

Sistemi Operativi

Compito Basic 2 settembre 2005

Si risponda ai seguenti quesiti in modo sintetico (evidenziando quindi tutti gli aspetti necessari per rispondere completamente alla domanda evitando di fornire dettagli insignificanti o non richiesti)

1. (a) Quali sono gli obiettivi di un sistema operativo?

L'obiettivo generale di un SO è realizzare una macchina astratta, offrendo funzionalità di alto livello, in modo da:

- eseguire programmi utente;
- rendere il sistema più facile da utilizzare e da programmare;
- utilizzare l'HW in modo sicuro ed efficiente.

- (b) Che cosa sono i programmi di sistema? Si fornisca qualche esempio.

I programmi di sistema sono programmi indipendenti dalle applicazioni, che richiamano i servizi del SO direttamente attraverso le system call e forniscono un ambiente per lo sviluppo e l'esecuzione dei programmi utente (o applicativi). Oltre al SO, i programmi di sistema comprendono l'interprete comandi, programmi per la gestione di file, per la scrittura di file (editor), programmi di supporto per i linguaggi di programmazione (compilatori, interpreti), programmi per la comunicazione (posta elettronica, trasferimento file, login remoto).

2. (a) Quali risorse condividono i thread di un processo e quali sono private?

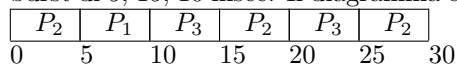
Le risorse condivise dai thread di un processo comprendono: codice eseguibile, variabili globali, risorse richieste al SO. Le risorse private di ogni thread comprendono: program counter, registri, stack con variabili locali, stato di esecuzione.

- (b) Si descriva il diagramma a stati di un processo e si diano esempi di eventi che causano transizioni da uno stato all'altro.

Si tratta del diagramma a cinque stati dei processi (vedi lucidi). Esempi di eventi che causano transizioni:

- ready \rightarrow running: processo selezionato dallo scheduler
- running \rightarrow ready: prelazione
- running \rightarrow waiting: il processo ha richiesto un'operazione di I/O e va in attesa del suo completamento
- waiting \rightarrow ready: completata l'operazione di I/O il processo torna in coda ready
- running \rightarrow terminated: il processo termina l'esecuzione

3. I processi P_1 , P_2 , P_3 arrivano in coda ready agli istanti 0, 0, 10 msec rispettivamente e hanno CPU burst di 5, 15, 10 msec. Il diagramma di GANTT relativo all'esecuzione dei tre processi è il seguente:



Con quali dei seguenti algoritmi di scheduling è compatibile il diagramma sopra?

- A. Scheduling basato su priorità statica
- B. RR $q=5$ msec
- C. Scheduling non-preemptive
- D. Scheduling basato su due code A e B a diversa priorità statica con prelazione tra code e priorità di A maggiore di B; A è RR $q=5$ msec, B è RR $q=10$ msec

Sistemi Operativi

Compito Basic 2 settembre 2005

E. Nessuno dei precedenti

E

4. (a) Cos'è una race condition?

- (b) Può avvenire in sistemi interattivi con scheduling FCFS?

5. (a) Qual è l'obiettivo dell'algoritmo del banchiere? Come riesce a garantirlo?

Garantire l'assenza di deadlocks. Ci riesce in modo aggressivo negando risorse che potrebbero portare il sistema in uno stato non sicuro. Richiede di sapere *a priori* il numero massimo di risorse che un processo può richiedere durante il funzionamento.

- (b) Si tratta di un algoritmo computazionalmente pesante o leggero? In quali situazioni viene applicato?

Leggero, basta scandire una tabella di numeri.

In contesti molto ben definiti dove si riesce a sapere quale sia il numero massimo di risorse che un processo può richiedere durante il funzionamento.

6. Si consideri la seguente tabella dei segmenti con la corrispondente tabella delle pagine in un sistema con memoria virtuale a segmentazione con paginazione con pagine da 256B.

Segm.	Base	Limite
0	0x74C0FE	0x30
1	0x74C174	0x10
2	0x74C010	0x50
3	0x700008	0x12

...	...
0x74C0	0xB30
0x74C1	0x999
...	...

Si calcolino gli indirizzi fisici corrispondenti agli indirizzi logici: A. $\langle 5, 0x68 \rangle$. B. $\langle 3, 0x1A \rangle$; C. $\langle 0, 0x20 \rangle$; D. $\langle 1, 0x36 \rangle$; E. $\langle 2, 0x4C \rangle$;

A. errore (segmento inesistente) B. errore (fuori segmento) C. 9991E D. errore (fuori segmento) E. B305C

7. Un computer ha quattro frame. Trovare dei valori adatti a x e y da mettere nella seguente tabella (che riporta per ogni frame l'istante di caricamento, di ultimo riferimento e il reference bit) affinché la pagina che verrebbe liberata dall'algoritmo FIFO e da LRU sia rispettivamente 3 e 2. Qual'è la pagina liberata da CLOCK (il puntatore all'ultimo frame controllato sia sul 2)?

Frame	Car.	Rif.	R
0	110	183	1
1	160	190	0
2	x	y	1
3	90	180	1

Sistemi Operativi

Compito Basic 2 settembre 2005

$x < y$ visto che non si può riferire prima di aver caricato. $x > 90$ altrimenti FIFO sceglierebbe
2. $y < 180$ altrimenti LRU sceglierebbe 3. Quindi $90 < x < y < 180$. CLOCK: 1.

8. (a) Si descriva cosa si intende per interruzioni precise ed imprecise.

- (b) A seconda dei casi è sempre possibile che il sistema operativo riesca a salvare nelle sue strutture uno stato coerente? E a quale costo?

9. Qual è la principale differenza nell'organizzazione delle directories fra HFS (Mac OS) e i vari file systems dei sistemi MS-DOS/UNIX? In quale situazione questa (profonda) scelta progettuale cambia radicalmente gli effetti e cosa succede invece negli altri casi?

Le directories in HFS sono organizzate come un B-tree.

Quando si eseguono ricerche di files all'interno dell'intero file system la struttura a B-tree accelera enormemente la ricerca. Per operazioni di altro genere invece l'overhead sulle modifiche del B-tree tende a svantaggiare leggermente.

10. Il meccanismo di file servicing NFS della Sun è Stateful (a stato) o Stateless (senza stato)?

Tale scelta compiuta dai progettisti è giustificata in termini di performance o dovuta ad altri fattori?

Se è stata compiuta la scelta più performante, a quale altre caratteristiche si è rinunciato? Se è stata compiuta la scelta meno performante, quali accorgimenti sono stati adottati in modo da alleviare il problema?