

Nome e Cognome:

Matricola:

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

Spazio riservato alla correzione

1. *10 punti*

Scrivere uno script che:

- 1) crei un comando che consenta di cancellare dalla directory corrente tutti i file il cui nome cominci con la cifra 2,3,4;
- 2) mandi in esecuzione tale comando;
- 3) visualizzi su standard output il nome dello script.

2. 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** .

3. 15 punti

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

```
(1) void handler1(void);
(2) void handler2(void);
(3) int main(void)
(4) { char array[5]="Dopo";
(5)     atexit(handler1);
(6)     printf("Prima \n");
(7)     fork( );
(8)     fork( );
(9)     atexit(handler2);
(10)    write(1,array,5);
(11)    exit(0);}

(10) void handler1(void)
(11) { printf("Handler-1\n"); }
(12) void handler2(void)
(13) { printf("Handler-2\n"); }
```

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

b) Che cosa contiene `FILE` dando `a.out > FILE`. Motivare la risposta.

c) Supponendo di sostituire nel programma precedente la linea (9) con `_exit(0);` dire, che cosa si ottiene su standard output dando `a.out`? Motivare la risposta.

4. 10 punti

Si consideri il seguente programma e si supponga di compilarlo.

```
#include<sys/types.h>
#include<fcntl.h>
#include<unistd.h>

int main(){
    if (access("tentativo.txt", R_OK)<0)
        printf("access error per tentativo.txt");
    else
        printf("access OK\n");

    if (open("prova.txt", O_RDONLY)<0)
        printf("open error per prova.txt");
    else
        printf("open OK\n");

    if (open("risultato.txt", O_RDWR)<0)
        printf("open error per risultato.txt");
    else
        printf("open OK\n");

    exit(0);
}
```

Se fosse

```
-rwxrwxrwx 1 rescigno 10932 Jun 4 10:45 a.out
-r----- 1 rescigno 1891   Jun 4 09:45 prova.txt
-r----- 1 studente 1891   Jun 4 09:45 tentativo.txt
-r----- 1 rescigno 1891   Jun 4 09:45 risultato.txt
```

a) supponendo che si sia loggato **studente**,

1) dire che cosa succede dando **a.out**;

2) assumendo di settare il set-user-id di **a.out**, dire se ci sono cambiamenti dando **a.out**. *In tutti i casi le risposte vanno giustificate.*

5. 15 punti

In un hard disk sono necessari 4 byte per contenere l'indirizzo di un blocco; inoltre, ciascun blocco ha una capacità di 1Kb.

Si assuma che un file F , costituito da 10 blocchi sia memorizzato su tale hard disk e che gli attributi del file siano già presenti in memoria principale.

Assumendo accesso diretto, dire, giustificando la risposta, quanti accessi a memoria sono necessari per **modificare i byte da 2000 a 2080 di F** per ciascuna delle diverse tecniche di memorizzazione di seguito riportati:

- 1) allocazione contigua;
- 2) allocazione concatenata;
- 3) indicizzata (si ricordi che il blocco indice é il primo blocco di un file su disco);

6. 15 punti

Si assuma che un SO usi:

- 30 bit per un indirizzo fisico
- 32 bit per un indirizzo logico
- frame di 4KB

Giustificando le risposte, dire

- (a) Quanti bit uso per l'offset?
- (b) Con quanti bit identifico un frame?
- (c) Quanti frame abbiamo in memoria fisica?
- (d) Con quanti bit identifico una pagina?
- (e) Quante pagine abbiamo nello spazio di indirizzamento logico?
- (f) Se un processo usa tutte le pagine quanto é grande in byte la page table?

7. 10 punti

Si consideri la seguente stringa di riferimenti a pagine

1 2 3 4 1 2 5 2 1 3 2 3

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

- (a) LRU con 4 frame
- (b) FIFO con 4 frame

8. 10 punti

In un sistema **time sharing con priorità** sono presenti 3 processi P1,P2,P3 nel seguente stato:

- P1 in esecuzione;
- P2 pronto;
- P3 in attesa di una operazione di I/O.

I processi P1 e P3 hanno uguale priorità, e più alta della priorità del processo P2.

Descrivere come cambia lo stato del sistema (cioé lo stato dei processi) se a partire dalla situazione data si verificano nell'ordine tutti e soli i seguenti eventi:

- a) termina l'operazione di I/O del processo in attesa
- b) trascorre un quanto di tempo
- c) il processo in esecuzione chiede una operazione di I/O
- d) il processo in esecuzione chiede una operazione di I/O
- e) termina l'operazione di I/O del processo in attesa

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA