Sistemi Operativi 2017.06.26 (computer pari)

Quiz navigation	Question 1		
1 2 3 4 5 6	Not yet answered Not graded		Dati studente
7 Finish attempt	Flag question	Inserisci qu	i tuoi dati, compila subito questa par
Start a new preview		Cognome	
		Nome	
		Matricola	
		email	
		Numero Computer	
		Ordinamento (509, 270, erasmus,ecc.)	
		Quanti CFU?	○ 5 cfu ○ 6 cfu

Question 2

Not yet answered Marked out of 6.00

Flag question

Edit question

Gestione della memoria: working set

Considera un processo P a cui vogliamo applicare il metodo del working set. L'immagine di P è composta da 110 pagine partizionate in 3 gruppi come segue

gruppo	numero di pagine			
Α	20			
В	40			
С	50			
totale	110			

L'esecuzione di P avviene per fasi. Ciascuna fase dura un certo numero di istanti di tempo ed accede a certi gruppi di pagine. Ecco il susseguirsi delle fasi.

fase	durata	inizio	fine	gruppi acceduti
1	300	0	299	A
2	300	300	599	АВ
3	300	600	899	AC
4	20	900	919	Α
5	500	920	1419	AC

Supponi che sia delta=100 e che gli accessi nel tempo siano uniformemente distribuiti sulle pagine. Rispondi brevemente alle seguenti domande.

- 1. Fase 2, quant'è la taglia del WS a regime?
- Transitorio tra fasi 1 e 2. Quanti istanti devono trascorrere perché WS arrivi regime?
 Spiega.
- 3. Fase 3, quant'è la taglia del WS a regime?
- Transitorio tra fasi 2 e 3. Quanti istanti devono trascorrere perché WS arrivi regime?
 Spiega
- Transitorio tra fasi 2 e 3. Qual'è la taglia massima raggiunta dal WS durante il transitorio? A che istante si raggiunge? Spiega
- 6. Fase 4. Qual'è la taglia del WS durante la fase 4? Spiega

Paragraph		₩	

Compito pari - turno 1

Rispondi punto per punto	_
Rispondi punto per punto	
1.	ı
2.	ı
3.	ı
4.	
5.	ı
6.	•
Path: p » em	

Question 3

Not yet answered Marked out of 6.00

Flag question
Edit question

I/O e interazioni tra processi

Supponi che ci siano quattro processi A, B, C, e D in esecuzione che fanno accesso allo stesso disco.

- A effettua in sequenza tante letture da posizioni consecutive su disco.
- B effettua in sequenza tante letture da posizioni random su disco.
- C effettua in sequenza tante scritture in posizioni consecutive su disco.
- **D** effettua in sequenza tante **scritture** in posizioni random su disco.

Rispondi ai seguenti punti

- 1. Supponi che il sistema abbia uno scheduler del disco tipo elevator senza particolari accortezze, quali processi sarebbero avvantaggiati e perché?
- che tipo di accortezze possono essere prese, a livello di scheduler del disco per migliorare la fairness del sistema?



Question 4

Not yet answered Marked out of 6.00

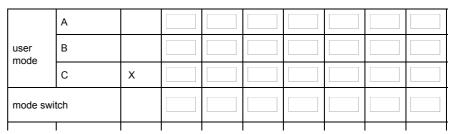
Flag question
Edit question

Sequenza degli eventi nel SO

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

- A, inizialmente in blocco, è I/O bound. Ciascun I/O viene servito in 80ms. E' inizialmente in blocco a causa di un I/O che verrà servito a tempo t=40 ms.
- B, inizialmente in blocco, è cpu bound. Genera un major page fault ogni 30ms servito in 50ms. E' inizialmente in blocco a causa di un page fault che verrà servito a tempo t=20ms.
- C, inzialmente running, 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.



Compito pari - turno 1

kernel	sched./disp.								
mode	system call								
	interrupt handler per page fault								
	interrupt handler per I/O								
	interrupt handler per quanto scaduto								
	running	С							
stati processi	ready								
	blocked	A B							
note tempi									
altre note									

Question 5

Not yet answered Marked out of 6.00

Flag question

Edit question

Espressioni Regolari (grep/egrep)

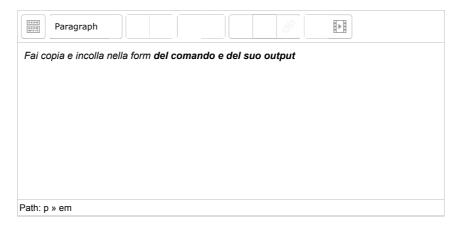
Considera il file di testo di esempio toBe.txt. Fornire una linea di comando basata su grep o egrep per stampare tutte le righe in cui siano verificate tutte le seguenti regole.

- 1. Il primo carattere è una consonante maiuscola.
- 2. L'ultimo carattere è ';'
- 3. Sia presente il carattere '.' o il carattere ':' seguito da uno spazio, seguito da due caratteri qualsiasi e quindi da un'altro spazio.

Usando l'esempio il comando deve stampare:

Devoutly to be wish'd. To die, to sleep;

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



Question 6

Not yet answered Marked out of 6.00

Flag question

Edit question

Awk

Si consideri il file voti.txt in cui ogni riga contiene il nome di ogni alunno e i voti dei compiti in classe, secondo il seguente formato

NomeAlunno voto1 voto2 voto3 ...

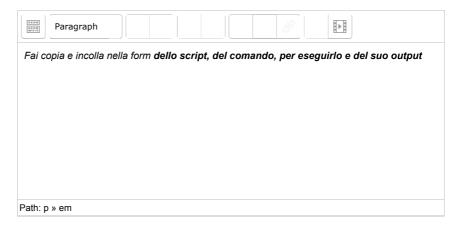
Il numero di compiti può variare da un alunno all'altro ma ve ne è almeno uno.

Si scriva uno script in awk che produca in output una tabella in cui, per i soli alunni che abbiano la media dei voti strettamente superiore a 24, si mostri voto minimo e voto massimo, secondo il seguente formato

NomeAlunno Minimo Massimo

Suggerimento: il numero dei campi di un record è dato dal valore della variabile NF.

Fai copia e incolla nella form dello script, del comando, per eseguirlo e del suo output



Question 7

Not yet answered Marked out of 6.00



Debugging

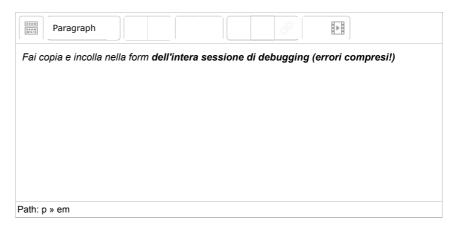
Si consideri il file fattoriale.c che contiene l'implementazione di del fattoriale nelle versioni iterativa, fact_iter(), e ricorsiva, fact_rec().

Si compili con il comando: gcc -Wall -g fattoriale.c -o fattoriale

Si consideri l'esecuzione di ./fattoriale con parametro 10. Eseguire le seguenti attività nel debugger.

- 1. imporre un breakpoint all'inizio di fact_iter()
- 2. eseguire con parametro 10
- 3. imporre un watchpoint che blocchi l'esecuzione all'interno di fact_iter() non appena r>100
- 4. continuare l'esecuzione
- 5. quando il watchpoint ferma l'esecuzione, mostrare il valore di r e di k
- 6. mostrare lo stack
- 7. imporre un breakpoint sul "return" di fact_rec() (riga 13)
- 8. condizionare il breakpoint appena inserito in modo che si fermi solo se r>100. Il comando da usare è "condition" consultare "help condition" per i dettagli.
- 9. continuare l'esecuzione
- 10. quando il breakpoint ferma l'esecuzione mostrare lo stack e i valori di n ed r
- 11. spostarsi sul frame #3 e mostrare i valori di n ed r

Fai copia e incolla nella form dell'intera sessione di debugging (errori compresi!)



Next

Moodle Docs for this page
You are logged in as Maurizio Pizzonia (Log out)

SOpari20170626