Main Memory Sheet

Q1) Consider a simple system running a single process. The size of physical frames and logical pages is 16 bytes. The RAM can hold 3 physical frames. The virtual addresses of the process are 6 bits in size. The program generates the following 20 virtual address references as it runs on the CPU: 0, 1, 20, 2, 20, 21, 32, 31, 0, 60, 0, 0, 16, 1, 17, 18, 32, 31, 0, 61. (Note: the 6-bit addresses are shown in decimal here.) Assume that the physical frames in RAM are initially empty and do not map to any logical page.

a. Translate the virtual addresses above to logical page numbers referenced by the process. That is, write down the reference string of 20 page numbers corresponding to the virtual address accesses above. Assume pages are numbered starting from 0, 1, ...

b. Calculate the number of page faults generated by the accesses above, assuming a FIFO page replacement algorithm. You must also correctly point out which page accesses in the reference string shown by you in part (a) are responsible for the page faults.

c. Repeat (b) above for the LRU page replacement algorithm.

d. What would be the lowest number of page faults achievable in this example, assuming an optimal page replacement algorithm were to be used? Repeat (b) above for the optimal algorithm.

Q2) Consider a system with a 6 bit virtual address space, and 16 byte pages/frames. The mapping from virtual page numbers to physical frame numbers of a process is (0,8), (1,3), (2,11), and (3,1). Translate the following virtual addresses to physical addresses. Note that all addresses are in decimal. You may write your answer in decimal or binary.

(a) 20

(b) 40

Q3) Consider a system with 8-bit virtual and physical addresses, and 16 byte pages. A process in this system has 4 logical pages, which are mapped to 3 physical pages in the following manner:

logical page 0 maps to physical page 6, 1 maps to 3, 2 maps to 11, and logical page 5 is not mapped to any physical page yet. All the other pages in the virtual address space of the process are marked invalid in the page table. The MMU is given a pointer to this page table for address translation. Further, the MMU has a small TLB cache that stores two entries, for logical pages 0 and 2. For each virtual address shown below, describe what happens when that address is accessed by the CPU. Specifically, you must answer what happens at the TLB (hit or miss?), MMU (which page table entry is accessed?), OS (is there a trap of any kind?), and the physical memory (which physical address is accessed?). You may write the translated physical address in binary format. (Note that it is not implied that the accesses below happen one after the other; you must solve each part of the question independently using the information provided above.

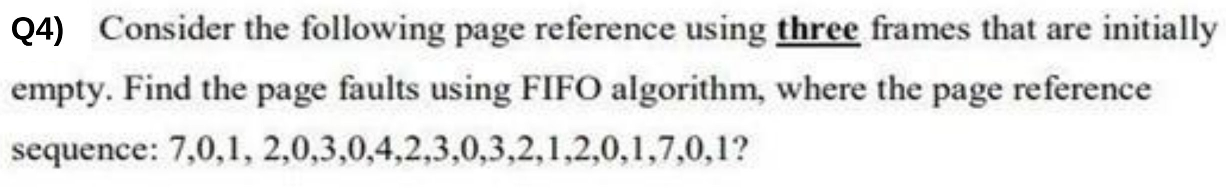
(a) Virtual address 7

(b) Virtual address 20

(c) Virtual address 70

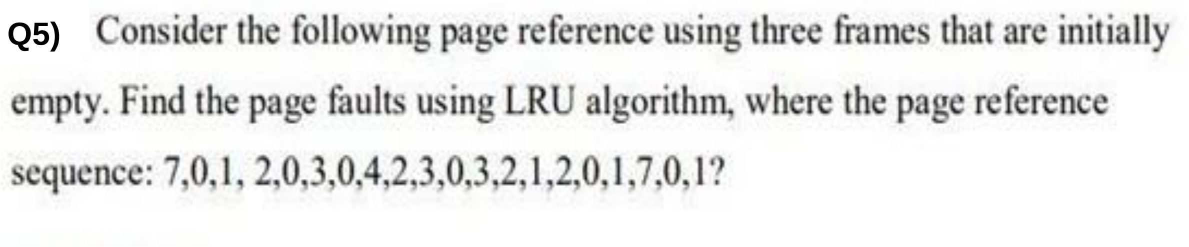
(d) Virtual address 80

Q4)



Consider the following page reference using **three** frames that are initially empty. Find the page faults using FIFO algorithm, where the page reference sequence: 7,0,1, 2,0,3,0,4,2,3,0,3,2,1,2,0,1,7,0,1?

Q5)



Consider the following page reference using three frames that are initially empty. Find the page faults using LRU algorithm, where the page reference sequence: 7,0,1, 2,0,3,0,4,2,3,0,3,2,1,2,0,1,7,0,1?

Q6) Logical Address Space (LAS) = 128KB, Physical Address Space (PAS) = 512KB and page size = 16KB.

Calculate:

a. Number of bits for Logical Address (LA)

b. Number of bits for Physical Address (PA)

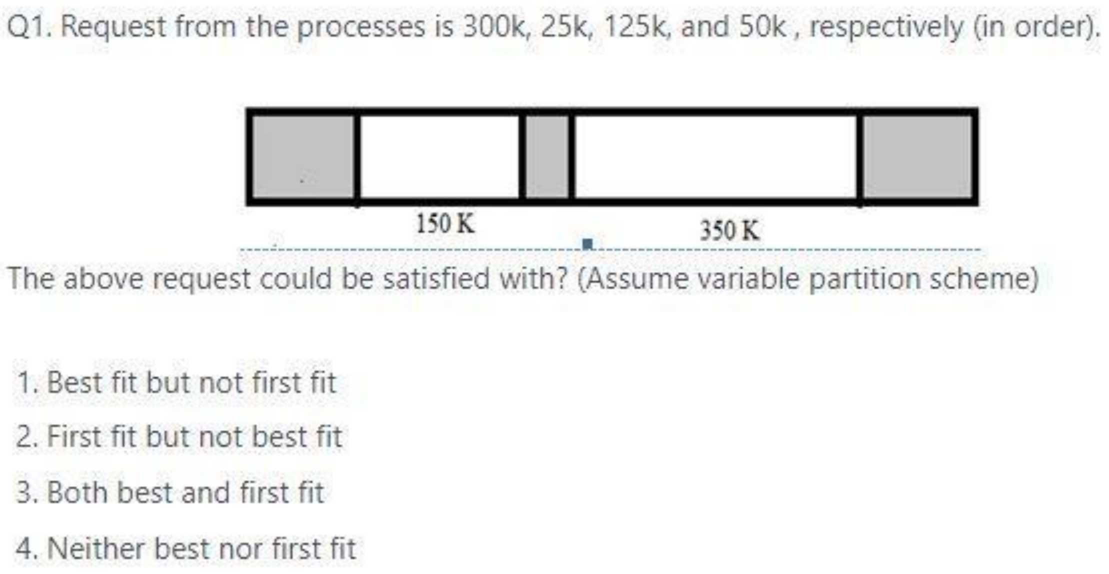
c. Number of pages in LAS or process.

d. Number Frames in main memory or PAS.

e. Page Table size.

Q7) Given memory partitions of 100K, 500K, 200K, 300K and 600K (in order), how would each of the First-fit, Best-fit, and Worst-fit algorithms place processes of 212K, 417K, 112K and 426K (in order)? Which algorithm makes the most efficient use of memory?

Q8)



Q1. Request from the processes is 300k, 25k, 125k, and 50k , respectively (in order).

The above request could be satisfied with? (Assume variable partition scheme)

1. Best fit but not first fit
2. First fit but not best fit
3. Both best and first fit
4. Neither best nor first fit

Q9) Consider the following system:

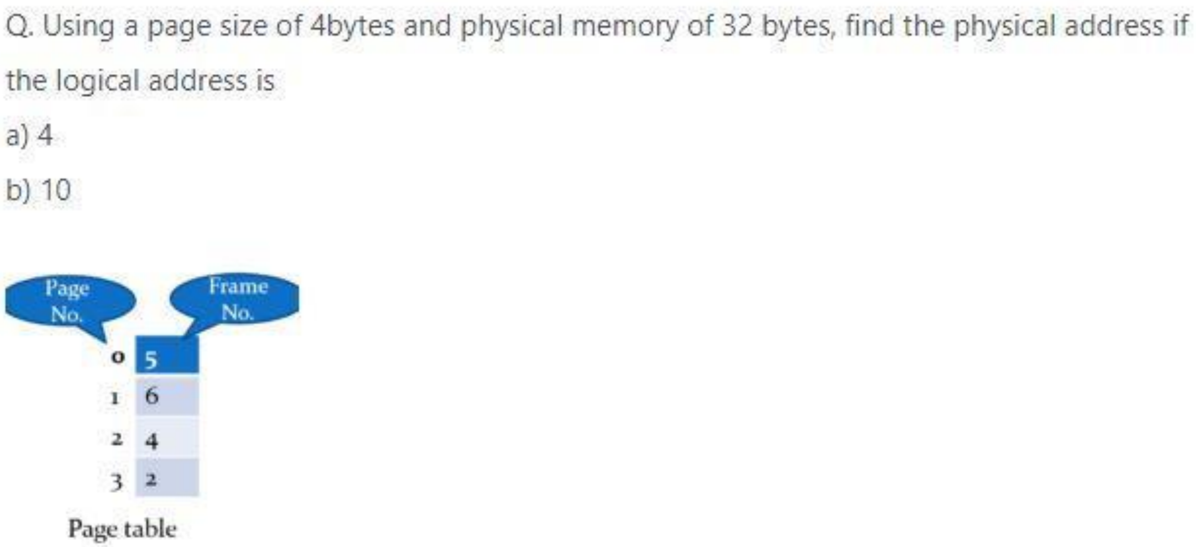
Number of pages = 2k
Page size = 4k words
Physical address = 18 bits
Calculate the logical address space (LAS) and number of frames.

Q10) Consider a system with LA = 32 bits, physical address space (PAS) = 64 MB and page size is 4 KB. The memory is byte addressable. Page table entry is 2 bytes. What is the approximate page table size?

Q11) Consider a system with TLB access time 20 ns and main memory access time 100 ns. Calculate effective memory access time if TLB hit ratio is 95%.

Q12) Consider a system with page fault service time (s) = 100 ns, main memory access time (M) = 20 ns, and page fault rate (P) =65%. Calculate effective memory access time.

Q13)



Q. Using a page size of 4bytes and physical memory of 32 bytes, find the physical address if the logical address is

a) 4

b) 10

Page No.	Frame No.
0	5
1	6
2	4
3	2

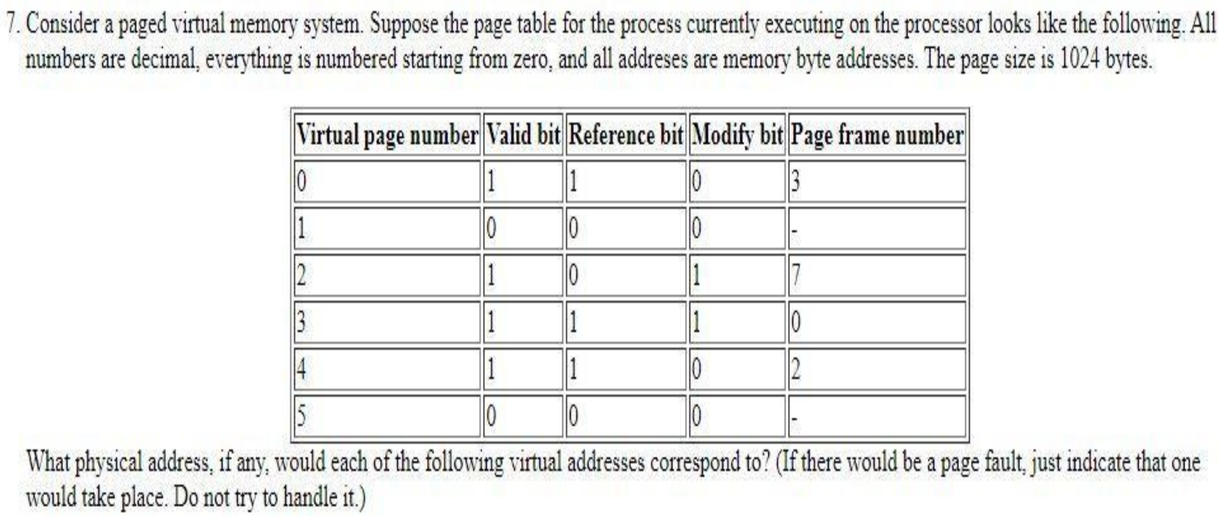
Page table

Q14) **Answer each question by marking a "T" or "F" next to the statement.**

1. External fragmentation is a problem with paged memory systems.
2. Internal fragmentation is a problem with paged memory systems.
3. Memory compaction is employed to counteract internal fragmentation.
4. Memory compaction is employed to counteract external fragmentation.
5. Dynamic relocation requires memory compaction.
6. Memory compaction requires dynamic relocation.
7. First-Fit is a replacement algorithm.
8. Best-Fit generally causes less fragmentation than First-Fit.
9. A function of a linker is to combine several object modules into a single load module.
10. A function of a linker is to replace absolute references in an object module by symbolic references to locations other modules.
11. Demand paging is a placement policy.
12. Hashing is a fetch policy.
13. The last process activated is most likely to have its entire working set resident.
14. The process that caused the most recent page fault is a good choice to suspend (swap the entire process out) when suspension is called for.
15. The 50% criterion is a rule used to decide which page to replace.
16. Fixed allocation with global replacement is not a workable combination for a resident set management scheme.
17. The term TLB stands for Translation Lookaside Buffer.
18. TLB is a cache for page table entries.
19. Without paging, a segmented memory system suffers external fragmentation.
20. Memory segmentation can be used for protection, allowing different privileges to a process for different memory segments.

Q15) Explain the differences between the meanings of the terms logical address, relative address, and physical address.

Q16)



7. Consider a paged virtual memory system. Suppose the page table for the process currently executing on the processor looks like the following. All numbers are decimal, everything is numbered starting from zero, and all addreses are memory byte addresses. The page size is 1024 bytes.

Virtual page number	Valid bit	Reference bit	Modify bit	Page frame number
0	1	1	0	3
1	0	0	0	-
2	1	0	1	7
3	1	1	1	0
4	1	1	0	2
5	0	0	0	-

What physical address, if any, would each of the following virtual addresses correspond to? (If there would be a page fault, just indicate that one would take place. Do not try to handle it.)