

Prova d'esame 7 Luglio 2010

Università di Salerno

Nome e Cognome:

Matricola:

1	2	3	4	5	tot
/40	/10	/20	/20	/10	/100

Spazio riservato alla correzione

1. *40 punti*(a) (*8 punti*) Scrivere uno script tale che invocato visualizzi in successione su standard-output:

- 1) il messaggio "Ecco la lista dei file di testo"
- 2) la lista dei file presenti nella cwd che terminano per `.txt`
- 3) il messaggio "Il numero totale di file é : "
- 4) il numero di file presenti nella cwd.

- (b) (*15 punti*) Scrivere un programma C che ottenga esattamente quanto richiesto al punto (a) e dove
- il punto 1) sia effettuato mediante una system call;
 - il punto 2) sia effettuato mediante una delle funzioni **exec**;
 - il punto 3) sia effettuato mediante una funzione di libreria;
 - il punto 4) sia effettuato mediante una delle funzioni **exec**.

- (c) (12 punti) Sia `File` un file in cui sia presente la parola `Prima` e sia `prog1.out` l'eseguibile di cui al punto (b). Scrivere 2 possibili programmi C in cui si simuli `prog1.out >> File`.

- (c) (5 punti) Si supponga di modificare l'ultima linea del programma di cui al punto (b) sostituendo `exit(0)` con `_exit(0)` e sia `prog2.out` l'eseguibile così ottenuto. Dire che cosa si ottiene in `File1` dopo che si é effettuato `prog2.out > File1`

2. 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("prova.txt", R_OK)<0)
        printf("access error per prova.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("prova.txt", O_WRONLY)<0)
        printf("open error per prova.txt");
    else
        printf("open OK\n");

    exit(0);
}
```

Se fosse

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

- 1) supponendo che si sia loggato **studente**, dire che cosa succede dando **a.out**.
- 2) supponendo che si sia loggato **rescigno**, dire che cosa succede dando **a.out**.
- 3) supponendo che si sia loggato **studente** e che sia settato il set-user-id di **a.out** dire che cosa succede dando **a.out**.

In tutti i casi le risposte vanno giustificate.

3. 20 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 piú piccolo di quello logico.

a) 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)

b) É necessario usare una paginazione a piú livelli? (motivate la vostra risposta)

c) Se il sistema adotta una paginazione su piú livelli, i processi gireranno in media piú o meno velocemente che nel caso in cui venisse adottata una paginazione ad un solo livello? (motivate la vostra risposta)

4. 20 punti

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

Processo	T. di Arrivo	Burst
P_1	0	10
P_2	1	8
P_3	2	6
P_4	11	3

a) (7 punti) Calcolare il turnaround medio ed il waiting time medio per i processi nel caso sia usato l'algoritmo di scheduling SJF non preemptive. Riportare il diagramma di Gantt usato per il calcolo.

b) (*13 punti*) Calcolare il turnaround medio ed il waiting time medio per i processi nel caso sia usato l'algoritmo di scheduling SJF preemptive (shortest remaining time first). Riportare il diagramma di Gantt usato per il calcolo.

5. 10 punti

In un sistema time sharing un processo pu essere interrotto in modo forzato prima della scadenza del suo quanto di tempo

- (a) no, mai
- (b) s, se ci sono altri processi in stato di attesa
- (c) solo se c un processo pronto con priorit pi alta
- (d) solo se c un processo pronto con priorit pi bassa

Dire **se, perché, ed in quali casi** le 4 precedenti risposte sono corrette.

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA