

Nome e Cognome:

Matricola:

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

Spazio riservato alla correzione

1. 10 punti

Scrivere uno script che prendendo da linea di comando il *nome di una nuova directory* da creare:

- 1) crei una sottodirectory all'interno della propria home directory il cui nome sia uguale a quello dato da linea di comando;
- 2) sposti tutti i file con estensione .x presenti nella home directory, nella directory appena creata;
- 3) mostri su standard output il numero di file spostati nella nuova directory.

2. *20 punti*

Sia **F** un file lungo 6 byte contenente i caratteri **abcdef**.

a) Scrivere un programma C che **in successione**

- crei un hard link **HF** ad **F**, utilizzando una delle funzioni **exec**;
- legga il contenuto di **HF** mostrandolo sullo standard output, utilizzando una delle funzioni **exec**.

b) Dire, giustificando la risposta, che cosa si ottiene sullo standard output eseguendo il programma del punto a).

c) Supponendo di rimuovere **F**, dire, giustificando la risposta, che cosa si ottiene mandando in esecuzione `cat HF`.

3. 10 punti

É noto che il segnale SIGINT é definito come segue

```
#define SIGINT 2      /* Interrupt (ANSI). Default action: exit */
```

Si supponga che a.out sia l'eseguibile corrispondente al seguente programma C

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

(a) Dire che cosa si ottiene dando a.out nell'ipotesi che al "*processo padre*" arrivi il segnale SIGINT durante l'esecuzione dello sleep. Motivare la risposta.

(b) Dire che cosa si ottiene dando a.out nell'ipotesi che al "*processo figlio*" arrivi il segnale SIGINT durante l'esecuzione dello sleep. Motivare la risposta.

4. 10 punti

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

```
void exit1(void);
void exit2(void);

int main(void)
{ char array[6]="Hello ";
  atexit(exit1);
  write(1,array,6);
  if (fork()==0)
      { atexit(exit2);
        printf("Ciao \n");
        _exit(0);}
  else { wait( );
        exit(0);}
}

void exit1(void)
{ printf("Exit Handler 1\n"); }
void exit2(void)
{ printf("Exit Handler 2\n"); }
```

- (a) dire che cosa succede dando `a.out`. Motivare la risposta.
- (b) dire che cosa contiene `FILE` dando `a.out > FILE`. Motivare la risposta.

5. 15 punti

Volendo adottare l'algoritmo LRU per la sostituzione delle pagine, dire:

(a) Quale dovrebbe essere il numero di frame che assicura il *minimo* numero di page fault. Giustificare la risposta.

(b) Quale dovrebbe essere il numero di frame che assicura il *massimo* numero di page fault. Giustificare la risposta.

Si consideri la seguente stringa di riferimenti a pagine

1 2 3 2 4 2 1 5 5 6 2 1 3 2 7 6

Si contino i page fault e gli accessi a disco in ciascuno dei tre casi (i frame sono inizialmente vuoti):

(c) LRU con 4 frame

(d) LRU con 2 frame

6. 15 punti

Un filesystem simile a Unix ha gli i-node fatti nel modo seguente:

attributi

ind. blocco 0

ind. blocco 1

ind. blocco 2

ind. blocco 3

ind. blocco 4

ind. blocco 5

ind. blocco indirizzi indirezione singola

ind. blocco indirizzi indirezione doppia

Se i blocchi del disco hanno capacità di 1 Kb e l'indirizzo dei blocchi occupa 32 bit,

a) quale é l'ampiezza massima del file memorizzabile utilizzando questo i-node? (Giustificare la risposta)

b) Se **paperino** é un file di dimensione $1Mb = 2^{20}$ byte, quanti blocchi servono in totale per la memorizzazione di **paperino** (compresi eventuali blocchi indice)? (Giustificare la risposta)

7. 10 punti

Quattro processi arrivano al tempo indicato e consumano la quantità di CPU indicata nella tabella sottostante)

processo	tempo di arrivo	burst
P_1	0	13
P_2	2	9
P_3	2	7
P_4	12	4

(a) Calcolare il turnaround medio e il waiting time medio per i processi nel caso l'algoritmo di scheduling sia SJF preemptive (shortest remaining time first). Riportare il diagramma di GANTT usato per il calcolo.

(b) SJF preemptive può causare problemi di starvation? E SJF non preemptive?

8. 10 punti

La macchina su cui viene installato un SO usa 32 bit per scrivere l'indirizzo fisico, e lo spazio di indirizzamento fisico é 4 volte più piccolo di quello logico.

Se le pagine sono da 2^{16} byte, quanto può essere grande (in numero di byte), al massimo, la page table di un processo? (motivate la vostra risposta)

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA