

**Tempo totale a disposizione: 60 minuti.****QUESITI & ESERCIZI (max 26 punti)****max 2 punti/quesito/esercizio**

1) Si indichi la funzione del comando **chmod**, riportando la sintassi e le opzioni di base del comando. Si scriva infine il comando per visualizzare la pagina della **sezione 1** del manuale relativa al comando.

2) Sia **~** la *current working directory* dell'utente loggato. Si scriva un comando per copiare la directory **~/esame** e tutto il suo contenuto all'interno della directory **/tmp** (la directory **esame** si suppone non vuota).

3) Si modifichi l'account dell'utente **ciccio** assegnandogli il nuovo nome di login **pippo** e gruppo primario **pippo**, supposto quest'ultimo già esistente. Quale utente può effettuare tale modifica?

4) Siano dati il file **elenco\_InfAut** e il file **elenco\_EleTlc** con struttura:

NOME COGNOME MATRICOLA CDL

File **elenco\_InfAut**

**Maria Rossi 123456 Inf&Aut**

**Gianni Bianchi 987654 Inf&Aut**

...

File **elenco\_EleTlc**

**Concetta Verdi 234567 Ele&Tlc**

**Ruggiero Giallini 765432 Ele&Tlc**

...

Si crei un unico file **elenco** contenente i soli campi COGNOME MATRICOLA CDL degli studenti elencati nei due precedenti file, ordinati per cognome.

5) Si dica cos'è la **shell**, quali funzioni svolge, dove si colloca all'interno dell'architettura di un sistema operativo UNIX e si citino alcune shell più diffuse.

6) Supposto di adottare un algoritmo di merito, si stabilisca l'ordine crescente di priorità (**P<sub>i</sub>**) nel prossimo intervallo statistico  $\Delta T$  per task i cui contatori di time slice esauriti (**n<sub>i</sub>**) e time slice assegnati (**N<sub>i</sub>**) siano i seguenti:

Task	n <sub>i</sub>	N <sub>i</sub>
1	2	X
2	X	9
3	3	Y+1
4	6	7
5	Y-1	8
6	4+W	7
7	5-Z	5
8	W	X-1
9	5+Z	9
10	3+W	8

Se la mediana attesa è 0.40, quale sarà la retroazione da apportare al time slice?

7) Si consideri un process scheduler che usi l'algoritmo round robin con time slice pari a 10 ms. Se i *burst time* B<sub>i</sub> dei 4 processi in esecuzione sono i seguenti:

B<sub>1</sub> = X8 ms; B<sub>2</sub> = 23 ms; B<sub>3</sub> = 3S ms; B<sub>4</sub> = Y ms;

descrivere la successione di tempi di CPU (diagramma di Gantt) relativi all'esecuzione dei 4 processi.

8) Specificare qual è la parte di *dispatch latency* risparmiata da un processo *n-threaded* rispetto a quella di n processi *single-threaded* cooperanti.

9) Si assuma che la tabella delle aree libere per un memory manager a partizionamento dinamico della memoria (con algoritmo best-fit) sia la seguente:

Area libera	Dimensione	Primo byte	Status bit
1	4Mb	22Mb	1
2	6Mb	12Mb	1
3	6Mb	X0Mb	1
4	8Mb	44Mb	0
5	10Mb	78Mb	0
6			0
7			0

Specificare come cambia tale tabella se prima viene allocato un processo che richiede 5 Mb e poi termina un processo di 4 Mb con primo byte a (X0 - 4) Mb.

- 10) Quale meccanismo parallelo alla MMU (o Dynamic Address Translator) viene usato, nel caso della paginazione reale, per la traduzione degli indirizzi da logici a fisici? E a quale inconveniente del DAT pone riparo?
- 11) Se  $p$  è la probabilità di page fault,  $T_{pf}$  il suo tempo medio di servizio e  $T_{am}$  il tempo di accesso alla memoria, qual è l'espressione con cui calcolare il tempo di accesso effettivo  $T_{eff}$  per una memoria a demand-paging?
- 12) Si consideri la seguente *snapshot* di un sistema
- |                | Alloc.  | Max     | Available |
|----------------|---------|---------|-----------|
|                | A B C D | A B C D | A B C D   |
| P <sub>0</sub> | 1 0 1 2 | 1 4 1 2 | 1 5 2 0   |
| P <sub>1</sub> | 1 0 0 0 | 1 7 5 0 |           |
| P <sub>2</sub> | 1 3 5 4 | 2 3 5 6 |           |
| P <sub>3</sub> | 0 3 3 2 | 0 X 5 2 |           |
| P <sub>4</sub> | 0 0 1 4 | 0 6 Y 6 |           |
- Il sistema è in uno stato ammissibile? Perché?  
Se, nello stato indicato, arrivasse dal processo P<sub>1</sub> una richiesta per (0,4,2,0), potrebbe essere garantita immediatamente? Perché?
- 13) Spiegare cosa significa che un'istruzione viene eseguita atomicamente.
- 14) Quali sono le caratteristiche secondo cui instaurare il communication link fra processi cooperanti?
- 15) Come funziona una trasmissione connection-oriented e come una connection-less?
- 16) Perché un thread viene anche chiamato "processo a peso leggero"?
- 17) Si consideri una rete, basata su collegamenti punto a punto, costituita da (X+Y) nodi. Quale sarà il numero minimo di collegamenti per garantire almeno 2 percorsi alternativi tra 2 nodi qualunque della rete?
- 18) Supposto che un pacchetto abbia una dimensione di 1500 bytes e che il suo header lo occupi per il 35%, a quanto ammonterà (in byte) il *payload*?

### 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.	L'algoritmo di scheduling della CPU noto come " <i>priorità dinamica</i> " è di tipo nonpreemptive.
2.	Le procedure pubbliche di un <i>monitor</i> sono mutuamente esclusive.
3.	Un <i>deadlock</i> si può determinare anche potendo requisire le risorse detenute da un processo.
4.	LINUX distingue tra <i>processi</i> e <i>thread</i> .
5.	Una <i>race condition</i> si verifica quando 2 o più processi tentano di accedere simultaneamente alla stessa risorsa.
6.	Nei <i>sistemi a memoria virtuale</i> , quest'ultima è separata dalla memoria logica.



10) Quale meccanismo parallelo alla MMU . . . . .

14) Quali sono le caratteristiche . . . . .

11) Se  $p$  è la probabilità di page fault, . . . . .

15) Come funziona una trasmissione . . . . .

12) Si consideri la seguente *snapshot* . . . . .

16) Perché un thread viene anche chiamato . . . . .

13) Spiegare cosa significa che un'istruzione . . . . .

17) Si consideri una rete, basata . . . . .

18) Supposto che un pacchetto . . . . .

### 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.**

	Affermazione	SV	SF
1.	L'algoritmo di scheduling della CPU noto come " <i>priorità dinamica</i> " è di tipo nonpreemptive.		
2.	Le procedure pubbliche di un <i>monitor</i> sono mutuamente esclusive.		
3.	Un <i>deadlock</i> si può determinare anche potendo requisire le risorse detenute da un processo.		
4.	LINUX distingue tra <i>processi</i> e <i>thread</i> .		
5.	Una race condition si verifica quando 2 o più processi tentano di accedere simultaneamente alla stessa risorsa.		
6.	Nei sistemi a memoria virtuale, quest'ultima è separata dalla memoria logica.		