Habib University

Dhanani School of Science and Engineering

Computer Architecture
EE 371 / CS 330 / CE 321 (Spring 2023)

Homework 4

Total marks: 100	**Marks obtained:**

Student Name:	**Student ID:**	**Section:**

Purpose:

The purpose of this assignment is to help you in understanding the performance of cache operations.

Instructions:

1. This assignment should be done in pairs.
2. All questions should be answered in **black ink only**.
3. Scan your answer sheet and upload it on HU LMS before the due date.

Grading Criteria:

1. Your assignments will be checked by instructor/TA.
2. You can also be asked to give a viva where you will be judged whether you understood the question yourself or not. If you are unable to answer correctly to the question you have attempted right, you may lose your marks.
3. Zero will be given if the assignment is found to be plagiarized.
4. Untidy work will result in reduction of your points.

Submission policy:

1. No submission will be accepted after the instructor releases the solution on HU LMS.

CLO Assessment:

This assignment assesses students for the following course learning outcomes.

Course Learning Outcomes		CLO Assessed
CLO 1	***Explain*** the role of ISA in modern processors and instruction encodings and assembly language programming	
CLO 2	***Explain*** the architecture and working of a single cycle processor	
CLO 3	***Design*** the architecture to mitigate issues of a pipelined processor	
CLO 4	***Analyze the*** performance of cache operations	✓

Question 1 [10 Marks]

Caches are important to providing a high-performance memory hierarchy to processors. Below is a list of 64-bit memory address references, given as word addresses.
0x03, 0xb4, 0x2b, 0x02, 0xbf, 0x58, 0xbe, 0x0e, 0xb5, 0x2c, 0xba, 0xfd

(a) **[05 Marks]** For each of these references, identify the binary word address, the tag, and the index given a direct-mapped cache with 16 one-word blocks. Also list whether each reference is a hit or a miss, assuming the cache is initially empty. First insertion is already done so that you may get the idea.

Word Address	Binary Address	Tag	Index	Hit/Miss
0x03	0000 0011	0	3	M
0xb4				
0x2b				
0x02				
0xbf				
0x58				
0xbe				
0x0e				
0xb5				
0x2c				
0xba				
0xfd				

(b) **[05 Marks]** For each of these references, identify the binary word address, the tag, the index, and the offset given a direct-mapped cache with two-word blocks and a total size of eight blocks. Also list if each reference is a hit or a miss, assuming the cache is initially empty.

Word Address	Binary Address	Tag	Index	Offset	Hit/Miss
0x03	0000 0011	0	1	1	M
0xb4					
0x2b					
0x02					
0xbf					
0x58					
0xbe					
0x0e					
0xb5					
0x2c					
0xba					
0xfd					

Question 2 [10 Marks]

For a direct-mapped cache design with a 64-bit address, the following bits of the address are used to access the cache.

Tag	Index	Offset
63-10	9-5	4-0

a) **[03 Marks]** What is the cache block size (in words)?
b) **[03 Marks]** How many blocks does the cache have?
c) **[04 Marks]** What is the ratio between total bits required for such a cache implementation over the data storage bits?

Question 3 [15 Marks]

Considering the address size of 64-bits, fill in the data for difference types of caches:

	Blocks	Data per block	Sets	Associativity -ways	Tag Bits	Index Bits	Offset Bits
Fully Associative Cache	8	8 words	--				
Direct Mapped Cache	16	8 words	--				
Set Associative Cache	32	8 words	4				
Direct Mapped Cache	64	8 words	--				
Set Associative Cache	128	8 words	32				
Set Associative Cache	256	8 words	32				
Fully Associative Cache	512	8 words	--				
Direct Mapped Cache	1024	8 words	--				
Set Associative Cache	2048	8 words	64				
Direct Mapped Cache	4096	8 words	--				

Question 4 [05 Marks]

Assume the miss rate of an instruction cache is 4% and the miss rate of the data cache is 6%. If a processor has a CPI of 3 without any memory stalls, and the miss penalty is 100 cycles for all misses, determine how much faster a processor would run with a perfect cache that never missed. Assume the frequency of all loads and stores is 26%.

Question 5 [10 Marks]

We are given 4 arrays of size 6. Each element in an array is of 32 bytes i.e., one word. Following is the data stored in the array:

A = (25, 48, 43, 30, 47, 36)
B = (16, 29, 35, 38, 32, 41)
C = (24, 33, 5, 39, 10, 14)
D = (23, 7, 11, 44, 42, 22)

The array data is arranged in main memory as follows:

00000	A[0]
00001	A[1]
00010	A[2]
00011	A[3]
00100	A[4]
00101	A[5]
00110	
00111	
01000	B[0]
01001	B[1]
01010	B[2]
01011	B[3]
01100	B[4]
01101	B[5]
01110	
01111	
10000	C[0]
10001	C[1]
10010	C[2]
10011	C[3]
10100	C[4]
10101	C[5]
10110	
10111	
11000	D[0]
11001	D[1]
11010	D[2]
11011	D[3]
11100	D[4]
11101	D[5]
11110	
11111	

We are given a direct mapped cache which contains 8 blocks (each block will contain one word). Insert the following elements in cache one by one and also mention whether it was a hit or a miss.

Assume that the first block of the cache will be populated by the first element of the array and so on. First insertion is already done so that you may get the idea.

Data to be Inserted	**Hit/Miss**	**Cache Index**							
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
A[0]	M	A[0]							
A[1]									
A[2]									
A[1]									
A[5]									
B[5]									
B[4]									
B[3]									
B[3]									
B[4]									
D[1]									
D[2]									
D[3]									
D[4]									
C[3]									
C[2]									
C[4]									
C[2]									

What is the Hit Ratio and the Miss Ratio in the above case?

Question 6 [10 Marks]

Whenever an element from an array is accessed, it is most probable that some other remaining elements of the array are also accessed. Repeat the same task as in Question 5 but this time design a cache with 4 blocks in which each block can accommodate 2 words. The first insertion is done again so that you may get the idea.

Data to be Inserted	**Hit/Miss**	**Cache Index**							
		0		**1**		**2**		**3**	
A[0]	M	A[0]	A[1]						
A[1]									
A[2]									
A[1]									
A[5]									
B[5]									
B[4]									
B[3]									
B[3]									
B[4]									
D[1]									
D[2]									
D[3]									
D[4]									
C[3]									
C[2]									
C[4]									
C[2]									

What is the Hit Ratio and the Miss Ratio in this case? Is it better than the previous? Does loading the whole array into the cache helps us in accessing the elements fast?

Question 7 [20 Marks]

a) **[05 Marks]** Repeat Question 5, this time using a fully associative cache containing 8 blocks. Use LRU replacement scheme for eviction.

b) **[05 Marks]** Repeat Question 6, this time using a fully associative cache containing 4 blocks in which each block can accommodate 2 words. Use LRU replacement scheme for eviction.

c) **[05 Marks]** Compare the Hit & Miss Ratio for both the cases (i.e., loading a single element vs. loading two elements) for both the caches. Which cache helps us in accessing elements faster? and in which case? [Hint: To answer which cache helps us in accessing the elements faster, compare the Hit Ratios for both the caches]

d) **[05 Marks]** Repeat Question 6, this time using a two-way set associative cache with 2 sets of 2 blocks. Use LRU replacement scheme for eviction.

Question 8 [20 Marks]

In this exercise, we will examine how replacement schemes affect miss rates. Assume a two-way set associative cache with four one-word blocks. Consider the following word address sequence: 0, 1, 2, 3, 4, 2, 3, 4, 5, 6, 7, 0, 1, 2, 3, 4, 5, 6, 7, 0.

a) **[05 Marks]** Assuming an LRU replacement scheme, which accesses are hits?
b) **[05 Marks]** Most Recently Used (MRU) is a cache replacement scheme which removes the most recently used items first. A MRU scheme is good in situations in which the older an item is, the more likely it is to be accessed. Assuming an MRU (most recently used) replacement scheme, which accesses are hits?
c) **[05 Marks]** Describe an optimal replacement scheme for this sequence. Which accesses are hits using this policy?
d) **[05 Marks]** Describe why it is difficult to implement a cache replacement scheme that is optimal for all address sequences.