

Nome e Cognome:

Matricola:

1	2	3	4	5	6	7	8	tot
/8	/15	/15	/12	/10	/10	/10	/20	/100

Spazio riservato alla correzione

1. *8 punti*

Scrivere uno script che avendo da linea di comando il nome di una directory presente nella directory corrente:

- 1) crei un comando che consenta di copiare dalla directory corrente alla directory specificata tutti i file il cui nome cominci con la lettera 'a';
- 2) mandi in esecuzione tale comando;
- 3) visualizzi su standard output il nome della directory specificata.

## 2. 15 punti

Sia `FF` un file che contenga le 4 lettere `abdc` tale che

```
-rw--w---- 1 rescigno 4 Jun 4 09:45 FF
```

Siano `HFF` e `SFF`, rispettivamente, hard link e symbolic link a `FF`.

a) Dire, giustificando la risposta, quali sono i permessi di `HFF` e di `SFF`;

b) Sia `leggo.out` l'eseguibile di un programma C che legge (utilizzando `open` e `read`) il file `FF`

```
-r-x--x--x 1 rescigno 567 Jun 5 09:45 leggo.out
```

b1) Dire giustificando la risposta che cosa si ottiene se l'utente `studente` manda in esecuzione `leggo.out`.

b2) Assumendo che **rescigno** setti il set-user-id di **leggo.out**, dire giustificando la risposta che cosa si ottiene se l'utente **studente** manda in esecuzione **leggo.out**.

c) Assumendo di aver eliminato il file **FF**, dire, giustificando le risposte, che cosa si ottiene dando

**cat HFF**

**cat SFF**

## 3. 15 punti

a) Si consideri il seguente programma C e sia `a.out` il suo eseguibile

```
(1)  int main(void)
(2)  { printf("Prima");
(3)      execl("/bin/echo","echo","Durante",NULL );
(4)      printf("Dopo");
(5)      exit(0); }
```

Che cosa si ottiene su standard output dando `a.out`? Motivare la risposta.

b) Scrivere un programma C che mostri su standard output:

Prima

Durante

Dopo

dove **Durante** sia scritto utilizzando una delle funzioni **exec** .

## 4. 12 punti

Si assuma che il segnale SIGKILL sia definito come segue

```
#define SIGKILL 9      /* Interrupt (ANSI). Default action: _exit */
```

Dato il seguente programma C, il cui eseguibile é a.out

```
(1) void handler(int);
(2) void exit1(void);

(3) int main(void)
(4) { char array[6]="Hello ";
(5) atexit(exit1);
(6) printf("Ciao");
(7) if (fork( ) == 0) signal(SIGINT, handler);
(8) sleep(30);
(9) write(1,array,6);
(10) exit(0);}

(11) void handler(int signum)
(12) { printf("Signal Handler\n");
      exit(0);}

(13) void exit1(void)
(14) { printf("Exit Handler 1\n"); }
```

(a) Dire che cosa succede dando a.out senza l'arrivo di alcun segnale. Motivare la risposta.

(b) nell'ipotesi che arrivi il segnale **SIGINT al processo figlio** durante l'esecuzione dello sleep. Motivare la risposta.

(c) nell'ipotesi che arrivi il segnale **SIGKILL al processo padre** durante l'esecuzione dello sleep. Motivare la risposta.

## 5. 10 punti

Un hard disk é formattato in blocchi da 128 byte. Sia A un file di 1024 byte. Si assuma che il disco adotti una allocazione concatenata.

a) Dire quanti blocchi sono necessari per allocare A. Specificare quale é lo spreco di memoria dovuto alla frammentazione interna.

b) Quante operazioni di lettura/scrittura sono necessarie per cancellare il terzo blocco del file, assumendo che il numero del primo blocco sia già in RAM (motivate la risposta)?

c) É possibile la frammentazione esterna?



## 6. 10 punti

Sia dato un disco di 32 Gb diviso in blocchi ampi 2 Kb. Si assuma che la struttura per tener traccia dei blocchi liberi sia del tipo **bitmap**. Si calcoli il numero di blocchi richiesto dalla struttura bitmap.

## 7. 10 punti

Cinque processi arrivano al tempo indicato, consumano la quantità di CPU indicata e hanno le priorità indicate nella tabella sottostante:

Processo	T. di Arrivo	Burst	Priorità
$P_1$	0	1	3
$P_2$	2	5	5
$P_3$	3	2	2
$P_4$	6	3	4
$P_5$	7	1	1

Calcolare il turnaround medio ed il waiting time medio per i processi nel caso sia usato l'algoritmo di *scheduling Round Robin con prelazione e con quanto di tempo di ampiezza 2*

## 8. 20 punti

In un sistema la memoria fisica ? divisa in  $2^{22}$  frame, un indirizzo logico ? scritto su 33 bit, e all'interno di una pagina, l'offset va da 0 a 111111111.

*Giustificando le risposte,*

(a) indicare la struttura dell'indirizzo fisico e dell'indirizzo logico.

(b) Assumendo che le tabelle delle pagine includono anche il *bit di validità* ed il *bit di modifica*, dire

1) Quanto è grande in byte la page table di un processo che usa tutte le pagine;

2) Quanti **page fault** e quanti **accessi a disco** ci sono da parte di un processo  $P_1$  a partire dall'istante  $t$  in cui la situazione della tabella delle pagine é la seguente

pagina	frame	bit validit�	bit di modifica
1	0	1	1
2	1	1	0
3	2	1	0
4	x	0	0
5	x	0	0

ed il riferimento alle pagine gi  presenti   stato fatto nell'ordine 1, 2, 3

e l'algoritmo utilizzato per la selezione della pagina vittima sia LRU con 3 frame,

ed i riferimenti a pagine da parte del processo  $P_1$  in istanti successivi a  $t$  sono i seguenti

- scrittura pagina 1
- lettura pagina 2
- lettura pagina 3
- lettura pagina 4
- lettura pagina 5
- scrittura pagina 2
- scrittura pagina 4
- scrittura pagina 2
- lettura pagina 1
- lettura pagina 3
- scrittura pagina 3
- lettura pagina 4

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA