

Iniziato	venerdì, 6 settembre 2024, 08:36
Stato	Completato
Terminato	venerdì, 6 settembre 2024, 09:12
Tempo impiegato	36 min. 39 secondi
Valutazione	18,60 su un massimo di 20,00 (93%)

Domanda 1

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

L'algoritmo di seconda chance percorre la lista dei frame allocati a un processo e, quando ne incontra uno che ha bit di riferimento pari a 1, setta il bit a 0 e lo usa per caricare la nuova pagina

Scegli una risposta:

- Vero
 Falso ✓

La risposta corretta è 'Falso'.

Domanda 2

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

Nel problema dei "lettori e scrittori" è richiesto che tutti i processi coinvolti accedano in mutua esclusione alle risorse condivise

Scegli una risposta:

- Vero
 Falso ✓

La risposta corretta è 'Falso'.

Domanda 3

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

Il meccanismo di "copiatura su scrittura" è efficace contro la starvation

Scegli una risposta:

- Vero
 Falso ✓

La risposta corretta è 'Falso'.

Domanda 4

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

Il buffer cache permette di rendere più veloce l'esecuzione dei processi mediante un meccanismo di write ahead

Scegli una risposta:

 Vero ✓ Falso

La risposta corretta è 'Vero'.

Domanda 5

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

Associa i seguenti nomi di algoritmi alla loro funzione:

namei

ricerca di un numero di inode



algoritmo ottimale

sostituzione di pagine



C-LOOK

scheduling del disco



protocollo basato su timestamp

transazioni concorrenti asincrone



Risposta corretta.

La risposta corretta è:

namei → ricerca di un numero di inode,

algoritmo ottimale → sostituzione di pagine,

C-LOOK → scheduling del disco,

protocollo basato su timestamp → transazioni concorrenti asincrone

Domanda 6

Parzialmente corretta

Punteggio ottenuto 1,60 su 2,00

Indica se le seguenti affermazioni inerenti lo scheduling della CPU sono vere o false:

L'aging permette di evitare la starvation

VERO



shortest jobs first è un algoritmo a priorità

VERO



Il tempo medio di attesa è minimo quando la politica è shortest jobs first

VERO



Lo scheduling first come first served soffre di starvation

FALSO



Round Robin è particolarmente adatto a sistemi in real time

VERO



Risposta parzialmente esatta.

Hai selezionato correttamente 4.

La risposta corretta è:

L'aging permette di evitare la starvation → VERO,

shortest jobs first è un algoritmo a priorità → VERO,

Il tempo medio di attesa è minimo quando la politica è shortest jobs first → VERO,

Lo scheduling first come first served soffre di starvation → FALSO,

Round Robin è particolarmente adatto a sistemi in real time → FALSO

Domanda 7

Risposta corretta

Punteggio ottenuto 2,00 su 2,00

Si consideri un sistema con TLB, in cui ciascun accesso alla RAM richiede 150 unità di tempo, l'hit ratio è 95% e ciascun accesso al TLB richiede 50 unità di tempo. Dire se le seguenti affermazioni sono vere o false:

in caso di TLB hit l'accesso al dato richiede 200 unità di tempo

VERO



il TLB miss è pari al 10%

FALSO



in caso di TLB miss l'accesso al dato richiede 300 unità di tempo

VERO



in caso di TLB hit l'accesso al dato richiede 300 unità di tempo

FALSO



in caso di TLB miss l'accesso al dato richiede 200 unità di tempo

FALSO



Risposta corretta.

La risposta corretta è:

in caso di TLB hit l'accesso al dato richiede 200 unità di tempo → VERO,

il TLB miss è pari al 10% → FALSO,

in caso di TLB miss l'accesso al dato richiede 300 unità di tempo → VERO,

in caso di TLB hit l'accesso al dato richiede 300 unità di tempo → FALSO,

in caso di TLB miss l'accesso al dato richiede 200 unità di tempo → FALSO

Domanda 8

Completo

Punteggio ottenuto 5,00 su 5,00

1) spiegare cos'è il vettore delle interruzioni e come venga usato per gestire gli eventi e 2) spiegare in quali circostanze occorre un evento di page fault

il vettore delle interruzioni è un vettore che contiene gli indirizzi degli event handler. quando si verifica un evento (interrupt) il dispatcher riceve questo evento lo riconosce e utilizza l'id dell'evento per trovare nel vettore delle interruzioni (il quale utilizza gli id come indice) l'indirizzo del giusto handler il quale è una serie di procedure da eseguire per gestire correttamente l'evento ricevuto. un esempio di interrupt può essere il page fault. il page fault avviene quando ad un processo serve un pezzo di codice, una variabile o altro ma la pagina su cui troviamo questo dato non è contenuta in ram al momento quindi bisogna caricare la pagina richiesta dal processo che in questo momento si trova in memoria secondaria e metterla in ram, se non c'è spazio bisogna trovare una pagina vittima da poter spostare in memoria secondaria per far spazio alla pagina che serve, ci sono diversi algoritmi per farlo come ad esempio l'algoritmo di seconda chance o last recently used LRU.

Commento:

Domanda 9

Completo

Punteggio ottenuto 4,00 su 5,00

1) spiegare cosa sono e dove sono memorizzati gli INODE e 2) spiegare l'allocazione concatenata dei blocchi ai file

gli INODE o FCB (file control block) sono degli oggetti salvati in memoria secondaria che contengono principalmente i metadati dei file come ad esempio id del proprietario, tipo di file, dimensioni del file, ultima data e ora di accesso e di modifica ecc. quando un file viene aperto il suo fcb viene aggiunto alla lista dei file control block che serve al sistema operativo per controllare quanti e quali file sono al momento aperti. se un file è in utilizzo da un processo viene caricato in ram anche il suo inode diventando un in-core inode il quale viene arricchito di altre informazioni utili al processo. per risalire ad un inode di un file si utilizza l'algoritmo namei che prende in input il path per arrivare al file.

i file possono essere salvati in memoria in diversi modi uno di questi è l'allocazione concatenata dei blocchi. praticamente il file è suddiviso a blocchi collegati l'uno con l'altro possono essere anche non contigui ma avere blocchi di altri file tra i propri.

Commento:

1) ok

2) risposta un po' approssimativa, servirebbe qualche dettaglio in più