

- 1. Si descriva cosa si intende per *interruzione precisa*.
- 2. Descrivere le differenze fra le seguenti modalità di I/O:
 - Programmed I/O (PIO),
 - Interrupt-driven I/O,
 - Direct Memory Access (DMA).
- 3. Spiegare brevemente come funziona il meccanismo di allocazione concatenata.

Risposta:

1. Un'interruzione si dice precisa quando gode delle seguenti quattro proprietà:
 - (a) il program counter viene salvato in un posto noto,
 - (b) tutte le istruzioni che precedono quella puntata dal program counter sono state completamente eseguite,
 - (c) nessuna delle istruzioni che seguono quella puntata dal program counter è stata eseguita,
 - (d) lo stato di esecuzione dell'istruzione puntata dal program counter è noto.
2. Mentre con la modalità Programmed I/O (I/O a interrogazione ciclica) il processore manda un comando di I/O e poi attende che l'operazione sia terminata, testando lo stato del dispositivo con un loop busy-wait (polling), con la modalità Interrupt-driven I/O, una volta inviato il comando di I/O, il processo viene sospeso fintanto che non arriva un interrupt a segnalare il completamento dell'operazione. Durante la sospensione del processo, la CPU può mandare in esecuzione altri processi o thread. Di fondamentale importanza è il vettore di interrupt che consente di selezionare la routine di gestione opportuna per ogni tipo di interrupt. Ovviamente la prima modalità è efficiente soltanto nel caso in cui la velocità del dispositivo di I/O sia paragonabile a quella della CPU. La modalità DMA richiede un controller DMA e funziona in questo modo: la CPU imposta i registri del controller DMA specificando il tipo di azione di I/O, l'indirizzo di memoria ed il conteggio di byte da trasferire. Poi i dati vengono trasferiti senza più richiedere l'intervento della CPU; infatti il controller del dispositivo di I/O riceve le richieste di lettura o scrittura da parte del controller DMA a cui notifica il completamento dell'operazione una volta che ha trasferito il byte da/verso l'indirizzo di memoria corretto (specificato dal controller DMA). A questo punto il controller DMA incrementa l'indirizzo di memoria comunicandolo sul bus e decrementa il conteggio dei byte da trasferire, ripetendo la richiesta di lettura o scrittura al controller del dispositivo fintanto che il conteggio dei byte non raggiungerà lo zero. Soltanto a questo punto verrà inviato un interrupt alla CPU che potrà far ripartire il processo sospeso. Siccome il controller DMA deve bloccare il bus per consentire i trasferimenti dal controller del dispositivo alla memoria, se anche la CPU ha bisogno di accedere al bus dovrà aspettare, venendo così rallentata.

3. Con l'allocazione concatenata ogni file è costituito da una lista concatenata di blocchi che possono essere sparsi dovunque nel disco. Ogni entry di directory contiene un puntatore al primo ed all'ultimo blocco del file in questione. Ogni blocco contiene un puntatore al blocco successivo. Quindi, volendo aggiungere dei dati ad un file, è sufficiente determinare un blocco libero tramite il sistema di gestione dello spazio libero su disco e concatenare il blocco alla fine del file, aggiornando i puntatori. In questo modo i file possono crescere fintanto che c'è dello spazio libero sul disco. Gli svantaggi sono che parte dello spazio di ogni blocco deve essere dedicato alla memorizzazione del puntatore al blocco successivo e che il sistema risulta efficiente solo per accessi di tipo sequenziale (in quanto, per accedere al blocco *i*-esimo, bisogna seguire la catena di puntatori). La traduzione da indirizzo logico (LA) ad indirizzo fisico si effettua quindi nel modo seguente: si divide LA per la dimensione del blocco diminuita dallo spazio occupato dal puntatore al blocco successivo. Il quoziente rappresenta la posizione occupata dal blocco nella lista concatenata, mentre il resto incrementato di uno rappresenta l'offset all'interno del blocco stesso.
- 1. Spiegare brevemente la differenza fra servizi di rete e servizi distribuiti.
 2. Descrivere cosa si intende per sistema multiprocessore UMA. È un sistema strettamente o debolmente accoppiato?

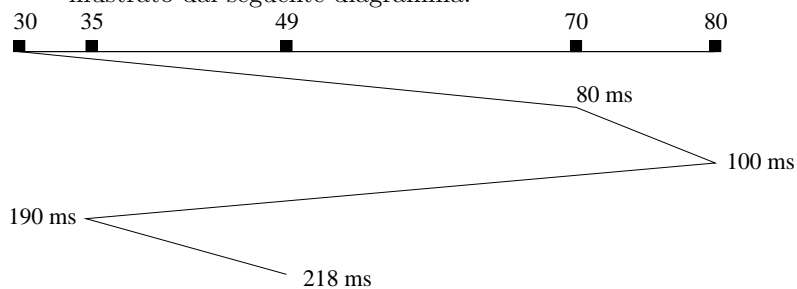
Risposta:

1. I servizi di rete offrono ai processi le funzionalità necessarie per stabilire e gestire le comunicazioni tra i nodi di un sistema distribuito (es.: l'interfaccia fornita dalle socket). In sostanza gli utenti devono essere consapevoli della struttura del sistema e devono indirizzare esplicitamente le singole macchine. I servizi distribuiti invece sono modelli comuni (paradigmi di comunicazione) trasparenti che offrono ai processi una visione uniforme, unitaria del sistema distribuito stesso (es: file system remoto). I servizi distribuiti vanno quindi a formare il cosiddetto *middleware*, ovvero, uno strato software fra il sistema operativo e le applicazioni che uniforma la visione dell'intero sistema.
 2. La sigla UMA significa Uniform Memory Access; un sistema multiprocessore UMA gode della proprietà che il tempo di accesso a qualunque locazione di memoria è sempre lo stesso (indipendentemente dal fatto che si faccia riferimento a memoria locale o di un altro nodo del sistema). Si tratta di un sistema strettamente accoppiato dato che la memoria è condivisa.
- (*) Si consideri un disco gestito con politica C-LOOK. 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 70, 80, 35, 49 come illustrato dal seguente diagramma:



2. Il tempo d'attesa medio è $\frac{(40*2-0)+(50*2-70)+((50+45)*2-50)+((50+45+14)*2-40)}{4} = \frac{80+30+140+178}{4} = 107 \text{ ms}$.

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