Susmitha Shailesh

CS383 Homework 4

5.1.1

2 64-bit integers can be stored in a 16-byte cache block.

5.1.2

I, J, and B[I][0] exhibit temporal locality.

5.1.3

A[I][J] exhibits spatial locality.

5.2.1.

Word Address	Binary Address	Tag	Index	Hit/Miss
0x03	00000011	0	3	Miss
0xb4	10110100	b	4	Miss
0x2b	00101011	2	b	Miss
0x02	00000010	0	2	Miss
0xbf	10111111	b	f	Miss
0x58	01011000	5	g	Miss
0xbe	10111110	b	e	Miss
0x0e	00001110	0	e	Miss
0xb5	10110101	b	5	Miss
0x2c	00101100	2	c	Miss
0xba	10110101	b	a	Miss
0xfd	11111101	f	d	Miss

5.2.2

Word Address	Binary Address	Tag	Index	Offset	Hit/Miss

0x03	00000011	0	1	1	Miss
0xb4	10110100	b	2	0	Miss
0x2b	00101011	2	5	1	Miss
0x02	00000010	0	1	0	Hit
0xbf	10111111	b	7	1	Miss
0x58	01011000	5	4	0	Miss
0xbe	10111110	b	6	0	Hit
0x0e	00001110	0	7	0	Miss
0xb5	10110101	b	2	1	Hit
0x2c	00101100	2	6	0	Miss
0xba	10110101	b	5	0	Miss
0xfd	11111101	f	6	1	Miss

5.2.3

			Cache 1		Cache 2		Cache 3	
Word Address	Binary Address	Tag	Index	Hit/ Miss	Index	Hit/ Miss	Index	Hit/ Miss
0x03	0000 0011	0x00	3	Miss	1	Miss	0	Miss
0xb4	1011 0100	0x16	4	Miss	2	Miss	1	Miss
0x2b	0010 1011	0x05	3	Miss	1	Miss	0	Miss
0x02	0000 0010	0x00	2	Miss	1	Miss	0	Miss
0xbf	1011 1111	0x17	7	Miss	3	Miss	1	Miss
0x58	0101	0x0b	0	Miss	0	Miss	0	Miss

	1000							
0xbe	1011 1110	0x17	6	Miss	3	Hit	1	Hit
0x0e	0000 1110	0x01	6	Miss	3	Miss	1	Miss
0xb5	1011 0101	0x16	5	Miss	2	Hit	1	Miss
0x2c	0010 1100	0x05	4	Miss	2	Miss	1	Miss
0xba	1011 0101	0x17	2	Miss	1	Miss	0	Miss
0xfd	1111 1101	0x1F	5	Miss	2	Miss	1	Miss

5.5.1

The cache block size is 4 words. Because there are two bits for the block offset and each can be either a 0 or 1, there are 2^2 spaces for words.

5.5.2

The cache has 32 blocks. There are 5 bits that account for index. Because each of those bits can be a 0 or 1, there are 2^5 blocks in each cache.

5.5.3

Data storage bits: $\frac{32\ blocks}{1\ cache} * \frac{4\ words}{1\ block} * \frac{8\ bytes}{1\ word} = 8192\ bits$

Total number of bits for cache implementation:

$8192\ data\ storage\ bits\ +\ (54\ tag\ bits\ *\ 32)\ +\ (1\ valid\ bit\ *\ 32)\ = 9952\ bits$

$\frac{9952}{8192} = 1.215$

5.5.4

Byte Address	Binary Address	Tag	Index	Offset	Hit/ Miss	Bytes Replaced
0x00	0000 0000	0x0	0x00	0x00	Miss	

	0000					
0x04	0000 0000 0010	0x0	0x00	0x04	Hit	
0x10	0000 0001 0000	0x0	0x00	0x10	Hit	
0x84	0000 1000 0100	0x0	0x04	0x04	Miss	
0xe8	0000 1110 1000	0x0	0x07	0x08	Miss	
0xa0	0000 1010 0000	0x0	0x05	0x00	Miss	
0x400	0100 0000 0000	0x1	0x00	0x00	Miss	0x00-0x1F
0x1e	0000 1000 1100	0x0	0x00	0x1e	Miss	0x400-0x41F
0x8c	0000 1000 1100	0x0	0x04	0x0c	Hit	
0xc1c	1100 0001 1100	0x3	0x00	0x1c	Miss	0x00-0x1F
0xb4	0000 1011 0100	0x0	0x05	0x14	Hit	
0x884	1000 1000 0100	0x2	0x05	0x04	Miss	0x80-0x9f

5.5.5

4 *hits* / 12 *total* = 3.33

5.5.6

<4, 2, Mem[0x880]-Mem[0x89f]>

<0, 3, Mem[0xC00]-Mem[0xC1F]>

<5, 0, Mem[0x0A0]-Mem[0x0Bf]>

<7, 0, Mem[0x0e0]-Mem[0x0ff]>

5.9.1

$0.04 * (20 * 8) = 6.4$

$0.03 * (20 * 16) = 9.6$

$0.02 * (20 * 32) = 12.8$

$0.015 * (20 * 64) = 19.2$

$0.01 * (20 * 128) = 25.6$

The optimal block size latency is B = 8.

5.9.2

$0.04 * (24 + 8) = 1.28$

$0.03 * (24 + 16) = 1.2$

$0.02 * (24 + 32) = 1.12$

$0.015 * (24 + 64) = 1.32$

$0.01 * (24 + 128) = 1.52$

The optimal block size latency is B = 32.

5.9.3

For constant miss latency, it would have whichever number of blocks has the lowest miss rate.

So, the optimal block size is 128 blocks.

5.11.1

48 x 5 = 196 bytes

48 / 6 = 8 lines

2 words , 1 bit offset

8, n = 3, index

Tag: 60 bits

Index: 3 bits

Offset: 1 bit

	Cache 1			Cache 2			Cache 3		
	Tag	Data	Hit/ Miss	Tag	Data	Hit/ Miss	Tag	Data	Hit/ Miss
0									
1									
2									
3									
4									
5									
6									

5.11.2

Word Address	Binary Address	Tag	Index	Offset	Hit/ Miss	Way0	Way1	Way2
0x03	0000 0011	0x0	1	1	Miss	T1 = 0		
0xb4	1011 0100	0xb	2	0	Miss	T1=0		
0x2b	0010 1011	0x2	5	1	Miss	T1=0 T2=b T5=2		
0x02	0000 0010	0x0	1	0	Hit	T1=0 T2=b T5=2		
0xbe	1011 1110	0xb	7	0	Miss	T1=0 T2=b T5=2		

						T7=b		
0x58	0101 1000	0x5	4	0	Miss	T1=0 T2=b T4=5 T5=2 T7=b		
0xbf	1011 1111	0xb	7	1	Hit	T1=0 T2=b T4=5 T5=2 T7=b		
0x0e	0000 1110	0x0	7	0	Miss	T1=0 T2=b T4=5 T5=2 T7=b	T7=0	
0x1f	0001 1111	0x1	7	1	Miss	T1=0 T2=b T4=5 T5=2 T7=b	T7=0	T7=1
0xb5	1011 0101	0xb	2	1	Hit	T1=0 T2=b T4=5 T5=2 T7=b	T7=0	T7=1
0xbf	1011 1111	0xb	7	1	Hit	T1=0 T2=b T4=5 T5=2 T7=b	T7=0	T7=1
0xba	1011 1010	0xb	5	0	Miss	T1=0 T2=b T4=5 T5=2 T7=b	T7=2 T5=b	T7=1
0x2e	0010 1110	0x2	7	0	Miss	T1=0 T2=b	T7=2 T5=b	T7=1

						T4=5 T5=2 T7=b		
0xce	1100 1110	0xc	7	0	Miss	T1=0 T2=b T4=5 T5=2 T7=b	T7=2 T5=b	T7=c

5.11.3

8 x 4 = 32 bytes

8/1 = 8 blocks

2^0 = 1, 0 offset

Tag: 64 bits

Index: 0 bits

Offset: 0 bits

T	D	H / M	T	D	H / M	T	D	H / M	T	D	H / M	T	D	H / M	T	D	H / M	T	D	H / M	T	D	H / M

5.13.1

MTBF = MTTR + MTTF

3 years + 1 day = 1096 days

5.13.2

Availability = MTTF / (MTTF + MTTR)

$\frac{1096}{1097}$ = 0.99

5.13.3

When the MTTR reaches 0, the availability reaches 1. That is a potential replacement time. This is a feasible scenario in some cases.

5.13.4

When the MTTR gets higher, the availability does decrease. However, if the MTTF is significantly greater than the MTTR, the specific value of the MTTR does not matter as much and the availability is not drastically impacted.