Computer Organization HW4

1. For a 256 Byte data cache, 8 word per block, direct mapped design with a 32-bit address, beginning from power on, the following **byte-addressed** cache references are recorded.

Address											
Hex	00	04	10	84	E8	A0	400	1E	8C	C1C	B4	884
Dec	0	4	16	132	232	160	1024	30	140	3100	180	2180

   a) Calculate the range of bits used for Tag, index and offset (for example, offset: 1-0, index: 3-2, etc)
   b) For each reference, list (1) its tag, index, and offset, (2) whether it is a hit or a miss, and (3) which bytes were replaced (if any).
   c) What is the hit ratio?
   d) List the final state of the cache, with each valid entry represented as a record of <index, tag, data>. For example, <0, 3, Mem[0x00]-Mem[0x1F]>.

2. This exercise examines the effect of different cache designs, specifically comparing associative caches to the direct-mapped caches. For these exercises, refer to the sequence of **word address** shown below. 0x03, 0xb4, 0x2b, 0x02, 0xbe, 0x58, 0xbf, 0x0e, 0x1f, 0xb5, 0xbf, 0xba, 0x2e, 0xce
   a) For a three-way set associative cache with two-word blocks and a total size of 48 words. which bits represent index and which bits represent tag in 32bit memory address (for example, index: 1-0, tag: 31-2)
   b) Using the given sequence, show the final cache contents for a three way set associative cache with two word blocks and a total size of 24 words. Use LRU replacement. For each reference identify the index bits, the tag bits, the offset bits, and if it is a hit or a miss. For invalid blocks use NA as content.
   c) For a fully associative cache with one word blocks and a total size of 8 words, using the given sequence, show the final cache contents for a fully associative cache with one word blocks and a total size of 8 words. Use LRU replacement. For each reference identify the index bits, the tag bits, and if it is a hit or a miss.

3. Multilevel caching is an important technique to overcome the limited amount of space that a first-level cache can provide while still maintaining its speed. Consider a processor with the following parameters and the miss rate are global miss:

Base CPI, No Memory Stalls	Processor Speed	Main Memory Access Time	First-Level Cache Miss Rate per Instruction**	Second-Level Cache, Direct-Mapped Speed	Miss Rate with Second-Level Cache, Direct-Mapped	Second-Level Cache, Eight-Way Set Associative Speed	Miss Rate with Second-Level Cache, Eight-Way Set Associative
1.5	2 GHz	100 ns	7%	12 cycles	3.5%	28 cycles	1.5%

   Calculate the CPI for the processor in the table using:
   a) only a first-level cache
   b) a second-level direct-mapped cache, and
   c) a second-level eight-way set associative cache.
   d) How do the CPI value calculated in b) change if all miss rates in table become local miss rate?

4. This exercise examines the single error correcting, double error detecting (SEC/DED) Hamming code.
   a) What is the minimum number of parity bits required to protect a 128-bit word using the SEC/DED code?
   b) Consider a SEC code that protects 8 bit words with 4 parity bits. If we read the value 0x375, is there an error? If so, correct the error.

5. Virtual memory uses a page table to track the mapping of virtual addresses to physical addresses (in unit of byte). This exercise shows how this table must be updated as addresses are accessed. The following data constitute a stream of virtual byte addresses as seen on a system. Assume 4 KiB pages, a four entry fully associative TLB, and true LRU replacement. If pages must be brought in from disk, increment the next largest page number. i.e. If the current maximum physical page number is 12, then the next physical page brought in from disk has a page number of 13.

Decimal	4669	2227	13916	34587	48870	12608	49225
hex	0x123d	0x08b3	0x365c	0x871b	0xbee6	0x3140	0xc049

TLB

Valid	Tag	Physical Page Number	Time Since Last Access
1	0xb	12	4
1	0x7	4	1
1	0x3	6	3
0	0x4	9	7

Page table

Valid	Physical Page or in Disk
1	5
0	Disk
0	Disk
1	6
1	9
1	11
0	Disk
1	4
0	Disk
0	Disk
1	3
1	12

   a) For each access shown above, fill up the following table, and then show the final state of the TLB and Page table

Address	Virtual Page Number	TLB Hit/Miss, Page table Hit/Miss, Page fault
4669		
…		

   b) Fill up the following table for each access, but this time use 4 KiB pages and a two-way set associative TLB, and then show the final state of the TLB (Assume the first two rows of previous TLB are set-index 0 and last two rows are set-index 1, and true LRU replacement)

Address	Virtual Page	Index	Tag	TLB Hit/Miss, Page table Hit/Miss, Page fault
4669				
…				