



# Sistemi Operativi

## 7 luglio 2011

### Compito A

3. Si consideri un sistema con memoria paginata a un livello, la cui page table sia mantenuta in memoria principale. Il tempo di accesso alla memoria principale sia  $t = 30ns$ .
- Qual è il tempo effettivo di accesso alla memoria?
  - Aggiungendo un TLB, con tempo di accesso  $\epsilon = 1ns$ , quale hit rate dobbiamo avere per un degrado delle prestazioni del 20% rispetto a  $t$ ?
  - E con una paginazione a due livelli?

**Risposta:**

- (2 punti) Il tempo effettivo di accesso alla memoria è  $2t$ , ovvero, 60 ns; infatti sono necessari 30 ns per accedere alla page table e 30 ns per accedere alla locazione nel frame fisico in memoria.
- (3 punti) Un degrado del 20% rispetto a  $t$  significa un EAT pari a  $1,2 \cdot t$ , ovvero, 36 ns. Quindi si ha quanto segue ( $\alpha$  rappresenta l'hit rate):

$$\begin{aligned} EAT &= \epsilon + \alpha t + (1 - \alpha)2t \\ 36 &= 1 + 30\alpha + (1 - \alpha) \cdot 60 \\ 36 &= 61 - 30\alpha \end{aligned}$$

da cui si ricava  $\alpha = \frac{25}{30} = \frac{5}{6} \approx 0,833$  (83,3%).

- (3 punti) Con una paginazione a due livelli si ha quanto segue:

$$\begin{aligned} EAT &= \epsilon + \alpha t + (1 - \alpha)3t \\ 36 &= 1 + 30\alpha + (1 - \alpha) \cdot 90 \\ 36 &= 91 - 60\alpha \end{aligned}$$

da cui si ricava  $\alpha = \frac{55}{60} = \frac{11}{12} \approx 0,917$  (91,7%).

4. Si consideri un processo che generi la seguente stringa di riferimenti alle pagine virtuali:

0 2 1 0 4 0 2 1 1 0 5 3 2

- Se il processo ha 4 frame, gestiti LRU, quanti page fault vengono generati?
- Qual è il numero minimo di frame necessario per minimizzare i page fault?

**Risposta:**

- (3 punti) Simuliamo il funzionamento di LRU nel caso della reference string data:

	0	2	1	0	4	0	2	1	1	0	5	3	2
		0	2	1	0	4	0	2	2	1	0	5	3
			0	2	1	1	4	0	0	2	1	0	5
					2	2	1	4	4	4	2	1	0
											4	2	1
												4	4
	P	P	P		P						P	P	P

Si verificano quindi sette page fault.

- (2 punti) Il minimo numero di page fault è 6 page fault (perché il processo accede a 6 pagine). Per determinare il numero minimo di frame per avere solo 6 page fault, si può sfruttare la *distance string*, che nel caso in questione risulta essere la seguente:

$\infty \ \infty \ \infty \ 3 \ \infty \ 2 \ 4 \ 4 \ 1 \ 3 \ \infty \ \infty \ 5$

Si ricorda che la distance string rappresenta la distanza fra la posizione di una pagina nel modello e la prima posizione, ovvero, quella nella prima riga della matrice (contando anche la casella di partenza) nel momento in cui la pagina stessa viene riferita. Se una pagina non è presente nella matrice, allora la sua distanza, quando viene riferita è  $\infty$ .

Indichiamo ora con  $C_i$  il numero di volte che il numero  $i$  compare nella distance string; nel caso in questione abbiamo:  $C_1 = 1$ ,  $C_2 = 1$ ,  $C_3 = 2$ ,  $C_4 = 2$ ,  $C_5 = 1$ ,  $C_\infty = 6$ . Indicando poi con  $m$  il numero di frame e con  $n$  il numero più grande che compare nella distance string,

# Sistemi Operativi

## 7 luglio 2011

### Compito A

indichiamo con  $F_m = \sum_{k=m+1}^n C_k + C_\infty$  il numero di page fault che si verificano con  $m$  frame e con la reference string data. L'intuizione è la seguente: se ho a disposizione  $m$  frame i page fault saranno provocati dai riferimenti a pagine che "distanano" almeno  $m + 1$  dal top della matrice e dal numero di  $\infty$  (ovvero da riferimenti a pagine non ancora presenti nel modello). Nel nostro caso abbiamo:  $F_1 = 12$ ,  $F_2 = 11$ ,  $F_3 = 9$ ,  $F_4 = 7$ ,  $F_5 = 6$ ; quindi il numero minimo di frame che minimizza i page fault è 5.

5. Si spieghi cos'è un file e quali sono le principali operazioni su di esso che un S.O. deve implementare.

**Risposta:** (3 punti) Un file è insieme di informazioni correlate a cui è stato assegnato un nome. Esso è inoltre la più piccola porzione unitaria di memoria logica secondaria allocabile dall'utente o dai processi di sistema. Le operazioni fondamentali sui file che un sistema operativo deve implementare sono le seguenti:

**Creazione:** allocazione dello spazio sul dispositivo e collegamento di tale spazio al file system.

**Cancellazione:** staccare il file dal file system e deallocare lo spazio assegnato al file.

**Apertura:** caricare alcuni metadati dal disco nella memoria principale, per velocizzare le chiamate seguenti.

**Chiusura:** deallocare le strutture allocate nell'apertura.

**Lettura:** dato un file e un puntatore di posizione, i dati da leggere vengono trasferiti dal media in un buffer in memoria.

**Scrittura:** dato un file e un puntatore di posizione, i dati da scrivere vengono trasferiti sul media.

**Append:** versione particolare di scrittura.

**Riposizionamento (seek):** non comporta operazioni di I/O.

**Troncamento:** azzerare la lunghezza di un file, mantenendo tutti gli altri attributi.

**Lettura dei metadati:** leggere le informazioni come nome, timestamp, ecc.

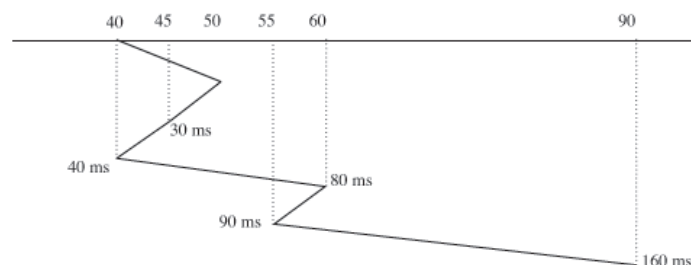
**Scrittura dei metadati:** modificare informazioni come nome, timestamps, protezione, ecc.

6. Si consideri un disco gestito con politica SSTF. Inizialmente la testina è posizionata sul cilindro 40; lo spostamento ad una traccia adiacente richiede 2 ms. Al driver di tale disco arrivano richieste per i cilindri 90, 45, 40, 60, 55, rispettivamente agli istanti 0 ms, 20 ms, 30 ms, 40 ms, 80 ms. Si trascuri il tempo di latenza.

1. In quale ordine vengono servite le richieste?
2. Il tempo di attesa di una richiesta è il tempo che intercorre dal momento in cui è sottoposta al driver a quando viene effettivamente servita. Qual è il tempo di attesa medio per le cinque richieste in oggetto?

**Risposta:**

1. (3 punti) Le richieste vengono servite nell'ordine 45, 40, 60, 55, 90:



2. (1 punto) Il tempo di attesa medio per le cinque richieste in oggetto è  $\frac{(160-0)+(30-20)+(40-30)+(80-40)+(90-80)}{5} = \frac{160+10+10+40+10}{5} = \frac{230}{5} = 46 \text{ ms}$ .