

Sistemi Operativi

20 giugno 2013

Compitino 2 A

Si risponda ai seguenti quesiti, giustificando le risposte.

Gli esercizi e le domande marcate con l'asterisco (*) devono essere svolti soltanto da chi ha in piano di studi l'esame di Sistemi Operativi da 9 o 12 CFU.

1. In un sistema che usa paginazione, l'accesso al TLB richiede 100ns, mentre l'accesso alla memoria richiede 450ns. Quando si verifica un page fault, si perdono 6ms per caricare la pagina che si sta cercando in memoria. Se il page fault rate è il 2% e il TLB hit il 80%, indicare l'EAT ai dati.

Risposta: La formula per il calcolo dell'Effective Access Time (EAT) è la seguente:

$$\begin{aligned} EAT &= t_{TLB} + \underbrace{\alpha \cdot t_{RAM}}_{\text{TLB hit}} + (1 - \alpha) \cdot \left[\underbrace{(1 - p) \cdot 2t_{RAM}}_{\text{pagina in memoria}} + \underbrace{p \cdot (2t_{RAM} + t_{LOAD})}_{\text{page fault}} \right] \\ &= t_{TLB} + \alpha \cdot t_{RAM} + (1 - \alpha) \cdot \overset{\text{TLB miss}}{[2t_{RAM} + p \cdot t_{LOAD}]} \\ &= t_{TLB} + \underbrace{t_{RAM}}_{\text{dati}} + (1 - \alpha) \cdot \left[\underbrace{t_{RAM}}_{\text{page table}} + p \cdot t_{LOAD} \right] \end{aligned}$$

dove:

- t_{TLB} è il tempo di accesso al TLB (100 ns = 10^{-4} ms),
- t_{RAM} è il tempo di accesso alla memoria (450 ns = $4,5 \cdot 10^{-4}$ ms),
- t_{LOAD} è il tempo di caricamento della pagina che si sta cercando in memoria (6 ms),
- α è la probabilità di un TLB hit (hit rate): 0,8,
- p è la probabilità di page fault (con necessità di caricare la pagina dal disco): 0,02.

Quindi, sostituendo i valori numerici, otteniamo quanto segue:

$$\begin{aligned} EAT &= 10^{-4} + 4,5 \cdot 10^{-4} + 0,2 \cdot [4,5 \cdot 10^{-4} + 0,02 \cdot 6] \\ &= 5,5 \cdot 10^{-4} + 0,2 \cdot [4,5 \cdot 10^{-4} + 0,12] \\ &= 5,5 \cdot 10^{-4} + 0,9 \cdot 10^{-4} + 0,024 \\ &= 6,4 \cdot 10^{-4} + 0,024 \\ &= 0,02464 \text{ ms} \end{aligned}$$

2. Descrivere la differenza fra le seguenti modalità di I/O:

1. Programmed I/O (PIO),
2. Interrupt-driven I/O,
3. DMA.

Risposta: 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.

Sistemi Operativi

20 giugno 2013

Compitino 2 A

3. Si spieghi cos'è un *file* e quali sono le principali operazioni su di esso che un sistema operativo deve implementare.

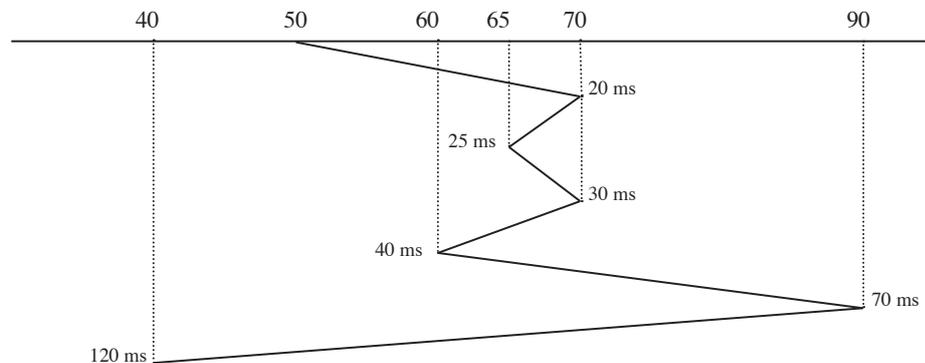
Risposta: un file è insieme di informazioni correlate a cui è stato assegnato un nome. Esso è inoltre la più piccola porzione unitaria di memoria logica secondaria allocabile dall'utente o dai processi di sistema. Le operazioni fondamentali sui file che un sistema operativo deve implementare sono le seguenti: creazione, cancellazione, apertura, chiusura, lettura, scrittura, append, riposizionamento (seek), troncamento, lettura/scrittura dei metadati.

4. Si consideri un disco gestito con politica SSTF. 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, 60, 40, rispettivamente agli istanti 0 ms, 20 ms, 30 ms, 50 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. Le richieste vengono soddisfatte nell'ordine: 65, 60, 90, 40, come risulta dal seguente diagramma:



2. Il tempo di attesa medio per le quattro richieste in oggetto è

$$\frac{(25-20)+(40-30)+(70-0)+(120-50)}{4} = \frac{5+10+70+70}{4} = \frac{155}{4} = 38,75 \text{ ms.}$$

5. Si descriva la tecnologia RAID in generale e si descriva in particolare il livello 1.

Risposta: l'architettura RAID è stata introdotta per migliorare l'affidabilità e le prestazioni del sistema dei dischi in un sistema di calcolo. Infatti, aumentando il numero di dischi fisici presenti in un calcolatore, diminuisce proporzionalmente il tempo medio fra un guasto e l'altro. Quindi il sistema RAID introduce degli schemi di ridondanza dei dati per consentire il funzionamento dei dischi anche in situazioni critiche. Al livello 1 (mirroring o shadowing) i contenuti di ogni disco sono interamente duplicati in un altro disco del sistema, garantendo un'eccellente resistenza ai crash, alte performance in lettura, ma non in scrittura (dove le prestazioni sono le stesse di un sistema non-RAID).

6. * In merito all'implementazione del file system, spiegare brevemente come funziona il meccanismo di allocazione *indicizzata* e indicare (giustificandone la ragione) se soffre o meno di frammentazione *esterna*.

Risposta: nel caso dell'allocazione indicizzata si mantengono tutti i puntatori ai blocchi di un file in una tabella indice, memorizzata a sua volta in un blocco su disco (blocco indice). In questo modo per accedere all'i-esimo blocco del file è sufficiente seguire l'i-esimo puntatore del blocco indice. Inizialmente (in fase di creazione del file) tutti gli indici sono inizializzati con un valore (-1) indicante che non vi è nessun blocco dati puntato. Per allocare un nuovo blocco al file, è sufficiente assegnare l'indirizzo di un blocco dati disponibile alla prima entry del blocco indice marcata con -1. Così, sacrificando un blocco per la tabella indice ed organizzando lo spazio in blocchi di dimensione uguale, è facile implementare l'accesso random ai file ed evitare il problema della frammentazione esterna.

Sistemi Operativi
20 giugno 2013
Compitino 2 A

Punteggi:

- 9/12 CFU: 1 (6 pt.), 2 (5 pt.), 3 (4 pt.), 4.1 (4 pt.), 4.2 (2 pt.), 5 (5 pt.), 6 (5 pt.)
- 6 CFU: 1 (7 pt.), 2 (6 pt.), 3 (5 pt.), 4.1 (5 pt.), 4.2 (2 pt.), 5 (6 pt.)