

Question 1.

Given ~~1 KB~~ ^{1 MB}

1 KB cache

Direct-Mapped

line size = 64 bit.

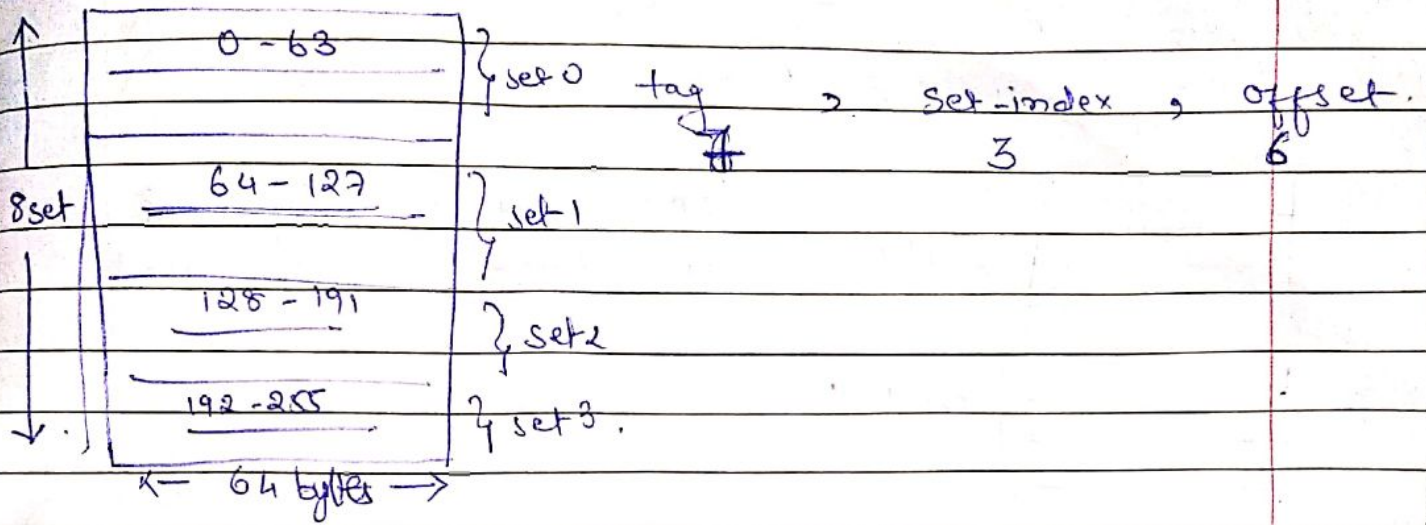
↑	0 - 63 / 1024 - 1053 / 0x0000 - 0x0003			
	64 - 127			
	128 - 191			
	192 - 255			
	128			
16 Line		tag	index	offset
		6 bit	4 bit	6 bit
		000...	0000	000000
↓				
← 64 bit Bytes →		tag	index	offset
1	0x0001	Miss.	000 - 0000	000000
2	0x0002	Hit.	"	" 000000
3	0x0003	Hit.	"	" 000001
65	0x0041	Miss.	000 - 0001	000000
129	0x0051	Miss.	000 - 0010	000000
161	0x00A1	Hit.	000 - 0010	100000
193	0x00C1	Miss.	000 - 0011	000000
1025	0x0401	Miss.	000 - 1000	000000
1026	0x0402	Hit.	000 - 1000	000001
1	0x0001	Miss.	0000 - 0000	000001
2	0x0002	Hit.	0000 - 0000	000000
3	0x0003	Hit.	0000 - 0000	000001

Cache Hits = 6.

$$\text{Hit ratio} = 6/12 = 1/2$$

Question 2

Now cache is 2-way Associative



	Int	bin	Hit/Miss
0x0001	1	000--0-0000-000001	Miss
0x0002	2	000--0-0000-000010	Hit
0x0003	3	000--0-0000-000011	Hit
0x0041	65	00--0-0001-000001	Miss
0x0081	129	00--0-0010-000001	Miss
0x00A1	161	00--0-0010-100001	Hit
0x00C1	193	00--0-0011-000001	Miss
0x0401	1025	00--1-0000-000001	Miss
0x0402	1026	00--1-0000-000010	Hit
0x0001	1	000--0-0000-000001	Hit
0x0002	2	000--0-0000-000010	Hit
0x0003	3	000--0-0000-000011	Hit

Cache hits :- 7

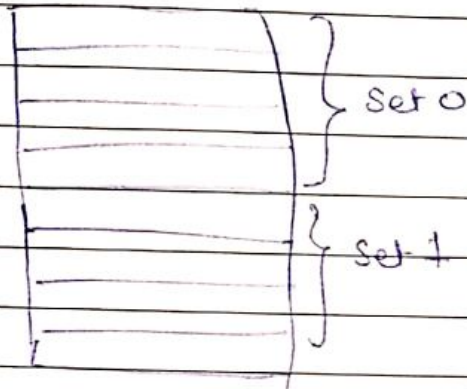
Hit ratio :- $\frac{7}{12}$

Question 3

128 B cache.

Line size 16 B

4-way Associative.



LRU: Least Recently Used

LFU: Least frequently used.

When LFU performs better than LRU

→ LFU captures pattern.

If there is no such pattern (random) LRU performs better.

Let say A, B, C, D, E have same set-index diff
Tag.

a) LRU better than LFU

Let's consider.

A, B, C, D, A, A, A, B, B, B, C, D, E, E, C, D

This case we can see LFU will replace C when E comes, but LRU replace A. ~~the~~ In this case LRU performed better than LFU. As there is no pattern.

b) LRU performs better LRU

lets there is Loop. with some condition

C, C, A, B, C, D, A, B, A, B, A, B, E, B, C, D, A, A, B, A, B.

In this case

When first time E is read we can see.

LRU will replace C, but LRU will replace D as it is only used which is more reasonable as it is least used and from series of read it is also not read frequently afterward.

Freq.

A: - 5

B: - 5

C: - 4

D: - 2

E: - 1

When 2nd Time D comes LRU replace A, but LRU replaces E, which again Good choice according to given series of operation.

Question 4:

My Answer, I believe same PA can be understood as extension of SA where whole cache is one just one set. So by using PA it just gives me more space lines to store for particular set/index. It will still perform ^{same} equally for similar test case that we did in question 3.

Question 5:

Cache: 8 words.

LRU replacement Scheme

Line: 2 word

		Content at end
↑	(20, 21), (28, 29)	28, 29
4 line	22/23,	22, 23
	24/25, 30/31	30, 31
↓	26/27, (20, 21)	20, 21

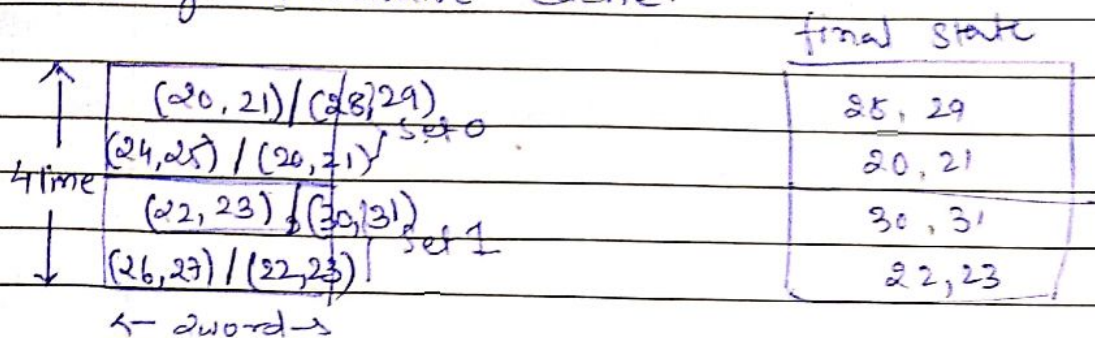
← 2 word →

20	Miss	27	Hit
21	Hit	28	Miss
22	Miss	29	Hit
23	Hit	22	Hit
24	Miss	30	Miss
25	Hit	21	Miss
26	Miss	23	Hit
		31	Hit

Hit rate := $8/15$

Question 6

2-way associative cache.



let say if address is divisible by 4 its set 0
else set 1.

20	Miss	24	Hit
21	Hit	25	Hit
22	Miss	30	Miss
23	Hit	21	Miss
24	Miss	23	Miss
25	Hit	31	Hit
26	Miss		
27	Hit		
28	Miss		

$$\text{Hit ratio} = 7/15$$