	1
4	100	1
3	000	1
2	110	1
1	001	1
0	010	1

Page Table

1	0	0	1	0	0	1	0	1	0	0	0	0	1	0

Virtual Address

Correct Answer: 10100101000010

Feedback: There are 16 rows in the page table, so the first 4 bits in the virtual address are shaded as the index bits. Binary 1001 is equal to 9 in decimal. so, the first three bits of physical address should be 101. The offset bits are copied directly from the virtual address. Therefore the physical address should be 10100101000010.

Given the page table of process A and the swap file, please enter the page numbers mapped to each block in the swap file correspondingly. If a block in swap file is not used, enter “free”. If a block is used by page 3 of process A, please present if by A(3). Separate each page number with a comma and do not add any spaces. For example, one possible answer you could enter is A(0),free,A(1),free.

Page Table of process A

Page #	In-memory?	Frame#	Disk#
0	Y	0	-
1	N	-	B1
2	N	3	B3
3	Y	-	-

Swap File

Bitmap: 0101			

Swap File

Bitmap: 0101			
B0	B1	B2	B3

Solution: free,A(1),free,A(2)
Other Correct Answer: free, A(1), free, A(2)
Feedback: Based on the bitmap, the first and third blocks are empty. So they should be filled with “free”. In the page table, page 1 is mapped to the B1 in the swap file and page 2 is mapped to the B3 in the. Therefore the answer should be free, A(1), free, A(2).

In a 32-bit operating system with an 8 KB page size, there are 9 bits in the top-level table and 10 bits in the second-level table. If a process uses 128 MB virtual memory, how many entries it will accept in the top-level table? Please enter your response as a number in the space provided below. Do not use any spaces or punctuation.

Solution: 16
Feedback: There are 10 bits in the second-level table and the page size is 8 KB. So, each second-level table can index 8 MB memory. The process needs 128 MB memory, so it needs to be able to index 16 second-level tables. Therefore, there will be 16 entries in the top-level page table.

When will MMU evict an entry from TLB? Select all that apply.

A. Correct. When a requested virtual page number is missed in the TLB and TLB is full.
Feedback: After a TLB miss, MMU will evict one of the entry from TLB and replaces with the new entry from the page table. So this option is correct.

B. Incorrect. When a requested virtual page number is hit in the TLB.
Feedback: MMU will not evict any entry after TLB hit since MMU does not access page table after all. So this option is not correct.

C. Correct. When an entry in the page table is evicted from memory to swap file and the same address is also evicted from the TLB.
Feedback: During a page fault, MMU will use page replacement policy to swap a page from memory to disk, if the address of the page is also stored in TLB. It should be evicted as well. So this option is correct.

Is the following statement true or false?

Inverted page tables takes up less space than regular page tables.
A. Correct. True.
B. Incorrect. False.

Solution: The size of a regular page table grows with the size of the virtual address space. If we have a large virtual address space (such as in a 64 bit architecture), the page table will become huge. Inverted page tables size depends on physical memory size and is therefore smaller.

Is the following statement true or false?

Swapping a page is an expensive process because in addition to swapping in the new page we need to write the eviction candidate back to disk.

A. Correct. True
B. Incorrect. False

Solution: Swapping is an expensive operation and will even block the current process. If a page in memory is “dirty” (ie. contains recent writes), and this block is selected for eviction, the changes need to be written to disk and the new page needs to be brought to memory. All of these operations can make page swapping an expensive process.

Is the following statement true or false?

FIFO replacement scheme is a theoretically optimal algorithm.
A. Incorrect. True
B. Correct. False

Solution: According to the FIFO scheme, on a page fault, the frame that has been in memory the longest is replaced. This is not an optimal algorithm, in fact FIFO suffers from Bélády's anomaly, the phenomenon in which increasing the number of page frames results in an increase in the number of page faults for certain memory access patterns. In FIFO, the page fault may or may not increase as the page frames increase, but in Optimal algorithms, as the page frames increase the page fault decreases.

The second chance algorithm is looking for a candidate to evict. It looks at the front of the queue and encounters a page A with reference bit set to 1. What is the next step? Select the best answer
A. Incorrect. Evict A.
B. Incorrect. Decrement the reference bit and evict A.
C. Correct. Decrement the reference bit and add to back of queue, look for another candidate.
D. Incorrect. Increment the reference bit and add to back of queue, look for another candidate.

Solution:The second chance algorithm looks in the queue for the first page with reference bit set to 0, this page is the candidate for eviction. If it finds a page with reference bit set to 1 it will decrement this value to 0 and add the to the back of the queue.

Given the 8-bit counter for four pages provided below, which of the following should be evicted if the LRU Aging scheme is applied. Select the best answer.
The 8-bit counter for 4 pages are provided below.

A. Incorrect. Page A - 100001
B. Correct. Page B - 001010
C. Incorrect. Page C - 010100

Solution: The page with the highest value of the 8-bit counter is the page that was most recently accessed.

In this example Page A was the page that was most recently accessed and Page B was accessed least recently, therefore Page B would be the candidate for eviction.

Which of the following page types is most likely to be evicted? Select the best answer.
A. Incorrect. Swappable
B. Incorrect. Unreclaimable
C. Correct. Discardable

Solution: Discardable Pages are not in-use and should be evicted first.

Which of the following page types is least likely to be evicted? Select the best answer.
A. Incorrect. Swappable
B. Incorrect. Unreclaimable
C. Incorrect. Discardable

Solution: Unreclaimable Pages refers to the kernel stack. These cannot be evicted.

Is the following statement true or false?

Consider a page p stored in physical frame f. When accessing page p, the corresponding TLB will only get updated in the event of a TLB miss for p.
(multiple choice question with ONE correct answer) → 1 point
A. Incorrect. True
B. Correct. False
It is possible for a TLB update to happen even when there is a TLB hit, because the dirty bit and reference bit can get updated.

Is the following statement true or false?

The best-fit memory allocation scheme always results in less fragmentation and satisfies more block requests compared to the first-fit scheme.
(multiple choice question with ONE correct answer) → 1 point
C. Incorrect. True
D. Correct. False
In practice, it turns out that there is no fixed consensus on which is the best technique. This is highly dependent on the workload.

Is the following statement true or false?

The TLB is motivated by the fact that accessing the disk is much slower than accessing main memory.
(multiple choice question with ONE correct answer) → 1 point
E. Incorrect. True

F. Correct. False
TLB is used to solve the problem that page tables are stored in memory, which is much slower than accessing cache. TLB itself is a small cache that we maintain in hardware.

4. Is the following statement true or false?

The LRU page replacement policy is a stack algorithm.
(multiple choice question with ONE correct answer) → 1 point

G. Correct. True
H. Incorrect. False

LRU is an example of a stack algorithm, because whenever we increase the amount of memory, we never throw away the last m, most recently used pages.

5. Is the following statement true or false?

When the degree of multiprogramming increases, the CPU utilization will always increase and eventually saturate at 100%.
(multiple choice question with ONE correct answer) → 1 point

I. Incorrect. True
J. Correct. False

When we run too many applications, too many page faults can happen which will cause processes to block, and the CPU utilization will plunge at that point in time.

6. Which is the following correctly reflects the maximum number of entries in an inverted page table?
(multiple choice question with ONE single correct answer) → 1 point

A. Correct. Number of page frames in physical memory
B. Incorrect. Number of virtual pages of the currently running processes
C. Incorrect. Total number of virtual pages of all processes (blocked, ready, running).

Solution:
Inverted page table only indexes active pages currently in physical memory.

7. In a 64-bit machine that has 4 KB page size, how many entries are there in the page table?

Please enter your answer as an exponent of the number 2.
For example: if your answer is 1048576, enter 2^20 as your answer in the space provided below. Do not use spaces or punctuation. → 1 point
(free response question with numeric response)

Correct answer: 2^52
Alternate correct answer: 2 ^ 52
Alternate correct answer: 2 ^52
Alternate correct answer: 2^ 52

Explanation:
Each entry contains a page of size 4 KB. 4 KB = 4 * 2^10 B = 2^12 B. Thus we need to use the lower 12 bits of the virtual address to index into the page.

For a 64-bit machine, each address contains 64 bits. Thus the remaining (64-12=52) bits are used to index into the page table, which means the table has 2^52 entries