Homework #2 – 28 pts
Submit your homework to TEACH by Wed. 10/17/2018, at 11:59pm

Your submission should be comprised of one item: a .pdf file containing answers to the questions. Legible, handwritten solutions are acceptable for undergraduate students (472). The handwritten solutions must be electronically scanned and submitted as a PDF file. Graduate students (572) must compose their solutions in MS Word, LaTeX, or some other word processor and submit the results as a PDF file.

1. (6 pts) Suppose the existence of a direct-mapped cache design with a 32-bit address. Assume that each address maps to a single byte. Assume the cache scheme uses the following breakdown of the address bits:

Tag	Index	Offset
31-9	8-5	4-0

   a. (2 pts) What is the cache block size (in words)?
      5 bits allocated to the byte offset. 2^5 = 32 bytes → 32/4 = 8 words
   b. (2 pts) How many entries does the cache have?
      4 bits allocated to the index. 2^4 = 16 entries (i.e. indices)
   c. (2 pts) Including space for the valid bits, tags, and actual block data, how many bits would be required to actually implement this hypothetical cache?
      For each entry:
      32 bytes of data = 256 bits
      1 valid bit for each entry
      23 bits for each tag
      ➔ Adds up to 280 bits per entry (index)
      ➔ 280 bits * 16 entries = 4480 total bits required

2. (10 pts) Using the cache design from problem 1, assume the following byte-addressed cache references are recorded. Assume that the cache is initially empty. Also assume that accesses occurred from left to right (e.g. address 0 was requested, then address 4, followed by address 16, etc).

Byte Address											
0	4	16	132	232	160	1024	30	140	3100	180	2180

Byte Address	Block address $\text{floor}\left(\frac{byte\ address}{block\ size\ in\ bytes}\right)$	Block Index Block address % num_of_indices	Tag	Hit?
0	0	0	00000000 00000000 0000000	M
4	0	0	00000000 00000000 0000000	H
16	0	0	00000000 00000000 0000000	H
132	4	4	00000000 00000000 0000000	M
232	7	7	00000000 00000000 0000000	M
160	5	5	00000000 00000000 0000000	M
1024	32	0	00000000 00000000 0000010	M
30	0	0	00000000 00000000 0000000	M
140	4	4	00000000 00000000 0000000	H
3100	96	0	00000000 00000000 0000110	M
180	5	5	00000000 00000000 0000000	H
2180	68	4	00000000 00000000 0000100	M

a. (4 pts) What was the number of cache hits?
4

b. (6 pts) Create a table to show the final state of the cache including the value of each Index, Valid bit, and Tag.

Index	Valid	Tag
0	1	00000000 00000000 0000110
1	0	?
2	0	?
3	0	?
4	1	00000000 00000000 0000100
5	1	00000000 00000000 0000000
6	0	?
7	1	00000000 00000000 0000000
8	0	?
9	0	?
10	0	?
11	0	?
12	0	?
13	0	?
14	0	?
15	0	?

3. (6 pts) Suppose the existence of a different direct-mapped cache design with a 32-bit address. Assume that each address maps to a single byte. Assume this cache scheme uses the following breakdown of the address bits:

Tag	Index	Offset
31-6	5-3	2-0

   a. (2 pts) What is the cache block size (in words)?
      3 bits allocated to the byte offset. 2^3 = 8 bytes → 8/4 = 2 words
   b. (2 pts) How many entries does this cache have?
      3 bits allocated to the index. 2^3 = 8 entries (i.e. indices)
   c. (2 pts) Including space for the valid bits, tags, and actual block data, how many bits would be required to actually implement this hypothetical cache?
      For each entry:
         8 bytes of data = 64 bits
         1 valid bit for each entry
         26 bits for each tag
         ➔ Adds up to 91 bits per entry (index)
      91 bits * 8 entries = 728 total bits required

4. (6 pts) Again assuming an initially empty cache, repeat the steps described in problem 2.

Byte Address	Block address floor($\frac{byte\ address}{block\ size\ in\ bytes}$)	Block Index Block address % num_of_indices	Tag	Hit ?
0	0	0	00000000 00000000 00000000 00	M
4	0	0	00000000 00000000 00000000 00	H
16	2	2	00000000 00000000 00000000 00	M
132	16	0	00000000 00000000 00000000 10	M
232	29	5	00000000 00000000 00000000 11	M
160	20	4	00000000 00000000 00000000 10	M
1024	128	0	00000000 00000000 00000100 00	M
30	3	3	00000000 00000000 00000000 00	M
140	17	1	00000000 00000000 00000000 10	M
3100	387	3	00000000 00000000 00001100 00	M
180	22	6	00000000 00000000 00000000 10	M
2180	272	0	00000000 00000000 00001000 10	M

a. (2 pts) What was the number of cache hits?
1

b. (4 pts) Create a table to show the final state of the cache including the value of each Index, Valid bit, and Tag.

Index	Valid	Tag
0	1	00000000 00000000 00001000 10
1	1	00000000 00000000 00000000 10
2	1	00000000 00000000 00000000 00
3	1	00000000 00000000 00001100 00
4	1	00000000 00000000 00000000 10
5	1	00000000 00000000 00000000 11
6	1	00000000 00000000 00000000 10
7	0	?

5. This question is not graded