

- 1. Un sistema di gestione della memoria con paginazione di quale tipo di frammentazione può soffrire? E un sistema con segmentazione?
- 2. Cos'è l'*anomalia di Belady*? Si citi un algoritmo (o, alternativamente, una classe di algoritmi) di rimpiazzamento delle pagine per cui *non* si presenta questo problema.
- 3. Si descriva il funzionamento dell'algoritmo basato sul working set model per il rimpiazzamento delle pagine.
- 4. (*) Supponendo di avere un sistema con quattro frame e sette pagine, adottando una politica di rimpiazzamento basata sul working set model, quanti page fault si verificheranno con la reference string seguente, assumendo $\Delta = 4$ e di mantenere in memoria esattamente il solo working set?

0 1 6 2 3 2 6 1 2 3 1

(Si assuma che i quattro frame siano inizialmente vuoti.)

Risposta:

1. Un sistema di gestione della memoria con paginazione può soffrire del problema di frammentazione interna; infatti è difficile che le immagini dei processi in memoria abbiano dimensioni che siano multipli esatti della dimensione delle pagine. Quindi l'ultima pagina sarà soltanto parzialmente utilizzata.
Un sistema con segmentazione invece può soffrire del problema di frammentazione esterna; infatti i segmenti hanno dimensioni variabili e, caricando e rimuovendo processi dalla memoria, può capitare che si formino delle zone di memoria libera tra un segmento e l'altro così piccole da non consentire l'allocazione di un nuovo segmento.
2. 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 non soffre di questo problema è l'algoritmo LRU che associa ad ogni pagina il tempo di ultimo riferimento in modo da rimpiazzare in caso di bisogno quella utilizzata meno recentemente. In generale tutti gli algoritmi di stack, ovvero gli algoritmi tali che $M(m, r) \subseteq M(m + 1, r)$ (dove $M(m, r)$ rappresenta l'insieme di pagine presenti in memoria in un sistema con m frame in seguito al riferimento di indice r nella reference string in considerazione), non soffrono dell'anomalia di Belady.
3. L'algoritmo per il rimpiazzamento delle pagine basato sul working set model sfrutta il concetto località dei riferimenti, ovvero, la teoria che prevede che un processo durante ogni fase della propria esecuzione faccia riferimento ad un sottoinsieme ristretto delle proprie pagine. Tale sottoinsieme viene definito il working set del processo. Fissato un numero k , l'algoritmo di rimpiazzamento basato sul working set seleziona come pagina da rimpiazzare una di quelle non presenti nell'approssimazione del working set corrente data dagli ultimi k riferimenti.
4. Simuliamo il funzionamento dell'algoritmo basato sul working set model:

0	1	6	2	3	2	6	1	2	3	1
0	1	6	2	3	2	6	1	2	3	1
	0	1	6	2	3	2	6	1	2	3
		0	1	6	6	3	2	6	1	2
			0	1			3		6	
P	P	P	P	P			P		P	

Si verificano quindi sette page fault.

- 1. Elencare i problemi che derivano dall'aver delle interruzioni imprecise.
- 2. Elencare i passi eseguiti dai driver dei dispositivi per servire le richieste.
- 3. Spiegare brevemente come funziona il meccanismo di allocazione indicizzata.

Risposta:

1. Nel caso in cui si abbiano interruzioni imprecise è difficile riprendere l'esecuzione in modo esatto in hardware: la CPU si limita a "ri-versare" sullo stack tutta l'informazione relativa allo stato corrente, lasciando che sia il sistema operativo a capire che cosa debba essere fatto. In questo modo rallentano le fasi di ricezione degli interrupt e di context-switch/ripristino dell'esecuzione, provocando grosse latenze.
2. I passi fondamentali sono i seguenti:
 - (a) controllare i parametri passati,
 - (b) organizzare le richieste in una coda di operazioni che sono soggette a scheduling,
 - (c) eseguire le operazioni, accedendo al controller del dispositivo,
 - (d) mettere il processo in modalità wait (nel caso di I/O interrupt-driven) o attendere la fine dell'operazione in busy-wait,
 - (e) controllare lo stato dell'operazione tramite il controller del dispositivo,
 - (f) restituire il risultato.
3. L'allocazione indicizzata raggruppa tutti i puntatori ai blocchi (sparsi nel disco) di un file in un *blocco indice*. Quindi l'i-esimo elemento di tale blocco è un puntatore al blocco i-esimo del file ed è sufficiente mantenere nella directory il puntatore al blocco indice per accedere ai blocchi dati del file. Nel caso di un file vuoto (ad esempio appena creato) tutti i puntatori del blocco indice saranno inizializzati con un valore speciale che non punta ad alcun blocco dati sul disco (solitamente -1). Al momento di allocare un nuovo blocco dati per il file è sufficiente aggiornare il primo puntatore "libero" con l'indirizzo del nuovo blocco restituito dal sistema di gestione dello spazio libero su disco. La traduzione da indirizzo logico (LA) ad indirizzo fisico avviene dividendo LA per la dimensione del blocco: il quoziente rappresenta l'offset nel blocco indice, mentre il resto rappresenta l'offset nel blocco dei dati indicato dall'indice. Per file di grosse dimensioni un singolo blocco indice può rivelarsi insufficiente; quindi ci sono due varianti:

- (a) schema concatenato: l'ultimo indirizzo del blocco indice è un puntatore ad un ulteriore blocco indice e così via fin quando è necessario;
- (b) indice a più livelli: ogni indirizzo del blocco indice punta a un blocco indice di secondo livello che punta ai blocchi dati (volendo si possono inserire più livelli).

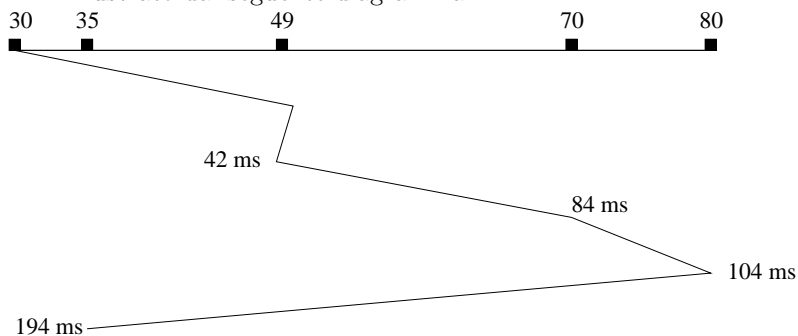
- 1. Descrivere cosa si intende per sistema distribuito.
- 2. Descrivere cosa si intende per sistema multiprocessore NUMA. È un sistema strettamente o debolmente accoppiato?

Risposta:

1. Un sistema distribuito è un insieme di unità di elaborazione debolmente accoppiate, ovvero, che non condividono clock e memoria e che comunicano tramite diverse linee di comunicazione (es.: reti locali o geografiche).
 2. La sigla NUMA significa Non Uniform Memory Access; come per l'architettura UMA vi è un singolo spazio di indirizzi comune a tutte le CPU, ma in questo caso l'accesso alla memoria locale è più veloce dell'accesso alla memoria remota. Ciò consente di ottenere sistemi scalabili con un gran numero di CPU. Trattandosi di un sistema multiprocessore con memoria comune, è un sistema strettamente accoppiato.
- (*) Si consideri un disco gestito con politica SSTF. Inizialmente la testina è posizionata sul cilindro 30, ascendente; lo spostamento ad una traccia adiacente richiede 2 ms. Al driver di tale disco arrivano richieste per i cilindri 70, 49, 35, 80, rispettivamente agli istanti 0 ms, 40 ms 50 ms, 70 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. L'ordine in cui vengono servite le richieste è 49, 70, 80, 35 come illustrato dal seguente diagramma:



2. Il tempo d'attesa medio è $\frac{(21*2-40)+(42*2-0)+(52*2-70)+(97*2-50)}{4} = \frac{2+84+34+144}{4} = 66 \text{ ms.}$

Il punteggio attribuito ai quesiti è il seguente: 2, 2, 3, 5, 3, 3, 3, 3, 3, 5 (totale: 32).