Assignment 7 (Due on June 7, 2023)

1. In this exercise, we will examine space/time optimizations for page tables. The following list provides parameters of a virtual memory system.

Virtual Address(in bits)	Physical DRAM Installed	Page Size	PTE Size (byte)
46	128GiB	8KiB	8

I. For a single-level page table, how many page table entries (PTEs) are needed? (10 pts)

2^46 / 2^13 = 2^33

II. How much physical memory is needed for storing the page table? (10 pts)

2^33 x 2^3 = 2^36 bytes = 64 GiB

2. The following table shows the contents of a 4-entry TLB.

EntryID	Valid	VA Page	Modified	Protection	PA Page
1	1	140	1	RW	30
2	0	40	0	RX	34
3	1	200	1	RO	32
4	1	280	0	RW	21

I. Under what scenarios would entry 2’s valid bit be set to zero? (6 pts)

Page Fault and TLB Invalidation

II. What happens when an instruction writes to VA page 80? (6 pts)

TLB write miss

III. What happens when an instruction writes to VA page 200? (6 pts)

Access Violation: entry 3 is RO / write protection

3. Assume that the write-back invalidating snooping protocol is adopted to a multi-core computer architecture. In this architecture, a processor can address 64KB main memory and have a 4-way set-associative write-back

cache with 64 sets and 16 bytes per line. The settings for this architecture is listed as follows:

- A LRU replacement policy is implemented on the cache.
- Two caches for two processors are empty at the beginning, and the memory is accessed by the following sequence.

Please show the information of the tag, set, line number, directory-based state, data for non-empty cache lines, and the contents of memory by using Tables 1 and 2 format after each step of data accesses.

(32 pts, 4 pts for each step)

[1]. Processor 0 reads from location 0D3C
[2]. Processor 1 writes 0110 to location 0D34
[3]. Processor 0 reads from location 0D38
[4]. Processor 1 writes 1111 to location 1D34
[5]. Processor 0 writes 0001 to location 1D3C
[6]. Processor 0 reads from location 0D3C
[7]. Processor 1 writes 1110 to location 2D30
[8]. Processor 0 reads from location 2D3C

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C

Table 1. The information in caches

	Memory Data			
Address	0	4	8	C
0D3x	0000	0000	0000	0000
1D3x	0000	0000	0000	0000
2D3x	0000	0000	0000	0000

Table 2. Initial state of memory

1.

Processor 0 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0000	0000	0000

Processor 1 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C

Memory

	Memory Data			
Address	0	4	8	C
0D3x	0000	0000	0000	0000
1D3x	0000	0000	0000	0000
2D3x	0000	0000	0000	0000

2.

Processor 0 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	0	0000	0000	0000	0000

Processor 1 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000

Memory

	Memory Data			
Address	0	4	8	C
0D3x	0000	0000	0000	0000
1D3x	0000	0000	0000	0000
2D3x	0000	0000	0000	0000

3.

Processor 0 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000

Processor 1 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000

Memory

	Memory Data			
Address	0	4	8	C
0D3x	0000	0110	0000	0000
1D3x	0000	0000	0000	0000
2D3x	0000	0000	0000	0000

4.

Processor 0 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000

Processor 1 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000
000111	010011	1	1	0000	1111	0000	0000

Memory

	Memory Data			
Address	0	4	8	C
0D3x	0000	0110	0000	0000
1D3x	0000	0000	0000	0000
2D3x	0000	0000	0000	0000

5.

Processor 0 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000
000111	010011	1	1	0000	1111	0000	0001

Processor 1 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000

000011	010011	1	0	0000	1111	0000	0000

Memory

	Memory Data			
Address	0	4	8	C
0D3x	0000	0110	0000	0000
1D3x	0000	1111	0000	0000
2D3x	0000	0000	0000	0000

6.

Processor 0 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000
000111	010011	1	1	0000	1111	0000	0001

Processor 1 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000
000011	010011	1	0	0000	1111	0000	0000

Memory

	Memory Data			
Address	0	4	8	C
0D3x	0000	0110	0000	0000
1D3x	0000	1111	0000	0000
2D3x	0000	0000	0000	0000

7.

Processor 0 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0000	0000	0000
000111	010011	1	1	0000	1111	0000	0001

Processor 1 cache

Tag				Data RAM			

Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000
000011	010011	1	0	0000	1111	0000	0000
001011	010011	2	1	1110	0000	0000	0000

Memory

	Memory Data			
Address	0	4	8	C
0D3x	0000	0110	0000	0000
1D3x	0000	1111	0000	0000
2D3x	0000	0000	0000	0000

8.

Processor 0 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0000	0000	0000
000111	010011	1	1	0000	1111	0000	0001
001011	010011	2	1	1110	0000	0000	0000

Processor 1 cache

Tag				Data RAM			
Tag	Set	Way	Valid bit	0	4	8	C
000011	010011	0	1	0000	0110	0000	0000
000011	010011	1	0	0000	1111	0000	0000
001011	010011	2	1	1110	0000	0000	0000

Memory

	Memory Data			
Address	0	4	8	C
0D3x	0000	0110	0000	0000
1D3x	0000	1111	0000	0000
2D3x	1110	0000	0000	0000

4. If the floating-point (FP) instructions of an application on a specific processor ***C1*** consumes 70% of the total execution time. Moreover, in the

same application, 30% of the floating-point time is taken to do square root calculations.

I. A new processor ***C2*** is developed. This new processor could either enhance the performance of FP instructions by a factor of 1.4 or alternatively increase the performance of the square root operation by a factor of 16. Which is the better design for the aforementioned application? (14 pts)

Let's say if the total execution time takes 100 seconds, then the FP instruction takes 70 seconds, and the square root operation takes 30 seconds.

If C2 improves FP instructions by 1.4, then FP instructions take 70 / 1.4 = 50 seconds. That is 20 second faster than before.

If C2 improves square root operations by 16, then the square root operations take 30 / 16 = 1.875 seconds. That is 28.125 second faster than before.

So, the increasement of the performance of the square root operation is the better design.

II. If we decide to parallelize the code within the application. What speedup can be reached if we can perfectly parallelize 90% of the code and run the applications on a system with 16 cores? (8 pts) What fraction of the code has to be parallelized if we want to get a speedup of 10 on the same system? (8 pts)

Parallelize 0.9 of the code:
Speedup = 1 / (0.1 + 0.9 / 16) = 6.4

10 = 1 / (1-x + x / 16) => x = 96%
96% of the code need to be parallelized.