CSCE 410-500 Homework #2

Alexia Perez
alexia_perezv@tamu.edu
List of people you worked with on homework – None.

Question 1: Consider the following segment table:

Segment	Base	Length
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96

What are the physical addresses for the following logical addresses?
hint: the logical address is formatted as: segment number, offset

a) 1, 10
   a. Since the offset (10) < segment length (14), physical address is Base + Offset. Therefore, physical address = 2300 + 10 = 2310

b) 4, 112
   a. Since the offset (112) > segment length (96), this will cause a segmentation fault.

Question 2: A computer has four-page frames. The time of loading, time of last access and the R(referenced) and M(modified) bits are shown below (the times are in clock ticks).

Page	Loaded	Last Ref. (ticks)	R bit	M bit
0	126	280	1	0
1	230	265	0	1
2	140	270	0	0
3	110	285	1	1

(a) Which page will Not Recently Used (NRU) replace?
   a. NRU will replace page 2 since both R and M are 0.

(b) Which page will FIFO replace?
   a. FIFO replaces the page for which arrival time is *least.* Therefore, it will replace page 3.

(c) Which page will Least Recently Used (LRU) replace?
   a. LRU will replace page 1 since its referenced time is least of all.

(d) Which page will second chance replace?
   a. The second chance algorithm replaces the page for which the R bit is 0 and with the least arrival time. Therefore, page 2 will be replaced.

Question 3: If you implement a correct solution to the multiple producer-consumer problem, why should your output (and therefore your order of producers and consumers executing) vary between runs?

In most producer-consumer problem solutions there will be some shared memory or critical section which needs to be updated by both the consumer and the producer. Therefore, even if we implement a correct solution to this problem, our output may very between runs due to the output depending on the order of the producers and the consumers entering the critical section. If we order them correctly by using synchronization mechanisms the final output will be the same, but the intermediate runs may vary.

Question 4: Why are output files for the printer normally spooled on disk before being printed?

Output files for printers are usually spooled (meaning they are put in a temporary location while the printing job is completing) to allow the user to continue doing work while the files are being printed. This does not speed up the process of printing the files, but it helps the user's productivity.

Question 5: What is the working set page replacement algorithm? How does the working set algorithm work? Explain locality of reference with respect to memory references and how this applies to the working set page replacement algorithm.

The working set page replacement algorithm is an approach used by many paging systems that keeps track of each process' working set and makes sure that it is in memory before letting the process run. Below is a detailed explanation to better understand what this means:

"The set of pages that a process is currently using is called its working set (Denning, 1968a; Denning, 1980). If the entire working set is in memory, the process will run without causing many faults until it moves into another execution phase (e.g., the next pass of the compiler). If the available memory is too small to hold the entire working set, the process will cause many page faults and run slowly since executing an instruction takes a few nanoseconds and reading in a page from the disk typically takes 10 milliseconds. At a rate of one or two instructions per 10 milliseconds, it will take ages to finish. A program causing page faults every few instructions is said to be thrashing (Denning, 1968b).

In a multiprogramming system, processes are frequently moved to disk (i.e., all their pages are removed from memory) to let other processes have a turn at the CPU. The question arises of what to do when a process is brought back in again. Technically, nothing need be done. The process will just cause page faults until its working set has been loaded. The problem is that having 20, 100, or even 1000 page faults every time a process is loaded is slow, and it also wastes considerable CPU time, since it takes the operating system a few milliseconds of CPU time to process a page fault. Therefore,

many paging systems (including the working set algorithm) try to keep track of each process' working set and make sure that it is in memory before letting the process run."

Locality of reference is used in the working set page algorithm, which means that "during any phase of execution, the process references only a relatively small fraction of its pages."

Sources:

Question #2:

Tanenbaum, A., & Boschung, H. T. (2018). *Modern Operating Systems*. Pearson.

Question #3:

If you implement a correct solution to the multiple producer-consumer problem, why should your output (and therefore your order of producers and consumers executing) vary between runs? bartleby. (2022, February 3). Retrieved February 5, 2022, from https://www.bartleby.com/questions-and-answers/if-you-implement-a-correct-solution-to-the-multiple-producer-consumer-problem-why-should-your-output/9e5e8abc-7558-4700-ab11-f04ea2504841

Question #5:

Page Replacement Algorithms. InformIT. (2002, February 8). Retrieved February 5, 2022, from https://www.informit.com/articles/article.aspx?p=25260&seqNum=9#:~:text=The%20working%20set%20is%20the,working%20set%20at%20time%20t.&text=Having%20this%20information%20also%20immediately,working%20set%20and%20evict%20it.