

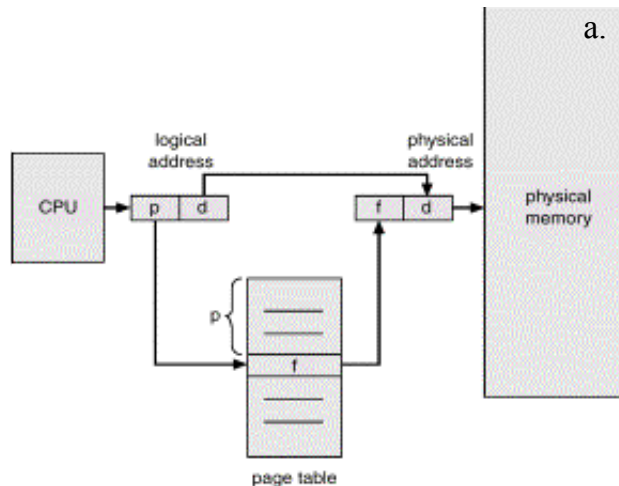
Quiz #3 (November 25, 2003)**Name:** Solution for your reference

COP4610 CGS5765 Recitation Section: _____ Last 4 SSN: _____

COP4610, Department of Computer Science, Florida State University, Fall 2003

Points: 100 points (Maximum 110 points)

1. (60 points) Suppose that we have a computer system with a **32-bit** logical address using paging for memory management. The page size is **4K** bytes and a page table entry in all page tables (page tables at different levels in multi-level paging) is **4** bytes.



- a. (15 points) Draw a diagram to illustrate how to translate a logical address to the corresponding physical address.

To translate a logical address to the corresponding physical address, we divide the logical address into page number and offset, we use the page number as index to the page table and the entry gives the corresponding frame number. We concatenate frame number with the offset, which gives us the corresponding physical address.

- b. (25 points) Suppose that single-level paging is used, calculate the size in bytes of the page table, and number of bits in each field in the logical address.

Here the page size is $4K = 2^{12}$ and so the offset is 12 bits, the page number is $32 - 12 = 20$ bits. The page table in bytes is $2^{20} * 4 = 2^{24} = 4$ Mega bytes.

- c. (10 points) Suppose we use a two-level paging, how many memory cycles do we need to fetch an instruction? How to improve the performance?

To map a logical address in a two-level paging, we need two memory cycles to find the corresponding entries in the page tables. Plus one more cycle to fetch the instruction, we need in total three memory cycles to fetch an instruction. To improve the performance, we use a translation lookaside buffer to avoid accessing page tables

- d. (10 points as extra credit points) Suppose we use two-level paging and arrange for all of the page tables to fit into a single page frame. How will the bits of the logical address be divided up then? Justify your answer.

Each page frame is 4 K bytes = 1 K page table entries and the number of bits in both page number fields can not have more than 10 bits. In this example, we have 20 bits in total and 10 bits in each level. So a logical address will be divided up as

10 10 12

2. (50 points, 25 points each) In a demand paging system, consider the following page reference string:

1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6

Further assume that there are four frames in the system, where all of them are initially empty. Answer the following questions.

- a. For LRU (least recently used) page replacement algorithm, fill the table where each column represents the frames after each page reference and indicate a page fault using a * in the corresponding entry. Then specify how many page faults it generates.

	1*	1	1	1	1	1	1	1	1	1	1	1	1	6*	6	6	6	6	6	6
		2*	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
			3*	3	3	3	5*	5	5	5	5	3*	3	3	3	3	3	3	3	3
				4*	4	4	4	6*	6	6	6	6	7*	7	7	7	1*	1	1	1
	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6

It generated 10 page faults.

- b. For the optimal page replacement algorithm, fill the table where each column represents the frames after each page reference and indicate a page fault using a * in the corresponding entry. Then specify how many page faults it generates.

	1*	1	1	1	1	1	1	1	1	1	1	1	7*	7	7	7	1*	1	1	1
		2*	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
			3*	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
				4*	4	4	5*	6*	6	6	6	6	6	6	6	6	6	6	6	6
	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6

It generated 8 page faults.