

Sistemi Operativi

23 giugno 2011

Compito A

Si risponda ai seguenti quesiti, giustificando le risposte.

- (a) In quali situazioni può essere attivato lo scheduling della CPU?
- (b) Quando un algoritmo di scheduling è preemptive? Quali sono i vantaggi e gli svantaggi di un algoritmo preemptive?

Risposta:

- (3 punti) Lo scheduling della CPU può essere attivato nelle seguenti circostanze:
 - un processo viene creato ed entra nella coda dei pronti;
 - un processo passa dallo stato di esecuzione allo stato di attesa;
 - un processo passa dallo stato di esecuzione allo stato pronto;
 - un processo passa dallo stato di attesa allo stato pronto;
 - un processo termina.
 - (3 punti) Un algoritmo di scheduling si dice preemptive se può interrompere l'esecuzione di un processo a favore di un altro processo e può quindi essere attivato ogni volta che un processo passa nella coda dei pronti (coda ready), oltre che ovviamente anche in altri casi. I vantaggi di un algoritmo preemptive sono essenzialmente dei tempi di risposta migliori e la garanzia che nessun processo riesca a monopolizzare la CPU senza rilasciarla. Gli svantaggi sono relativi alla condivisione dei dati fra processi; infatti, se un processo sta manipolando dei dati utilizzati anche da altri processi e viene prelazionato c'è il rischio che questi rimangano in uno stato inconsistente e generino così degli errori. Per evitare tutto ciò è necessario un attento uso di primitive come mutex e semafori per garantire un accesso esclusivo e corretto alle risorse condivise.
2. Si consideri un sistema con scheduling 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.
- Di quale tipo di algoritmo di scheduling si tratta?
 - Nelle code A, B, C entrano i seguenti processi:

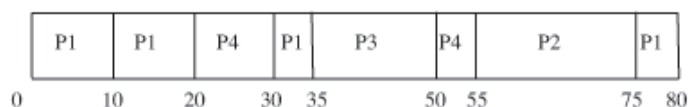
	coda	arrivo	burst
P_1	A	0	30ms
P_2	C	5	20ms
P_3	B	15	15ms
P_4	A	20	15ms

Si determini:

- il diagramma di GANTT relativo all'esecuzione dei quattro processi;
- il tempo di attesa medio;
- il tempo di reazione medio.

Risposta:

- (2 punti) Si tratta di un algoritmo di scheduling a code multiple con feedback negativo.
- (4 punti) Assunzione: un processo prelazionato viene posto all'inizio della sua coda.
 - Il diagramma di Gantt è il seguente:



- Tempo di attesa medio = $\frac{50+50+20+20}{4} = \frac{140}{4} = 35$ ms.
- Tempo di reazione medio = $\frac{0+50+20+0}{4} = \frac{70}{4} = 17,5$ ms.

Sistemi Operativi
23 giugno 2011
Compito A

3. Si descriva cosa si intende per anomalia di Belady e si diano esempi di algoritmi che ne sono affetti e non.

Risposta: (4 punti) Per anomalia di Belady si intende il fenomeno per cui, nonostante si incrementi la memoria fisica disponibile e quindi il numero di frame totali, non è detto che i page fault diminuiscano. Un algoritmo di rimpiazzamento delle pagine che soffre di questo problema è l'algoritmo FIFO (First-In First-Out), mentre LRU (Least Recently Used) e tutti gli algoritmi di stack ne sono immuni.

4. 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 $t = 30ns$.

- (a) Qual è il tempo effettivo di accesso alla memoria?
(b) Aggiungendo un TLB, con tempo di accesso $\epsilon = 1ns$, quale hit rate dobbiamo avere per un degrado delle prestazioni del 3% rispetto a t ?
(c) E con una paginazione a due livelli?

Risposta:

1. (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.
2. (3 punti) Un degrado del 3% rispetto a t significa un EAT pari a $1,03 \cdot t$, ovvero, 30,9 ns. Quindi si ha quanto segue (α rappresenta l'hit rate):

$$\begin{aligned} EAT &= \epsilon + \alpha t + (1 - \alpha)2t \\ 30,9 &= 1 + 30\alpha + (1 - \alpha) \cdot 60 \\ 30,9 &= 61 - 30\alpha \end{aligned}$$

da cui si ricava $\alpha = \frac{30,1}{30} = 1,003$ (100%).

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

$$\begin{aligned} EAT &= \epsilon + \alpha t + (1 - \alpha)3t \\ 30,9 &= 1 + 30\alpha + (1 - \alpha) \cdot 90 \\ 30,9 &= 91 - 60\alpha \end{aligned}$$

da cui si ricava $\alpha = \frac{60,1}{60} = 1,001$ (100%).

5. Quando un'interruzione viene definita *precisa* (si elenchino le quattro condizioni)?

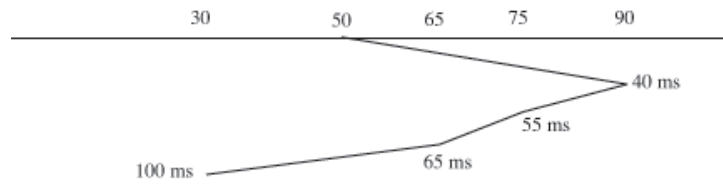
Risposta: (4 punti) Un'interruzione si dice precisa quando gode delle seguenti quattro proprietà:

1. il program counter viene salvato in un posto noto,
 2. tutte le istruzioni che precedono quella puntata dal program counter sono state completamente eseguite,
 3. nessuna delle istruzioni che seguono quella puntata dal program counter è stata eseguita,
 4. lo stato di esecuzione dell'istruzione puntata dal program counter è noto.
6. Si consideri un disco gestito con politica LOOK. Inizialmente la testina è posizionata sul cilindro 50; lo spostamento ad una traccia adiacente richiede 1 ms. Al driver di tale disco arrivano richieste per i cilindri 90, 65, 75, 30, rispettivamente agli istanti 0 ms, 20 ms, 40 ms, 55 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 quattro richieste in oggetto?

Risposta:

1. (3 punti) Le richieste vengono servite nell'ordine 90, 75, 65, 30:

Sistemi Operativi
23 giugno 2011
Compito A



2. (2 punti) Il tempo di attesa medio per le quattro richieste in oggetto è
- $$\frac{(40-0)+(65-20)+(55-40)+(100-55)}{4} = \frac{40+45+15+45}{4} = \frac{145}{4} = 36,25 \text{ ms.}$$