CSE 2431/5431 **HOMEWORK 3** **SPRING 2025**

Instructor: Luan Duong, Ph.D.

Electronic Submission: 11:59 pm, Monday Mar 24, 2025

Point: 25 points (5% of total grade)

Notes:

K = 2^{10}; M=2^{20}; G=2^{30}; index always starts from 0 (e.g. page 0 is the first page)
All memory is byte addressable (unless otherwise stated in the question)
If a number has no special mark (i.e. 3251), it is a decimal number.
If a number ends with 'b' (i.e. 01100000b), it is a binary number.
If a number starts with '0x' (i.e. 0xFF10), it is a hexadecimal number.

Question 1: Paging [11 points]

Let's try paging again! Suppose 16-bit addresses are used for both virtual address and physical address. Suppose page/frame size is 256 bytes.

1) How many bits are used for page number and how many bits are used for the offset? **[0.5 point]**

2) What is the maximum number of pages a process can have? **[0.5 point]**

3) Suppose each entry in the page table takes 4 bytes (including frame number, valid bit, and some other bits). Suppose an OS uses an array to store the page table. What is the size of a page table? **[0.5 point]**

4) Suppose the first 3 pages of a process are mapped to frames 100 to 102, and the last 3 pages of the process are mapped to frames 5 to 7. All other pages are invalid. Draw the page table (include valid bit and frame number). **[2 points]**

5) Translate the following virtual addresses to physical addresses (show how you get the answers): **[2 points]** Note: same as your mid-journey test, you do not need to convert them to decimal numbers. Use the simplified calculation.

a) 0x01CC

b) 0x02FF

c) 0x0301

d) 0xFF01

6) Now suppose the OS is using two-level page table. Draw the page table. Suppose frames 8 to 99 are free so that you can allocate space for the page table there. Also suppose the page table directory always takes several consecutive full frames (e.g. If it is 2 bytes, it takes one frame; if it is 257 bytes, it takes 2 consecutive frames.) **[3 points]**

7) What is the size of this two-level page table? [0.5 point]

8) Repeat 5) on the two-level page table and show how you get the answers. [2 points]

Question 2: Page Replacement Policy [7.2 points]

Agents, please fill in the blank by tracing the following Paging Algorithms and determine if the page fault happens for each page reference. Suppose you only have 3 frames in your main memory and the system starts from a 'fresh'/empty memory. If there is a tie, choose the **lowest-numbered page**.

Page Reference Stream: 4, 2, 3, 4, 1, 3, 2, 4, 5, 4, 3, 2

FIFO Algorithm [2.4 points]												
	4	2	3	4	1	3	2	4	5	4	3	2
Frame 1												
Frame 2												
Frame 3												
Page Faults?												

Optimal Algorithm [2.4 points]												
	4	2	3	4	1	3	2	4	5	4	3	2

Frame 1												
Frame 2												
Frame 3												
Page Faults												

LRU (Least Recently Used) Algorithm [2.4 points]												
	4	2	3	4	1	3	2	4	5	4	3	2
Frame 1												
Frame 2												
Frame 3												
Page Faults												

Question 3: **Disk Writing** **[6.8 points]**

For a **6,000-RPM** disk, assume its **average seek time** is 5 msec, its **average data transfer speed** is 100 MB/sec, and the **sector size** is 512 bytes. (In this problem, **assume** that 1 MB = 1×10^6 bytes; 1 msec = 0.001 sec; throughput means the amount of data transferred divided by time.) [Ref: Lecture Topic 6a, Slide 47]

1) How long does it take to access **1MB** of contiguous data on average? What is the average throughput (bytes read per second) if the user accesses 1MB for each I/O read operation? Assume the data are perfectly aligned on the disk [3.4 points]

2) How long does it take to access **100 MB** of contiguous data (on average)? What is the average throughput (bytes read per second) if the user accesses 100 MB for each I/O read operation? Assume the data are perfectly aligned on the disk. [3.4 points]