Virtual Memory Problem Set

Not to be turned in. Self-studying only.
Solutions will be released soon. Have fun!!!

Problem 1

Say you are building a hypothetical machine that supports virtual memory. Programmers on this machine see a memory space of 2^16 bytes, and the actual physical memory size is 8K bytes.

1. If the page size is 512 bytes, how many bits are used for the physical page number (PPN), and how many bits are used for the virtual page number (VPN)?

2. Suppose each page table entry (PTE) contains, in addition to the PPN, a Valid bit, a modified bit, and two bits of access control. What is the size of the page table?

Problem 2

An ISA supports an 8-bit, byte-addressable virtual address space. The corresponding physical memory has only 128 bytes. Each page contains 16 bytes. So there are 8 pages in the physical memory. A simple, one-level translation scheme is used and the page table resides in physical memory. The initial contents of the physical memory are shown below.

Physical Page Number	**Physical Page Contents**
0	Empty
1	Virtual Page 13
2	Virtual Page 5
3	Virtual Page 2
4	Empty
5	Virtual Page 0

6	Empty
7	Page Table

A three-entry Translation Lookaside Buffer that uses LRU replacement is added to this system. (Note: LRU policy is used to select virtual pages for replacement in physical memory) Initially, this TLB contains the entries for virtual pages 0, 2, and 13. Assume this machine executes a program that issues the following references to the virtual memory:

References (to virtual pages): 0, 13, 5, 2, 14, 14, 13, 6, 6, 13, 15, 14, 15, 13, 4, 3.

1. List the references that generate a TLB hit
2. List the references that generate a page fault.
3. What is the hit rate of the TLB for this sequence of references?
4. At the end of this sequence, what three entries are contained in the TLB?
5. At the end of this sequence, what are the contents of the 8 physical frames?

Problem 3

An instruction is said to generate a page fault if a page fault occurs at any time during the processing of that instruction.

How many page faults could possibly be generated by the following x86 instruction: `addq %rax,(%rbx)`? Assuming a one-level translation scheme.

Relevant Problems in Past Exams:

- Problem 4 in Spring 2018 Final (challenging)
- Problem 5 in Spring 2019 Final
- Problem 4 in Spring 2020 Final