

Esercizio cache

Soluzione

Si consideri una gerarchia di memoria con:

- una memoria centrale costituita da 256K parole da 16-bit (unità indirizzabile).
- una cache di 4K parole (da 16 bit), set associativa a 4 vie, con linee da 64 parole (da 16 bit)

Si assuma che:

- il bus di sistema sia di ampiezza uguale ad una parola
- il tempo di accesso alla cache sia 10 volte più breve rispetto al tempo di accesso alla memoria centrale.
- la cache sia inizialmente vuota
- la CPU carichi 4352 parole (da 16 bit) dalle locazioni 0,1,2,3...,4351, in ordine, e che ripeta tale caricamento per altre 14 volte.

a) Specificare il numero di bit per i campi in cui un indirizzo di memoria centrale viene suddiviso.

Soluzione:

Poiché la memoria centrale possiede 256K (cioè 2^{18}) parole, occorrono 18 bit per rappresentare un generico indirizzo. Questi bit vengono suddivisi nei campi tag, set e parola. La dimensione del campo parola è determinato dalla dimensione del blocco, che è di $64 = 2^6$ parole. Quindi il campo parola è costituito da 6 bit (i meno significativi dell'indirizzo). Poiché ogni set della cache contiene 4 linee, considerando che in totale la cache contiene 4K (cioè 2^{12}) parole e quindi $2^{12}/2^6 = 2^6$ linee, si avranno $2^6/2^2 = 2^4$ set. Pertanto il campo set sarà costituito da 4 bit, e per differenza il campo tag sarà costituito da $18-4-6=8$ bit.

b) Assumendo una politica di rimpiazzo LRU per i blocchi, stimare il fattore di velocizzazione (**speedup**: rapporto fra il tempo impiegato senza cache e il tempo impiegato con la cache) risultante con l'uso della cache.

Soluzione:

Struttura della Cache

Set 0	Set 1	Set 2	Set 3	Set 4	Set 5	Set 6	Set 7	Set 8	Set 9	Set10	Set11	Set12	Set13	Set14	Set15
linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0	linea 0
linea	linea	linea	linea	linea	linea	linea	linea	linea	linea						

Set 0	Set 1	Set 2	Set 3	Set 4	Set 5	Set 6	Set 7	Set 8	Set 9	Set 10	Set 11	Set 12	Set 13	Set 14	Set 15
48/32	49/33	50/34	51/35	4	5	6	7	8	9	10	11	12	13	14	15
64/48	65/49	66/50	67/51	20	21	22	23	24	25	26	27	28	29	30	31
16/0/64	17/1/65	18/2/66	19/3/67	36	37	38	39	40	41	42	43	44	45	46	47
32/16	33/17	34/18	35/19	52	53	54	55	56	57	58	59	60	61	62	63

Inizialmente i blocchi 0..3 sono caricati nella linea 2 di ogni set e di seguito si hanno miss per tutte le linee dei Set 0..2, come nel caso precedente:

$$N_{miss_3} = 4 + 4 + 4 + 4 + 4 = 20$$

Le iterazioni successive (da 4 a 15) si comporteranno similmente, portando il numero totale di miss a:

$$N_{miss} = 68 + 20 \cdot 14 = 348$$

$$\begin{aligned} \text{Tempo totale con cache} &= 4352 \cdot 15 \cdot t + 348 \cdot (64 \cdot 10 \cdot t) \\ &= 65280 \cdot t + 222720 \cdot t \\ &= 288000 \cdot t \end{aligned}$$

$$\text{Speedup} = \frac{\text{Tempo totale senza cache}}{\text{Tempo totale con cache}} = \frac{652800 \cdot t}{288000 \cdot t} = 2.26$$

c) Risolvere il punto (b) assumendo per il rimpiazzo dei blocchi la politica del blocco più recentemente riferito (MRU).

Soluzione:

La formula generale data al punto b) rimane valida, mentre cambia il numero di miss totale dovuto alla diversa politica di rimpiazzo dei blocchi.

Prima Iterazione (MRU)

Stato della cache prima della iterazione Stato della cache dopo l' iterazione

Set 0	Set 1	Set 2	Set 3	...
0	1	2	3	...
16	17	18	19	...
32	33	34	35	...
48/64	49/65	50/66	51/67	...

64 miss iniziali e 4 rimpiazz i secondo la politica MRU (le linee che contengono i blocchi più recentemente riferite sono le linee 3 di ogni set).
(inizialmente la cache è vuota)

$$N_{miss_1} = 64 + 4 = 68$$

Seconda Iterazione (MRU)

Stato della cache **prima** della iterazione

Set 0	Set 1	Set 2	Set 3	. . .
0	1	2	3	...
16	17	18	19	...
32	33	34	35	...
64	65	66	67	...

Riferimenti ai blocchi da 0 a 47 sono hit successivi, ma i blocchi da 48 a 51 sono miss. I blocchi più recentemente usati si trovano nelle linee 2 di ogni set.

Stato della cache **dopo** l' iterazione

Set 0	Set 1	Set 2	Set 3	...
0	1	2	3	...
16	17	18	19	...
32/48	33/49	34/50	35/51	...
64	65	66	67	...

Quindi i blocchi 48..51 inducono miss e sono posti nelle linee 2 di ogni set. Come risultato abbiamo: $N_{miss_2} = 4$

Terza Iterazione (MRU)

Stato della cache **prima** della iterazione

Set 0	Set 1	Set 2	Set 3	...
0	1	2	3	...
16	17	18	19	...
48	49	50	51	...
64	65	66	67	...

I blocchi più recentemente usati si trovano nelle linee 1 di ogni set.

Stato della cache **dopo** l' iterazione

Set 0	Set 1	Set 2	Set 3	...
0	1	2	3	...
16/32	17/33	18/34	19/35	...
48	49	50	51	...
64	65	66	67	...

Quindi i blocchi 32..35 inducono miss e sono posti nelle linee 1 di ogni set. Pertanto: $N_{miss_3} = 4$

Quarta Iterazione (MRU)

Stato della cache **prima** della iterazione

Set 0	Set 1	Set 2	Set 3	...
0	1	2	3	...
32	33	34	35	...
48	49	50	51	...
64	65	66	67	...

I blocchi più recentemente usati si trovano nelle linee 0 di ogni set.

Stato della cache **dopo** l' iterazione

Set 0	Set 1	Set 2	Set 3	...
0/16	1/17	2/18	3/19	...
32	33	34	35	...
48	49	50	51	...
64	65	66	67	...

Quindi i blocchi 16..19 inducono miss e sono posti nelle linee 0 in ogni set. Pertanto: $N_{miss_4} = 4$

Quinta Iterazione (MRU)

Stato della cache **prima** della iterazione

Set 0	Set 1	Set 2	Set 3	...
16	17	18	19	...
32	33	34	35	...

Stato della cache **dopo** l' iterazione

Set 0	Set 1	Set 2	Set 3	...
16	17	18	19	...
32	33	34	35	...

48	49	50	51	...	48/64	49/65	50/66	51/67	...
64	65	66	67	...	64/0	65/1	66/2	67/3	...

I blocchi più recentemente usati si trovano nelle linee 3 di ogni set.

I miss occorrono all'inizio e alla fine della iterazione raddoppiando il numero di miss: $N_{miss_5} = 8$

Le iterazioni dalla 6 alla 15 si comportano similmente a quelle appena viste.

Iterazione 1 \rightarrow numero miss = 68

Iterazioni 2,3,4,6,7,8,10,11,12,14,15 \rightarrow numero miss = 4

Iterazioni 5,9,13 \rightarrow numero miss = 8

Quindi in totale abbiamo:

$$N_{miss} = 68 + 11 \cdot 4 + 3 \cdot 8 = 136$$

$$\begin{aligned} \text{Tempo totale con cache} &= 4352 \cdot 15 \cdot t + 136 \cdot (64 \cdot 10 \cdot t) \\ &= 65280 \cdot t + 87040 \cdot t \\ &= 152320 \cdot t \end{aligned}$$

$$\text{Speedup} = \frac{\text{Tempo totale senza cache}}{\text{Tempo totale con cache}} = \frac{652800 \cdot t}{152320 \cdot t} = 4.28$$