

Sistemi Operativi

4 febbraio 2013

Compito

Si risponda ai seguenti quesiti, giustificando le risposte.

1. (a) Che cosa si intende per processi CPU-bound e processi I/O-bound? Quali effetti si possono avere se viene data la precedenza a processi CPU-bound?
- (b) Si diano esempi di algoritmi di scheduling della CPU che privilegiano i processi I/O-bound.

Risposta:

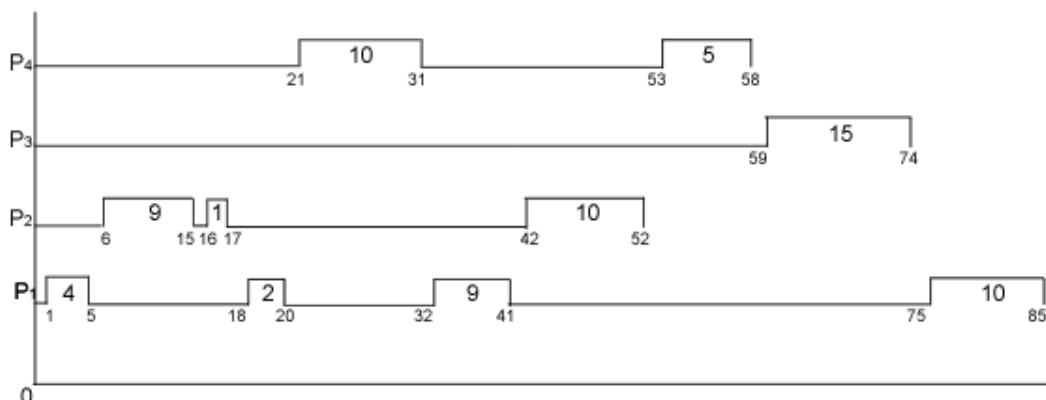
- (a) I processi possono essere classificati come I/O-bound quando presentano lunghi periodi di I/O e brevi periodi di calcolo. Al contrario si parla di processi CPU-bound quando si hanno lunghi periodi di intensiva computazione e pochi (possibilmente lunghi) cicli di I/O. Privilegiando i processi CPU-bound si allungano i tempi di risposta percepiti dagli utenti che stanno eseguendo dei processi interattivi (I/O-bound).
 - (b) In generale una classe di algoritmi di scheduling della CPU che privilegiano i processi I/O-bound è costituita dagli scheduling con code multiple e con retroazione (feedback) in cui la coda a maggior priorità viene riservata ai processi interattivi, mentre quelli CPU-bound vengono posizionati nelle code a minor priorità. Fra i sistemi operativi di uso comune l'algoritmo di scheduling di Linux è un esempio di algoritmo che favorisce i processi interattivi (I/O-bound) dato che assegna un bonus di priorità dinamica a questi ultimi. Anche lo scheduling di Windows favorisce i processi I/O-bound dato che sceglie sempre dalla coda a priorità maggiore, ma la priorità di un thread utente può essere temporaneamente maggiore di quella base (per effetto delle "spinte") per thread che attendono dati di I/O (spinte fino a +8) e per dare maggiore reattività a processi interattivi (+2).
2. Si consideri un sistema con scheduling della CPU a priorità con tre code, A, B, C, di priorità decrescente, con prelazione tra code. Le code A e B sono round robin con quanto di 10 e 15 ms, rispettivamente; la coda C è FCFS. Se un processo nella coda A o B consuma il suo quanto di tempo, viene spostato in fondo alla coda B o C, rispettivamente.

Nelle code A, B, C entrano i seguenti processi:

	codice	arrivo	burst
P_1	B	0	25ms
P_2	A	5	20ms
P_3	C	15	15ms
P_4	A	20	15ms

Si determini il diagramma di GANTT relativo all'esecuzione dei quattro processi, assumendo che il tempo di latenza del kernel sia pari a 1 ms.

Risposta: Il diagramma di GANTT relativo all'esecuzione dei quattro processi è il seguente:



3. Si vuole evitare che all'interno di un locale ci siano contemporaneamente più di N utenti. Si completi il seguente monitor, che gestisce l'accesso e l'uscita dal locale.

Sistemi Operativi

4 febbraio 2013

Compito

```
monitor Locale
  condition notfull,notempty;
  integer spaces, capacity;

  procedure enter();
  begin
    ...
    ...
    ...
  end;

  procedure exit();
  begin
    ...
    ...
    ...
  end;

  spaces := N;
  capacity := N;
end monitor;
```

Risposta:

```
monitor Locale
  condition notfull,notempty;
  integer spaces, capacity;

  procedure enter();
  begin
    while (spaces = 0) do wait (notfull);
    spaces := spaces - 1;
    signal (notempty);
  end;

  procedure exit();
  begin
    while (spaces = capacity) do wait (notempty);
    spaces := spaces + 1;
    signal (notfull);
  end;

  spaces := N;
  capacity := N;
end monitor;
```

4. Si consideri la seguente situazione, dove P_0, P_1, P_2, P_3, P_4 sono cinque processi in esecuzione, C è la matrice delle risorse correntemente allocate, Max è la matrice del numero massimo di risorse da assegnare ad ogni processo e A è il vettore delle risorse disponibili:

	<u>C</u>				<u>Max</u>			
	A	B	C	D	A	B	C	D
P_0	0	0	1	2	0	0	1	2
P_1	1	0	0	0	1	7	5	0
P_2	1	3	5	4	2	3	5	6
P_3	0	6	3	2	0	6	5	2
P_4	0	0	1	4	0	6	5	6

Sistemi Operativi

4 febbraio 2013

Compito

Available (A)			
A	B	C	D
1	5	2	0

- (a) Calcolare la matrice R delle richieste.
 (b) Il sistema è in uno stato sicuro?
 (c) La richiesta (0 4 2 0) da parte del processo P_1 può essere soddisfatta immediatamente?

Risposta:

- (a) $R = MAX - C$:

				R
A	B	C	D	
0	0	0	0	
0	7	5	0	
1	0	0	2	
0	0	2	0	
0	6	4	2	

- (b) Il sistema è in uno stato sicuro in quanto esiste la sequenza di esecuzione sicura P_0 (nuovo valore di $A = (1\ 5\ 3\ 2)$), P_2 (nuovo valore di $A = (2\ 8\ 8\ 6)$), P_1 (nuovo valore di $A = (3\ 8\ 8\ 6)$), P_3 (nuovo valore di $A = (3\ 14\ 11\ 8)$) e P_4 (nuovo valore di $A = (3\ 14\ 12\ 12)$).
 (c) Se la richiesta (0 4 2 0) da parte del processo P_1 viene soddisfatta immediatamente, si hanno i seguenti nuovi valori per le matrici C , R e A :

C				R			
A	B	C	D	A	B	C	D
0	0	1	2	0	0	0	0
1	4	2	0	0	3	3	0
1	3	5	4	1	0	0	2
0	6	3	2	0	0	2	0
0	0	1	4	0	6	4	2

$$A = (1\ 1\ 0\ 0)$$

La richiesta può essere accettata in quanto esiste la sequenza di esecuzione sicura P_0 (nuovo valore di $A = (1\ 1\ 1\ 2)$), P_2 (nuovo valore di $A = (2\ 4\ 6\ 6)$), P_1 (nuovo valore di $A = (3\ 4\ 6\ 6)$), P_3 (nuovo valore di $A = (3\ 10\ 9\ 8)$) e P_4 (nuovo valore di $A = (3\ 10\ 10\ 12)$).

5. Si consideri un sistema con memoria paginata a due livelli, in cui sia la page table esterna che quella interna necessarie alla risoluzione dell'indirizzo siano mantenute in memoria principale. Il tempo di accesso alla memoria principale sia $t = 60ns$.
- (a) Qual è il tempo effettivo di accesso alla memoria (EAT: Effective Access Time)?
 (b) Aggiungendo un TLB (Translation Look-aside Buffer), con tempo di accesso $\epsilon = 1ns$, quale hit rate (α) dobbiamo avere per un degrado delle prestazioni del 5% rispetto a t ?

Risposta:

- Il tempo effettivo di accesso alla memoria è $3t$, ovvero, 180 ns; infatti sono necessari 60 ns per accedere alla page table esterna, 60 ns per accedere alla page table interna ed infine 60 ns per accedere alla locazione nel frame fisico in memoria.
- Un degrado del 5% rispetto a t significa un EAT pari a $1,05 \cdot t$, ovvero, 63 ns. Quindi si ha quanto segue (α rappresenta l'hit rate):

$$\begin{aligned} EAT &= \epsilon + \alpha t + (1 - \alpha)3t \\ 63 &= 1 + 60\alpha + (1 - \alpha) \cdot 180 \\ 63 &= 181 - 120\alpha \end{aligned}$$

da cui si ricava $\alpha = \frac{118}{120} = 0,98$ (98%).

Sistemi Operativi

4 febbraio 2013

Compito

6. Nell'ambito dell'implementazione del filesystem, si spieghi come funziona lo schema di allocazione dei blocchi del disco noto come *allocazione concatenata*. In particolare, si illustri come tradurre l'indirizzo logico nel corrispondente indirizzo fisico.

Risposta: Lo schema di allocazione concatenata prevede che ogni blocco dati abbia una parte di esso riservata alla memorizzazione dell'indirizzo del blocco successivo (l'ultimo blocco viene marcato con un indirizzo non valido, e.g., -1). In questo modo l'implementazione di un file consiste in una lista di blocchi concatenati ed è sufficiente memorizzare per ogni file nella directory entry il blocco iniziale e quello finale. I pregi consistono essenzialmente nella semplicità dell'implementazione, mentre gli aspetti negativi sono la fragilità dello schema (se si corrompe un puntatore della catena, si perde tutta la parte del file da quel punto in poi) e la mancanza di supporto dell'accesso diretto (seek); infatti per accedere al blocco i -esimo bisogna prima scorrere la catena di puntatori dei precedenti $i-1$ blocchi.

Per tradurre un indirizzo logico LA nel corrispondente indirizzo fisico, è sufficiente eseguire la divisione intera di LA per $B - k$ dove B è la dimensione del blocco e k è lo spazio necessario a memorizzare un indirizzo:

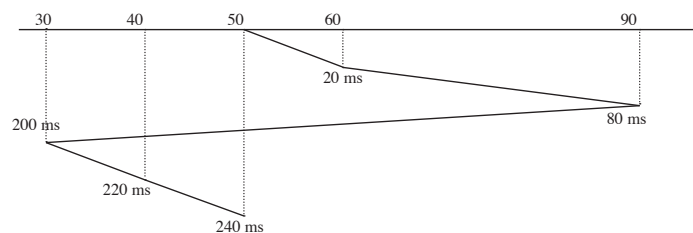
$$Q = \frac{LA}{(B - k)}$$

il quoziente rappresenta il numero del blocco da accedere (0 indica il primo blocco della catena), mentre l'offset all'interno di quest'ultimo è rappresentato dal resto della divisione precedente sommato a k .

7. Si consideri un disco gestito con politica C-LOOK (LOOK *circolare*) con direzione di servizio verso tracce con numero *crescente*. Inizialmente la testina è posizionata sul cilindro 50; lo spostamento ad una traccia adiacente richiede 2 ms. Al driver di tale disco arrivano richieste per i cilindri 90, 60, 50, 30, 40, rispettivamente agli istanti 0 ms, 10 ms, 30 ms, 60 ms, 70 ms. Si trascuri il tempo di latenza.
- (a) In quale ordine vengono servite le richieste?
- (b) 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 60, 90, 30, 40, 50:



2. (1 punto) Il tempo di attesa medio per le cinque richieste in oggetto è
- $$\frac{(20-10)+(80-0)+(200-60)+(220-70)+(240-30)}{5} = \frac{10+80+140+150+210}{5} = \frac{590}{5} = 118 \text{ ms.}$$