

Tra le ragioni per cui un sistema operativo può finire in una condizione di thrashing troviamo:

Scegli un'alternativa:

- ☒ a. un elevato numero di processi attivi rispetto alla dimensione della memoria principale, e l'adozione di una politica di allocazione globale o locale dei frame ✓
- ☐ b. un elevato numero di processi attivi rispetto alla dimensione della memoria principale, e l'adozione di un algoritmo di sostituzione delle pagine di tipo FIFO
- ☐ c. un elevato numero di processi CPU bound rispetto alla dimensione della memoria principale, e l'adozione di una politica di allocazione globale o locale dei frame
- ☐ d. un elevato numero di processi I/O bound rispetto alla dimensione della memoria principale, e l'adozione di una politica di allocazione globale o locale dei frame

Risposta corretta.

La risposta corretta è: un elevato numero di processi attivi rispetto alla dimensione della memoria principale, e l'adozione di una politica di allocazione globale o locale dei frame

In un sistema operativo che adotta uno scheduling senza diritto di prelazione, quattro processi arrivano al tempo indicato e consumano la quantità di CPU indicata nella tabella sottostante

Processo	T. di arrivo	Burst
Pa	0	5
Pb	2	3
Pc	4	2
Pd	6	1

Se si usa l'algoritmo di scheduling non preemptive che fornisce le migliori prestazioni possibili per schedulare i 4 processi in tabella:

il waiting time medio è: ✓

il turnaround medio è: ✓

il diagramma di GANTT è: ✓

l'algoritmo usato per le risposte precedenti può soffrire di starvation?

✓

In quale caso l'accesso in lettura ad un file memorizzato su un sistema RAID è più veloce che se il file fosse memorizzato su un normale hard disk?

Scegli un'alternativa:

- ☒ a. quando il file è memorizzato su due o più blocchi appartenenti a strip contenuti su dischi diversi del RAID ✓
- ☐ b. quando il file è memorizzato su uno più blocchi appartenenti a strip contenuti sullo stesso disco del RAID
- ☐ c. quando il RAID usato è di tipo 01/10, poiché in questo caso si possono sfruttare i dischi di mirroring
- ☐ d. sempre, dato che i sistemi RAID sono stati pensati proprio per fornire maggiore velocità di accesso ai file (oltre che maggiore affidabilità)

Risposta corretta.

La risposta corretta è: quando il file è memorizzato su due o più blocchi appartenenti a strip contenuti su dischi diversi del RAID

Secondo quanto visto a lezione, quale/quali dei seguenti comandi Unix modifica il valore del *link counter* dell'index-node associato al file cartella X? (si assuma di avere i permessi per eseguire tutti i comandi e di essere posizionati dentro a X, a sua volta contenuta in una generica cartella user/tmp)

- 1) ln ../X Z
- 2) mkdir K
- 3) ln X Y
- 4) ln -s ../X W

Scegli un'alternativa:

- ☒ a. il comando 2) ✓
- ☐ b. i comandi 1) e 2)
- ☐ c. i comandi 2) e 4)
- ☐ d. i comandi 2) 3) e 4)

In quale/i caso/i un processo in coda di ready viene fatto passare allo stato running?

Scegli un'alternativa:

- ☒ a. quando lo scheduler della CPU lo seleziona per entrare in esecuzione. Ad esempio, nel caso di FCFS, quando il processo si trova in cima alla coda di ready ✓
- ☐ b. quando il sistema operativo ha terminato le operazioni necessarie ad amministrare la vita del processo, ad esempio ha allocato il processo in RAM e inizializzato il suo PCB
- ☐ c. mai, è il sistema operativo che sposta i processi da uno stato all'altro.
- ☐ d. quando il processo è entrato in RQ arrivando da una coda di waiting, perché significa che è disponibile l'informazione che il processo stava attendendo, e la sua computazione può ripartire

Un sistema operativo può adottare una paginazione semplice oppure una paginazione a più livelli. Che implicazioni ha questa scelta?

Scegli un'alternativa:

- ☐ a. La paginazione a più livelli è una scelta obbligata quando le tabelle delle pagine possono raggiungere grandi dimensioni, per cui sarebbe difficile allocarle in uno spazio contiguo di RAM. D'altra parte la paginazione a più livelli è molto più vantaggiosa per la traduzione degli indirizzi da logici a fisici soprattutto in assenza di un TLB
- ☐ b. La paginazione a più livelli è una scelta obbligata in caso di spazi logici molto piccoli, per i quali la paginazione semplice produrrebbe troppo spreco di spazio in RAM. D'altra parte la paginazione semplice rende più efficiente la traduzione degli indirizzi da logici a fisici quando non si possa sfruttare il TLB
- ☐ c. La paginazione a più livelli è una scelta obbligata quando dobbiamo ricorrere ad una Tabella delle Pagine Invertita, che altrimenti sarebbe troppo difficile da allocare in RAM. D'altra parte la paginazione semplice rende più efficiente la traduzione degli indirizzi da logici a fisici quando non si possa sfruttare il TLB
- ☒ d. La paginazione a più livelli è una scelta obbligata quando le tabelle delle pagine possono raggiungere grandi dimensioni, per cui sarebbe difficile allocarle in uno spazio contiguo di RAM. D'altra parte la paginazione semplice rende più efficiente la traduzione degli indirizzi da logici a fisici quando non si possa sfruttare il TLB ✓

il problema della sezione critica può essere risolto in molti modi diversi, ma qualsiasi soluzione deve rispettare alcune condizioni fondamentali, che possono essere così descritte:

Scegli un'alternativa:

- ☒ a. quando un processo vuole entrare nella sua sezione critica deve riuscire a farlo in un tempo finito; non si possono mai trovare due o più processi contemporaneamente in una sezione critica; qualsiasi processo riuscirà prima o poi ad entrare in sezione critica in un tempo finito.
- ☐ b. quando un processo vuole entrare nella sua sezione critica deve riuscire a farlo in un tempo finito; non si possono mai trovare due o più processi contemporaneamente in una sezione critica; qualsiasi processo deve rimanere in sezione critica per un tempo finito.
- ☐ c. quando un processo è in sezione critica deve rimanerci per un tempo finito; non si possono mai trovare due o più processi contemporaneamente in una sezione critica; qualsiasi processo riuscirà prima o poi ad entrare in sezione critica in un tempo finito.
- ☐ d. quando un processo vuole entrare nella sua sezione critica deve riuscire a farlo in un tempo finito; nella sezione critica di deve essere sempre almeno un processo, e mai più di uno; qualsiasi processo riuscirà prima o poi ad entrare in sezione critica in un tempo finito. ✗

Di un sistema è noto che la tabella delle pagine più grande del sistema occupa esattamente 2 frame, il numero di un frame è scritto su 2 byte usando usando tutti i bit a disposizione, e nel sistema sono presenti in media 4 processi che insieme producono una frammentazione interna complessiva media di 4 Kilobyte.

lo spazio logico del sistema è grande:

4 Megabyte

 ✓

lo spazio fisico del sistema è grande:

128 Megabyte

 ✓

nessuno dei valori proposti

non si può ricavare dai dati del problema

8 Megabyte

64 Megabyte

2 Megabyte

256 Megabyte

Dopo l'esecuzione dei seguenti comandi in un ambiente Unix (come visti a lezione):

```
1: cd /tmp
2: mkdir newfolder
3: echo "ciao" > pippo // crea un nuovo file di nome pippo contenente la stringa ciao
4: cd newfolder
5: ln -s ../pippo paperino
6: ln -s /tmp/newfolder folder2
7: cp ../pippo topolino
8: echo "salve" >> topolino // aggiunge "salve" a fondo file
9: rm pippo
10: cat paperino // cat stampa il contenuto del file passato come argomento
11: mkdir folder3
```

Scegli un'alternativa:

- ☐ a.
 - 1. il link-counter dell'i-node di *topolino* è: 2
 - 2. il link counter di *newfolder* è: 3
 - 3. l'output del comando 10 è: "ciao" seguito da "salve"
 - 4. il link counter di tmp è: aumentato di 1
- ☐ b.
 - 1. il link-counter dell'i-node di *topolino* è: 1
 - 2. il link counter di *newfolder* è: 3
 - 3. l'output del comando 10 è: "ciao"
 - 4. il link counter di tmp è: aumentato di 2
- ☐ c.
 - 1. il link-counter dell'i-node di *topolino* è: 3
 - 2. il link counter di *newfolder* è: 3
 - 3. l'output del comando 10 è: "ciao" seguito da "salve"
 - 4. il link counter di tmp è: aumentato di 2
- ☒ d.
 - 1. il link-counter dell'i-node di *topolino* è: 1 ✓
 - 2. il link counter di *newfolder* è: 3
 - 3. l'output del comando 10 è: "ciao"
 - 4. il link counter di tmp è: aumentato di 1

In un sistema time sharing che usa un algoritmo di sostituzione delle pagine, tra le ragioni per cui si può verificare un context switch tra processi utente troviamo:

Scegli un'alternativa:

- ☐ a.
 1. il processo in CPU si addormenta in seguito all'esecuzione di una wait
 2. viene richiesta una operazione di I/O dal processo in CPU
 3. in una coda di wait entra un processo con priorità maggiore di quello che sta girando
 4. viene generata una trap dal processo in esecuzione
- ☒ b.
 1. il processo in CPU si addormenta in seguito all'esecuzione di una wait ✓
 2. in RQ entra un processo con priorità maggiore di quello che sta girando
 3. viene richiesta una operazione di I/O dal processo in CPU
 4. viene generata una trap dal processo in esecuzione
- ☐ c.
 1. il processo in CPU si addormenta in seguito all'esecuzione di una signal
 2. in RQ entra un processo con priorità maggiore di quello che sta girando
 3. viene richiesta una operazione di I/O dal processo in CPU
 4. viene generata una trap dal processo in esecuzione
- ☐ d.
 1. il processo in CPU si addormenta in seguito all'esecuzione di una wait
 2. in RQ entra un processo con priorità maggiore di quello che sta girando
 3. viene completata una operazione di I/O dal processo in CPU
 4. viene generata una trap dal processo in esecuzione