# Memory Hierarchy Assignment (Evaluation)

### CSCI 389: Computer Systems

## Spring 2022

This assignment is designed to test your understanding of the memory hierarchy. Feel free to collaborate wih others and use resources, but all code and writeups must be your own.

Due Date: Friday, March 4th at 10:00 am.

- 1. (10 points) **Average Memory Access Time.** Consider a system with an L1 cache and an L2 cache between the processor and memory. The system has the following characteristics:
  - The L1 cache takes 2 cycles to access.
  - The L1 cache has a 10% miss ratio.
  - The L2 cache takes 20 cycle to access.
  - The L2 cache has a 15% miss ratio.
  - Memory takes 220 cycles to access.
  - (a) (6 points) What is the average memory access time for this system?
  - (b) (4 points) What is the average memory access time if the miss ratio of the L1 cache decreases to 5%?

#### Α.

- Memory takes 220 cycles
- L2 takes 20 + 0.15(220) = 53
- L1 takes 2 + 0.1(53) = 7.3

Thus the average memory access time is 7.3 cycles

#### В.

- Memory takes 220 cycles
- L2 takes 20 + 0.15(220) = 53 cycles on average.
- L1 takes 2 + 0.05(53) = 4.65 cycles on average.

Thus the average memory access time is 4.65 cycles

2. (20 points) Caching Bits. Consider the following request sequence.

0x0000, 0x8400, 0x6082, 0x840C, 0x8411, 0x2148, 0x2110, 0x2420, 0x2100, 0x8403

For each part, you will draw a physical diagram, which consists of the tags and valid bits for each cache line. Clearly mark which cache lines are in which sets. In the area where data would normally go, simply denote the range of addresses covered.

- (a) (5 points) Provide a physical diagram showing how a direct-mapped cache with block size of 16 bytes will cache this sequence. Assume that the cache can store 128 bytes of data and runs LRU.
- (b) (5 points) Provide a physical diagram showing how a 2-way set associative cache with block size of 16 bytes will cache this sequence. Assume that the cache can store 128 bytes of data and runs LRU.

- (c) (5 points) Provide a physical diagram showing how a fully associative cache with block size of 16 bytes will cache this sequence. Assume that the cache can store 128 bytes of data and runs LRU.
- (d) (5 points) Provide a physical diagram showing how a fully associative cache with block size of 32 bytes will cache this sequence. Assume that the cache can store 128 bytes of data and runs LRU.

Α.

$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	Α.														
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	Hex	$\mathbf{r} \mid \mathbf{r}$	${ m ag}({ m First})$	t 9)	Set (N	/Iiddle	3)	Offset (I	Last 4	bits)					
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	0x000	00	0000 0000	0 0		000		(	0000						
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	0x840	00	1000 0100	0 0		000		0000							
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	0x608	32	0110 0000	0 1		000		0	0010						
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	0x840	$^{\circ}$ C	1000 0100	0 0				1	100						
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $						001		(	0001						
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	0x214					100									
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $															
$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$															
$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	0x840				0011										
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	Tag	Set	Offset		Tag	Set		Offset		Tag		Set		Offset	
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	000	Ø	Ø		000	0000 0	000	0000	_	000	10	00 010	0 0	0000	
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	001	Ø	Ø		001	Ø		Ø		001		Ø		Ø	
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	010	Ø	Ø		010	Ø		Ø		010		Ø		Ø	
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	011	Ø	Ø	$\longrightarrow$	011	Ø		Ø	$\longrightarrow$	011		Ø		Ø	$\longrightarrow$
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	100	Ø	Ø		100	Ø		Ø		100		Ø		Ø	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	101	Ø	Ø		101	Ø		Ø		101		Ø		Ø	
$\begin{array}{ c c c c c c c c c c }\hline \textbf{Tag} & \textbf{Set} & \textbf{Offset} \\ \hline 000 & 0110 0000 1 & 0010 \\ 001 & \emptyset & \emptyset & 001 & 0110 0000 1 & 0010 \\ 001 & \emptyset & \emptyset & 001 & 0110 0000 1 & 0001 \\ 011 & \emptyset & \emptyset & 010 & \emptyset & \emptyset & 010 & 001 & 0010 \\ 011 & \emptyset & \emptyset & 010 & \emptyset & \emptyset & 010 & \emptyset & 010 & 0000 1 \\ 010 & \emptyset & \emptyset & 010 & \emptyset & \emptyset & 010 & \emptyset & 001 \\ 011 & \emptyset & \emptyset & 0 & 011 & \emptyset & \emptyset & 0010 & 0010 & 0001 \\ 100 & \emptyset & \emptyset & 100 & \emptyset & \emptyset & 100 & 0010 0001 & 1000 \\ 101 & \emptyset & \emptyset & 101 & \emptyset & \emptyset & 100 & 0010 0001 & 1000 \\ 101 & \emptyset & \emptyset & 101 & \emptyset & \emptyset & 101 & \emptyset & \emptyset & 101 & \emptyset & \emptyset \\ 110 & \emptyset & \emptyset & 110 & \emptyset & \emptyset & 110 & \emptyset & \emptyset & 110 & \emptyset & \emptyset \\ 111 & \emptyset & \emptyset & 111 & \emptyset & \emptyset & 111 & \emptyset & \emptyset$		Ø	Ø					Ø				Ø		Ø	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	111	Ø	Ø		111	Ø		Ø		111		Ø		Ø	
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	Tag		Set	Offse	et	Tag		Set	Offse	et		Tag		Set	Offset
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	000	0110	0000 1	0010	<del></del>	000	011	0 0000 1	0010	)	=	000	011	0 0000 1	0010
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	001		Ø	Ø		001	011	0 0000 1	0001	Ĺ		001	011	0 0000 1	0001
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	010		Ø	Ø		010		Ø	Ø			010		Ø	Ø
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	011		Ø	Ø	$\longrightarrow$	011		Ø	Ø	_	$\rightarrow$	011		Ø	Ø
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	100		Ø	Ø		100		Ø	Ø			100	001	0 0001 0	1000
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	101		Ø	Ø		101		Ø	Ø			101		Ø	Ø
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	110		Ø	Ø		110		Ø	Ø			110		Ø	Ø
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	111		Ø	Ø		111		Ø	Ø			111		Ø	Ø
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	Tag		Set	Offse	et	Tag		Set	Offse	et_	_	Tag		Set	Offset
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	000	0110	0000 1	0010	)	000	001	0 0001 0	0000	)		000	100	0 0100 0	0011
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	001	0110	0000 1	0001		001	011	0 0000 1	0001	L		001	011	0 0000 1	0001
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	010	0010	0100 0			010	001	0 0100 0		)		010	001	0 0100 0	
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	011			Ø	$\longrightarrow$	011			Ø	_	$\rightarrow$	011			Ø
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$		0010			001						0 0001 0				
	101		Ø	Ø		101		Ø	Ø		101		Ø	Ø	
$111 \hspace{.1cm}   \hspace{.1cm} \emptyset \hspace{.1cm}   \hspace{.1cm} \emptyset \hspace{.1cm} 111 \hspace{.1cm}   \hspace{.1cm} \emptyset \hspace{.1cm}   \hspace{.1cm} \emptyset \hspace{.1cm}   \hspace{.1cm} \emptyset$	110									110		Ø	Ø		
	111		Ø	Ø		111		Ø	Ø			111		Ø	Ø

В.

Set

0110 0000 10

Tag

00

Offset

0010

Hex	Tag(First	10) Se	et (Middle	2) Offset (L	ast 4 bit	s)_			
0x000	0000 0000	00 00	00	00	000				
0x840	0   1000 0100	00 (	00	00	000				
0x608	2 0110 0000	10	00	00	010				
0x8400	C   1000 0100	00 (	00	1.	100				
0x841	1 1000 0100	00 (	01	00	0001				
0x214	8 0010 0001	. 01	00	10	000				
0x242	0 0010 0100	00 (	10	00	000				
0x210	0 0010 0001	. 00	00	00	000				
0x840	3 1000 0100	00 (	00	00	)11				
Tag	Set	Offset	Tag	Set	Offset				
00	0000 0000 00	0000	00	1000 0100 00	0000	-			
01	Ø	Ø	$\longrightarrow$ 01	Ø	Ø	$\longrightarrow$			

01		Ø	$\emptyset \longrightarrow$	01		Ø	Ø	$\longrightarrow$	01		Ø	$\emptyset$
10		Ø	Ø	10		Ø	Ø	Ì	10		Ø	Ø
11		Ø	Ø	11		Ø	Ø		11		Ø	Ø
	Tag	Set	Offset		Tag	Set		Offset		Tag	Set	Offset
_	00	1000 0100 0	00 1100	_	00	1000 0100 0	00	1100		00	0010 0001	01 1000
$\longrightarrow$	01	Ø	Ø	$\longrightarrow$	01	1000 0100 0	00	0001	$\longrightarrow$	01	1000 0100	0001
	10	Ø	Ø		10	Ø		Ø		10	Ø	Ø
	11	Ø	Ø		11	Ø		Ø		11	Ø	Ø
	Tag	$\mathbf{Set}$	Offset		Tag	Set		Offset		Tag	$\mathbf{Set}$	Offset
_	00	0010 0001 0	01 1000	_	00	0010 0001 0	01	0000	_	00	1000 0100	00 0011
$\longrightarrow$	01	1000 0100 0	0001	$\longrightarrow$	01	1000 0100 0	00	0001	$\longrightarrow$	01	1000 0100	0001
	10	0010 0100 0	0000		10	0010 0100 0	00	0000		10	0010 0100	0000
	11	Ø	Ø		11	<b>Ø</b>		Ø		11	Ø	Ø

С.

Hex	Tag(First 12)	Offset (Last 4 bits)
0x0000	0000 0000 0000	0000
0x8400	1000 0100 0000	0000
0x6082	0110 0000 1000	0010
0x840C	1000 0100 0000	1100
0x8411	1000 0100 0001	0001
0x2148	0010 0001 0100	1000
0x2420	0010 0100 0010	0000
0x2100	0010 0001 0000	0000
0x8403	1000 0100 0000	0011

Tag(First 12)	Offset (Last 4 bits)		Tag(First 12)	Offset (Last 4 bits)	
0000 0000 0000	0000		0000 0000 0000	0000	=
Ø	$\emptyset$		1000 0100 0000	0000	
Ø	$\emptyset$		Ø	Ø	
Ø	$\emptyset$	$\rightarrow$	Ø	Ø	$\rightarrow$
Ø	$\emptyset$		Ø	Ø	
Ø	$\emptyset$		Ø	Ø	
Ø	$\emptyset$		Ø	Ø	
Ø	$\emptyset$		Ø	Ø	

0x2148

0x2420

 $0\mathrm{x}2100$ 

0x8403

0010 0001 010

0010 0100 001

 $0010\ 0001\ 000$ 

1000 0100 000

Tag(First	12) Offset	(Last 4 l	bits)		Tag(First 12)	Offset (Last 4 bits)	
0000 0000 0	000	0000		:	0000 0000 0000	0000	=
1000 0100 0		0000			0110 0000 1000	0010	
0110 0000 1		0010			1000 0100 0000	1100	
Ø Ø	000	Ø		$\rightarrow$	000 0100 0000 Ø	Ø	$\rightarrow$
Ø		Ø			Ø	Ø	·
Ø		Ø			Ø	Ø	
						· .	
Ø Ø		Ø Ø			Ø Ø	Ø Ø	
Tag(First	12) Offset	(Last 4 l	bits)		Tag(First 12)	Offset (Last 4 bits)	
0000 0000 0	000	0000		:	0000 0000 0000	0000	=
0110 0000 1		0010			0110 0000 1000	0010	
1000 0100 0		1100			1000 0100 0000	1100	
0		Ø		$\rightarrow$	1000 0100 0001	0001	$\rightarrow$
Ø		Ø			0	0	
Ø		Ø			Ø	Ø	
Ø		Ø			Ø	Ø	
Ø		Ø			Ø	Ø	
Tag(First	12) Offset	(Last 4 l	bits)		Tag(First 12)	Offset (Last 4 bits)	
0000 0000 0	000	0000		:	0000 0000 0000	0000	=
0110 0000 1		0010			0110 0000 1000	0010	
1000 0100 0		1100			1000 0100 0000	1100	
1000 0100 0		0001		$\rightarrow$	1000 0100 0000	0001	$\rightarrow$
0010 0001 0		1000			0010 0001 0100	1000	
Ø	100	Ø			0010 0001 0100	0000	
Ø		Ø			0010 0100 0010	Ø	
Ø		Ø			Ø	Ø	
P		ν			<i>V</i>	~	
Tag(First		(Last 4 l	bits) =====	:	Tag(First 12)	Offset (Last 4 bits)	=
0000 0000 0		0000			0000 0000 0000	0000	
0110 0000 1		0010			0110 0000 1000	0010	
1000 0100 0		1100			1000 0100 0000	1100	
1000 0100 0		0001		$\rightarrow$	1000 0100 0001	0001	
0010 0001 0		1000			0010 0001 0100	1000	
0010 0100 0		0000			0010 0100 0010	0000	
0010 0001 0	000	0000			0010 0001 0000	0000	
Ø		Ø			1000 0100 0000	0011	
D.							
Hex T	ag(First 11)	Offset	(Last	5 1	bits)		
	000 0000 000		0 000				
	000 0100 000		0 000	0			
0x6082 0	110 0000 100		0 001				
0x840C 1	000 0100 000		0 110	0			
0x8411 1	000 0100 000		1 000	1			

0 1000

0 0000 0 0000

0 0011

Tag(First 12)	Offset (Last 4 bits)		Tag(First 12)	Offset (Last 4 bits)	_
0000 0000 000	0 0000	-	0000 0000 000	0 0000	_
Ø	$\emptyset$		1000 0100 000	0 0000	
Ø	Ø		Ø	Ø	
Ø	Ø	$\rightarrow$	Ø	Ø	$\rightarrow$
Ø	Ø		Ø	Ø	
Ø	Ø		Ø	$\emptyset$	
Ø	Ø		Ø	Ø	
Ø	Ø		Ø	Ø	
Tag(First 12)	Offset (Last 4 bits)	= :	Tag(First 12)	Offset (Last 4 bits)	=
0000 0000 000	0 0000		0000 0000 000	0 0000	
1000 0100 000	0 0000		1000 0100 000	0 0000	
0110 0000 100	0 0010		0110 0000 100	0 0010	
Ø	Ø	$\rightarrow$	1000 0100 000	0 1100	$\rightarrow$
Ø	Ø		Ø	Ø	
Ø	Ø		Ø	Ø	
Ø Ø	Ø Ø		Ø	Ø Ø	
V	ν		Ψ	۳	
Tag(First 12)	Offset (Last 4 bits)	= :	Tag(First 12)	Offset (Last 4 bits)	=
0000 0000 000	0 0000		0000 0000 000	0 0000	
1000 0100 000	0 0000		1000 0100 000	0 0000	
0110 0000 100 1000 0100 000	0 0010	$\rightarrow$	0110 0000 100 1000 0100 000	0 0010	$\rightarrow$
1000 0100 000 Ø	1 0001 Ø	,	0010 0001 010	1 0001 0 1000	,
Ø	Ø		Ø Ø	0 1000 Ø	
Ø	Ø		Ø	Ø	
Ø	Ø		Ø	Ø	
Tag(First 12)	Offset (Last 4 bits)		Tag(First 12)	Offset (Last 4 bits)	
0000 0000 000	0 0000	= :	0000 0000 000	0 0000	=
1000 0100 000	0 0000		1000 0100 000	0 0000	
0110 0000 100	0 0010		0110 0000 100	0 0010	
1000 0100 000	1 0001	$\rightarrow$	1000 0100 000	1 0001	$\rightarrow$
0010 0001 010	0 1000		0010 0001 010	0 1000	
$0010\ 0100\ 001$	0 0000		$0010\ 0100\ 001$	0 0000	
Ø	$\emptyset$		$0010\ 0001\ 000$	0 0000	
Ø	Ø		Ø	Ø	
Tag(First 12)	Offset (Last 4 bits)				
0000 0000 000	0 0000	=			
$1000\ 0100\ 000$	0 0000				
$0110\ 0000\ 100$	0 0010				
$1000\ 0100\ 000$	1 0001				
0010 0001 010	0 1000				
0010 0100 001	0 0000				
0010 0001 000	0 0000				
1000 0100 000	0 0011				

3. (20 points) Eviction Policies. Consider the following request sequence:

(a) (5 points) Provide a logical diagram that shows how an optimal fully associative cache of size 3 will cache this sequence.

- (b) (5 points) Provide a logical diagram that shows how a FIFO fully associative cache of size 3 will cache this sequence.
- (c) (5 points) Provide a logical diagram that shows how an LRU fully associative cache of size 3 will cache this sequence.
- (d) (5 points) Label each miss in your diagram for part c with its type.

For all sub questions **bold represents replacement** and *italics represents hit* 

Cache	1 (A)	2 (B)	3 (C)	4 (B)	5 (A)	6 (D)	7 (C)	8 (E)	9 (B)	10 (D)	11 (A)	12 (C)
1	A	A	A	A	A	D	D	D	D	D	A	$\mathbf{C}$
$\frac{2}{3}$	Ø	$\mathbf{B}$	В	B	В	В	В	В	B	В	В	В
3	Ø	Ø	$\mathbf{C}$	$\mathbf{C}$	$\mathbf{C}$	$\mathbf{C}$	C	${f E}$	$\mathbf{E}$	$\mathbf{E}$	$\mathbf{E}$	$\mathbf{E}$
В.	'											

Cache	1 (A)	2 (B)	3 (C)	4 (B)	5 (A)	6 (D)	7 (C)	8 (E)	9 (B)	10 (D)	11 (A)	12 (C)
1	A (1)	D (6)	A (11)	A (11)								
2	Ø	B (2)	B(2)	B(2)	B(2)	B(2)	B(2)	E (8)	E(8)	E(8)	E(8)	C(12)
3	Ø	Ø	C(3)	C(3)	C(3)	C(3)	C(3)	C(3)	B(9)	B(9)	B (9)	B(9)
	•											

 $\mathbf{C}.$ 

Cache	1 (A)	2 (B)	3 (C)	4 (B)	5 (A)	6 (D)	7 (C)	8 (E)	9 (B)	10 (D)	11 (A)	12 (C)
1	A (1)	A (1)	A (1)	A (1)	A (5)	A (5)	A (5)	E (8)	E (8)	E (8)	A (11)	A (11)
2	Ø	B (2)	B(2)	B(4)	B(4)	B(4)	C(7)	C(7)	C(7)	D (10)	D (10)	D(10)
3	Ø	Ø	C(3)	C(3)	C(3)	D (6)	D(6)	D(6)	B (9)	B (9)	B (9)	C (12)

D.

- Requests 1-3: Compulsory misses
- Request 6: Compulsory miss
- Request 7: Replacement miss
- $\bullet$  Request 8: Compulsory miss
- Request 9-10: Replacement miss
- Request 11-12: Capacity miss