

Preview Compito Pari Turno1

Start again

1

Dati studente

Inserisci qui i tuoi dati, compila subito questa parte.

Cognome

Nome

Matricola

2

Memory management

Traslation Lookaside Buffer (TLB).

- Descrivi cosa è e quale è il suo scopo
- Descrivi cosa succede durante un process switch al TLB in una normale architettura MAS (esempio architettura Pentium) e motiva la risposta.

Answer:

jlkj
jlkj

3

I/O

Considera 8 dischi da configurare in raid 51. Descrivi come aggredheresti i dischi (es. scrivi (1,2,3) (4,5,6) ecc.)

Mostra, nello schema di aggregazione da te scelto, come sono disposti i blocchi logici nei blocchi fisici compilando il seguente schema (indica con P una eventuale parità).

| Mostra | Disco 1 | Disco 2 | Disco 3 | Disco 4 | Disco 5 | Disco 6 | Disco 7 | Disco 8 |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| blocco1 | | | | | | | | |
| blocco2 | | | | | | | | |
| blocco3 | | | | | | | | |
| blocco4 | | | | | | | | |
| blocco5 | | | | | | | | |
| blocco6 | | | | | | | | |

Scrivi qui eventuali note

4

Scheduling delle attivita' all'interno del SO

Considera un sistema con architettura del kernel “execution within user process”. In tale sistema sono presenti tre processi: A, B, C, inizialmente tutti e tre ready nell'ordine A in testa, poi B, C in coda. La politica di scheduling è **round robin senza alcun tipo di preemption** con quanto di tempo pari a 40ms.

- **A** è I/O bound: cpu burst trascurabili, I/O burst di durata 13ms, nessun page fault.
- **B** è I/O bound: cpu burst trascurabili, I/O burst di durata 120ms, nessun page fault.
- **C** è puramente cpu bound e non provoca 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 | | | | | | | | | | | | | | | | | | | A |
| | B | | | | | | | | | | | | | | | | | | | B |
| | C | | | | | | | | | | | | | | | | | | | C |
| mode switch | | | | | | | | | | | | | | | | | | | | mode switch |
| kernel mode | disptaching | | | | | | | | | | | | | | | | | | | disptai |
| | system call | | | | | | | | | | | | | | | | | | | system |
| | interrupt handler per page fault | | | | | | | | | | | | | | | | | | | interru handle page f |
| | interrupt handler per I/O | | | | | | | | | | | | | | | | | | | interru handle I/O |
| | interrupt handler per quanto scaduto | | | | | | | | | | | | | | | | | | | interru handle quantc scadui |
| stati processi | running | | | | | | | | | | | | | | | | | | | running |
| | ready | | | | | | | | | | | | | | | | | | | ready |
| | block | | | | | | | | | | | | | | | | | | | block |
| note tempi | | | | | | | | | | | | | | | | | | | | |
| altre note | | | | | | | | | | | | | | | | | | | | |

Scripting

Il file di testo [radb.db.txt](#) contiene un record per ciascun prefisso IP registrato presso un certo registro di Internet. Nel file ciascun record è separato da una linea vuota, i campi sono su linee distinte, tranne il campo "descr" che è su più linee. I campi occupano sempre la stessa posizione. Per svolgere l'esercizio non è necessario conoscere il significato di tutti i campi. Suggerimenti: alcune volte, ma non sempre, conviene processare tale file con awk usando RS="" (stringa vuota) e FS="\n", ricorda che, in awk, gsub() è un efficace strumento di sostituzione.

5 

Il campo "route" contiene un prefisso. Mostra un comando che selezioni tutte e sole le linee del file del campo "route" in cui il prefisso ha il secondo numero che inizia per 1 o per 2 (es. 193.110.10.0 o 193.210.10.0).

Answer:

6 

Usa awk per costruire una linea di comando che prenda in input il file `radb.db.txt` e dia in output una tabella che abbia nella prima colonna il campo "mnt-by" e nella seconda la prima riga del campo "descr". Esempio

```
route: 198.136.200.0/24
descr: AlterNet route - AS 701
27, Jay Av.
NY AC27
origin: AS701
mnt-by: MAINT-AS701
changed: asp@uunet.uu.net 19950430
source: RADB
```

diventa

MAINT-AS701 AlterNet route - AS 701

Answer:

System programming

Scarica il file [prj.tar.gz](#), contiene il programma "fib" composto da più file con estensione ".c" e ".h". Tale programma inizializza una lista contenente una serie di numeri, derivati mediante una funzione, dai primi n numeri di fibonacci e ne stampa il contenuto (n è passato come parametro sulla linea di comando).

7 

Crea un Makefile per il progetto con i seguenti target badando a mettere le dipendenze opportune
– fib: crea l'eseguibile "fib" linkato **staticamente** con **simboli di debug**

- fib_static: eseguibile "fib_static" senza simboli di debug, creato a partire da fib
- main.i: il precompilato del file main.c
- clean: pulisce il progetto cancellando i file inutili (*.o, *.~)

8 

Answer:

Compila fib in maniera da poter usare il **debugger**. Considera una esecuzione di fib con parametro **20**. Considera il primo istante in cui la funzione **fib()** sta per ritornare un valore **>100**. Quante volte è stata invocata la funzione **fib()** fino a quel momento? Usa il debugger per rispondere. Fai copia e incolla nella form della sessione del debugger con i comandi che hai usato e i loro output.

Answer:

 [Moodle Docs for this page](#)You are logged in as [Admin User](#) ([Logout](#))