

NTUA DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING



Σειρά 3

Αρχιτεκτονική Υπολογιστών

Μαρίνα Φραγκούλη el22429
21/12/2024

Τμήμα 3 /

1. Μέρος Α	2
1) Εκφώνηση	2
2) Απάντηση	2
2. Μέρος Β	3
1) Κώδικας σε C	3
2) Υποερώτημα 1	3
3) Υποερώτημα 2	3
4) Υποερώτημα 3	16

1. ΜΕΡΟΣ Α

1) ΕΚΦΩΝΗΣΗ

Δίνεται επεξεργαστής με ένα επίπεδο κρυφής μνήμης με μέσο χρόνο πρόσβασης στη μνήμη 2.4 κύκλους ρολογιού. Πιο συγκεκριμένα, τα hits εξυπηρετούνται σε 1 κύκλο ενώ τα misses εξυπηρετούνται από την κύρια μνήμη σε 80 κύκλους. Σας ζητούν να προσθέσετε ένα δεύτερο επίπεδο κρυφής μνήμης ώστε η επιτάχυνση (speedup) του μέσου χρόνου πρόσβασης στη μνήμη να είναι ίση με 1.65. Ποιο το hit rate αυτής της L2, αν η πρόσβαση σε αυτή στοιχίζει 6 κύκλους;

2) ΑΠΑΝΤΗΣΗ

AMAT: Average Memory Access Time

HT: Hit Time συμπεριλαμβανομένου και του χρόνου που χρειάζεται για να προσδιοριστεί αν η προσπέλαση είναι ευστοχία ή αστοχία

MP: Miss Penalty συμπεριλαμβανομένου και του χρόνου προσπέλασης του μπλοκ, της μεταφοράς από το ένα επίπεδο στο άλλο, της εισαγωγής του επιπέδου που παρουσίασε την αστοχία και στην συνέχεια της μεταβίβασης του μπλοκ στο πρόγραμμα που το ζήτησε

HR: Hit Rate

MR: Miss Rate

$$AMAT_{L1} = HR_{L1} * HT_{L1} + MR_{L1} * MP_{L1} \quad (1)$$

$$AMAT_{L1+L2} = HR_{L1} * HT_{L1} + MR_{L1} * HR_{L2} * HT_{L2} + MR_{L1} * MR_{L2} * MP_{L2} \quad (2)$$

$$\frac{AMAT_{L1}}{AMAT_{L1+L2}} = 1.65 \quad (3)$$

$$(1): 2.4 = HR_{L1} * 1 + (1 - HR_{L1}) * 80 \rightarrow HR_{L1} = 0.982 \text{ και } MR_{L1} = 0.018$$

$$(3): AMAT_{L1+L2} = \frac{AMAT_{L1}}{1.65} \text{ που με αντικατάσταση από (1), (2):}$$

$$\frac{2.4}{1.65} = 0.982 * 1 + 0.018 * HR_{L2} * 6 + 0.018 * (1 - HR_{L2}) * 80$$

$$\rightarrow HR_{L2} = 0.726 \text{ και } MR_{L2} = 0.274$$

Άρα το hit rate αυτής της L2 είναι 0.726 ή 72.6%.

2. ΜΕΡΟΣ Β

1) ΚΩΔΙΚΑΣ ΣΕ C

```
int i, j;

double a[32][8], b[512];

for (i=0; i<8; i++)
    for(j=0; j<8; j++)
        a[i][j] = a[i+2][j]+a[i+1][j]+a[i][j]+b[i*8+j];
```

2) ΥΠΟΕΡΩΤΗΜΑ 1

Εκφώνηση:

A) Δείξτε το σχήμα διευθυνσιοδότησης, δηλαδή πόσα bits χρειάζονται τα πεδία tag, index και block_offset των διευθύνσεων μνήμης του επεξεργαστή.

Απάντηση:

Τα lines της cache είναι $\frac{256}{32}$ άρα χρειάζονται $\log_2 \frac{256}{32} = \log_2 8 = 3$. Έστω n το μέγεθος σε bits του πεδίου tag έχουμε n=3 bits

Αφού «Το μέγεθος του block είναι 32 bytes, ενώ η μικρότερη μονάδα δεδομένων που μπορεί να διευθυνσιοδοτηθεί είναι το 1 byte.» έχουμε ($2^5 = 32$) 5 bits block offset. Έστω m το μέγεθος σε bits του πεδίου block_offset έχουμε m=5 bits.

Για το μέγεθος του πεδίου ετικέτας tag ισχύει η παρακάτω εξίσωση (Patterson, Hennessy pg 456)

$$\#tag = 32 - (n + m) = 32 - (3 + 5) = 24 \text{ bits.}$$

31 30 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

Tag	index	block_ offset
-----	-------	------------------

Το μέγεθος του block δίνεται σε bytes οπότε αγνοούμε το μέγεθος της λέξης.

3) ΥΠΟΕΡΩΤΗΜΑ 2

Εκφώνηση:

B) Βρείτε το συνολικό αριθμό hits και misses για όλη την εκτέλεση του παραπάνω κώδικα.

Απάντηση:

Σύμφωνα με την εκφώνηση προσπέλαση στην κρυφή μνήμη έχουμε μόνο για τα στοιχεία του πίνακα. Επίσης κάθε στοιχείο του πίνακα είναι 8 bytes.

- 1 block = 32 bytes
- 1 στοιχείο = 8 bytes
- Πίνακας αποθηκευμένος κατά γραμμές
- Cache χώρος για 8 blocks των 4 στοιχείων του πίνακα το καθένα.
- Για τον πίνακα a
 - RAM για τα πρώτα 8*8 στοιχεία του a χρησιμοποιεί 16 blocks των 4 στοιχείων του πίνακα το καθένα. Με χρώματα φαίνεται ποια δύο block με στοιχεία του a πέφτουν στο ίδιο block της CACHE.
 - Για να πάμε από index 2D([i][j]) σε index 1D(k) χρησιμοποιούμε τον τύπο $k = (i * 8 + j)$
 - Επειδή έχουμε direct map:

Σε ένα block της cache θα απεικονίζονται 4 διαδοχικά στοιχεία του πίνακα
Πχ $a[i][j]$, $a[i][j+1]$, $a[i][j+2]$, $a[i][j+3]$

$$\begin{aligned} index &= (\text{Διεύθυνση block}) \bmod (\text{πλήθος μπλοκ cache}) \\ &= \left(\frac{\text{Διεύθυνση στοιχείου του πίνακα}}{4} \right) \bmod (\text{πλήθος μπλοκ cache}) \\ &= \frac{i * 8 + j}{4} \bmod 8 \end{aligned}$$

- Για τον πίνακα b μας νοιάζει για συγκεκριμένη τιμή των i, j ποιες τιμές του πίνακα έχουν το ίδιο index. Εδώ ο πίνακας είναι μίας διάστασης ωστόσο τα στοιχεία τα οποία καλεί ο κώδικας έχουν όλα Διεύθυνση στοιχείου του πίνακα $i * 8 + j$ και /4 για να πάμε από διεύθυνση στοιχείου σε διεύθυνση block πάλι λοιπόν

$$\begin{aligned} index &= (\text{Διεύθυνση block}) \bmod (\text{πλήθος μπλοκ cache}) \\ &= \left(\frac{\text{Διεύθυνση στοιχείου του πίνακα}}{4} \right) \bmod (\text{πλήθος μπλοκ cache}) \\ &= \frac{i * 8 + j}{4} \bmod 8 \end{aligned}$$

Το /4 εξηγείται και αφού αγνοούμε τα δύο τελευταία ψηφία της διεύθυνσης στο 2δικό και τα επόμενα 3 lsb είναι το index μας.

Το Write Allocate έχει να κάνει με την περίπτωση των write miss δηλαδή όταν πάμε να αλλάξουμε την τιμή μιας μεταβλητής και αυτή δεν βρίσκεται στην cache. Συγκεκριμένα με αυτή σε περίπτωση **write miss**, το block που περιέχει τα δεδομένα **φορτώνεται στην cache**, και μετά γίνεται η εγγραφή στη cache.

RAM

2D	1D	Space	Address
0.0	0	8 B	XXX000000
0.1	1	8 B	000001
0.2	2	8 B	000010
0.3	3	8 B	000011
0.4	4	8 B	000100
0.5	5	8 B	000101
0.6	6	8 B	000110
0.7	7	8 B	000111
1.0	8	8 B	001000
1.1	9	8 B	001001
1.2	10	8 B	001010
1.3	11	8 B	001011
1.4	12	8 B	001100
1.5	13	8 B	001101
1.6	14	8 B	001110
1.7	15	8 B	001111
2.0	16	8 B	010000
2.1	17	8 B	010001
2.2	18	8 B	010010
2.3	19	8 B	010011
2.4	20	8 B	010100
2.5	21	8 B	010101
2.6	22	8 B	010110
2.7	23	8 B	010111
3.0	24	8 B	011000
3.1	25	8 B	011001
3.2	26	8 B	011010
3.3	27	8 B	011011
3.4	28	8 B	011100
3.5	29	8 B	011101
3.6	30	8 B	011110
3.7	31	8 B	011111
4.0	32	8 B	100000
4.1	33	8 B	100001
4.2	34	8 B	100010
4.3	35	8 B	100011
4.4	36	8 B	100100
4.5	37	8 B	100101
4.6	38	8 B	100110
4.7	39	8 B	100111
5.0	40	8 B	101000
5.1	41	8 B	101001
5.2	42	8 B	101010
5.3	43	8 B	101011
5.4	44	8 B	101100

CACHE\$

Space	Address
32B	000
32B	001
32B	010
32B	011
32B	100
32B	101
32B	110
32B	111

Line 0
Line 1
Line 2
Line 3
Line 4
Line 5
Line 6
Line 7

Block 0

Block 1

Block 2

Block 3

Block 4

Block 5

Block 6

Block 7

Block 8

Block 9

Block 10

Block 11

5.5	45	8 B	101101
5.6	46	8 B	101110
5.7	47	8 B	101111
6.0	48	8 B	110000
6.1	49	8 B	110001
6.2	50	8 B	110010
6.3	51	8 B	110011
6.4	52	8 B	110100
6.5	53	8 B	110101
6.6	54	8 B	110110
6.7	55	8 B	110111
7.0	56	8 B	111000
7.1	57	8 B	111001
7.2	58	8 B	111010
7.3	59	8 B	111011
7.4	60	8 B	111100
7.5	61	8 B	111101
7.6	62	8 B	111110
7.7	63	8 B	111111

Block 12

Block 13

Block 14

Block 15

Τα πρώτα 8*8 στοιχεία του πίνακα *a* στην RAM

-	0	8 B	00000000
-	1	8 B	00000001
-	2	8 B	00000010
-	3	8 B	00000011
-	4	8 B	00000100
-	5	8 B	00000101
-	6	8 B	00000110
-	7	8 B	00000111

Τα στοιχεία του πίνακα *b* στην RAM

-	504	8 B	111111000
-	505	8 B	111111001
-	506	8 B	111111010
-	507	8 B	111111011
-	508	8 B	111111100
-	509	8 B	111111101
-	510	8 B	111111110
-	511	8 B	111111111

2D το index του αντίστοιχου στοιχείου το πίνακα όταν αυτός γράφεται στις 2 διαστάσεις

1D το index του αντίστοιχου στοιχείου το πίνακα όταν αυτός γράφεται στη μία διάσταση.

Address η διεύθυνση στην μνήμη.

Space πόση πληροφορία έχει μέσα.

Ακολουθούν πίνακες που δείχνουν τα hits και τα misses καθώς και τα περιεχόμενα της CACHE ανά γραμμή για μερικές τιμές των i, j .

i=0, j=0

a[2][0]	comp m	block a4	line 4
a[1][0]	comp m	block a2	line 2
a[0][0]	comp m	block a0	line 0

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

b[0]	comp m	blockb0	line 0
------	--------	---------	--------

	00	01	10	11
000	b[0]	b[1]	b[2]	b[3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

a[0][0]	write m	block a0	line 0
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

i=0, j=1

a[2][1]	hit	block a4	line 4
a[1][1]	hit	block a2	line 2
a[0][1]	hit	block a0	line 0

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

b[1]	con m	blockb0	line 0
------	-------	---------	--------

	00	01	10	11
000	b[0]	b[1]	b[2]	b[3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

a[0][1]	write m	block a0	line 0
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

i=0, j=2

a[2][2]	hit	block a4	line 4
a[1][2]	hit	block a2	line 2
a[0][2]	hit	block a0	line 0

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

b[2]	con m	block b0	line 0
------	-------	----------	--------

	00	01	10	11
000	b[0]	b[1]	b[2]	b[3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

a[0][2]	write m	block a0	line 0
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

i=0, j=3

a[2][3]	hit	block a4	line 4
a[1][3]	hit	block a2	line 2
a[0][3]	hit	block a0	line 0

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

b[3]	con m	block b0	line 0
------	-------	----------	--------

	00	01	10	11
000	b[0]	b[1]	b[2]	b[3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

a[0][3]	write m	block a0	line 0
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

i=0, j=4

a[2][4]	com m	block a5	line 5
a[1][4]	com m	block a3	line 3
a[0][4]	com m	block a1	line 1

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

b[4]	comp m	block b1	line 1
------	--------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	b[4]	b[5]	b[6]	b[7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

a[0][4]	write m	block a1	line 1
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

i=0, j=5

a[2][5]	hit	block a5	line 5
a[1][5]	hit	block a3	line 3
a[0][5]	hit	block a1	line 1

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

b[5]	con m	block b1	line 1
------	-------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	b[4]	b[5]	b[6]	b[7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

a[0][5]	write m	block a1	line 1
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

i=0, j=6

a[2][6]	hit	block a5	line 5
a[1][6]	hit	block a3	line 3
a[0][6]	hit	block a1	line 1

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

b[6]	con m	blockb1	line 1
------	-------	---------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	b[4]	b[5]	b[6]	b[7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

a[0][6]	write m	block a1	line 1
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

i=0, j=7

a[2][7]	hit	block a5	line 5
a[1][7]	hit	block a3	line 3
a[0][7]	hit	block a1	line 1

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

b[7]	con m	blockb1	line 1
------	-------	---------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	b[4]	b[5]	b[6]	b[7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

a[0][7]	write m	block a1	line 1
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110				
111				

i=1, j=0

a[3][0]	com m	block a6	line 6
a[2][0]	hit	block a4	line 4
a[1][0]	hit	block a2	line 2

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

b[8]	com m	blockb2	line 2
------	-------	---------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	b[8]	b[9]	b[10]	b[11]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

a[1][0]	write m	block a2	line 2
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

i=1, j=1

a[3][1]	hit	block a6	line 6
a[2][1]	hit	block a4	line 4
a[1][1]	hit	block a2	line 2

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

b[9]	com m	blockb2	line 2
------	-------	---------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	b[8]	b[9]	b[10]	b[11]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

a[1][1]	write m	block a2	line 2
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

i=1, j=2

a[3][2]	hit	block a6	line 6
a[2][2]	hit	block a4	line 4
a[1][2]	hit	block a2	line 2

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

b[10]	con m	blockb2	line 2
-------	-------	---------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	b[8]	b[9]	b[10]	b[11]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

a[1][2]	write m	block a2	line 2
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

i=1, j=3

a[3][3]	hit	block a6	line 6
a[2][3]	hit	block a4	line 4
a[1][3]	hit	block a2	line 2

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

b[11]	con m	blockb2	line 2
-------	-------	---------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	b[8]	b[9]	b[10]	b[11]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

a[1][3]	write m	block a2	line 2
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

i=1, j=4

a[3][4]	com m	block a7	line 7
a[2][4]	hit	block a5	line 5
a[1][4]	hit	block a3	line 3

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111	a[3][4]	a[3][5]	a[3][6]	a[3][7]

b[12]	com m	block 3	line 3
-------	-------	---------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	b[12]	b[13]	b[14]	b[15]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111	a[3][4]	a[3][5]	a[3][6]	a[3][7]

a[1][4]	write m	block a3	line 3
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111	a[3][4]	a[3][5]	a[3][6]	a[3][7]

Από τον κώδικα καταλαβαίνουμε ότι είτε η CACHE είναι κενή είτε έχει άλλο block με στοιχεία του a θα έχω την πρώτη φορά που καλώ κάποιο στοιχείο από αυτό το block compulsory miss. Επίσης θα έχω compulsory miss ή conflict miss όταν καλώ να διαβάσω από την cache block με στοιχεία του b (compulsory την πρώτη φορά που φέρνω αυτό το block του b από την μνήμη και μετά θα έχω 3 φορές conflict αφού 4 στοιχεία του b σε ένα block αφού θα έχω εκεί στοιχεία του a). Τέλος έχω write miss όταν πάω να αλλάξω την τιμή του στοιχείου a[i][j].

Για i=0

Κάθε 4 αλλαγές τιμή του j έχω 3 com misses (στοιχεία a). Το j αλλάζει 8 φορές άρα $3 \cdot 8 / 4 = 6$ misses

+ Για κάθε αλλαγή του j έχω 1 (con ή com) miss και 1 write miss. Το j αλλάζει 8 φορές άρα $2 \cdot 8 = 16$ misses

Δηλαδή #misses=22 misses

Τα υπόλοιπα από τα $5 \cdot 8 = 40$ requests μας είναι hit.

Δηλαδή #hits= 40-22=18 hits

Για i≠0

Κάθε 4 αλλαγές τιμή του j έχω 1 com misses (στοιχεία a). Το j αλλάζει 8 φορές άρα $1 \cdot 8 / 4 = 2$ misses

+ Για κάθε αλλαγή του j έχω 1 (con ή com) miss και 1 write miss. Το j αλλάζει 8 φορές άρα $2 \cdot 8 = 16$ misses

Δηλαδή για κάθε τιμή του i ($i > 0$) έχουμε $16 + 2 = 18$ misses

Τα υπόλοιπα από τα $5 \cdot 8 = 40$ requests μας είναι hit.

Δηλαδή για κάθε τιμή του i ($i > 0$) έχουμε $40 - 18 = 22$ hits

Έχουμε 7 φορές να αλλάζει η τιμή του i (για $i \neq 0$)

Επομένως συνολικά έχουμε

#misses=22+7*18=148 και

#hits= 18+7*22=172

Το μοτίβο το οποίο ακολουθούν τα hits και τα misses φαίνεται και στον παρακάτω πίνακα.

	$a[i+2][j]$	$a[i+1][j]$	$a[i][j]$	$b[i*8+j]$	$a[i][j]$
$i=0, j=0$	m	m	m	m	m
$i=0, j=1$	h	h	h	m	m
$i=0, j=2$	h	h	h	m	m
$i=0, j=3$	h	h	h	m	m
$i=0, j=4$	m	m	m	m	m
$i=0, j=5$	h	h	h	m	m
$i=0, j=6$	h	h	h	m	m
$i=0, j=7$	h	h	h	m	m
$i=1, j=0$	m	h	h	m	m
$i=1, j=1$	h	h	h	m	m
$i=1, j=2$	h	h	h	m	m
$i=1, j=3$	h	h	h	m	m
$i=1, j=4$	m	h	h	m	m
$i=1, j=5$	h	h	h	m	m
$i=1, j=6$	h	h	h	m	m
$i=1, j=7$	h	h	h	m	m
$i=2, j=0$	m	h	h	m	m

Επίσης για επαλήθευση μπορούμε να σκεφτούμε εμείς συνολικά κάνουμε $64*5=320$ αιτήματα και όλα είναι είτε hit είτε miss ($148+172=320$)

4) ΥΠΟΕΡΩΤΗΜΑ 3

Εκφώνηση:

Γ) Τι αλλάζει αν η cache είναι write-no-allocate; Υπολογίστε ξανά το συνολικό αριθμό hits και misses για την εκτέλεση του παραπάνω κώδικα.

Απάντηση:

Write-Allocate

Σε περίπτωση **write miss**, το block που περιέχει τα δεδομένα **φορτώνεται στην cache**, και μετά γίνεται η εγγραφή στη cache.

No Write-Allocate

Σε περίπτωση **write miss**, τα δεδομένα **γράφονται απευθείας στη RAM** και δεν φορτώνονται στη cache.

Επομένως τώρα, το μοτίβο το οποίο ακολουθούν τα hits και τα misses φαίνεται και στον παρακάτω πίνακα.

	$a[i+2][j]$	$a[i+1][j]$	$a[i][j]$	$b[i*8+j]$	$a[i][j]$
$i=0, j=0$	m	m	m	m	m
$i=0, j=1$	h	h	m	m	m
$i=0, j=2$	h	h	m	m	m
$i=0, j=3$	h	h	m	m	m
$i=0, j=4$	m	m	m	m	m
$i=0, j=5$	h	h	m	m	m
$i=0, j=6$	h	h	m	m	m
$i=0, j=7$	h	h	m	m	m
$i=1, j=0$	m	h	h	m	m
$i=1, j=1$	h	h	m	m	m
$i=1, j=2$	h	h	m	m	m
$i=1, j=3$	h	h	m	m	m
$i=1, j=4$	m	h	h	m	m
$i=1, j=5$	h	h	m	m	m
$i=1, j=6$	h	h	m	m	m
$i=1, j=7$	h	h	m	m	m
$i=2, j=0$	m	h	h	m	m

Ακολουθούν πίνακες που δείχνουν τα hits και τα misses καθώς και τα περιεχόμενα της CACHE ανά γραμμή για μερικές τιμές των i, j .

i=0, j=0

a[2][0]	comp m	block a4	line 4
a[1][0]	comp m	block a2	line 2
a[0][0]	comp m	block a0	line 0

	0	1	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

b[0]	comp m	blockb0	line 0
------	--------	---------	--------

	0	1	10	11
0	b[0]	b[1]	b[2]	b[3]
1				
10	a[1][0]	a[1][1]	a[1][2]	a[1][3]
11				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

a[0][0]	write m	block a0	line 0
---------	---------	----------	--------

	0	1	10	11
000	b[0]	b[1]	b[2]	b[3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

i=0, j=1

a[2][1]	hit	block a4	line 4
a[1][1]	hit	block a2	line 2
a[0][1]	con m	block a0	line 0

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

b[1]	con m	blockb0	line 0
------	-------	---------	--------

	00	01	10	11
000	b[0]	b[1]	b[2]	b[3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

a[0][1]	write m	block a0	line 0
---------	---------	----------	--------

	00	01	10	11
000	b[0]	b[1]	b[2]	b[3]
001				
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011				
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101				
110				
111				

i=1, j=0

a[3][0]	com m	block a6	line 6
a[2][0]	hit	block a4	line 4
a[1][0]	hit	block a2	line 2

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

b[8]	com m	blockb2	line 2
------	-------	---------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	b[8]	b[9]	b[10]	b[11]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

a[1][0]	write m	block a2	line 2
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	b[8]	b[9]	b[10]	b[11]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

i=1, j=1

a[3][1]	hit	block a6	line 6
a[2][1]	hit	block a4	line 4
a[1][1]	con m	block a2	line 2

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	a[1][0]	a[1][1]	a[1][2]	a[1][3]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

b[9]	con m	blockb2	line 2
------	-------	---------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	b[8]	b[9]	b[10]	b[11]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

a[1][1]	write m	block a2	line 2
---------	---------	----------	--------

	00	01	10	11
000	a[0][0]	a[0][1]	a[0][2]	a[0][3]
001	a[0][4]	a[0][5]	a[0][6]	a[0][7]
010	b[8]	b[9]	b[10]	b[11]
011	a[1][4]	a[1][5]	a[1][6]	a[1][7]
100	a[2][0]	a[2][1]	a[2][2]	a[2][3]
101	a[2][4]	a[2][5]	a[2][6]	a[2][7]
110	a[3][0]	a[3][1]	a[3][2]	a[3][3]
111				

Για $i=0$

Κάθε 4 αλλαγές τιμή του j έχω 2 com misses (στοιχεία a). Το j αλλάξει 8 φορές άρα $2 \cdot 8 / 4 = 4$ misses

+ Για κάθε αλλαγή του j έχω 1 (con ή com στοιχείο a) miss, 1 (con ή com στοιχείο b) miss και 1 write miss. Το j αλλάξει 8 φορές άρα $3 \cdot 8 = 24$ misses

Δηλαδή #misses = 28 misses

Τα υπόλοιπα από τα $5 \cdot 8 = 40$ requests μας είναι hit.

Δηλαδή #hits = $40 - 28 = 12$ hits

Για $i \neq 0$

Από τον πίνακα μετράμε 24 misses και 16 hits

#misses = $28 + 7 \cdot 24 = 196$ και

#hits = $12 + 7 \cdot 16 = 124$

Επίσης για επαλήθευση μπορούμε να σκεφτούμε εμείς συνολικά κάνουμε $64 \cdot 5 = 320$ αιτήματα και όλα είναι είτε hit είτε miss ($196 + 124 = 320$)