CS4328: Homework #4

Due on April 28, 2024 (firm with no extensions)

Mina Guirguis

You may discuss this problem set with other students. However, you must write up your answers on your own. You must also write the names of other students you discussed any problem with. Each problem has a different weight. Please state any assumptions you are making in solving a given problem. Late assignments will not be accepted with prior arrangements. Assignments are due in class.

Problem 1

Consider a simple paging system with the following parameters: 2^{32} bytes of physical memory; page size of 2^{12} bytes; 2^{16} pages of logical address space. Answer the following questions: **[10 pts]**

(a) How many bits are in a logical address?

$2^{12} + 2^{16} = 2^{28}$ # of bits in logical address is 28

(b) How many bytes are in frame?

The size of the frame is the same size as page so value for byte number is 2^{12} bytes

(c) How many bits in the physical address specify a frame?

$\frac{2^{32}}{2^{12}} = 2^{20}$ number of bits required to specify a frame is 20 within physical address

(d) How many entries in the page table

with number of entries in page table being number of pages in virtual memory, the number of entries within page table = 2^{16}

(e) How many bits in each page table entry? Assume each page table entry includes a valid/invalid bit.

Each page table has 21 bits since number of bits required to specify frame location in the physical address is 20 (part c). The number of bits in each table is one on top of that value

Problem 2

Consider a paging system with page table stored in memory, answer the following **[3 pts each]**:

(a) If a memory reference takes 0.2 microseconds, how long does a paged memory reference take?

Time = checking page lookup table + Accessing instruction from memory
- instruction address from table = 0.2 Nsec because it same amount that memory reference takes

Total time = 0.2 + 0.2 = 0.4 microseconds

(b) If we add 64 associative registers and 75% of all page table references are found in the associative registers, what is the effective memory reference time? Assume that checking a page table entry in the associative registers takes 10 nanosecond. 0.2 Nsec to access ? 10 nanosec to search page table

equation = 75% @ beginning = $0.75 \cdot (0.2\,Nsec + 10\,nsec) = 1.575 \times 10^{-7}$ sec

= 25% @ remaining = $0.25 \cdot 0.4\,Nsec = 1 \times 10^{-7}$ sec

Total time: $1.575 \times 10^{-7} + 1 \times 10^{-7} = 2.575 \times 10^{-7}$ sec or 0.2575 Nsec

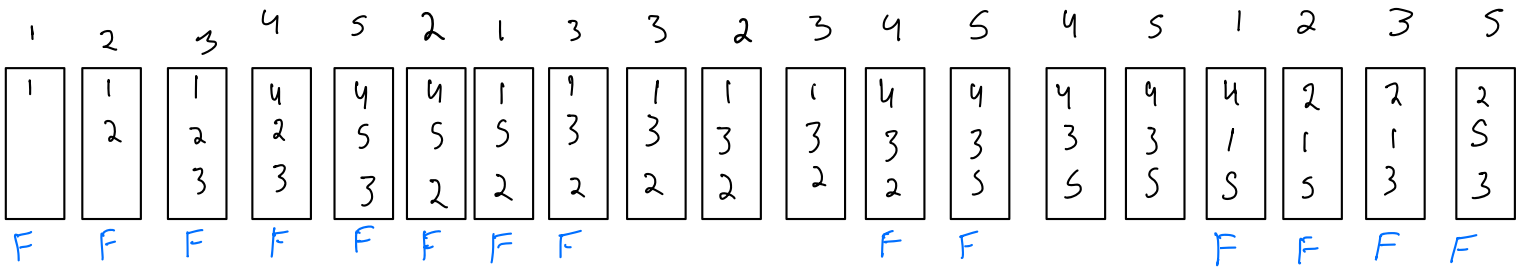
Problem 3

Consider the following sequence of page references (each element represents a page number in a virtual memory system):

1 2 3 4 5 2 1 3 3 2 3 4 5 4 5 1 2 3 5 Assume only 3 frames are available and that they were initially empty.

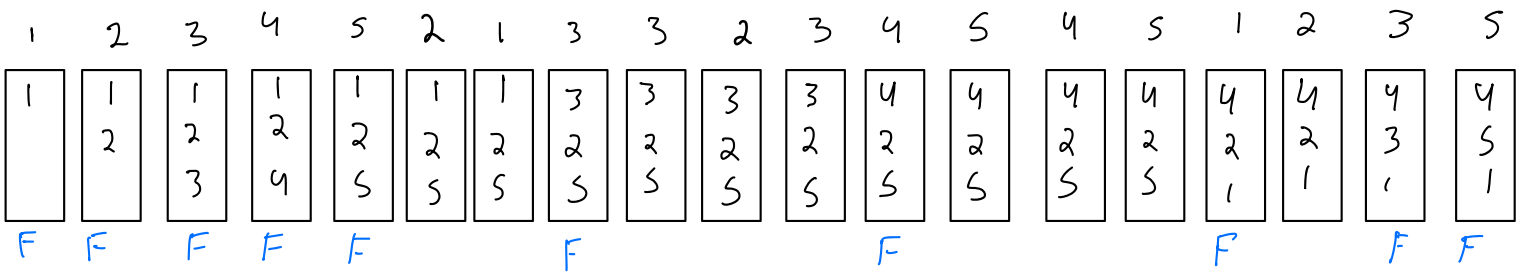
Show how many page faults would occur under each of the following policies:

(a) FIFO. **[3 pts]**



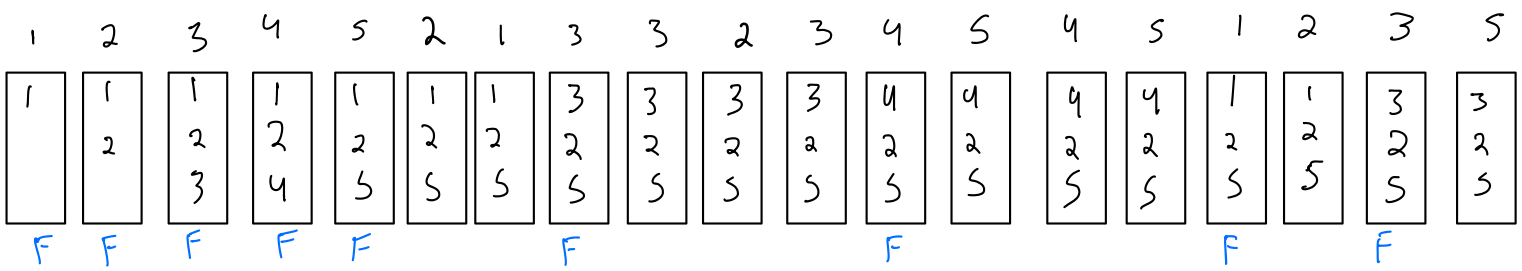
14 total page faults

(b) LRU. **[3 pts]**



10 total page faults

(c) Optimal. **[3 pts]**



9 total page faults

Problem 4

Consider a page reference string for a process with a working set of M frames initially all empty. The page reference string is of length P with N distinct page numbers in it. For any page replacement algorithm, answer the following [**3 pts each**]:

(a) What is the lower bound on the number of page faults?

N will be lower bound number for this case

(b) What is the upper bound on the number of page faults?

P will be upper bound case based on the algorithm