

2.6. La memòria d'un ordinador és adreçable per bytes, i la longitud de la paraula és de 32 bits. Un programa consisteix en dos bucles nidats (nested-loops): un petit bucle interior i un bucle exterior molt més gran.

L'estructura general del programa es mostra a la figura 1. Les adreces de memòria decimal mostren la localització dels dos bucles i el principi i final del programa total. Totes les ubicacions de memòria a les diverses seccions del programa, 8-52, 56-136, 140-240, etc., contenen instruccions per executar-les en seqüències. El programa s'executarà en una computadora que tingui una memòria cau d'instruccions organitzada amb mapejat directe amb els següents paràmetres:

- Mida de la memòria cau 1KiB
- Mida del bloc de 128 bytes

El "miss penalty" en el caché d'instruccions és 80τ , on τ és el temps d'accés de la memòria cau. Calculeu el temps total necessari per a obtenir instruccions durant l'execució del programa a la Figura 1. Supposeu que, després de portar un bloc de MP, triguem τ en portar la dada de la caché.

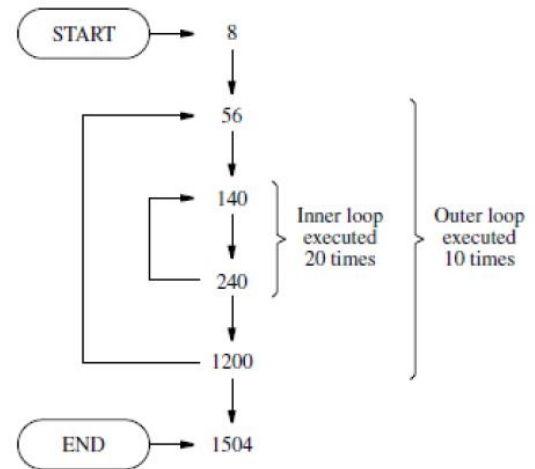


Figura 1.

-> 0x38(56)	00	000	0111000-----	
-> 0x8C(140)	00	001	0001100-----	
-> 0xF0(240)	00	001	1110000-----	
-> 0x4B0(1200)	01	001	0110000-----	

8 → 56 → 140 → 240 → 1200 → 1504

$$\text{mida cache} = 1\text{kB} = 2^{10} \text{ Bytes}$$

$$\text{mida paraula (B)} = 32 \text{ bits} = 4 \text{ Bytes} = 2^2$$

$$\text{mida bloc} = 128 \text{ Bytes} = 2^7 \text{ Bytes}$$

$$\text{Miss Penalty caché instruccions} = 80\tau$$

$$\tau = \text{temps accés a caché}$$

Temps total necessari per a obtenir instruccions durant l'execució del programa?

$$\text{Total instruccions} = 1504 - 8 = 1496 \text{ instruccions}$$

$$\text{num blocs (C)} = \frac{\text{mida cache}}{\text{mida bloc}} = \frac{2^{10}}{2^7} = 2^3 \text{ blocs/línies}$$

$$\text{mida bloc} = 2^{W+B} \rightarrow 2^7 = 2^{W+2} \rightarrow W = 5$$

$$\text{en MP: } K = 1; S = C$$

$$S = C = 2^3 \rightarrow i = 3$$

**** TABLA HIT/MISS ****

Total misses = 48

Miss Penalty (M) = 80τ

$$T_{ac} = \tau$$

$$\text{Hit Rate (h)} = 1 - \text{Miss Rate} = 1 - \frac{48}{1496} = 1 - 0.032 = 0.968$$

$$t_{misses} = 48 \times 80\tau$$

$$t_{START} = (56 - 8) \times \tau = 48 \cdot \tau$$

$$t_{INNER LOOP} = (244 - 140) \times 20 \times 10 \times \tau = 20800 \cdot \tau$$

$$t_{OUTTER LOOP} = ((1204 - 56) - (244 - 140)) \times 10 \times \tau = 10440 \cdot \tau$$

$$t_{END} = (1508 - 1204) \times \tau = 304 \cdot \tau$$

$$t_{total} = t_{START} + t_{INNER LOOP} + t_{OUTTER LOOP} + t_{END} + t_{misses}$$

$$t_{total} = 48 \cdot \tau + 20800 \cdot \tau + 10440 \cdot \tau + 304 \cdot \tau + 48 \times 80\tau = 35432 \cdot \tau$$

MP

TAG	i	W	B
?	3	5	2

t_{med} : Temps Mig d'accés a la informació

T_{ac} : Temps d'Accés a la caché

h : Hit Rate

M : Miss Penalty

**** TABLA HIT/MISS ****

Byte Adress	Binary Adress	TAG	Index	Offset	Hit/Miss
8	... 0000 0000 1000	... 00	00 0	000 1000	Miss
...					<i>*hits*</i>
→ 56	... 0000 0011 1000	... 00	00 0	011 1000	Hit
...					<i>*hits*</i>
128	... 0000 1000 0000	.. 00	00 1	000 0000	Miss
...					<i>*hits*</i>
140	... 0000 1000 1100	... 00	00 1	000 1100	Hit
...					<i>*hits*</i>
240	... 0000 1111 0000	... 00	00 1	111 000	Hit
...					<i>*hits*</i>
256	... 0001 0000 0000	... 00	01 0	000 0000	Miss
...					<i>*hits*</i>
384	... 0001 1000 0000	... 00	01 1	000 0000	Miss
...					<i>*hits*</i>
512	... 0010 0000 0000	... 00	10 0	000 0000	Miss
...					<i>*hits*</i>
640	... 0010 1000 0000	... 00	10 1	000 0000	Miss
...					<i>*hits*</i>
768	... 0011 0000 0000	... 00	11 0	000 0000	Miss
...					<i>*hits*</i>
896	... 0011 1000 0000	... 00	11 1	000 0000	Miss
...					<i>*hits*</i>
1024	... 0100 0000 0000	... 01	00 0	000 0000	Miss
...					<i>*hits*</i>
1200	... 0100 1011 0000	... 01	00 1	011 0000	Miss
→ inici loop:					
56	... 0000 0011 1000	... 00	00 0	011 1000	Miss
...					<i>*hits*</i>
128	... 0000 1000 0000	... 00	00 1	000 0000	Miss
...					<i>*hits*</i>
256	... 0001 0000 0000	... 00	01 0	000 0000	Hit
...					<i>*hits*</i>
384	... 0001 1000 0000	... 00	01 1	000 0000	Hit
...					<i>*hits*</i>
512	... 0010 0000 0000	... 00	10 0	000 0000	Hit
...					<i>*hits*</i>
640	... 0010 1000 0000	... 00	10 1	000 0000	Hit
...					<i>*hits*</i>
768	... 0011 0000 0000	... 00	11 0	000 0000	Hit
...					<i>*hits*</i>
896	... 0011 1000 0000	... 00	11 1	000 0000	Hit
...					<i>*hits*</i>
1024	... 0100 0000 0000	... 01	00 0	000 0000	Miss
...					<i>*hits*</i>
1200	... 0100 1011 0000	... 01	00 1	011 0000	Miss
...					<i>*hits*</i>
1280	... 0101 0000 0000	... 01	01 0	000 0000	Miss
...					<i>*hits*</i>
1408	... 0101 1000 0000	... 01	01 1	000 0000	Miss
...					<i>*hits*</i>
1504	... 0101 1110 0000	... 01	01 1	110 0000	Hit

20 times
(0 miss)

9 times
(4 x 9 miss)

Total misses = 10 + (4 × 9) + 2 = **48 misses**