

**Domanda 1**

Risposta corretta

Punteggio ottenuto 2,0 su 2,0

In un sistema time-sharing che non usa un algoritmo di sostituzione delle pagine, un processo in esecuzione può passare ad uno stato di wait. Ad esempio perché:

Scegli un'alternativa:

- a. perché ha eseguito una wait su un semaforo con valore minore di zero oppure perché ha completato una operazione di I/O
- b. perché ha eseguito una wait su un semaforo con valore minore di zero oppure perché ha richiesto una operazione di I/O
- c. perché ha richiesto una operazione di I/O oppure perché ha indirizzato una pagina non presente in RAM
- d. perché ha completato una operazione di I/O oppure perché ha indirizzato una pagina non presente in RAM

**Domanda 2**

Risposta corretta    Punteggio ottenuto 2,0 su 2,0

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'algorithmo 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'algorithmo usato per le risposte precedenti può soffrire di starvation?

si

no

18/4

19/4

(0) ... Pa ... (5) ... Pd ... (6) ... Pc ... (8) ... Pb ... (11)

no

(0) ... Pa ... (5) ... Pc ... (7) ... Pd ... (8) ... Pb ... (11)

2

3

**Domanda 3**

Risposta corretta

Punteggio ottenuto 2,0 su 2,0

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

- 1) `ln X ../X`
- 2) `cp X Y`
- 3) `ln X X`
- 4) `rm X Y`

Scegli un'alternativa:

- a. i comandi 2) e 3)
- b. i comandi 1) e 3)
- c. i comandi 3) e 4)
- d. i comandi 1) e 4)

**Domanda 4**

Risposta corretta

Punteggio ottenuto 2,0 su 2,0

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:

lo spazio fisico del sistema è grande:

☐ 4 Megabyte☐ non si può ricavare dai dati del problema☐ nessuno dei valori proposti☐ 8 Megabyte☐ 2 Megabyte☐ 128 Megabyte☐ 64 Megabyte☐ 256 Megabyte

**Domanda 5**

Risposta corretta

Punteggio ottenuto 2,0 su 2,0

del codice statico e del codice staticamente rilocabile possiamo dire che:

Scegli un'alternativa:

- a. nei sistemi operativi moderni non vengono usati perché renderebbero impossibile l'implementazione della memoria virtuale
- b. nei sistemi operativi moderni non vengono usati perché per girare in modo efficiente richiederebbero dell'hardware specifico che ne limiterebbe la portabilità
- c. nei sistemi operativi moderni non vengono usati perché non permetterebbero una implementazione efficiente della paginazione della memoria
- d. nei sistemi operativi moderni non vengono usati perché producono codice molto meno efficiente di quello dinamicamente rilocabile

**Domanda 6**

Risposta errata

Punteggio ottenuto 0,0 su 2,0

nel sistema operativo Unix un link simbolico:

Scegli un'alternativa:

- a. viene implementato allocando un nuovo index node in cui viene scritto il pathname (assoluto o relativo che sia) usato come primo argomento del comando "ln -s".
- b. è l'unico tipo di link implementabile fra cartelle, dato che se fossero permessi i link fisici si potrebbero creare cicli all'interno del file system
- c. è una sorta di puntatore ad un file (regolare o cartella che sia), ma di tipo particolare, infatti la rimozione del puntatore comporta anche la rimozione del file
- d. permette di creare collegamenti veloci e che non occupano spazio aggiuntivo sull'hard disk a file e cartelle di cui si deve fare un uso frequente

**Domanda 7**

Risposta corretta

Punteggio ottenuto 2,0 su 2,0

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 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
  
- 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 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
  
- 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

**Domanda 8**

Risposta corretta

Punteggio ottenuto 2,0 su 2,0

In un sistema operativo moderno che implementa la memoria virtuale, quando si verifica un page fault e non c'è spazio in RAM occorre scegliere una pagina vittima. Dei criteri di scelta visti a lezione possiamo dire che:

Scegli un'alternativa:

- a. scegliere la pagina che è stata riferita da più tempo è un ottimo criterio, ma all'atto pratico non si può implementare in modo efficiente
- b. scegliere una pagina che è stata modificata di recente è un buon criterio, nel caso in cui ci sia bisogno di salvare periodicamente lo stato della computazione del processo
- c. scegliere la pagina che verrà riferita più in là nel tempo è sicuramente il criterio migliore, ma richiede hardware dedicato offerto solo dalle CPU di fascia alta
- d. scegliere la pagina che è entrata in RAM da più tempo è un buon criterio, e infatti viene adottato da molti sistemi operativi



## Domanda 9

Risposta corretta

Punteggio ottenuto 2,0 su 2,0

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 ../pippo paperino
6: ln ../newfolder folder2
7: ln -s paperino 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 *paperino* è: 2
  - 2. il link counter di *tmp* è: aumentato di 2
  - 3. l'output del comando 10 è: "ciao" seguito da "salve"
  - 4. il comando 6 da come risultato: un errore perché non sono ammessi hard link tra cartelle
- b.
  - 1. il link-counter dell'i-node di *paperino* è: 1
  - 2. il link counter di *tmp* è: aumentato di 2
  - 3. l'output del comando 10 è: "ciao"
  - 4. il comando 6 da come risultato: un nuovo collegamento alla cartella newfolder
- c.
  - 1. il link-counter dell'i-node di *paperino* è: 2
  - 2. il link counter di *tmp* è: aumentato di 1
  - 3. l'output del comando 10 è: "ciao" seguito da "salve"
  - 4. il comando 6 da come risultato: un errore perché non sono ammessi hard link tra cartelle
- d.
  - 1. il link-counter dell'i-node di *paperino* è: 2
  - 2. il link counter di *tmp* è: 2
  - 3. l'output del comando 10 è: no such file or directory
  - 4. il comando 6 da come risultato: un errore perché non sono ammessi hard link tra cartelle

**Domanda 10**

Risposta corretta

Punteggio ottenuto 2,0 su 2,0

il problema della sezione critica può essere risolto in molti modi diversi, ma alcune soluzioni sono migliori di altre. Una qualità auspicabile per una corretta soluzione al problema della sezione critica è:

Scegli un'alternativa:

- a. che due processi possano stare contemporaneamente in sezione critica se devono solo leggere e non modificare le variabili condivise della sezione critica
- b. che il processo che sta uscendo dalla sezione critica possa decidere quale processo può prendere il suo posto, in modo da evitare situazioni di deadlock
- c. che garantisca che il processo in sezione critica esca dalla sezione critica in un tempo finito, in modo che gli altri processi non soffrano di starvation
- d. che non sia basata sul busy waiting, in modo che un processo che cerca di entrare in una sezione critica occupata da un altro processo non sprechi tutto il suo quanto di tempo