

CACHE E ASSOCIATIVITA'

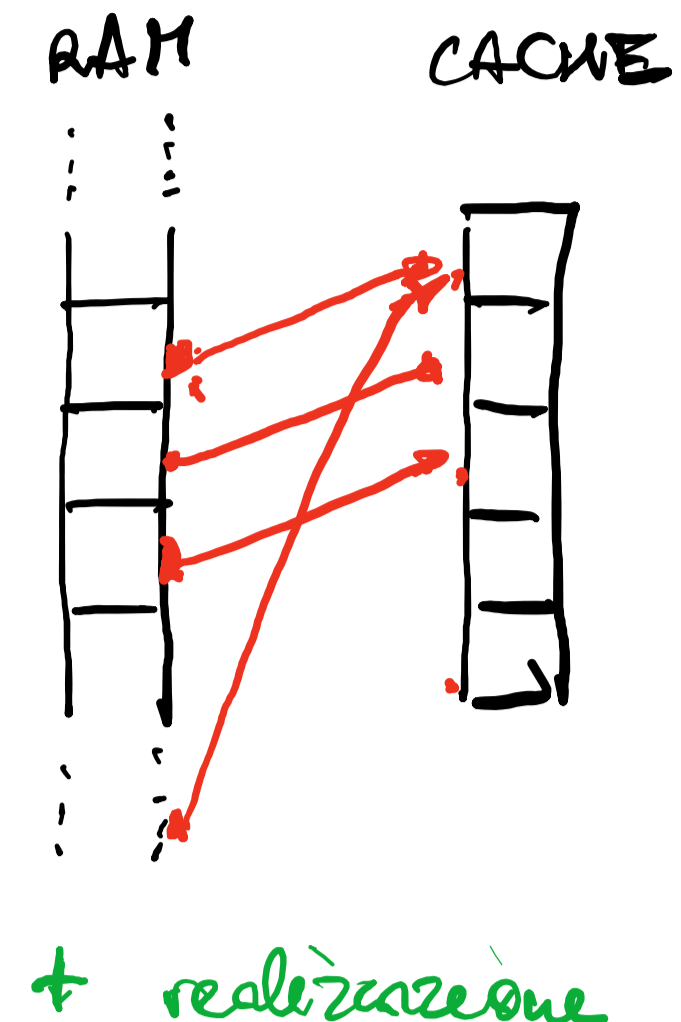
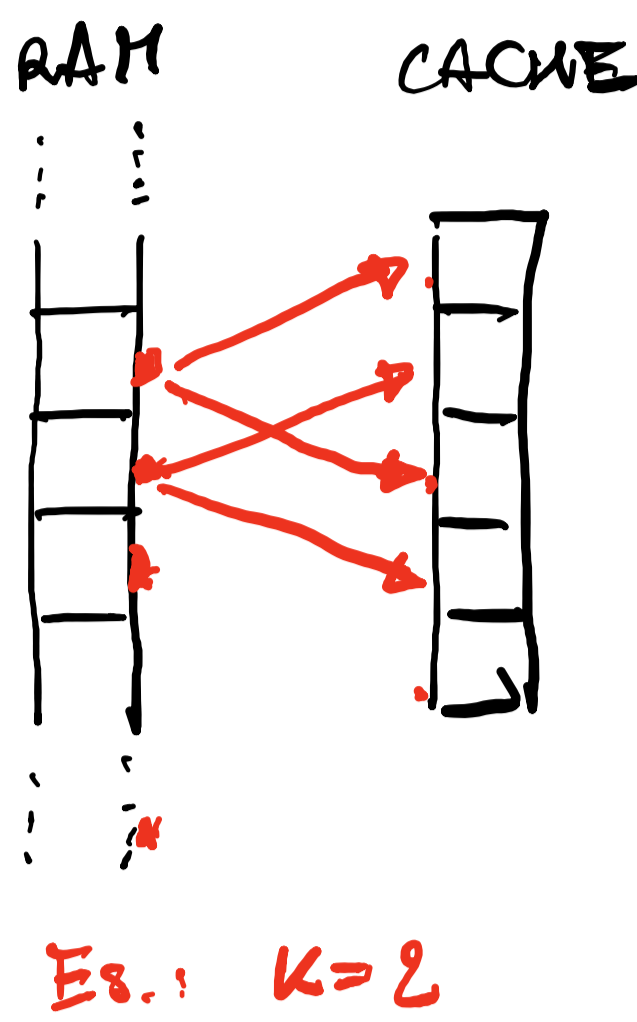
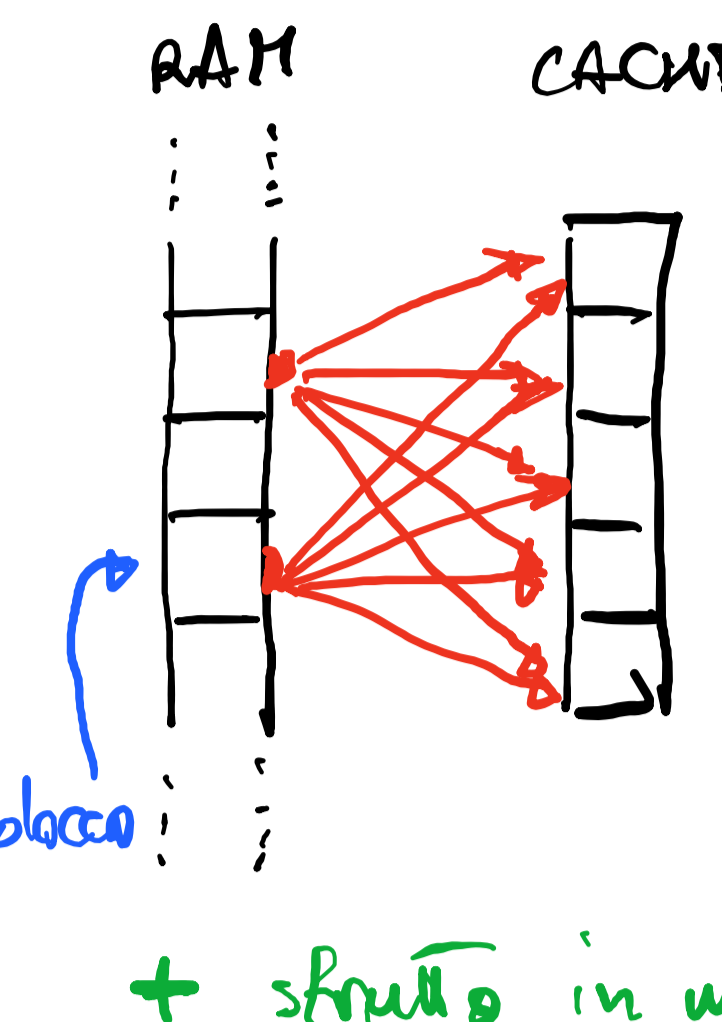
Esempio di funzionamento di una cache con 4 linee da 64 byte ciascuna.

Indirizzo acceduto	numero blocco	Contenuto della cache	
20	0	0(0)	COLD CACHE MISS
464	7	0(2) 7(0)	" " "
128	2	0(2) 7(1) 2(0)	" " "
36	0	0(0) 7(2) 2(2)	CACHE HIT
608	9	0(2) 7(3) 2(2) 9(0)	COLD CACHE MISS
360	5	0(2) 5(0) 2(3) 9(2)	CAPACITY CACHE MISS
204	3	0(3) 5(1) 3(0) 9(2)	CAPACITY CACHE MISS

CACHE COMPLETAMENTE ASSOCIATIVA

CACHE ASSOCIATIVE A-K-VIE

CACHE MAPPING DIRETTO



+ sfruttato in modo efficiente lo spazio in cache
- realizzazione completa

Es.: k=2

+ realizzazione semplice
- meno efficiente

A seconda del tipo di cache adottato abbiamo diversi risultati per ogni accesso in memoria:

- CACHE HIT**: il dato è presente in cache
- COLD CACHE MISS**: il dato non è presente in cache e questa è (parzialmente vuota)
- CAPACITY CACHE MISS**: il dato non è presente in cache e la cache è piena => necessaria selezione di una vittima
- CONFLICT CACHE MISS**: il dato non è presente in cache e confligge con altri dati presenti nelle k-vie

Es.: 25/7/2019 domanda 2

k=2 linee=4 dim=32 byte

ADDR	BLOCK	CACHE 0	CACHE 1	CACHE 2	CACHE 3	
105	3	-	-	3(0)	-	COLD
782	24	24(0)	-	3(1)	-	"
386	12	24(1)	12(0)	3(2)	-	"
1356	42	42(0)	12(1)	3(3)	-	CONFLICT
11	0	42(1)	0(0)	3(4)	-	"
324	10	10(0)	0(1)	3(5)	-	"
265	8	10(1)	8(0)	3(6)	-	"

risultato: 10, 8, 3, -

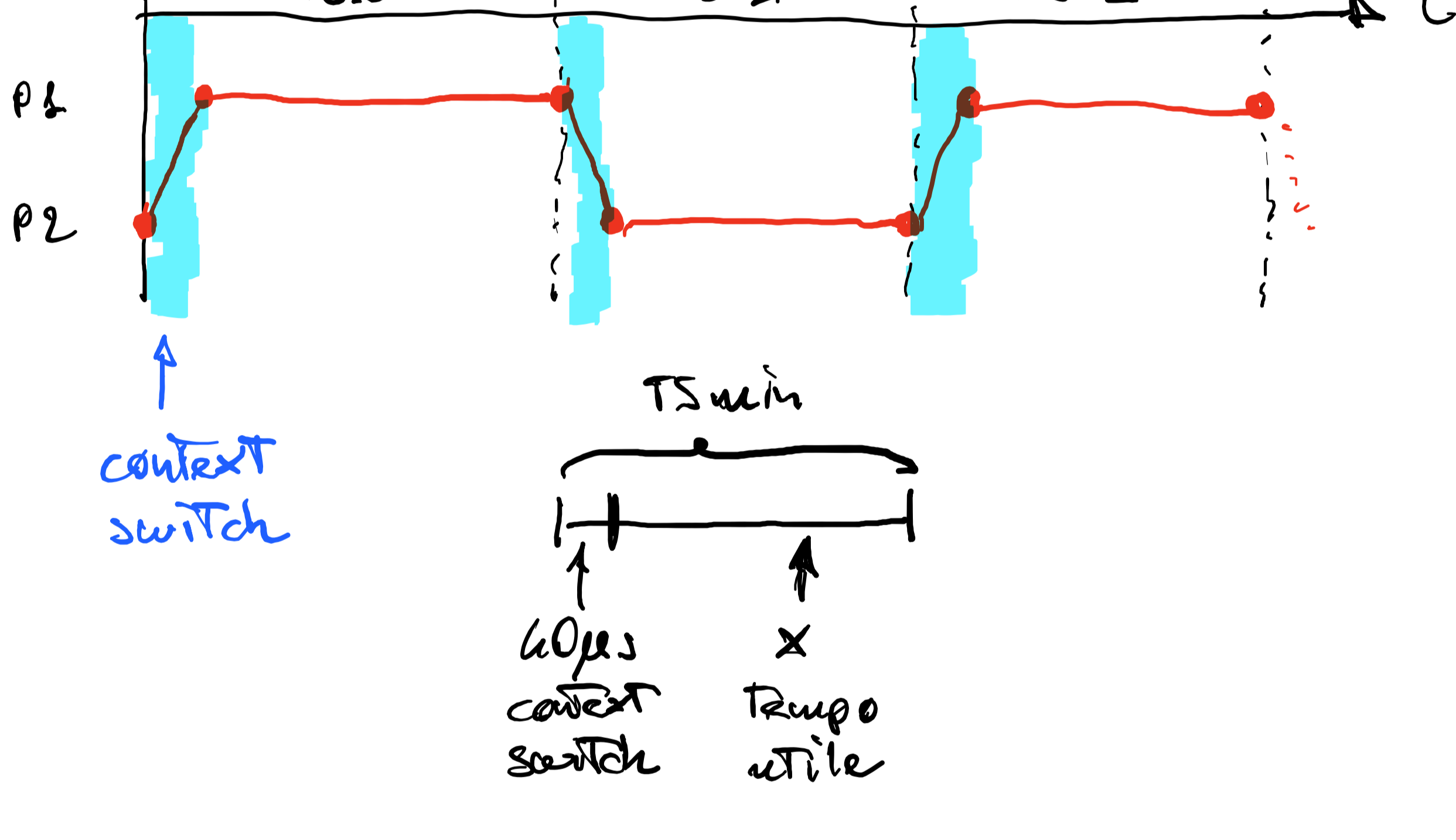
Es. 25/7/2019 domanda 1

A -> k = 3,5x P -> s = 1,4x S = $\frac{1}{\frac{\alpha}{k} + 1 - \alpha}$

$1,4 = \frac{1}{\frac{\alpha}{3,5} + 1 - \alpha} \Rightarrow \frac{3,5}{\alpha + 3,5 - 3,5\alpha} = \frac{3,5}{3,5 - 2,5\alpha}$

$2,5\alpha = 3,5 - \frac{3,5}{1,4} \approx 0,4 \rightarrow \alpha = 40\%$

Es. 25/7/2019 domanda 3



$40 = 0,02 * TS_{min} = 0,02 * (40 + x)$
 $x = \frac{40}{0,02} - 40 = 1,96 \text{ msec}$

Es. 25/7/2019 domanda 4

spazio indirizzabile dai processi -> 2^{20}
 Tabella delle pagine -> 2KB = 2048 byte = 2^{11}
 ogni entry -> 32bit = 4 byte = 2^2
 numero righe nella Tabella = $\frac{\text{dim. Tabella}}{\text{dim. entry}} = \frac{2^{11}}{2^2} = 2^9$
 dimensione pagina = $\frac{\text{dim. RAM}}{\text{num. pagine}} = \frac{2^{20}}{2^9} = 2^{11} = 2KB$

ATTENZIONE 2KB \neq 2048 byte

POTENZE DI 10			POTENZE DI 2		
10^3 byte	kilobyte	KB	2^{10} byte	kilobyte	kiB
10^6	megabyte	MB	2^{20}	Milobyte	MiB
10^9	giga...	GB	2^{30}	Gilobyte	GiB
10^{12}	Tera	TB
10^{15}	petra	PB			
10^{18}	exa	EB			
10^{21}	zetta	ZB			
10^{24}	yotta	YB			

Es. 16/9/2023 domanda 4

```
typedef struct S {
    char x;
    char *y;
    short z;
} S;
```

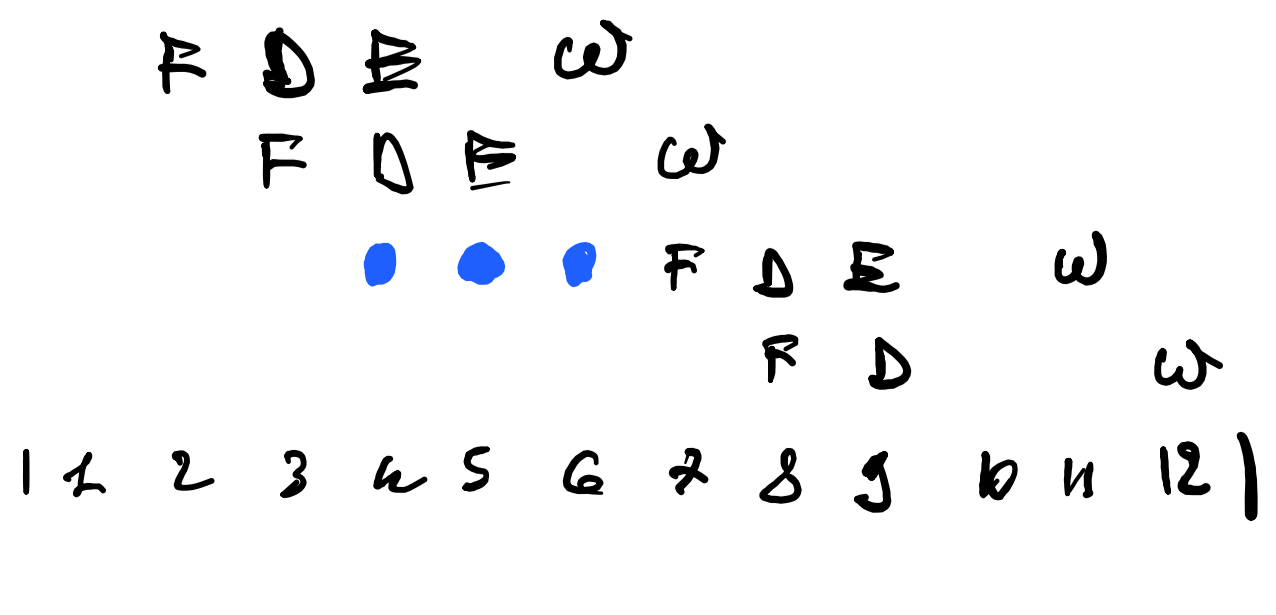
sizeof(S) = 12 byte di padding = 5
 ordine di ottimizzazione dimensione

```
typedef struct S {
    char *y;
    short z;
    char x;
} S;
```

sizeof(S) = 8 byte di padding = 1

Es. 16/9/2023 domanda 2

```
movl $5, %eax
addl $10, %edx
subl $10, %ecx
incl %edx
movl $3, %edi
```



```
movl $5, %eax
addl %edi, %edx
subl $10, %ecx
movl $3, %edi
incl %edx
```

