

Esercizi proposti per il corso di “progetto di sistemi operativi”

Esercizi sulla gestione della memoria e sulla memoria virtuale

- (1) Supponiamo che il meccanismo hardware di traduzione degli indirizzi preveda la presenza di un registro base RB e di un registro limite RL, per cui ogni indirizzo logico viene tradotto secondo il seguente schema:

if indirizzo_logico > RL **then** TRAP

else indirizzo_fisico = indirizzo_logico + RB

Quale delle seguenti tecniche di gestione della memoria può essere implementata?

- partizioni fisse
 - partizioni variabili
 - paginazione
 - segmentazione paginata
- (2) Si consideri un sistema paginato con tabella delle pagine memorizzata in memoria centrale. Si supponga che lo spazio degli indirizzi virtuali sia di 16 pagine di 512 parole, la memoria fisica sia di 64 frame, e ogni descrittore di pagina logica nella tabella delle pagine occupi una parola.
- Quanti bit compongono un indirizzo logico (o virtuale) ?
 - Quanti bit compongono un indirizzo fisico?
 - Nell'ipotesi che il sistema in esame abbia un grado di multiprogrammazione costante uguale ad n , calcolare l'overhead di memoria dovuto alla paginazione (ovvero la quantità totale di memoria utilizzata per mantenere le tabelle delle pagine più la memoria sprecata a causa della frammentazione).
- (3) Si supponga di avere una memoria fisica di dimensione 4096Kbyte, e di utilizzare uno schema di gestione a partizioni variabili in cui l'unità minima di allocazione della memoria è di 64 byte (ovvero lo spazio di memoria viene allocato in multipli di 64 byte). Si considerino le seguenti strutture per mantenere traccia dello spazio libero disponibile:
- una *bit map*
 - una lista linkata in cui ogni nodo rappresenta una partizione libera; i nodi della lista sono costituiti da due campi: (**dimensioni della partizione, puntatore alla partizione successiva**) e sono memorizzati nei primi byte della partizione che rappresentano.

Per ciascuna di queste due alternative si valuti la quantità di spazio che è necessario riservare per mantenere la struttura e si mettano in luce vantaggi e svantaggi. Quale di queste alternative può essere usata vantaggiosamente per gestire lo spazio libero su disco?

- (4) Data la seguente stringa di riferimenti alle pagine di un processo

0, 1, 2, 1, 3, 4, 1, 2, 0, 1

Calcolare il numero di page fault, indicando a fronte di ciascun riferimento l'eventuale vittima, utilizzando

- l'algoritmo FIFO con una memoria fisica di 4 frames

- l'algoritmo della seconda scelta con una memoria fisica di 4 frames
- l'algoritmo LRU con una memoria fisica di 3 o 4 frames

(5) Si consideri la seguente sequenza di riferimenti in memoria nel caso di un programma di 460 parole:

10, 11, 104, 170, 73, 309, 185, 245, 246, 434, 458, 364

- Si determini la stringa dei riferimenti supponendo che la dimensione delle pagine sia di 100 parole. Sapendo che la memoria principale a disposizione per il programma è di 200 parole e che si utilizza come algoritmo di rimpiazzamento il FIFO, determinare quanti page fault si verificheranno.
- Si ripeta il precedente esercizio supponendo che la dimensione delle pagine sia ridotta a 50 parole mentre la dimensione della memoria a disposizione e l'algoritmo di rimpiazzamento rimangano invariati.

Osservazione: normalmente la dimensione delle pagine è una potenza di due (sapete spiegare perchè?). In questo esercizio si è usata una dimensione diversa per facilitare il calcolo delle pagine a partire dagli indirizzi indicati.

(6) In un sistema che usa uno schema di gestione della memoria a partizioni variabili, la lista delle partizioni libere contiene, nell'ordine, una partizione da 100K, una da 500K, una da 200K, una da 300K e una da 600K. Come verrebbero allocate le seguenti richieste (sottoposte al sistema nell'ordine indicato) e utilizzando gli algoritmi First fit, Best fit e Worst fit?
212K, 417K, 112K e 426K

(7) Si consideri un sistema che usa la paginazione su richiesta. Sono state rilevate le seguenti utilizzazioni medie delle risorse del sistema

- $U(\text{CPU}) = 20\%$
- $U(\text{DISCO}) = 97.7\%$ (si tratta del disco a supporto della paginazione)
- $U(\text{I/O DEVICES}) = 5\%$

Quali dei seguenti interventi possono migliorare l'utilizzo della CPU, e perchè?

- installare una CPU più veloce
- installare un disco più grande a supporto della paginazione
- incrementare il livello di multiprogrammazione
- decrementare il livello di multiprogrammazione

(8) Data la seguente stringa di riferimenti a pagina: 1234531154323345241345 calcolare il demand point per la strategia Page Fault Frequency con parametro di controllo $C=3$.

Esercizi sullo scheduling della CPU

- (1) Si consideri il seguente insieme di processi: ad ognuno è associata la durata del prossimo CPU-burst e la priorità.

PROC	CPU-burst	PRIOR
P1	10	3
P2	1	1
P3	2	3
P4	1	4
P5	5	2

Si suppone che i processi siano arrivati tutti al tempo 0 nell'ordine del corrispondente indice. Si illustri attraverso diagrammi di Gantt l'ordine di esecuzione dei processi utilizzando gli algoritmi FCFS, SJF, priorità (numeri piccoli = priorità alta) e Round Robin con quanto di tempo di 1 unità. Calcolare in ciascun caso il tempo di turnaround e il tempo di attesa di ogni processo. Quale algoritmo permette di ottenere il minor tempo medio di attesa?

- (2) Si consideri il seguente algoritmo di scheduling basato su priorità dinamiche (ovvero che cambiano nel tempo). In questo sistema valori numerici più grandi corrispondono a priorità più alte. Quando un processo è nella ready queue, la sua priorità cambia con un tasso x ; quando invece è running la sua priorità cambia con un tasso y . Quando un processo entra nella ready queue gli viene assegnata priorità 0. Assegnando opportuni valori ai parametri x ed y è possibile ottenere diversi algoritmi di scheduling. Quali algoritmi di scheduling si ottengono nei due casi seguenti?

- $y > x > 0$
- $x < y < 0$

Suggerimento: La priorità di un processo ready al tempo $t+dt$ (assumendo che sia rimasto ready in tutto l'intervallo $(t, t+dt)$) è $prio(t+dt) = prio(t) + x dt$, per un processo running invece sarà $prio(t+dt) = prio(t) + y dt$. Si suggerisce di provare a rappresentare in un grafico l'andamento nel tempo delle priorità dei processi ready e running.