CSCI 564: Homework 4

Name: _____ CWID: _____

For each of the following questions, please show all of your work, and explain your answers.

1. **Virtual and Physical Addresses.** Consider these three configurations:

   1. 32-bit operating system, 4 KiB pages, 1 GiB of RAM
   2. 32-bit operating system, 16 KiB pages, 2 GiB of RAM
   3. 64-bit operating system, 16 KiB pages, 16 GiB of RAM

   (a) (15 points) Fill out the following table showing how many bits are needed for the virtual address, physical address, virtual page number, physical page number, and offset in each of the above configurations.

	Virtual Address	**Physical Address**	**Virtual Page #**	**Physical Page #**	**Offset**
1					
2					
3					

   (b) (5 points) What are some advantages of using a larger page size?

   (c) (5 points) What are some disadvantages of using a larger page size?

2. (25 points) **Using the TLB.** As described in the textbook and explained in class, virtual memory uses a page table to track the mapping of virtual addresses to physical addresses. To speed up this translation, modern processors implement a cache of the most recently used translations called the *translation lookaside buffer (TLB)*. This exercise shows how the page table and the TLB must be updated as addresses are accessed.

Initial TLB State (values are base-10)

Valid	Tag	Physical Page #	LRU
1	11	12	2
1	7	4	3
1	3	6	4
0	4	9	1

The LRU column works as follows: the *older* an entry is, the *lower* its LRU number will be. Each time an entry is used, its LRU number is set to 4 (since it is now the most-recently used), and the other numbers are adjusted downward accordingly.

Initial Page Table State (values are base-10)

Index	Valid	Physical Page or On Disk
0	1	5
1	0	Disk
2	0	Disk
3	1	6
4	1	9
5	1	11
6	0	Disk
7	1	4
8	0	Disk
9	0	Disk
10	1	3
11	1	12

Assume 4 KiB pages and a four-entry fully-associative TLB with an LRU replacement policy. If pages must be brought in from disk, give them the next largest unused page number (that is, start at 13).

Given the following list is a stream of virtual addresses seen on the system, the initial TLB, and initial page table, show the final state of the TLB and page table, and specify whether each memory address is a hit in the TLB (H), a hit in the page table (TLB miss, M) or a page fault (PF).

Address	Result (H, M, PF)
`0x0FFF`	
`0x7A28`	
`0x3DAD`	
`0x3A98`	
`0x1C19`	
`0x1000`	
`0x22D0`	

Final TLB State

Valid	Tag	Physical Page #	LRU

Final Page Table State

Index	Valid	Physical Page or On Disk
0		
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		