

## II Prova in Itinere 6 Giugno 2014

Università di Salerno

Nome e Cognome:

Matricola:

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

Spazio riservato alla correzione

## 1. 10 punti

Scrivere uno script che prendendo da linea di comando **il nome di una directory**, che supponiamo sia presente nella home directory, ed **una stringa** realizzi in ordine quanto segue:

- 1) sposti tutti i file presenti nella home directory che contengono **stringa** all'interno della directory specificata;
- 2) mostri sullo standard output l'elenco dei file presenti nella directory specificata dopo averli ordinati in ordine lessicografico inverso
- 3) definire il comando **list** che visualizza sullo standard output l'elenco dei file presenti nella directory specificata dopo averli ordinati in ordine lessicografico inverso;
- 4) mandi in esecuzione l'ultimo comando dato.

## 2. 18 punti

Sia **elenco** un file contenuto nella cwd. Scrivere un programma C che *in successione*

- 1) mostri sullo standard output il contenuto di **elenco**, utilizzando una delle funzioni **exec**;
- 2) mostri sullo standard output il contenuto di **elenco** ordinato in ordine lessicografico, utilizzando una delle funzioni **exec**;
- 3) scriva la parola **Fatto** sullo standard output.

## 3. 12 punti

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

```
(1) void handler(int);
(2) void exit1(void);
(3) void exit2(void);

(4) int main(void)
(5) { char array[6]="Hello\n";
(6)   atexit(exit1);
(7)   write(1,array,6);
(8)   if (fork( )==0)   atexit(exit2);
(11)  fork( );
(11)  printf("Ciao\n");
(12)  exit(0);}

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

- (a) dire che cosa succede dando `a.out`. Motivare la risposta.
- (b) modificando la linea (14) con `_exit(0)`; dire che cosa succede dando `a.out`. Motivare la risposta.
- (c) dopo aver effettuato la modifica in (b), dire che cosa contiene `FILE` dando `a.out > FILE`. Motivare la risposta.



## 4. 10 punti

É noto che il segnale SIGKILL é 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) int main(void)  
(3) { char array[6]="Hello ";  
(4) printf("Ciao");  
(5) signal(SIGINT, handler);  
(7) sleep(30);  
(8) write(1,array,6);  
(9) exit(0);}  
  
(10) void handler(int signum)  
(11) { printf("Handler\n");  
      exit(0);}
```

- (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 durante l'esecuzione dello sleep. Motivare la risposta.

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

## 5. 12 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

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.

6. 25 punti

Si assuma che un SO usi:

- 32 bit per un indirizzo fisico
- 34 bit per un indirizzo logico
- frame di 1KB

Giustificando le risposte, dire

- (a) Quanti bit uso per l'offset?
- (b) Con quanti bit identifico un frame?
- (c) Con quanti bit identifico una pagina?



(d) 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 è la seguente

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

ed il riferimento alle pagine già presenti è stato fatto nell'ordine 1, 2, 3 ed i riferimenti a pagine da parte del processo  $P_1$  in istanti successivi a  $t$  sono i seguenti

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

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

(g) Se per un processo  $P_2$  non si verificano mai page fault, qual 'é, in nanosecondi, il tempo medio di accesso in RAM del sistema se viene usato un TLB con un tempo di accesso di 5 nanosecondi, un hit-ratio del 90% e un tempo di accesso in RAM di 0,075 microsecondi? (é sufficiente riportare l'espressione aritmetica che fornisce il risultato finale) (ricorda 1microsecondo=1000 nanosecondi).

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA