Gebze Technical University
Department of Computer Engineering
CSE 312 Spring 2023
Homework 2

In this project, you will design and implement a simulated and simplified virtual memory management system and a number of pagereplacement algorithms in C/C++. This project is completely independent of your other homework/exam assignments.

Since this is a simulation, we will use an integer C array for your physical memory as in **Fig 3-9** in your textbook. As expected, your virtual memory may be much larger as shown in the same figure. You will keep your data that does not fit in your physical memory on a disk file.

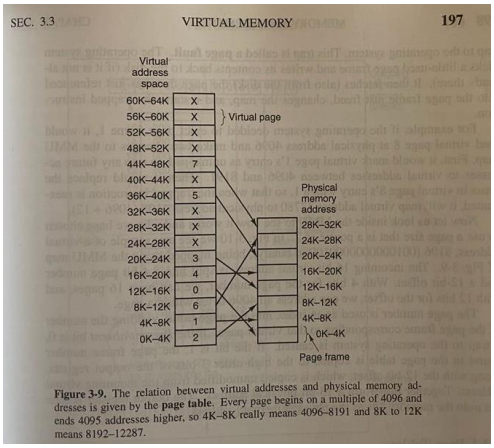

Figure 3-9. The relation between virtual addresses and physical memory addresses is given by the **page table**. Every page begins on a multiple of 4096 and ends 4095 addresses higher, so 4K–8K really means 4096–8191 and 8K to 12K means 8192–12287.

Your simplified system will do simple integer array arithmetic with 2 different multiplication algorithms (matrix (e.g.: 1000 by 3) - vector (e.g.: 3 by 1) Multiplication, vector (e.g.: 1 by 1000)-vector^T Multiplication) and an array summation algorithm (Array Summation) 2 different search algorithms (linear search and binary search).

After filling the entire virtual memory with random integers, you will use C/C++ threads, one for each multiplication and summation operations. Each thread will work on a different part of the virtual array with equal size of the virtual memory. When both of them done multiplication, another thread will make the array summation of the results of both. Also, it will make the vector summarization of both. Then, you will need threads for remaining 2 search algorithms that they will wait until the all the arithmetic is done and they do search operations on the main result array. Search 5 numbers 2 of which are not in the array.

The above step will only access your virtual memory using only by c++ functions (get and set).

Design what you will need as parameters for this functions.

If the data is available in the physical memory, it is handled immediately. If the data is not available, then the page replacement algorithm has to be executed. The page replacement statistics can be easily calculated in these functions

PART 1

Design a page table structure that makes it possible to implement Second-Chance (SC), Least-Recently-Used (LRU), Working set clock (WSClock) algorithms. Note that since we are simulating the hardware, we can add any features as we like to the page table including updating the table at every array reference for LRU.

In your project report, draw figures and explain each field of the page table entries like Fig 3-11. Your report should include references to your implementation code (line numbers, C files, etc.) in your report.

PART 2

In this part, your program will be named **operateArrays** and it will have the following command line structure

operateArrays 12 5 10 LRU inverted 10000 diskFileName.dat

which defines a frame size of 2^{12} (4096) integers, 2^5 (32) physical frames, 2^{10} (1024) virtual frames, uses LRU as page replacement algorithm, inverted page table, and **diskFileName.dat** as the disk file name. In other words, this system has a physical memory that can hold 4096*32=131.072 integers and has a virtual memory that can hold 4096*1024= 4.194.304 integers. This command prints the page table on the screen at every 10000 memory accesses.

At the end of the program run, your statistics will include the following for each thread of the program

- Number of reads
- Number of writes
- Number of page misses
- Number of page replacements
- Number of disk page writes
- Number of disk page reads
- Estimated working set function w for each thread as a table

In this part, you will implement all necessary functions including get/set, page replacement, and all program phases. Note that you will use the virtual memory only for array storage. For other temporary sorting data (such as indexes), you will use C variables.

Write your report for this part that discusses how you implemented the get/set methods, your page replacement algorithms, and allocation policy issues. Do not forget to include a discussion about your backing store (Section 3.6.5 of the textbook)

PART 3

For this part, physical memory can hold 16K integers and the virtual memory can hold 128K integers. Include a detailed discussion about each of these tasks in your report and list all the

data for the graphs in your report. You may use tools like MS Excel to draw the graphs.

- Write a program that changes the page frame size in a loop for the given system to find the optimal page size for each sorting algorithm. The best frame size is the one that causes the smallest number of page replacements.
- (Bonus) Write another program that finds the best page replacement algorithm for each sort method. You should run many different configurations to make a decision.