

Sistemi Operativi 2016.07.21 (computer dispari)

Home ► Courses ► Miscellaneous ► SODispari20160721 ► General ► Compito dispari - turno 1 ► Preview

Quiz navigation

1 2 3 4 5 6

7

Finish attempt ...

Start a new preview

Administration

- Quiz administration
- Edit settings
 - Group overrides
 - User overrides
 - Edit quiz
 - Preview**
 - Results
 - Locally assigned roles
 - Permissions
 - Check permissions
 - Filters
 - Logs
 - Backup
 - Restore
 - Question bank

Course administration

Switch role to...

My profile settings

Site administration

Search

Question 1

Not yet answered

Not graded

Flag question

Edit question

Dati studente

Inserisci qui i tuoi dati, compila subito questa parte.

Quanti CFU?

☒ 5 cfu ☐ 6 cfu

Cognome

Nome

Matricola

email

Numero Computer

Ordinamento (509, 270, erasmus,ecc.)

Question 2

Not yet answered

Not graded

Flag question

Edit question

Memory management

Rispondi brevemente, ma punto per punto, alle seguenti domande.

1. Cosa è il page buffer?
2. Che obiettivi si prefigge?
3. Qual'è il ruolo dell page buffer nella risoluzione dei page fault?
4. Descrivi due situazioni diverse in cui il page buffer deve cedere un frame/pagina.

Paragraph

1.

2.

3.

4.

Path: p

Question 3

Not yet answered

Not graded

Flag question

Edit question

RAID

Supponi che un disco di marca X abbia un Mean Time Between Failures (MTBF, cioè il valore atteso del tempo di guasto) pari a T.

1. Immagina di avere un array RAID 0 composto da 3 dischi X . Qual'è il suo MTBF? perché?
2. Immagina di avere un array RAID 5 composto da 4 dischi X. Qual'è il tempo medio (nel senso di valore atteso) per osservare l'array che va in stato degradato? perché?
3. Supponi che l'array RAID 5 in un certo istante vada in stato degradato e che non venga ripristinato il disco rotto. Qual'è il tempo medio (nel senso di valore atteso) che deve trascorrere, a partire da quando va in stato degradato, per osservare l'array che va in stato guasto? perché?

Paragraph

1.

2.

3.

Path: p

Question 4

Not yet answered
Not graded

Flag question
Edit question

Sequenza degli eventi nel SO

In un sistema sono presenti tre processi: A, B, C. La politica di scheduling è **preemptive con priorità (A>B>C)**.

- **A** è inizialmente running, a t=40ms effettua una lettura da disco che sarà servita in 40ms, poi diventa i/o bound, ciascun i/o servito in 40ms.
- **B** è inizialmente in blocco a causa di un page fault che verrà servito a t=20, poi è cpu bound ma genera un page fault ogni 30ms di esecuzione che saranno ciascuno in 40ms.
- **C** è inizialmente ready è cpu bound e non genera page faults.

Il processore esegue di volta in volta A, B, C, e inoltre, con tempi trascurabili, mode switching, dispatching, system call e interrupt handlers. Mostra schematicamente, nella seguente tabella, l'ordine con cui tali attività vengono eseguite (una sola croce per ciascuna colonna). Indica anche quali processi sono running, quali ready e quali bloccati in ciascun istante come indicato nell'esempio.

| | | | | | | | | | |
|----------------|--------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| user mode | A | X | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | B | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | C | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| mode switch | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| kernel mode | sched./disp. | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | system call | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | interrupt handler per page fault | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | interrupt handler per I/O | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | interrupt handler per quanto scaduto | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| stati processi | running | A | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | ready | C | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | block | B | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| note tempi | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| altre note | | <input type="text"/> | | | | | | | |

Question 5

Not yet answered
Not graded

Flag question
Edit question

Espressioni Regolari

Nel file di testo [divina.txt](#), è contenuto il testo del primo libro della Divina Commedia (l'Inferno). Considera ciascuna riga divisa in *parole* separate da spazi.

Scrivere un comando linux che **restituisca** tutte le righe per cui

- la **prima parola** contenga esattamente **3 caratteri**,
- primo carattere** sia la lettera "N" (enne maisucola) ,
- la riga **finisca** per uno dei seguenti caratteri di punteggiatura: i due punti (:), la virgola (,) o il punto e virgola (;).

Inoltre le righe così selezionate dovranno essere ordinate (in ordine lessicografico) rispetto alla seconda parola.

Ad esempio:

*Non parrebbe di là poi meraviglia,
Non per aver a sé di bene acquisto,
Noi procedemmo più avanti allotta,
Noi salavam per una pietra fessa,
Noi siam di voglia a muoverci sì pieni,
Non ti dee oramai parer più forte,
...*

Fai copia e incolla nella form del comando e del suo output

Paragraph

Fai copia e incolla nella form del comando e del suo output

Path: p » em » span.s1

Question 6

Not yet answered
Not graded

Flag question
Edit question

Si consideri l'output del comando **ps aux**. Si scriva un comando, una pipeline di comandi, o uno script che sull'output dell'esecuzione di **ps aux**, per ogni utente, calcoli il massimo delle percentuali di cpu e di memoria utilizzata per i processi di tale utente.

Il comando **ps aux** ritorna un record in cui le informazioni richieste sono indicate sulle colonne "USER", "%CPU" e "%MEM" .

Ad esempio, si ipotizzi il seguente output al comando **ps aux**:

| USER | PID | %CPU | %MEM | VSZ | RSS | TT | STAT | STARTED |
|---------------|------|------|------|---------|---------|----|------|---------|
| TIME COMMAND | | | | | | | | |
| rdevirgilio | 5134 | 8,4 | 6,7 | 4569472 | 562076 | ?? | S | 7:02pm |
| 6:28.95 / | | | | | | | | |
| rdevirgilio | 1699 | 1,8 | 22,6 | 4464316 | 1899036 | ?? | S | Lun12pm |
| 35:38.65 / | | | | | | | | |
| _windowserver | 194 | 0,7 | 0,7 | 3681072 | 57756 | ?? | Ss | Sab12pm |
| 85:38.60 / | | | | | | | | |
| rdevirgilio | 1633 | 0,5 | 0,3 | 2684992 | 26028 | ?? | S | Lun12pm |
| 0:27.03 / | | | | | | | | |
| rdevirgilio | 4019 | 0,4 | 3,5 | 4450064 | 292544 | ?? | S | Mer02pm |
| 28:21.75 / | | | | | | | | |
| rdevirgilio | 363 | 0,2 | 0,2 | 2572088 | 17492 | ?? | S | Sab12pm |
| 7:22.73 / | | | | | | | | |
| root | 5435 | 0,0 | 0,1 | 2496840 | 4488 | ?? | Ss | 10:07pm |
| 0:00.05 / | | | | | | | | |

Il risultato del comando o script sarà:

rdevirgilio 8,4 22,6
_windowserver 0,7 0,7
root 0,0 0,1

Fai copia e incolla del comando/srcipt e del suo output nella form.

Paragraph

Fai copia e incolla del comando/script e del suo output nella form.

Path: p » em » span.s1

Question 7

Not yet answered

Not graded

Flag question

Edit question

Debugging

Considera il codice del seguente file [divisione.c](#) e compila tale file con il comando
`gcc -g -o divisione divisione.c`

Non ci sono errori (solo alcuni warning) nella compilazione, ma eseguendo il programma con input "8 9" (otto spazio nove) l'esecuzione termina improvvisamente.

pizzonia@pisolo\$./divisione

Floating point exception (core dumped)

Esegui il debugger tramite il comando

`gdb test`

ed esegui i seguenti passi nell'ambiente di debugging nell'ordine specificato

- Inserisci un breakpoint su "main"
- fai "single step" fino alla chiamata di "div" e entra dentro
- continua (senza single step!) finché il programma non dà errore
- stampa i valori di x e y nel contesto di div
- mostra lo stack
- stampa i valori di x e y nel contesto di main

Paragraph

Path: p

Next

[Moodle Docs for this page](#)

You are logged in as [Maurizio Pizzonia](#) ([Log out](#))

S0dispari20160721