

Iniziato	giovedì, 12 febbraio 2026, 12:11
Stato	Completato
Terminato	giovedì, 12 febbraio 2026, 13:11
Tempo impiegato	59 min. 51 secondi
Valutazione	20,67 su un massimo di 30,00 (68,89%)

Domanda 1

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

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

Risposta corretta.

La risposta corretta è: perché ha eseguito una wait su un semaforo con valore minore di zero oppure perché ha richiesto una operazione di I/O

Domanda 2

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

Su un hard disk che adotta una allocazione concatenata (senza FAT) è memorizzato un file A della dimensione di 0x8000 byte, e si sa che nell'ultimo blocco di A sono presenti 32 byte del file. Si sa inoltre che per scrivere il numero di un blocco vengono usati 28 bit, arrotondati al minimo numero di byte necessario. Quanto è grosso l'hard disk?

Scegli un'alternativa:

- a. 512 Gigabyte
- b. 1 Terabyte ✓
- c. 2 Terabyte
- d. 4 Terabyte

Risposta corretta.

La risposta corretta è: 1 Terabyte

Domanda 3

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

Gli algoritmi di sostituzione delle pagine forniscono ovviamente prestazioni diverse, anche in base alla particolare sequenza di riferimento su cui devono operare. Alcuni degli algoritmi visti a lezione poi, hanno anche comportamenti anomali e controidintuitivi. In particolare:

Scegli un'alternativa:

- a. per alcuni di questi algoritmi si è osservato che, usando una sostituzione globale delle pagine (cioé il numero di frame assegnati al processo può variare nel tempo), per alcune specifiche sequenze di riferimento, aumentando il numero di frame assegnati ad un processo, il numero di page fault generati dagli altri processi aumenta
- b. per alcuni di questi algoritmi si è osservato che, usando una sostituzione globale delle pagine (cioé il numero di frame assegnati al processo può variare nel tempo), per qualsiasi sequenza di riferimento, aumentando il numero di frame assegnati ad un processo, il numero di page fault generati dagli altri processi aumenta
- c. per alcuni di questi algoritmi si è osservato che, usando una sostituzione locale delle pagine (cioé assegnando un numero fisso di frame al processo), per qualsiasi sequenza di riferimento, aumentando il numero di frame assegnati, il numero di page fault non diminuisce
- d. per alcuni di questi algoritmi si è osservato che, usando una sostituzione locale delle pagine (cioé assegnando un numero fisso di frame al processo), per alcune specifiche sequenze di riferimento, aumentando il numero di frame assegnati, il numero di page fault aumenta anziché rimanere uguale o diminuire

Risposta corretta.

La risposta corretta è: per alcuni di questi algoritmi si è osservato che, usando una sostituzione locale delle pagine (cioé assegnando un numero fisso di frame al processo), per alcune specifiche sequenze di riferimento, aumentando il numero di frame assegnati, il numero di page fault aumenta anziché rimanere uguale o diminuire

Domanda 4

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

In hard disk grande 512 Gigabyte, per scrivere il numero di un blocco vengono usati 28 bit, arrotondati al minimo numero di byte necessario. L'hard disk adotta una allocazione indicizzata semplice, e di un file A si sa che nel suo blocco indice 28 byte vengono usati per tenere traccia dei blocchi di dati di A. Quanto può essere grande al massimo A?

Scegli un'alternativa:

- a. 14 Kilobyte ✓
- b. 10 Kilobyte
- c. 16 Kilobyte
- d. 12 Kilobyte

Risposta corretta.

La risposta corretta è: 14 Kilobyte

Domanda 5

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

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	8
Pb	2	8
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?

✓

Risposta corretta.

La risposta corretta è:

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	8
Pb	2	8
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: [4]

il turnaround medio è: [35/4]

il diagramma di GANTT è: [(0) ... Pa ... (8) ... Pd ... (9) ... Pc ... (11) ... Pb ... (19)]

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

Domanda 6

Risposta non data

Punteggio max.: 2,00

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 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
- c. 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
- d. è l'unico tipo di link implementabile fra cartelle, dato che se fossero permessi i link fisici si potebbero creare cicli all'interno del file system

Risposta errata.

La risposta corretta è: 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".

Domanda 7

Parzialmente corretta

Punteggio ottenuto 1,67 su 2,00

si consideri l'esecuzione della seguente porzione di codice che utilizza la system call fork:

```
int a, b, c, d, n, pid1, pid2, pid3;
a = 60, b = 70, c = 80, d = 90;
n = fork();
if ( n == 0)
    {a = 65; b = 75;
    pid1 = getppid();
    printf("%d", pid1);
    exit(0);}
else
    {c = 85; d = 95;
    pid2 = getpid();
    printf("%d", pid2);
    pid3 = wait(NULL);
    exit(0);}
```

il valore della variabile a vista dal processo figlio subito prima della sua exit è ✓

il valore della variabile c vista dal processo figlio subito prima della sua exit è ✓

il valore della variabile b vista dal processo padre subito prima della sua exit è: ✓

il valore della variabile d vista dal processo padre subito prima della sua exit è: ✓

all'esecuzione delle due printf vale la seguente relazione: pid1 ✓ pid2

del risultato della wait possiamo dire che: pid2 ✗ pid3

Risposta parzialmente esatta.

Hai selezionato correttamente 5.

La risposta corretta è:

si consideri l'esecuzione della seguente porzione di codice che utilizza la system call fork:

```
int a, b, c, d, n, pid1, pid2, pid3;
a = 60, b = 70, c = 80, d = 90;
n = fork();
if ( n == 0)
    {a = 65; b = 75;
    pid1 = getppid();
    printf("%d", pid1);
    exit(0);}
else
    {c = 85; d = 95;
    pid2 = getpid();
    printf("%d", pid2);
    pid3 = wait(NULL);
    exit(0);}
```

il valore della variabile a vista dal processo figlio subito prima della sua exit è [65]

il valore della variabile c vista dal processo figlio subito prima della sua exit è [80]

il valore della variabile b vista dal processo padre subito prima della sua exit è: [70]

il valore della variabile d vista dal processo padre subito prima della sua exit è: [95]

all'esecuzione delle due printf vale la seguente relazione: pid1 [=] pid2

del risultato della wait possiamo dire che: pid2 [<] pid3

Domanda 8

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

In un sistema operativo un indirizzo fisico è scritto su 26 bit, l'offset più grande in una pagina è FFFF, lo spazio logico è il doppio di quello fisico, e nel sistema possono essere presenti contemporaneamente al massimo 256 processi.

Se il sistema adottasse una Inverted Page Table, quanto sarebbe grande questa tabella? (selezionate l'opzione di risposta che riporta il ragionamento aritmetico e il risultato corretti)

Scegli un'alternativa:

- a. Ogni entry della IPT è grande 10 bit, arrotondati a 2 byte, e dunque la IPT sarà grande $2 * 2^{11} = 4$ Kbyte (circa)
- b. Ogni entry della IPT è grande 10 bit, arrotondati a 2 byte, e dunque la IPT sarà grande $3 * 2^{10} = 2$ Kbyte (circa)
- c. Ogni entry della IPT è grande 19 bit, arrotondati a 3 byte, e dunque la IPT sarà grande $3 * 2^{11} = 6$ Kbyte (circa)
- d. Ogni entry della IPT è grande 19 bit, arrotondati a 3 byte, e dunque la IPT sarà grande $3 * 2^{10} = 3$ Kbyte (circa) ✓

Risposta corretta.

La risposta corretta è: Ogni entry della IPT è grande 19 bit, arrotondati a 3 byte, e dunque la IPT sarà grande $3 * 2^{10} = 3$ Kbyte (circa)

Domanda 9

Risposta errata

Punteggio ottenuto 0,00 su 2,00

GESTIONE DEI PROCESSI

Si consideri questa variante del problema dei produttori e consumatori:

Semaphore mutex = 1; full = 0; empty = N;

codice consumatore:

```
repeat
  wait(empty);
  <preleva dato dal buffer>
  signal(full)
  <consuma dato>
forever
```

codice produttore:

```
repeat
  <producি dato>
  wait(full)
  wait(mutex);
  <inserisci dato nel buffer>
  signal(mutex);
  signal(empty)
forever
```

Scegli un'alternativa:

- a. La soluzione proposta funziona a condizione che i consumatori accedano al buffer condiviso i momenti diversi, visto che il prelievo di un dato non è protetto dal semaforo di mutua esclusione
- b. La soluzione proposta non funziona in nessun caso, perché i semafori coinvolti nelle operazioni di sincronizzazione non sono usati in modo corretto
- c. La soluzione proposta funziona, e corrisponde al caso in cui è presente un solo consumatore, visto che il prelievo di un dato non è protetto dal semaforo di mutua esclusione
- d. La soluzione proposta non funziona, visto che il prelievo di un dato non è protetto dal semaforo di mutua esclusione e quindi due consumatori potrebbero prelevare contemporaneamente lo stesso dato

✗

Risposta errata.

La risposta corretta è: La soluzione proposta non funziona in nessun caso, perché i semafori coinvolti nelle operazioni di sincronizzazione non sono usati in modo corretto

Domanda 10

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

Con il termine thrashing intendiamo:

Scegli un'alternativa:

- a. un problema dei sistemi operativi che implementano la memoria virtuale in cui i processi hanno a disposizione poco spazio in RAM e passano più tempo in coda di ready che a portare avanti la loro computazione
- b. un problema dei sistemi operativi che implementano la memoria virtuale in cui i processi hanno a disposizione poco spazio in RAM e passano più tempo a generare page fault che a portare avanti la loro computazione ✓
- c. un problema dei sistemi operativi che non implementano la paginazione della memoria in cui i processi hanno a disposizione poco spazio in RAM e passano più tempo a generare page fault che a portare avanti la loro computazione
- d. un problema dei sistemi operativi che implementano la memoria virtuale in cui i processi hanno a disposizione poco spazio nell'area di swap e passano più tempo a generare page fault che a portare avanti la loro computazione

Risposta corretta.

La risposta corretta è: un problema dei sistemi operativi che implementano la memoria virtuale in cui i processi hanno a disposizione poco spazio in RAM e passano più tempo a generare page fault che a portare avanti la loro computazione

Domanda 11

Risposta errata

Punteggio ottenuto 0,00 su 2,00

Un sistema ha un tempo di accesso in RAM di 70 ns, adotta un TLB con un tempo di accesso di 10 ns e un hit rate del 95%, e usa un algoritmo di rimpiazzamento delle pagine. Quando si verifica un hit la pagina indirizzata è sicuramente in RAM. Quando si verifica un miss, nel 20 % dei casi la pagina indirizzata non è in RAM e il page fault ha un costo totale di gestione di 1 microsecondo, indipendentemente dal valore del dirty bit. Qual è l'effective access time (eat) del sistema? (per semplicità in caso di miss si ignori il costo di interrogazione del TLB, e in caso di page fault si consideri solo il tempo di gestione del page fault)

Scegli un'alternativa:

- a. $mat = 98,6$ ns
- b. $mat = 83,6$ ns ✗
- c. $mat = 91,6$ ns
- d. $mat = 80,6$ ns

Risposta errata.

La risposta corretta è: $mat = 91,6$ ns

Domanda 12

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

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: cp 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: 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*
- b. 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
- c. 1. il link-counter dell'i-node di *paperino* è: 1
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
- d. 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"
4. il comando 6 da come risultato: un errore perché non sono ammessi hard link tra cartelle



Risposta corretta.

La risposta corretta è:

- 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"
- 4. il comando 6 da come risultato: un errore perché non sono ammessi hard link tra cartelle

Domanda 13

Risposta errata

Punteggio ottenuto -1,00 su 2,00

Riguardo ai link fisici nell'ambiente Unix, secondo quanto visto a lezione possiamo dire che:

Scegli un'alternativa:

- a. permettono un accesso più veloce ai file e occupano meno spazio dei link simbolici, e sono ammessi solo tra file regolari perché tra cartelle sono ammessi solo i link simbolici
- b. permettono un accesso più lento ai file e occupano più spazio dei link simbolici, e sono ammessi sia tra file regolari che tra cartelle, anche se con alcune restrizioni
- c. permettono un accesso più veloce ai file e occupano meno spazio dei link simbolici, e sono ammessi sia tra file regolari che tra cartelle, anche se con alcune restrizioni
- d. permettono un accesso più lento ai file e occupano più spazio dei link simbolici, e sono ammessi solo tra file regolari perché tra cartelle sono ammessi solo i link simbolici ×

Risposta errata.

La risposta corretta è: permettono un accesso più veloce ai file e occupano meno spazio dei link simbolici, e sono ammessi sia tra file regolari che tra cartelle, anche se con alcune restrizioni

Domanda 14

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

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. viene eseguita una wait su un semaforo con valore positivo dal processo in CPU
2. è stato terminato il processo running perché ha cercato di usare una istruzione privilegiata
3. il processo in esecuzione ha indirizzato una pagina non presente in RAM
4. il processo running termina di eseguire il suo codice
- b. 1. è stato terminato il processo running perché ha cercato di usare una istruzione privilegiata ✓
2. il processo in esecuzione ha indirizzato una pagina non presente in RAM
3. viene eseguita una wait su un semaforo con valore negativo dal processo in CPU
4. il processo running termina di eseguire il suo codice
- c. 1. viene eseguita una wait su un semaforo con valore negativo dal processo in CPU
2. è stato rimesso in RQ un processo che ha cercato di usare una istruzione privilegiata
3. il processo in esecuzione ha indirizzato una pagina non presente in RAM
4. il processo running termina di eseguire il suo codice
- d. 1. viene eseguita una wait su un semaforo con valore negativo dal processo in CPU
2. è stato terminato il processo running perché ha cercato di usare una istruzione privilegiata
3. è stato rimesso in RQ un processo che ha cercato di usare una istruzione privilegiata
4. il processo in coda di ready termina di eseguire il suo codice

Risposta corretta.

La risposta corretta è:

- 1. è stato terminato il processo running perché ha cercato di usare una istruzione privilegiata
- 2. il processo in esecuzione ha indirizzato una pagina non presente in RAM
- 3. viene eseguita una wait su un semaforo con valore negativo dal processo in CPU
- 4. il processo running termina di eseguire il suo codice

Domanda 15

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

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 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: ✓

Risposta corretta.

La risposta corretta è:

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 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]