

**Tempo totale a disposizione: 60 minuti.****QUESITI & ESERCIZI (max 26 punti)****IMPORTANTE**

I partecipanti a questa prova scritta sono invitati, nell'elaborato da consegnare, a specificare le informazioni richieste relative all'esame previsto dal proprio corso di studi.

**RACCOMANDAZIONI**

- curare la correttezza e l'appropriatezza del linguaggio e della grafia adoperati;
- evitare inutili e non richiesti allungamenti delle risposte, formulando risposte comprensibili, concise e compendiose;
- giustificare il perché delle asserzioni formulate;
- attenersi rigorosamente a quanto richiesto dal quesito/esercizio;
- non trascurare di dare risposta ad eventuali richieste multiple contenute nei quesiti/esercizi;

- 1) Scrivere un comando che restituisca l'elenco di file e directory, contenuti nella current working directory, il cui nome cominci per ^, s oppure a.
- 2) Scrivere un comando per copiare la cartella **dir1** e tutto il suo contenuto all'interno della cartella **dir2**.
- 3) Scrivere un comando che consenta di visualizzare dinamicamente i processi attivi sul sistema. Commentare l'output prodotto da tale comando, di cui di seguito è riportato un esempio.  

```
PID USER PR NI VIRT RES SHR S %CPU
%MEM TIME+ COMMAND
1983 root 15 -5 0 0 0 D 3.7 0.0
15:26.3 kjournald
6059 root 20 0 42320 14m 7376 S 3.7
3.0 1:29.96 Xorg
```
- 4) Definire cos'è la shell, citando alcune shell maggiormente diffuse.
- 5) La cartella **fileEliminati** contiene 8 file. Scrivere un comando per rimuovere tale cartella.
- 6) Descrivere il comando **yes > /dev/null** e come è possibile lanciarlo in *background*. Cosa si intende per esecuzione in *background*?
- 7) A cosa serve l'interrupt vector? e quale è lo scopo di una system call?
- 8) Come si può proteggere un SO da un programma che abbia un loop infinito o che non chiami mai il sistema operativo?
- 9) A cosa servono le code in un SO? Quali code ci sono generalmente in un SO? Come sono implementate? Quante code di device ci sono in un SO?
- 10) Determinare, nei SO UNIX-like a tre livelli di indicizzazione, il numero di blocchi di seconda indicizzazione per un file dopo 65000 operazioni di scrittura se i blocchi di indicizzazione vengono allocati all'occorrenza e quelli di dati sono preallocati 16 alla volta.
- 11) Un computer prevede indirizzi virtuali di 48 bit e indirizzi fisici di 32 bit. Ogni pagina ha una dimensione di  $2^{(1+S)}$ Kbyte. Da quanti elementi al massimo sarà costituita la External Page Map Table (EPMT) o SWAP file?

- 12) Perché l'algoritmo del banchiere (deadlock avoidance) si dice che è conservativo?
- 13) Si abbia un HD costituito da 200 cilindri (0..199), posizionato al cilindro 87, ultima richiesta precedentemente servita al cilindro T5 e con la seguente coda di richieste:  
92, 156, 26, 102, 74, 184, 55, 37  
In aggiunta a queste richieste, 2 nuove richieste per le tracce 3 e 180 arrivano quando metà delle richieste sono state servite. Indicare (in msec) il tempo totale impiegato dalla testina per una schedulazione con algoritmo dell'ascensore *C-LOOK*, se il tempo di spostamento è di Y msec/cyl.
- 14) Si assuma che lo scheduling della CPU avvenga secondo l'algoritmo di priorità dinamica "*process-merit*" e che i processi abbiano i seguenti valori di merito:  
P1 = 0.45   P2 = 0.81   P3 = 0.67   P4 = 0.54   P5 = 0.31  
P6 = 0.72   P7 = 0.59   P8 = 0.88   P9 = 0.91   P10 = 0.93  
Tra quali valori sarà compresa la *mediana*?  
Quale sarà l'effetto prodotto se la mediana attesa è pari a 0.Y0?
- 15) Si consideri uno scheduling *Round Robin* che prevede che la *Ready List* (i cui elementi sono puntatori ai PCB) possa avere un processo listato più volte. Quale è l'effetto? Quali vantaggi e quali svantaggi? Come si potrebbe ottenere gli stessi vantaggi senza replicare i puntatori?
- 16) Quali sono le caratteristiche secondo cui instaurare il *communication link* fra processi cooperanti?
- 17) Si supponga che un processo periodico in tempo reale sia pronto all'istante  $t=10$  sec, che la sua *deadline* sia pari a 1Y sec e il suo *computation time* sia di X sec. Se all'istante  $t=10$  sec sorge la necessità di eseguire un processo aperiodico con deadline uguale a quella del processo periodico e con *computation time*  $2 \cdot (1+W+Z)$  sec, sarà possibile garantirne la deadline? Motivare la risposta.
- 18) Cosa s'intende per busy waiting? E come si evita?

### AFFERMAZIONI (max 4 punti)

Si considerino le seguenti affermazioni.

Si barri la casella "Sicuramente Vera" (SV), se si è sicuri che l'affermazione è vera.

Si barri, invece, la casella "Sicuramente Falsa" (SF), se si è sicuri che l'affermazione è falsa.

**Per ogni risposta corretta 1 punto. Per ogni risposta errata -1 punto. Le affermazioni senza risposta comportano 0 punti.**

	Affermazione
1.	Due sezioni critiche possono essere eseguite contemporaneamente se non accedono o aggiornano le stesse variabili.
2.	L'algoritmo di Page Replacement noto come LRU segnala quando la sostituzione deve aver inizio.
3.	Un preemptive scheduler non può essere un real-time scheduler.
4.	Il contesto di un processo comprende almeno il Program Counter e l'informazione di stato del processo.
5.	La realizzazione di un semaforo richiede l'uso di speciali istruzioni di CPU.
6.	Un processo in tempo reale non dovrà essere eseguito il più velocemente possibile.
7.	Questa frase è falsa (paradosso del mentitore, VI secolo avanti Cristo).

POLITECNICO DI BARI

Specificare: *Corso di Laurea in* \_\_\_\_\_ *DM* \_\_\_\_  
*Esame di* \_\_\_\_\_ *CFU* \_\_\_\_

Cognome: \_\_\_\_\_ ; Nome: \_\_\_\_\_ ; matricola: \_\_\_\_\_

**Quesiti ed Esercizi**Dovunque appaiano, utilizzare i seguenti valori delle variabili indicate negli esercizi.

X = (numero di lettere che compongono il Cognome) - 2.

X = ..... (max 9);

Y = (numero di lettere che compongono il 1° Nome) - 2.

Y = ..... (max 9);

Z = 1 se X è pari; Z = 0 se X è dispari ;

Z = ..... ;

W = 1 se Y è pari ; W = 0 se Y è dispari ;

W = ..... ;

S = (penultima cifra del numero di Matricola).

S = ..... ;

T = (ultima cifra del numero di Matricola).

T = ..... ;

1) Scrivere un comando che restituisca . . . . .

7) A cosa serve l'interrupt vector? e quale . . . . .

2) Scrivere un comando per copiare . . . . .

8) Come si può proteggere un SO . . . . .

3) Scrivere un comando che consenta . . . . .

9) A cosa servono le code in un SO? Quali . . . . .

4) Definire cos'è la shell, citando alcune . . . . .

10) Determinare, nei SO UNIX-like . . . . .

5) La cartella **fileEliminati** contiene . . . . .6) Descrivere il comando **yes** > . . . . .

11) Un computer prevede indirizzi virtuali . . . . .

12) Perché l'algoritmo del banchiere . . . . .

15) Si consideri uno scheduling Round Robin . . . . .

13) Si abbia un HD costituito da . . . . .

16) Quali sono le caratteristiche secondo . . . . .

14) Si assuma che lo scheduling della CPU . . . . .

17) Si supponga che un processo periodico . . . . .

18) Cosa s'intende per busy waiting? E . . . . .

### **AFFERMAZIONI**

Si considerino le seguenti affermazioni.

Si barri la casella "Sicuramente Vera" (SV), se si è sicuri che l'affermazione è vera.

Si barri, invece, la casella "Sicuramente Falsa" (SF), se si è sicuri che l'affermazione è falsa.

**Per ogni risposta corretta 1 punto. Per ogni risposta errata -1 punto. Le affermazioni senza risposta comportano 0 punti.**

	<i>Affermazione</i>	SV	SF
1.	Due sezioni critiche possono essere eseguite contemporaneamente se non accedono o aggiornano le stesse variabili.		
2.	L'algoritmo di Page Replacement noto come LRU segnala quando la sostituzione deve aver inizio.		
3.	Un preemptive scheduler non può essere un real-time scheduler.		
4.	Il contesto di un processo comprende almeno il Program Counter e l'informazione di stato del processo.		
5.	La realizzazione di un semaforo richiede l'uso di speciali istruzioni di CPU.		
6.	Un processo in tempo reale non dovrà essere eseguito il più velocemente possibile.		
7.	Questa frase è falsa (paradosso del mentitore, VI secolo avanti Cristo).		