

## Lezione 9-bis

## ESERCIZI SULLA MEMORIA VIRTUALE

## Algoritmi di avvicinamento

- FIFO (viene sostituita la pagina in memoria presente da piu' tempo)
- LRU (viene sostituita la pagina che non e' usata da piu' tempo)
- LFU (viene sostituita la pagina meno frequentemente utilizzata)
- Seconda chance/CLOCK (varianti di FIFO)
- Working set / PFF numero di frame variabile



Per tutti obiettivo comune:  
Minimizzare il numero di page faults

## Esercizio 1

- Con riferimento ad un ambiente di gestione della **memoria virtuale** con paginazione su richiesta, si consideri un processo caratterizzato dalla seguente **stringa di riferimenti**

1 0 3 5 6 9 1 19 15 18 9 15 1 3 5 1 9 19 3

- Si illustri il comportamento degli algoritmi FIFO e LRU nel caso siano assegnati al processo 5 frame. Si calcoli il numero di page faults

## Soluzione FIFO

1 0 3 5 6 9 1 19 15 18 9 15 1 3 5 1 9 19 3

1	1	1	1	1	9	9	9	9	9
	0	0	0	0	0	1	1	1	1
		3	3	3	3	3	19	19	19
			5	5	5	5	5	15	15
				6	6	6	6	6	18

ok   ok   ok

3	3	3	3	3
1	5	5	5	5
19	19	1	1	1
15	15	15	9	9
18	18	18	18	19

ok

15 page faults



### Esercizio 3

- Nei due casi precedenti:
  - FIFO = 5 page fault
  - LRU = 4 page fault

qual'è il tempo di accesso effettivo della paginazione su richiesta se il tempo medio di servizio è

- 25 millisecondi ( $25 \times 10^{-3}$  sec) in caso di page fault
- 100 microsecondi ( $100 \times 10^{-6}$  sec) in caso di pagina presente in memoria?

### Soluzione

- Completivamente l'istruzione  $A[i*j] = A[2*i] + C[2*j]$  viene eseguita **512 volte**
- Ogni volta vengono fatti **3 accessi**
- Totale di **1536 riferimenti in memoria**.
- $EAT = (1 - p) \times t[\text{accesso alla memoria}] + p \times t[\text{page fault}]$

FIFO  $\rightarrow p = 5/1536 = 0.0032$

$$EAT = 0.9968 \times 100 \times 10^{-6} + 0.0032 \times 25000 \times 10^{-6} = 179.68 \times 10^{-6} \text{ sec}$$

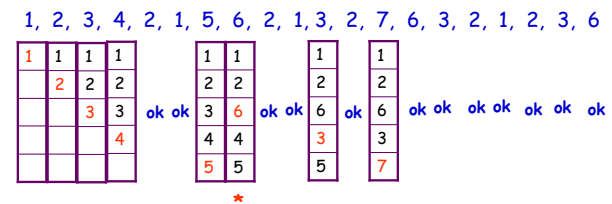
LRU  $\rightarrow p = 4/1536 = 0.0026$

$$EAT = 0.9974 \times 100 \times 10^{-6} + 0.0026 \times 25000 \times 10^{-6} = 164.74 \times 10^{-6} \text{ sec}$$

### Esercizio 4

- Considerate la seguente successione di riferimenti di pagine:  
1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 3, 2, 7, 6, 3, 2, 1, 2, 3, 6.
- Determinare il tempo di accesso effettivo della paginazione su richiesta per LRU con 5 blocchi, se:
  - il tempo medio di servizio di un **page fault senza salvataggio** della pagina avvicinata è di **80 millisecondi** ( $80 \times 10^{-3}$  sec),
  - il tempo medio di servizio di un **page fault con salvataggio** della pagina avvicinata è di **140 millisecondi** ( $140 \times 10^{-3}$  sec)
  - il tempo di **accesso alla memoria** è di **80 microsecondi** ( $80 \times 10^{-6}$  sec),
- nell'ipotesi che l'accesso alle pagine 1, 2, 3 sia sempre in scrittura.

### soluzione



\* Quando la pagina 6 prende il posto della pagina 3, quest'ultima deve essere salvata perché l'accesso precedente era in scrittura

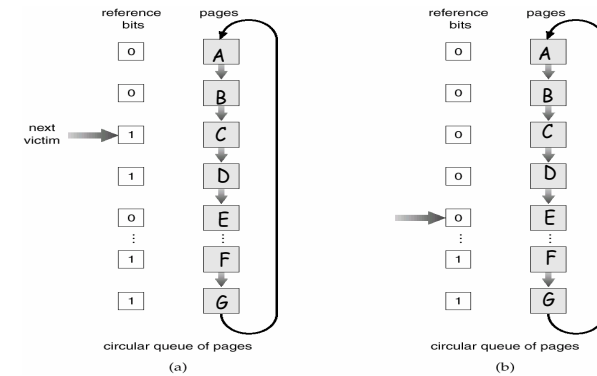
- 1 page fault con salvataggio della pagina sostituita
- 7 page fault senza salvataggio della pagina sostituita
- 12 accessi diretti senza page fault

$$EAT = (12/20 \times 80 + 7/20 \times 80000 + 1/20 \times 140000) \times 10^{-6} \text{ sec} = 35.048 \text{ msec}$$

## Clock algorithm

- Implementazione mediante un **bit di accesso**. (bit=1 se referenziata)
- Si scorrono le pagine presenti nella lista:
- Se **bit=1**:
  - Si pone il bit di riferimento a 0.
  - Si lascia la pagina in memoria.
  - Si rimpiazza la pagina successiva (in ordine di clock), in base alle stesse regole.
- Se **bit=0**
  - Si rimpiazza la pagina

## Clock algorithm



- pagina **C** **candidata** ad essere eliminata
- bit=1** → **C** viene salvata e si esamina la pagina **D**
- bit=1** → **D** viene salvata e si esamina la pagina **E** che viene eliminata

## Esercizio 5

- Data la seguente sequenza di **referimenti di pagine**  
**1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 3, 2, 7, 6, 3, 2, 1, 2, 3, 4**

Determinare **quanti page fault** avvengono utilizzando il **clock algorithm con 5 frame** e lancetta dell'orologio posizionata sul primo elemento

## soluzione

1	2	3	4	2	1	5	6	2	1	3	2	7	6	3	2	1	2	3	4
1	1	1	1	1	1*	1*	1	1	1*	1*	1*	1*	1*	1*	1*	1*	1*	1*	1
	2	2	2	2*	2*	2*	2	2*	2*	2*	2*	2*	2*	2*	2*	2*	2*	2*	2
		3	3	3	3	3	6	6	6	6	6	6	6*	6*	6*	6*	6*	6*	6
			4	4	4	4	4	4	4	3	3	3	3	3*	3*	3*	3*	3*	3
						5	5	5	5	5	5	7	7	7	7	7	7	7	4
					ok	ok			ok	ok		ok	ok	ok	ok	ok	ok	ok	
(0)							(1)			(2)		(3)							(4)

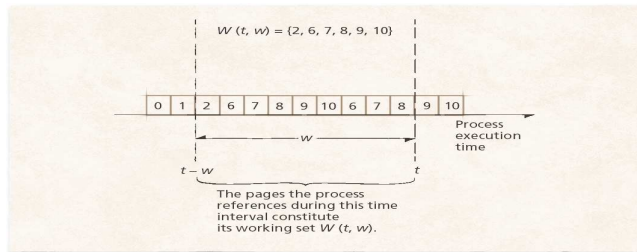
**9 P.F. !!**

- (0) all'inizio clock = 1
- (1) pag 1 e 2 bit=1 → si portano bit=0 e si avvicina pagina 3 → clock = 4
- (2) clock = 5
- (3) clock = 1
- (4) pag 1, 2, 6, 3 bit=1 → si portano bit=0 e si avvicina pagina 7 → clock = 1

## Working set

Il modello basato su working set cerca di favorire i processi che realizzano il principio di localita' temporale

Il **working set** di un processo  $W(t, w)$ , e' l'insieme delle pagine referenziate dal processo nell'intervallo di tempo  $[t-w, t]$



9-bis. Esercizi su memoria virtuale

17

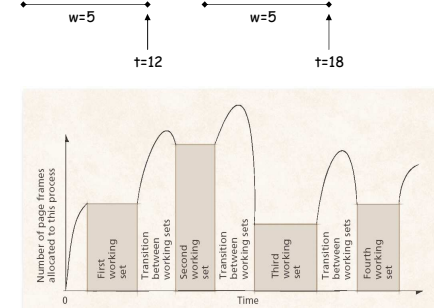
marco lapegna

## Esercizio 6 / soluzione

Si studi l'andamento dell'algoritmo basato su **working set**  $W(t, w)$  con  $w=5$  per la seguente stringa di riferimento alle pagine di memoria e si determini la **minima e la massima dimensione** del w.s.

1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 4, 5, 6, 4, 5, 6, 4, 5, 6

$W(1,5) = \{1\}$   $d=1$   
 $W(2,5) = \{1,2\}$   $d=2$   
 $W(3,5) = \{1,2,3\}$   $d=3$   
 $W(4,5) = \{1,2,3\}$   
 $W(5,5) = \{1,2,3\}$   
 ....  
 $W(12,5) = \{1,2,3\}$   $d=3$   
 $W(13,5) = \{1,2,3,4\}$   $d=4$   
 $W(14,5) = \{1,2,3,4,5\}$   $d=5$   
 $W(15,5) = \{2,3,4,5,6\}$   $d=5$   
 $W(16,5) = \{3,4,5,6\}$   $d=4$   
 $W(17,5) = \{4,5,6\}$   $d=3$   
 $W(18,5) = \{4,5,6\}$   $d=3$   
 ....  
 $W(21,5) = \{4,5,6\}$   $d=3$



9-bis. Esercizi su memoria virtuale

18

marco lapegna

## Esercizio 7

Determinare il numero di page faults per l'algoritmo del working set con  $w=5$  per le seguenti stringhe di riferimenti in memoria

S1 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 4, 5, 6, 4, 5, 6, 4, 5, 6

S2 1, 4, 1, 3, 2, 6, 1, 2, 3, 5, 4, 2, 3, 2, 4, 6, 5, 6, 1, 5, 3

In entrambi i casi:

4 riferimenti alle pagine 1, 2 e 3

3 riferimenti alle pagine 4, 5 e 6

9-bis. Esercizi su memoria virtuale

19

marco lapegna

## Soluzione

Stringa S1

$W(1,5) = \{1\}$   
 $W(2,5) = \{1,2\}$   
 $W(3,5) = \{1,2,3\}$   
 $W(4,5) = \{1,2,3\}$   
 $W(5,5) = \{1,2,3\}$   
 ....  
 $W(12,5) = \{1,2,3\}$   
 $W(13,5) = \{1,2,3,4\}$   
 $W(14,5) = \{1,2,3,4,5\}$   
 $W(15,5) = \{2,3,4,5,6\}$   
 $W(16,5) = \{2,3,4,5,6\}$   
 $W(17,5) = \{3,4,5,6\}$   
 $W(18,5) = \{4,5,6\}$   
 ....  
 $W(21,5) = \{4,5,6\}$

6 page faults

Stringa S2

$W(1,5) = \{1\}$   
 $W(2,5) = \{1,4\}$   
 $W(3,5) = \{1,4\}$   
 $W(4,5) = \{1,4,3\}$   
 $W(5,5) = \{1,4,3,2\}$   
 $W(6,5) = \{4,1,3,2,6\}$   
 $W(7,5) = \{1,3,2,6\}$   
 $W(8,5) = \{3,2,6,1\}$   
 $W(9,5) = \{2,6,1,3\}$   
 $W(10,5) = \{6,1,2,3,5\}$   
 $W(11,5) = \{1,2,3,5,4\}$   
 $W(12,5) = \{2,3,5,4\}$   
 $W(13,5) = \{3,5,4,2\}$   
 $W(14,5) = \{5,4,2,3\}$   
 $W(15,5) = \{4,2,3\}$   
 $W(16,5) = \{2,3,4,6\}$   
 $W(17,5) = \{3,2,4,6,5\}$   
 $W(18,5) = \{2,4,6,5\}$   
 $W(19,5) = \{4,6,5,1\}$   
 $W(20,5) = \{6,5,1\}$   
 $W(21,5) = \{5,6,1,3\}$

11 page faults

9-bis. Esercizi su memoria virtuale

20

marco lapegna

## Esercizio 8

- Supponiamo che il sistema di paginazione utilizzato dal sistema operativo assegni **3 frame** (blocchi di memoria) **da 512B** a ciascun processo e che l'algoritmo di sostituzione delle pagine sia **LRU**.
- Prendiamo in considerazione il seguente programma:
 

```
...
#define N 512
int a[N];
int i;
...
for (i=0; i < N/2; i++)
  a[i] = a[2*i] + a[N-i-1];
...
```
- Si risponda ai seguenti quesiti:
  - se la dimensione di un **intero è 4B**, qual è il **numero di page faults**?
  - In tal caso, se il tempo medio di servizio di un page fault è di 25 millisecondi ( $25 \cdot 10^{-3}$  sec) ed il tempo di accesso alla memoria di 100 microsecondi ( $100 \cdot 10^{-6}$  sec), qual è il **tempo di accesso medio** della paginazione su richiesta?

9-bis. Esercizi su memoria virtuale

21

marco lapegna

## soluzione

Le pagine sono da 512B e ogni intero è 4B (128 elementi a pagina):

0	1	2	3
---	---	---	---

a[0] ... a[127]   a[128] ... a[255]   a[256] ... a[383]   a[384] ... a[511]

for (i=0; i < N/2; i++) a[i] = a[2\*i] + a[N-i-1];

i = 0, ... 63	a[2*i] → 0 a[N-i-1] → 3 a[i] → 0	} 64 volte
i = 64, ... 127	a[2*i] → 1 a[N-i-1] → 3 a[i] → 0	
i = 128, ... 191	a[2*i] → 2 a[N-i-1] → 2 a[i] → 1	} 64 volte
i = 192, ... 255	a[2*i] → 3 a[N-i-1] → 2 a[i] → 1	

Totale  
riferimenti  
= 768

9-bis. Esercizi su memoria virtuale

22

marco lapegna

## soluzione

0	3	0	...	...	1	3	0	...	...	2	2	1	...	...	3	2	1	...	...
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	3	3	3
	3	3	3	3	3	3	3	3	3	3	3	1	1	1	1	1	1	1	1
						1	1	1	1	1	2	2	2	2	2	2	2	2	2

ok ok ok      ok ok ok ok      ok      ok ok      ok ok ok ok

- 6 page fault
- 762 riferimenti validi

$$EAT = (6/768 * 25 \cdot 10^{-3} + 762/768 * 100 \cdot 10^{-6}) \text{ sec} = 0.294 \cdot 10^{-3}$$

9-bis. Esercizi su memoria virtuale

23

marco lapegna

## Esercizio 9

- In un s.o. con paginazione su richiesta occorrono:
  - 8 msec in caso di **p.f. senza salvataggio** della pagina avvicendata
  - 20 msec in caso di **p.f. con salvataggio** della pagina avvicendata
  - 100 nsec in caso di **pagina presente** in memoria
- Supponendo che il **70% delle volte è necessario salvare** la pagina avvicendata, determinare il massimo valore del **p.f. rate p** per ottenere un **EAT al più di 200 nsec**

9-bis. Esercizi su memoria virtuale

24

marco lapegna

## soluzione

$$EAT = p T_{pf} + (1-p) T_{am} < 0.2 \cdot 10^{-6}$$



$$\begin{aligned} EAT &= p (0.7 \cdot 20 + 0.3 \cdot 8) \cdot 10^{-3} + (1-p) \cdot 0.1 \cdot 10^{-6} = \\ &= 1000p (1.4 + 2.4) \cdot 10^{-6} + (1-p) \cdot 0.1 \cdot 10^{-6} = \\ &= 3800p \cdot 10^{-6} + 0.1 \cdot 10^{-6} - 0.1 p \cdot 10^{-6} < 0.2 \cdot 10^{-6} \end{aligned}$$



$$3799.9p < 0.1$$



$$p < 0.1/3799.9 = 0.000026 = 2.6 \cdot 10^{-5}$$

(26 p.f. ogni 10000 riferimenti)