

=====
== CORSO DI SISTEMI OPERATIVI mod. A (gr. 1) ==
===== compito N. 1 del 200106 =====
=====

domanda N. 1:

Si determini la veridicità delle seguenti affermazioni:

- a) Uno dei vantaggi dei sistemi operativi basati su architettura a microkernel è la portabilità. Infatti il microkernel non dipende da una particolare piattaforma hardware, perché la maggior parte dei componenti è eseguito al di fuori del kernel.
- b) Nei moderni sistemi operativi tra cui Linux e Windows XP, il coordinamento delle varie attività avviene mediante una tecnica chiamata *polling*, che consiste nel richiedere continuamente, da parte della CPU, lo stato di tutti i dispositivi.

[A] a) falsa ; b) vera

[B] entrambe false

[C] entrambe vere

[D] a) vera ; b) falsa

domanda N. 2:

Si determini la veridicità delle seguenti affermazioni:

- a) I segnali sono il principale meccanismo di comunicazione tra processi. Essi infatti permettono ad un processo di porsi in uno stato di attesa di un dato, e di riprendere l'esecuzione solo dopo che un altro processo gli abbia esplicitamente inviato il dato. Tale modello di programmazione concorrente è chiamato *a scambio di messaggi*.
- b) Un vantaggio dei thread realizzati a livello del kernel rispetto all'implementazione a livello utente è che nel primo caso possono essere schedulati indipendentemente differenti thread di uno stesso processo permettendo una

maggior concorrenza nel caso di applicazioni multithreads con numerose operazioni di I/O.

- [A] entrambe false
- [B] a) falsa ; b) vera
- [C] a) vera ; b) falsa
- [D] entrambe vere

domanda N. 3:

Si determini il numero di processi generati (compreso quello mandato in esecuzione dalla linea di comando) dal seguente codice C, compilato e fatto eseguire su un sistema con il sistema operativo Unix:

```
main(){  
    int pid, i;  
    for (i = 0; i <= 3; i ++){  
        pid=fork();  
        if (pid == 0 && i/2 * 2 == i) exit(0);  
    }  
    exit(0);  
}
```

- [A] 8
- [B] 16
- [C] 13
- [D] 7

domanda N. 4:

Sia dato il seguente insieme di processi:

	T_A	T_C
P_1	0	9
P_2	2	6
P_3	4	5
P_4	5	2
P_5	8	1

dove T_A è il tempo di arrivo nella coda dei processi pronti e T_C è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling dello Shortest Job First con prelazione, si determini il tempo medio di esecuzione.

- [A] 9.0
- [B] 9.6
- [C] 9.4
- [D] 9.2

domanda N. 5:

Sia dato il seguente insieme di processi:

	T_A	T_C
P_1	0	5
P_2	3	4
P_3	5	7
P_4	10	6
P_5	14	5

dove T_A è il tempo di arrivo nella coda dei processi pronti e T_C è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo $q = 2$, si determini il tempo medio di esecuzione.

- [A] 12.4
- [B] 12.2
- [C] 12.0

[D] 12.6

domanda N. 6:

Siano dati tre processi che condividono una variabile x e i cui pseudo codici sono di seguito riportati:

P_1	P_2	P_3
$\text{wait}(S)$	$\text{wait}(R)$	$\text{wait}(T)$
$x = x - 1;$	$x = x + 2;$	<i>if</i> ($x > 0$) <i>then</i>
$\text{signal}(T);$	$\text{signal}(T);$	$\text{signal}(R);$
	$\text{wait}(R);$	<i>else</i>
	$x = x + 1;$	$\text{signal}(S);$
	$\text{signal}(T);$	<i>endif</i>
		$\text{wait}(T);$
		<i>print</i> $x;$

Si determini l'output del processo P_3 supponendo che inizialmente $x = -1$ e i tre semafori binari siano inizializzati rispettivamente $S = 0$, $R = 1$, $T = 0$.

- [A] 1
- [B] 0
- [C] -1
- [D] 2

domanda N. 7:

Si determini la veridicità delle seguenti affermazioni:

- a) La strategia di allocazione contigua della memoria first fit ha un overhead minore rispetto a quello delle strategie best-fit e worst-fit.
- b) Uno dei principali problemi delle strategie di allocazione basate su partizioni fisse è la frammentazione interna.

- [A] a) falsa ; b) vera

- [B] entrambe false
- [C] entrambe vere
- [D] a) vera ; b) falsa

domanda N. 8:

Si consideri un sistema di paginazione con tabella delle pagine in memoria centrale. Supponendo che un riferimento alla memoria centrale richieda 200 ns e che il tempo di accesso ai registri TLB richieda 20 ns, si determini la percentuale di successo nell'accesso ai TLB (hit ratio dei TLB) per ottenere un tempo medio di accesso di 240 ns.

- [A] 86%
- [B] 90%
- [C] 92%
- [D] 88%

domanda N. 9:

Si consideri un sistema con paginazione su due livelli. Per un processo che richiede 16 pagine, di seguito si riportano la pagina di primo livello (P') e le 4 pagine di secondo livello (P''_0, P''_1, P''_2 e P''_3), le quali hanno rispettivamente indirizzo iniziale 8192, 8704, 9216 e 9728:

P'		P''_0		P''_1		P''_2		P''_3	
0	8192	0	0	0	2048	0	4096	0	6144
1	8704	1	512	1	2560	1	4608	1	6656
2	9216	2	1024	2	3072	2	5120	2	7168
3	9728	3	1536	3	3584	3	5632	3	7680

Si determini l'indirizzo fisico corrispondente all'indirizzo logico $< 3, 0, 502 >$

- [A] 6646
- [B] 5187

[C] 977

[D] 3024

domanda N. 10:

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo First In First Out (FIFO), che assegna ad ogni processo 3 frame di 512 Byte e che richiede 2 Byte per la memorizzazione di un intero. Ponendo attenzione solo agli array A e B, e ricordando che il linguaggio C memorizza gli array 2-dimensionali per righe, si determini il numero di page faults del seguente frammento di codice C. ...

```
int A[1024], B[2][512];
...
N=512;
for (i=0; i<2; i++){
    for (j=0; j<N/2; j++){
        B[i][2*j] = A[i*N+j] + A[i*N+2*j];
    }
}
```

[A] 10

[B] 8

[C] 11

[D] 12

domanda N. 11:

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo least Recently Used (LRU), che assegna ad ogni processo 5 frame. Se determini il numero di page faults per la sequenza di riferimenti alle pagine:

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

[A] 7

[B] 8

[C] 9

[D] 10

domanda N. 12:

Sia dato un disco con 200 tracce e velocità di seek di 1 traccia per ms. All'istante t=0 il sistema operativo sta servendo una richiesta sulla traccia 100 e in coda ci sono richieste per le tracce (85; 90; 120; 150; 165; 195). Successivamente arrivano altre richieste all'istante t=60 per la traccia 135 e all'istante t=80 per la traccia 180. Si calcoli il tempo di ricerca complessivo per servire tutte le richieste secondo la politica LOOK, iniziando in ordine ascendente e trascurando la latenza rotazionale e il tempo di trasferimento.

[A] 195

[B] 180

[C] 205

[D] 230

domanda N. 13:

Si determini la veridicità delle seguenti affermazioni:

- a) L'algoritmo di scheduling del disco Shortest Seek Time First (SSTF), non è adatto ai sistemi interattivi di tipo generale perchè, in generale, esibisce elevati tempi medi di servizio.
- b) Tra le principali configurazioni RAID (livelli 0, 1 e 5) quella di livello 1 è la configurazione che assicura il miglior compromesso tra efficienza negli accessi al disco e affidabilità nella salvaguardia dei dati.

[A] a) falsa ; b) vera

[B] a) vera ; b) falsa

[C] entrambe vere

[D] entrambe false

domanda N. 14:

Si calcoli la probabilità di perdere i dati con un sistema RAID level 1 con 4 dischi, se il Mean Time to Failure (MTF) è di 1000 giorni e il Mean Time to Repair (MTR) è di 10 giorni.

[A] 0.00009

[B] 0.00001

[C] 0.00002

[D] 0.00003

domanda N. 15:

Si determini la veridicità delle seguenti affermazioni:

- a) Il principale svantaggio dello schema di allocazione contigua dei file è l'elevato numero di accessi al disco, soprattutto per quanto riguarda i blocchi logicamente posti alla fine del file.
- b) Nel caso di allocazione di un solo blocco libero, lo schema di gestione dello spazio libero mediante lista concatenata è più efficiente dello schema basato sulla bitmap.

[A] entrambe false

[B] a) falsa ; b) vera

[C] entrambe vere

[D] a) vera ; b) falsa

domanda N. 16:

Sia dato un file system che gestisca blocchi da 1024 byte e indirizzi da 64 bit mediante una allocazione indicizzata dei file. Si calcoli la dimensione massima di un file per questo file system supponendo che i blocchi indice contengono:

- 7 puntatori diretti a blocchi di dati
- 3 puntatori indiretti a blocchi di dati

[A] 166480 byte

[B] 220364 byte

[C] 400384 byte

[D] 363212 byte

=====
===== compito N. 1 del 200106 =====
=====

SVOLTO DA _____ MATRICOLA _____

domanda 1	
domanda 2	
domanda 3	
domanda 4	
domanda 5	
domanda 6	
domanda 7	
domanda 8	
domanda 9	
domanda 10	
domanda 11	
domanda 12	
domanda 13	
domanda 14	
domanda 15	
domanda 16	

- 1.5 punti per ogni risposta esatta
- 0 punti per ogni risposta non data
- 0.5 punti di penalita' per ogni risposta errata

FIRMA _____

=====
== CORSO DI SISTEMI OPERATIVI mod.A (gr. 1) ==
===== con el 200907 =====

domanda N. 1: (corretta)

Si determini la veridicità delle seguenti affermazioni:

- a) Il livello software che contiene le componenti fondamentali di un sistema operativo è chiamato kernel. Al kernel appartengono i moduli per: la gestione della memoria centrale, il controllo dei processi, la gestione dell'I/O e la gestione della memoria secondaria. A questo elenco di componenti fondamentali del kernel va aggiunto solo il sottosistema per la gestione delle interfacce grafiche e per l'interprete del linguaggio di comando.
- b) I sistemi operativi hanno lo scopo primario di permettere alle applicazioni software di interagire con l'hardware sottostante, e di gestire le risorse hardware e software del sistema in maniera efficiente.
- [A] entrambe false
[B] entrambe vere
[C] a) falsa ; b) vera
[D] a) vera ; b) falsa

domanda N. 2: (corretta)

Si determini la veridicità delle seguenti affermazioni:

- a) Il *Cambio di contesto* (detto anche Context Switch) è l'insieme delle operazioni necessarie ad arrestare l'esecuzione di un processo P_1 *running* e iniziare l'esecuzione di un processo P_2 *ready*. A tal fine il kernel deve prima memorizzare nel Process Control Block di P_1 il relativo contesto di esecuzione e caricare dal Process Control Block di P_2 il contesto del processo che deve andare in esecuzione.

b) Un cambio di contesto eseguito troppo frequentemente puo' deteriorare la performance complessiva del sistema. Nonostante ciò tale operazione è una operazione sempre presente in tutti i sistemi operativi con multiprogrammazione. ✓

[A] a) vera ; b) falsa

[B] entrambe vere

[C] a) falsa ; b) vera

[D] entrambe false

domanda N. 3:

Si determini il numero di processi generati (compreso quello mandato in esecuzione dalla linea di comando) dal seguente codice C, compilato e fatto eseguire su un sistema con il sistema operativo Unix:

```
main(){
    int pid, i;
    for (i = 0; i <= 3; i + +){
        pid=fork();
        if (pid == 0 && i > 1) exit(0);
    }
    exit(0);
}
```

[A] 9

[B] 8

[C] 10

[D] 12

domanda N. 4:

Sia dato il seguente insieme di processi:

	T_A	T_C
P_1	0	8
P_2	1	4
P_3	3	6
P_4	7	2
P_5	10	2

dove T_A è il tempo di arrivo nella coda dei processi pronti e T_C è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling dello Shortest Job First con prelazione, si determini il tempo medio di esecuzione.

[A] 8.0

[B] 8.4

[C] 8.6

[D] 8.2

domanda N. 5:

Sia dato il seguente insieme di processi:

	T_A	T_C
P_1	0	3
P_2	1	5
P_3	3	7
P_4	8	5
P_5	10	6

dove T_A è il tempo di arrivo nella coda dei processi pronti e T_C è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo $q = 2$, si determini il tempo medio di esecuzione.

[A] 13.6

[B] 14

[C] 14.2 $\Rightarrow \times$

[D] 13.8

domanda N. 6:

Siano dati tre processi che condividono una variabile x e i cui pseudo codici sono di seguito riportati:

P_1	P_2	P_3
$\text{wait}(R)$	$\text{wait}(S)$	$\text{wait}(T)$
$x = x+2;$	$x = x^*2;$	$x = x+1;$
$\text{if } (x > 0) \text{ then}$	$\text{signal}(T);$	$\text{signal}(S);$
$x = -1;$	$\text{wait}(S);$	$\text{wait}(T)$
endif	$x = -x;$	$x = x+1$
$\text{print } x;$	$\text{signal}(R);$	$\text{signal}(S);$

Si determini l'output del processo P_1 supponendo che inizialmente $x = 1$ e i tre semafori binari siano inizializzati rispettivamente $R = 0$, $S = 0$, $T = 1$.

[A] -2

[B] -1

[C] -3

[D] -4

domanda N. 7:

Si determini la veridicità delle seguenti affermazioni:

- Nessuna delle principali strategie di allocazione contigua della memoria con partizioni variabili (strategia first fit, strategia best-fit e strategia worst-fit) comporta problemi di frammentazione interna. F
- Il principale svantaggio della strategia first fit rispetto alle altre due strategie di allocazione contigua della memoria è il maggiore overhead. ✓

~~A~~ a) falsa ; b) vera

- [B] entrambe false
 [C] entrambe vere
 [D] a) vera ; b) falsa

domanda N. 8:

Sia dato un sistema che gestisce una memoria di 5120 byte mediante partizioni variabili. Si supponga che nella memoria siano già presenti 5 processi di dimensioni 740, 875, 250, 1450 e 490 byte rispettivamente allocati negli indirizzi fisici 2048, 3072, 2788, 0 e 4096. Si determini l'indirizzo in cui verrà allocato un ulteriore processo di 389 byte mediante la tecnica del *best fit*.

- [A] 3038
 [B] 3947
~~[C]~~ 4586
 [D] 1450

domanda N. 9:

Si consideri un sistema con paginazione su due livelli. Per un processo che richiede 16 pagine, di seguito si riportano la tabella delle pagine di primo livello (P') e le 4 tabelle di secondo livello (P''_0, P''_1, P''_2 e P''_3), le quali hanno rispettivamente indirizzo iniziale 8192, 8704, 9216 e 9728:

P'	P''_0	P''_1	P''_2	P''_3
0	8192	0	0	0
1	8704	1	2560	1
2	9216	2	3072	2
3	9728	3	3584	3

Si determini l'indirizzo fisico corrispondente all'indirizzo logico $< 2, 1, 248 >$

- [A] 1272
 [B] 3320

[C] 4856

[D] 6904

domanda N. 10:

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo First In First Out (FIFO), che assegna ad ogni processo 3 frame di 512 Byte e che richiede 2 Byte per la memorizzazione di un intero. Ponendo attenzione solo agli array A, B e C, e ricordando che il linguaggio C memorizza gli array 2-dimensional per righe, si determini il numero di page faults del seguente frammento di codice C. ...

int A[2][512], B[2][256], C[512];

...

```
N=256;  
for (i=0; i<2; i++){  
    for (j=0; j<N; j++){  
        C[i*N+j] = A[i][2*j] + B[i][j];  
    } }
```

[A] 12

[B] 8

[C] 10

[D] 11

domanda N. 11: 

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo Least Recently Used (LRU), che assegna ad ogni processo 5 frame. Se de-termini il numero di page faults per la sequenza di riferimenti alle pagine:

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

[A] 7

[B] 9

[C] 8

[D] 10

domanda N. 12:

Sia dato un disco con 200 tracce e velocità di seek di 1 traccia per ms. All'istante t=0 il sistema operativo sta servendo una richiesta sulla traccia 50 e in coda ci sono richieste per le tracce (15; 30; 105; 120; 52; 85). Successivamente arrivano altre richieste all'istante t=25 per la traccia 70 e all'istante t=150 per la traccia 45. Si calcoli il tempo di ricerca complessivo per servire tutte le richieste secondo la politica LOOK, iniziando in ordine ascendente e trascurando la latenza rotazionale e il tempo di trasferimento.

[A] 305

[B] 205

[C] 234

[D] 225

domanda N. 13:

Si determini la veridicità delle seguenti affermazioni:

- a) L'algoritmo di scheduling del disco First Come First Served (FCFS) è l'algoritmo meno utilizzato nei sistemi interattivi di tipo generale perché è quello che ha il più alto overhead per la gestione delle strutture dati necessarie all'algoritmo stesso.
- b) I principali vantaggi di un sistema RAID 0 sono la semplicità di realizzazione e l'elevata velocità di trasferimento.

[A] a) falsa ; b) vera

[B] entrambe false

[C] a) vera ; b) falsa

[D] entrambe vere

domanda N. 14:

Si consideri un disco che opera a 12000 rpm (giri al minuto), dove ogni traccia è composta da 80 settori di 512 bytes ognuno. Si determini la velocità di trasferimento necessaria per poter leggere o scrivere una traccia intera durante una rotazione del disco.

[A] circa 9600 KB/sec

[B] circa 19200 KB/sec

[C] circa 8000 KB/sec

[D] circa 16000 KB/sec

domanda N. 15:

Si determini la veridicità delle seguenti affermazioni:

- a) Lo schema di allocazione basato sulla File Allocation Table (FAT) di Microsoft è uno schema che, con poca richiesta di memoria (necessaria solo per la tabella) e con pochi accessi al disco assicura una gestione efficiente del file system anche se di grandi dimensioni. Per tale motivo è ancora alla base di tutti i file system dei sistemi operativi Microsoft.
- b) Se il numero di blocchi usati in un file system è molto elevato, la memoria richiesta per la gestione dei blocchi liberi mediante lista concatenata può essere inferiore a quella richiesta dalla bitmap, nonostante quest'ultima usi solo un bit per blocco.

[A] entrambe vere

[B] a) vera ; b) falsa

[C] entrambe false

[D] a) falsa ; b) vera

domanda N. 16:

Sia dato un file system di 2 MB che gestisca i file mediante allocazione tabellare (ad es. una File Allocation Table). Si determini la dimensione della tabella nel caso in cui i blocchi del disco siano di 1024 byte.

[A] ~ 6 KB

[B] ~ 64 KB

[C] ~ 16 KB

[D] ~ 2.7 KB

domanda N. 17:

Si determini la veridicità delle seguenti affermazioni:

- a) Si consideri un file formato da 100 record allocato su disco in forma contigua dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se l'ultimo record letto è il record 30 allora per cancellare il 70-mo record sono necessari 40 accessi in lettura e 1 in scrittura.
- b) Si consideri un file formato da 100 record allocato su disco in forma concatenata dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se l'ultimo record letto è il record 10, allora per cancellare il 35-mo record sono necessari 25 accessi in lettura e 25 in scrittura.

[A] entrambe false

[B] entrambe vere

[C] a) vera ; b) falsa

[D] a) falsa ; b) vera

[D] 6

domanda N. 3:

Sia dato il seguente insieme di processi:

	T_A	T_C
P_1	0	7
P_2	1	7
P_3	7	7
P_4	11	3
P_5	13	2

dove T_A è il tempo di arrivo nella coda dei processi pronti e T_C è il tempo di utilizzo della CPU.
Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo $q = 2$, si determini il tempo medio di esecuzione.

[A] 14.4

[B] 14.8

15 16

[D] 14.6

domanda N. 4:

Siano dati tre processi che condividono una variabile x e i cui pseudo codici sono di seguito riportati:

P_1	P_2	P_3
$wait(R)$	$wait(S)$	$wait(T)$
$x = x+3;$	$x = x-5;$	$x = x+2;$
$if(x==4)then$	$signal(R);$	$if(x >= 5)then$
$ signal(S);$	$wait(S);$	$ signal(S);$
$endif$	$signal(T);$	$else$
$wait(R);$		$ signal(R);$
		$endif$
		$wait(T);$
		$signal(R);$

Supponendo che i tre semafori binari R , S e T siano inizializzati rispettivamente a $R = 0$, $S = 0$ e $T = 1$, si determini quale valore iniziale della variabile x , tra quelli proposti nelle risposte, non comporti una situazione di stallo per i processi.

x = 4 16

[B] x = 1

[C] $x = 3$

[D] $x = 2$

domanda N. 5:

Si consideri un sistema di paginazione con tabella delle pagine in memoria centrale. Supponendo che un riferimento alla memoria centrale richieda 190 ns e che il tempo di accesso ai registri TLB richieda 25 ns, si determini la percentuale di successo nell'accesso ai TLB (hit ratio dei TLB) per ottenere un tempo medio di accesso di 272 ns.

[A] 68%

70% ok

[C] 73.5%

[D] 76.6666%

$$EAT = (\alpha + \beta) E + (\alpha + 2\beta)(E - 1)$$

$$TTL - \alpha \alpha = 25 \text{ ns}$$

$$M.C. - \alpha \beta = 190 \text{ ns}$$

$$EAT = 272 \text{ ns}$$

domanda N. 6:

Si consideri un sistema con paginazione su due livelli. Per un processo che richiede 16 pagine, ognuna di 512 byte, di seguito si riportano la tabella delle pagine di primo livello (P') e le 4 tabelle di secondo livello (P''_0, P''_1, P''_2 e P''_3), le quali hanno rispettivamente indirizzo fisico iniziale 0, 512, 1024 e 1536:

P'		P''_0		P''_1		P''_2		P''_3	
0	0	0	8192	0	10752	0	5632	0	9216
1	512	1	11776	1	7680	1	10240	1	3584
2	1024	2	5120	2	3072	2	13824	2	12800
3	1536	3	14336	3	6144	3	11264	3	14848

Si determini quale indirizzo fisico, tra quelli proposti, non appartiene allo spazio di indirizzamento del processo.

La risposta esatta è la D poichè con 512 byte

[A] 11266 **abbiamo allocazione a 2^9 . L'indirizzo 7666 non è**

[B] 14338 **raggiungibile da nessuno degli spazi aggiungendo al**

~~[C] 3074 NO~~ **più 9 gli altri sì**

~~[D] 7666~~

*← Ok perché l'indirizzo viene a 1680 ma
7666 < 7680 e non si possono dare altre
indirizzi.*

domanda N. 7:

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo First In First Out (FIFO), che assegna ad ogni processo 3 frame di 512 Byte e che richiede 2 Byte per

la memorizzazione di un intero. Ponendo attenzione solo agli array A, B e C, e ricordando che il linguaggio C memorizza gli array 2-dimensional per righe, si determini il numero di page faults del seguente frammento di codice C. ...

```
int A[2][512], B[512], C[512];
```
N=512;
for (i=0; i<2; i++){
 for (j=0; j<N; j++){
 B[j/2 + i*256] = A[i][j] + C[j];
 }
}
```

[A] 8

~~B~~ 1160

[C] 10

[D] 12

OK

#### domanda N. 8:

Sia dato un disco con 200 tracce e velocità di seek di 1 traccia per ms. All'istante t=0 il sistema operativo sta servendo una richiesta sulla traccia 50 e in coda ci sono richieste per le tracce (58; 15; 23; 25; 65; 55). Successivamente arrivano altre richieste all'istante t=7 per la traccia 49 e all'istante t=50 per la traccia 57. Si calcoli il tempo di ricerca complessivo per servire tutte le richieste secondo la politica Shortest Seek Time First (SSTF), trascurando la latenza rotazionale e il tempo di trasferimento.

[A] 98

~~B~~ 107 OK

[C] 65

[D] 117

#### domanda N. 9:

Si calcoli la probabilità di perdere i dati con un sistema RAID level 5 con 5 dischi, se il Mean Time to Failure (MTF) è di 1500 giorni e il Mean Time to Repair (MTR) è di 24.75 giorni.

[A] 0.00011

[B] 0.0002

[C] 0.00003

[D] 0.00009

~~MTR = 1600~~  
~~MTR = 23.04~~

164

$$P_{failure} = \frac{D(D-1)(MTR)}{2(MTF^2)}$$

$$P_{failure} = \frac{D}{MTF}$$

$$P_{failure} = \frac{D(MTR)}{2(MTF^2)}$$

**domanda N. 10:**

Sia dato un file system che gestisca blocchi da 256 byte e indirizzi da 32 bit mediante una allocazione indicizzata dei file. Si calcoli la dimensione massima di un file per questo file system supponendo che i blocchi indice contengono:

- 7 puntatori diretti a blocchi di dati
- 3 puntatori indiretti a blocchi di dati

[A] 266240 byte *No*

[B] 330240 byte

[C] 67584 byte

*[D] 50944 byte* *OK*

$$\begin{aligned} \text{Blocchi: } & 256 \text{ byte} \\ \text{Indirizzi: } & 4 \text{ byte} \\ 256/\text{a} = 64 & \text{ indirizzi} \times 32 \text{ bit} \end{aligned}$$

$$\begin{aligned} 7 \cdot 256 &= 1792 \\ 3 \cdot 64 \cdot 256 &= 49152 \\ &\underline{-} \\ &50944 \end{aligned}$$

**domanda N. 11:**

In un sistema operativo multimediale sono in esecuzione tre processi periodici *A*, *B* e *C* con le seguenti caratteristiche:

|          | Periodo | Burst |
|----------|---------|-------|
| <i>A</i> | 30      | 10    |
| <i>B</i> | 40      | 10    |
| <i>C</i> | 20      | 5     |

dove i tempi sono espressi in ms. Si determini l'istante in cui termina il 5-mo burst del processo *A* schedulando i processi con l'algoritmo Rate Monotonic Schedule (RMS).

[A] 140

[B] 145

*[C] 135* *OK*

*[D] 130* *No*

**domanda N. 12:**

Sia dato un sistema multimediale con algoritmo di scheduling RMS, dove già sono in esecuzione tre processi periodici *A*, *B* e *C* con caratteristiche

- A periodo = 20ms e burst di CPU = 5ms
- B periodo = 50ms e burst di CPU = 10ms
- C periodo = 30ms e burst di CPU = 3ms

Si determini quale può essere al più il burst di CPU di un quarto processo D il cui periodo è di 35 ms.

- [A] 4.46 ms  
[B] 9.73 ms  
[C] 7.245 ms *OK*  
[D] 6.755 ms

$$\text{se} \quad \text{burst} \leq \frac{1}{m} \quad m(t^{1/m} - 1) < 1$$

$$\frac{2}{35} \cdot \frac{10}{50} + \frac{2}{35} = 0,55$$

$$4(\sqrt[4]{2} - 1) = 0,456$$

$$\rightarrow 0,55 + \frac{x}{35} < 0,456 \rightarrow x < 1,665$$

domanda N. 13:

Si consideri un sistema operativo con un algoritmo di paginazione simile a quello di Windows XP con dimensione massima del working set di 3 frame, ed un tempo di permanenza tra le pagine in attesa di al più 3 unità di tempo. Si determini il numero di page faults e di soft faults per la seguente stringa di riferimento:

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

- [A] 9 page faults e 6 soft faults  
*OK*  
[B] 10 page faults e 5 soft faults *OK*  
[C] 9 page faults e 5 soft faults *NO*  
[D] 10 page faults e 6 soft faults

domanda N. 14:

Un piccolo sistema distribuito con file system AFS-3 (Andrew File System 3a versione) è composto da un file server S a cui accedono due client C1 e C2. A partire dall'istante 0 ms, il client C1 accede in lettura ad un file *test* memorizzato su S ad intervalli di tempo regolari di 15 minuti, mentre C2 vi accede in scrittura ogni 49 minuti. Si determini quante volte il file *test* viene trasferito da S a C1 nell'arco di 4 ore. Si consideri trascurabile il tempo di trasferimento del file dal server ai client.

- [A] 1  
[B] 12  
[C] 5  
*OK*

domanda N. 15:

Un sistema operativo impegna l'algoritmo del banchiere per controllare l'allocation di 7 unità

notra. I tre processi  $P_1$ ,  $P_2$  e  $P_3$  ne hanno necessità di al più 7, 3 e 5 rispettivamente per completare la loro esecuzione. Quante unità il sistema operativo può allocare in modo sicuro al processo  $P_1$ , se l'attuale stato di allocazione è il seguente: 2, 1 e 1 unità nastro sono allocate ai processi  $P_1$ ,  $P_2$  e  $P_3$  rispettivamente.

- [A] 0 ←
- [B] 1
- [C] 2 ○
- [D] 3

**domanda N. 16:**

Si consideri un file formato da 100 record allocato su disco in forma contigua dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se l'ultimo record letto è il record 10 quanti accessi in lettura e in scrittura sono necessari per cancellare il 75-esimo record?

- [A] 65 accessi in lettura e 1 in scrittura.
- [B] 1 accesso in lettura e 65 in scrittura.
- (C) 25 accessi in lettura e 25 in scrittura +10%
- [D] 1 accesso in lettura e 1 in scrittura no

---

**domanda N. 1:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 60    |
| $P_2$ | 0     | 120   |
| $P_3$ | 0     | 100   |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU espresso in ms. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 20$  ms, si determini quanto deve durare al piu' il cambio di contesto se si vuole che il tempo medio di completamento dei tre processi sia al piu' 290 ms.

- [A] 5.41 ms
- [B] 4.29 ms
- [C] 3.35 ms
- [D] 6.13 ms 

**domanda N. 2:**

Si determini il numero di processi generati (compreso quello mandato in esecuzione dalla linea di comando) dal seguente codice C, compilato e fatto eseguire su un sistema con il sistema operativo Unix:

```
main(){
 int pid, i;
 for (i = 0; i <= 3; i ++){
 pid=fork();
 if (pid == 0 && i > 1) exit(0);
 }
 exit(0);
}
```

- [A] 8
- [B] 10
- [C] 12 

**domanda N. 3:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 7     |
| $P_2$ | 1     | 7     |
| $P_3$ | 7     | 7     |
| $P_4$ | 11    | 3     |
| $P_5$ | 13    | 2     |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 2$ , si determini il tempo medio di esecuzione.

- [A] 14.4
- [B] 14.8
- [C] 15 
- [D] 14.6

**domanda N. 4:**

Siano dati tre processi che condividono una variabile  $x$  e i cui pseudo codici sono di seguito riportati:

| $P_1$                     | $P_2$        | $P_3$        |
|---------------------------|--------------|--------------|
| $wait(R)$                 | $wait(S)$    | $wait(T)$    |
| $x = 2*x;$                | $x = x*2;$   | $x = x+1;$   |
| $if (x > 0) \text{ then}$ | $signal(T);$ | $signal(S);$ |
| $x = x+1;$                |              | $wait(T)$    |
| $signal(S);$              |              | $x = x-1$    |
| $else$                    |              | $signal(R);$ |
| $x = x-1;$                |              |              |
| $signal(T);$              |              |              |
| $endif$                   |              |              |
| $wait(R);$                |              |              |
| $print x;$                |              |              |

Si determini l'output del processo  $P_1$  supponendo che inizialmente  $x = -1$  e i tre semafori binari siano inizializzati rispettivamente  $R = 1$ ,  $S = 0$ ,  $T = 0$ .

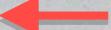
- [A] 5
- [B] 4

[C] -5 

[D] -4

**domanda N. 5:**

Si consideri un sistema di paginazione con tabella delle pagine in memoria centrale. Supponendo che un riferimento alla memoria centrale richieda 190 ns e che il tempo di accesso ai registri TLB richieda 25 ns, si determini la percentuale di successo nell'accesso ai TLB (hit ratio dei TLB) per ottenere un tempo medio di accesso di 272 ns.

[A] 70% 

[B] 73.5%

[C] 76.6666%

[D] 68%

**domanda N. 6:**

Si consideri un sistema con paginazione su due livelli. Per un processo che richiede 16 pagine, di seguito si riportano la tabella delle pagine di primo livello ( $P'$ ) e le 4 tabelle di secondo livello ( $P''_0$ ,  $P''_1$ ,  $P''_2$  e  $P''_3$ ), le quali hanno rispettivamente indirizzo iniziale 8192, 8704, 9216 e 9728:

| $P'$ |      | $P''_0$ |      | $P''_1$ |      | $P''_2$ |      | $P''_3$ |      |
|------|------|---------|------|---------|------|---------|------|---------|------|
| 0    | 8192 | 0       | 0    | 0       | 2048 | 0       | 4096 | 0       | 6144 |
| 1    | 8704 | 1       | 512  | 1       | 2560 | 1       | 4608 | 1       | 6656 |
| 2    | 9216 | 2       | 1024 | 2       | 3072 | 2       | 5120 | 2       | 7168 |
| 3    | 9728 | 3       | 1536 | 3       | 3584 | 3       | 5632 | 3       | 7680 |

Si determini l'indirizzo fisico corrispondente all'indirizzo logico  $< 0,3,408 >$

[A] 4731

[B] 1944 

[C] 7803

[D] 6267

**domanda N. 7:**

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo First In First Out (FIFO), che assegna ad ogni processo 3 frame di 512 Byte e che richiede 2 Byte per la memorizzazione di un intero. Ponendo attenzione solo agli array A, B e C, e ricordando che

il linguaggio C memorizza gli array 2-dimensionali per righe, si determini il numero di page faults del seguente frammento di codice C. ...

```
int A[512], B[2][256], C[2][512];
...
N=256;
for (i=0; i<2; i++){
 for (j=0; j<N; j++){
 C[i][2*j] = A[i*N+j] + B[i][j];
 }
}
```

[A] 11

[B] 12 

[C] 8

[D] 10

**domanda N. 8:**

Sia dato un disco con 2000 tracce. Una operazione di lettura di un blocco richiede 3 ms per lo spostamento tra due tracce adiacenti, 5 ms per la latenza rotazionale e 10 ms per il trasferimento di un blocco di dati. Si calcoli il tempo necessario per leggere un file costituito da 45 blocchi nel caso in cui questi ultimi siano distanti mediamente 15 tracce l'uno dall'altro.

[A] 2.7 sec 

[B] 12.15 sec

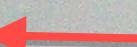
[C] 2.04 sec

[D] 8.44 sec

**domanda N. 9:**

Si consideri un disco che opera a 12000 rpm (giri al minuto), dove ogni traccia è composta da 160 settori di 1024 byte ognuno. Si determini la velocità di trasferimento necessaria per poter leggere o scrivere una traccia intera durante una rotazione del disco.

[A] circa 8000 KB/sec

[B] circa 32000 KB/sec 

[C] circa 16000 KB/sec

[D] circa 9600 KB/sec

**domanda N. 10:**

Qual'è la massima dimensione della partizione del disco per un sistema operativo MS-DOS che utilizza il file system FAT-16 con blocchi del disco da 32KB?

- [A] 1 GB
- [B] 4 GB
- [C] 512 MB
- [D] 2 GB 

**domanda N. 11:**

In un sistema operativo multimediale sono in esecuzione tre processi periodici *A*, *B* e *C* con le seguenti caratteristiche:

|          | <i>Periodo</i> | <i>Burst</i> |
|----------|----------------|--------------|
| <i>A</i> | 40             | 10           |
| <i>B</i> | 30             | 5            |
| <i>C</i> | 50             | 25           |

dove i tempi sono espressi in ms. Si determini quale processo è in esecuzione all'istante  $t=135$ , schedulando i processi con l'algoritmo Earliest Deadline First (EDF).

- [A] il processo *B*
- [B] il processo *C*
- [C] il processo *A* 
- [D] la CPU è inattiva

**domanda N. 12:**

Un piccolo server di una stazione di tv on demand usa uno schema di schedulazione del disco statica round per video con 25 frame/sec e 20 KB per frame. I flussi provengono da un disco con le seguenti caratteristiche:

- tempo medio di seek di 2 ms
- velocità di trasferimento di 420 Mbit/sec
- Latenza rotazionale trascurabile

Quanti flussi può supportare il server, supponendo che i frame sono memorizzati uno per blocco in tracce consecutive?

- [A] 11 flussi

[B] 10 flussi

[C] 16 flussi

[D] 18 flussi



**domanda N. 13:**

Si consideri un sistema operativo con un algoritmo di paginazione simile a quello di Windows XP con dimensione massima del working set di 3 frame, ed un tempo di permanenza tra le pagine in attesa di al più 3 unità di tempo. Si determini il numero di page faults e di soft faults per la seguente stringa di riferimento:

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

[A] 9 page faults e 7 soft faults



[B] 10 page faults e 7 soft faults

[C] 9 page faults e 8 soft faults

[D] 10 page faults e 8 soft faults

**domanda N. 14:**

Un piccolo sistema distribuito con file system AFS-3 (Andrew File System 3a versione) è composto da un file server S a cui accedono due client C1 e C2. A partire dall'istante  $t=0$ , il client C1 accede in lettura ad un file *test* memorizzato su S ad intervalli di tempo regolari di 27 minuti, mentre C2 vi accede in scrittura ogni 35 minuti. Si determini quante volte il file *test* viene trasferito da S a C1 nell'arco di 4 ore. Si consideri trascurabile il tempo di trasferimento del file dal server ai client.

[A] 1

[B] 6

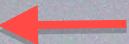
[C] 7



[D] 9

**domanda N. 15:**

Un sistema operativo impegna l'algoritmo del banchiere per controllare l'allocazione di 7 unità nastro. I tre processi  $P_1$ ,  $P_2$  e  $P_3$  ne hanno necessità di al più 7, 3 e 5 rispettivamente per completare la loro esecuzione. Quante unità il sistema operativo può allocare in modo sicuro al processo  $P_1$ , se l'attuale stato di allocazione è il seguente: 1, 2 e 1 unità nastro sono allocate ai processi  $P_1$ ,  $P_2$  e  $P_3$  rispettivamente.

- [A] 0
- [B] 2
- [C] 3
- [D] 1 

**domanda N. 16:**

Si consideri un file formato da 100 record allocato su disco in forma contigua dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se l'ultimo record letto e' il record 10 quanti accessi in lettura e in scrittura sono necessari per cancellare il 60-mo record?

- [A] 50 accessi in lettura e 50 in scrittura
- [B] 40 accessi in lettura e 40 in scrittura 
- [C] 40 accessi in lettura e 1 in scrittura.
- [D] 50 accessi in lettura e 1 in scrittura

SVOLTO DA \_\_\_\_\_ MATRICOLA \_\_\_\_\_

|            |   |
|------------|---|
| domanda 1  | D |
| domanda 2  | C |
| domanda 3  | C |
| domanda 4  | C |
| domanda 5  | A |
| domanda 6  | B |
| domanda 7  | B |
| domanda 8  | A |
| domanda 9  | B |
| domanda 10 | D |
| domanda 11 | C |
| domanda 12 | C |
| domanda 13 | A |
| domanda 14 | C |
| domanda 15 | D |
| domanda 16 | B |

- 1.5 punti per ogni risposta esatta
- 0 punti per ogni risposta non data
- 0.5 punti di penalita' per ogni risposta errata

FIRMA \_\_\_\_\_

=====  
==== CORSO DI SISTEMI OPERTIVI -9CFU (gr. 1) ====  
===== compito N. 10 del 220610 =====

**domanda N. 1:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 70    |
| $P_2$ | 0     | 110   |
| $P_3$ | 0     | 90    |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU espresso in ms. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 20$  ms, si determini quanto deve durare al piu' il cambio di contesto se si vuole che il tempo medio di completamento dei tre processi sia al piu' 280 ms.

[A] 3.33 ms 

[B] 4.41 ms

[C] 2.29 ms

[D] 1.35 ms

**domanda N. 2:**

Si determini il numero di processi generati (compreso quello mandato in esecuzione dalla linea di comando) dal seguente codice C, compilato e fatto eseguire su un sistema con il sistema operativo Unix:

```
main(){
 int pid, i;
 for (i = 0; i <= 3; i ++){
 pid=fork();
 if (pid == 0 && i == 2) exit(0);
 }
 exit(0);
}
```

[A] 8

[B] 10

[C] 14

**domanda N. 3:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 10    |
| $P_2$ | 3     | 6     |
| $P_3$ | 5     | 6     |
| $P_4$ | 9     | 4     |
| $P_5$ | 13    | 2     |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 2$ , si determini il tempo medio di esecuzione.

[A] 18.8

[B] 17.6

[C] 18.0 

[D] 18.4

**domanda N. 4:**

Siano dati tre processi che condividono una variabile  $x$  e i cui pseudo codici sono di seguito riportati:

| $P_1$               | $P_2$           | $P_3$        |
|---------------------|-----------------|--------------|
| $wait(R)$           | $wait(S)$       | $wait(T);$   |
| $x = x+2;$          | $x = x+4;$      | $x = x-5;$   |
| $if(x \geq 4) then$ | $if(x==1) then$ | $signal(S);$ |
| $signal(S);$        | $signal(T);$    | $wait(T);$   |
| $else$              | $endif$         | $signal(R);$ |
| $signal(T);$        | $wait(S);$      |              |
| $endif$             |                 |              |
| $wait(R);$          |                 |              |
| $signal(S);$        |                 |              |

Supponendo che i tre semafori binari  $R$ ,  $S$  e  $T$  siano inizializzati rispettivamente a  $R = 1$ ,  $S = 0$  e  $T = 0$ , si determini quale valore iniziale della variabile  $x$ , tra quelli proposti nelle risposte, non comporti una situazione di stallo per i processi.

[A]  $x = 3$

[B]  $x = 2$

[C]  $x = 0$  

[D]  $x = 1$

**domanda N. 5:**

Sia dato un sistema che gestisce una memoria di 5120 byte mediante partizioni variabili. Si supponga che nella memoria siano già presenti 5 processi di dimensioni 740, 875, 250, 1450 e 490 byte allocati rispettivamente a partire dagli indirizzi fisici 2048, 3072, 2788, 0 e 4096. Si determini l'indirizzo in cui verrà allocato un ulteriore processo di 389 byte mediante la tecnica del *worst fit*.

[A] 4586

[B] 3038

[C] 3947

[D] 1450 

**domanda N. 6:**

Si consideri un sistema con paginazione su due livelli. Per un processo che richiede 16 pagine, ognuna di 512 byte, di seguito si riportano la tabella delle pagine di primo livello ( $P'$ ) e le 4 tabelle di secondo livello ( $P''_0$ ,  $P''_1$ ,  $P''_2$  e  $P''_3$ ), le quali hanno rispettivamente indirizzo fisico iniziale 0, 512, 1024 e 1536:

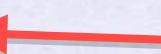
| $P'$ |      | $P''_0$ |       | $P''_1$ |       | $P''_2$ |       | $P''_3$ |       |
|------|------|---------|-------|---------|-------|---------|-------|---------|-------|
| 0    | 0    | 0       | 8192  | 0       | 10752 | 0       | 5632  | 0       | 9216  |
| 1    | 512  | 1       | 11776 | 1       | 7680  | 1       | 10240 | 1       | 3584  |
| 2    | 1024 | 2       | 5120  | 2       | 3072  | 2       | 13824 | 2       | 12800 |
| 3    | 1536 | 3       | 14336 | 3       | 6144  | 3       | 11264 | 3       | 14848 |

Si determini quale indirizzo fisico, tra quelli proposti, non appartiene allo spazio di indirizzamento del processo.

[A] 6146

[B] 9218

[C] 14850

[D] 3070 

**domanda N. 7:**

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo First In

First Out (FIFO), che assegna ad ogni processo 3 frame di 512 Byte e che richiede 2 Byte per la memorizzazione di un intero. Ponendo attenzione solo agli array A, B e C, e ricordando che il linguaggio C memorizza gli array 2-dimensional per righe, si determini il numero di page faults del seguente frammento di codice C. ...

```
int A[2][512], B[512], C[256];
...
N=512;
for (i=0; i<2; i++){
 for (j=0; j<N; j++){
 B[j/2 + i*256] = A[i][j] + C[j/2];
 } }
```

- [A] 8
- [B] 12
- [C] 10
- [D] 11

**domanda N. 8:**

Sia dato un disco con 200 tracce e velocità di seek di 1 traccia per ms. All'istante  $t=0$  il sistema operativo sta servendo una richiesta sulla traccia 50 e in coda ci sono richieste per le tracce (58; 15; 23; 25; 65; 55). Successivamente arrivano altre richieste all'istante  $t=7$  per la traccia 49 e all'istante  $t=50$  per la traccia 57. Si calcoli il tempo di ricerca complessivo per servire tutte le richieste secondo la politica Shortest Seek Time First (SSTF), trascurando la latenza rotazionale e il tempo di trasferimento.

- [A] 107
- [B] 65
- [C] 117
- [D] 98

**domanda N. 9:**

Si consideri un disco che opera a 12000 rpm (giri al minuto), dove ogni traccia è composta da 80 settori di 512 bytes ognuno. Si determini la velocità di trasferimento necessaria per poter leggere o scrivere una traccia intera durante una rotazione del disco.

- [A] circa 19200 KB/sec
- [B] circa 8000 KB/sec
- [C] circa 16000 KB/sec

**domanda N. 10:**

Qual'e' la massima dimensione della partizione del disco per un sistema operativo MS-DOS che utilizza il file system FAT-16 con blocchi del disco da 32KB?

- [A] 4 GB
- [B] 512 MB
- [C] 2 GB** 
- [D] 1 GB

**domanda N. 11:**

In un sistema operativo multimediale sono in esecuzione tre processi periodici *A*, *B* e *C* con le seguenti caratteristiche:

|          | <i>Periodo</i> | <i>Burst</i> |
|----------|----------------|--------------|
| <i>A</i> | 40             | 10           |
| <i>B</i> | 30             | 5            |
| <i>C</i> | 50             | 25           |

dove i tempi sono espressi in ms. Si determini quale processo è in esecuzione all'istante  $t=135$ , schedulando i processi con l'algoritmo Earliest Deadline First (EDF).

- [A] il processo *B*
- [B] il processo *C*
- [C] il processo *A*
- [D] la CPU è inattiva

**domanda N. 12:**

Si consideri un sistema multimediale con un disco gestito secondo l'algoritmo di scheduling EDF/SCAN, dove le richieste sono processate a gruppi di 6. Si supponga quindi che, all'istante  $t = 0$ , siano gia' in coda delle richieste per 3 filmati multimediali, le cui frequenze sono rispettivamente di 1 frame ogni 27 u.t. (unita' di tempo), 1 frame ogni 23 u.t. e 1 frame ogni 33 u.t. Si supponga inoltre, per semplicita', che per  $t = 0$  sono presenti tre richieste (una per ogni filmato) che si riferiscono rispettivamente alle tracce 0, 100 e 200 e che frame consecutivi di uno stesso filmato siano distanti 5 tracce l'uno dall'altro. Quante tracce sono attraversate dalla testina per processare le richieste che devono essere servite entro l'istante  $t=160$ ?

- [A] 680

[B] 500

[C] 580

[D] 620



**domanda N. 13:**

Si consideri un sistema operativo con un algoritmo di paginazione simile a quello di Windows XP con dimensione massima del working set di 3 frame, ed un tempo di permanenza tra le pagine in attesa di al più 3 unità di tempo. Si determini il numero di page faults e di soft faults per la seguente stringa di riferimento:

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

[A] 9 page faults e 7 soft faults

[B] 10 page faults e 7 soft faults

[C] 9 page faults e 8 soft faults

[D] 10 page faults e 8 soft faults

**domanda N. 14:**

Un piccolo sistema distribuito è composto da due nodi su cui sono in esecuzione due processi (denominati P1 e P2) che si sincronizzano per l'accesso ad una regione critica secondo l'algoritmo di Agrawala e Ricart. Il processo P1, a partire dall'istante  $t=0$ , esegue in sequenza per 3 volte una porzione di codice (da eseguirsi in mutua esclusione) che richiede 5 sec ed una porzione di codice (senza vincoli particolari) che richiede 3 sec. Il processo P2 invece, a partire da  $t=0$ , esegue in sequenza per 3 volte una porzione di codice (senza vincoli particolari) che richiede 4 sec ed una porzione di codice (da eseguirsi in mutua esclusione) che richiede 4 sec. Supponendo che il tempo necessario per mandare un messaggio da un nodo all'altro sia di 1 sec, determinare l'istante in cui il processo P1 termina la sua esecuzione.

[A] 24

[B] 36

[C] 32

[D] 30

**domanda N. 15:**

Un sistema operativo impegna l'algoritmo del banchiere per controllare l'allocazione di 7 unità nastro. I tre processi  $P_1$ ,  $P_2$  e  $P_3$  ne hanno necessità di al più 7, 3 e 5 rispettivamente per completare la loro esecuzione. Quante unità il sistema operativo può allocare in modo sicuro al processo  $P_1$ , se l'attuale stato di allocazione è il seguente: 2, 1 e 1 unità nastro sono allocate ai processi  $P_1$ ,  $P_2$  e  $P_3$  rispettivamente.

[A] 2

[B] 3

[C] 0



[D] 1

**domanda N. 16:**

Si consideri un file formato da 100 record allocato su disco in forma concatenata dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se l'ultimo record letto è il record 25, quanti accessi in lettura e in scrittura sono necessari per cancellare il 60-mo record?

[A] 1 accesso in lettura e 1 in scrittura

[B] 40 accessi in lettura e 40 in scrittura

[C] 35 accessi in lettura e 35 in scrittura

[D] 35 accessi in lettura e 1 in scrittura



===== CORSO DI SISTEMI OPERTIVI -9CFU (gr. 1) =====  
===== compito N. 10 del 220610 =====

SVOLTO DA \_\_\_\_\_ MATRICOLA \_\_\_\_\_

|            |   |
|------------|---|
| domanda 1  | A |
| domanda 2  | D |
| domanda 3  | C |
| domanda 4  | C |
| domanda 5  | D |
| domanda 6  | D |
| domanda 7  | A |
| domanda 8  | A |
| domanda 9  | B |
| domanda 10 | C |
| domanda 11 | C |
| domanda 12 | D |
| domanda 13 | A |
| domanda 14 | C |
| domanda 15 | C |
| domanda 16 | D |

- 1.5 punti per ogni risposta esatta
- 0 punti per ogni risposta non data
- 0.5 punti di penalita' per ogni risposta errata

FIRMA \_\_\_\_\_

=====  
==== CORSO DI SISTEMI OPERATIVI -9CFU (gr. 1) =====

===== compito N. 11 del 220610 =====

**domanda N. 1:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 60    |
| $P_2$ | 0     | 120   |
| $P_3$ | 0     | 100   |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU espresso in ms. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 20$  ms, si determini quanto deve durare al piu' il cambio di contesto se si vuole che il tempo medio di completamento dei tre processi sia al piu' 290 ms.

- [A] 5.41 ms
- [B] 4.29 ms
- [C] 3.35 ms
- [D] 6.13 ms

**domanda N. 2:**

Si determini il numero di processi generati (compreso quello mandato in esecuzione dalla linea di comando) dal seguente codice C, compilato e fatto eseguire su un sistema con il sistema operativo Unix:

```
main(){
 int pid, i;
 for (i = 0; i <= 3; i + +){
 pid=fork();
 if (pid == 0 && i > 1) exit(0);
 }
 exit(0);
}
```

- [A] 8
- [B] 10
- [C] 12

[D] 9

domanda N. 3:

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 7     |
| $P_2$ | 1     | 7     |
| $P_3$ | 7     | 7     |
| $P_4$ | 11    | 3     |
| $P_5$ | 13    | 2     |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 2$ , si determini il tempo medio di esecuzione.

- [A] 14.4
- [B] 14.8
- [C] 15
- [D] 14.6

domanda N. 4:

Siano dati tre processi che condividono una variabile  $x$  e i cui pseudo codici sono di seguito riportati:

| $P_1$             | $P_2$        | $P_3$        |
|-------------------|--------------|--------------|
| $wait(R)$         | $wait(S)$    | $wait(T)$    |
| $x = 2^*x;$       | $x = x^*2;$  | $x = x+1;$   |
| $if (x > 0) then$ | $signal(T);$ | $signal(S);$ |
| $x = x+1;$        |              | $wait(T)$    |
| $signal(S);$      |              | $x = x-1$    |
| $else$            |              | $signal(R);$ |
| $x = x-1;$        |              |              |
| $signal(T);$      |              |              |
| $endif$           |              |              |
| $wait(R);$        |              |              |
| $print x;$        |              |              |

Si determini l'output del processo  $P_1$  supponendo che inizialmente  $x = -1$  e i tre semafori binari siano inizializzati rispettivamente  $R = 1$ ,  $S = 0$ ,  $T = 0$ .

- [A] 5
- [B] 4

[C] -5

[D] -4

**domanda N. 5:**

- o Si consideri un sistema di paginazione con tabella delle pagine in memoria centrale. Supponendo che un riferimento alla memoria centrale richieda 190 ns e che il tempo di accesso ai registri TLB richieda 25 ns, si determini la percentuale di successo nell'accesso ai TLB (hit ratio dei TLB) per ottenere un tempo medio di accesso di 272 ns.

[A] 70%

[B] 73.5%

[C] 76.6666%

[D] 68%

**domanda N. 6:**

- o Si consideri un sistema con paginazione su due livelli. Per un processo che richiede 16 pagine, di seguito si riportano la tabella delle pagine di primo livello ( $P'$ ) e le 4 tabelle di secondo livello ( $P''_0, P''_1, P''_2$  e  $P''_3$ ), le quali hanno rispettivamente indirizzo iniziale 8192, 8704, 9216 e 9728:

| $P'$ |      | $P''_0$ |      | $P''_1$ |      | $P''_2$ |      | $P''_3$ |      |
|------|------|---------|------|---------|------|---------|------|---------|------|
| 0    | 8192 | 0       | 0    | 0       | 2048 | 0       | 4096 | 0       | 6144 |
| 1    | 8704 | 1       | 512  | 1       | 2560 | 1       | 4608 | 1       | 6656 |
| 2    | 9216 | 2       | 1024 | 2       | 3072 | 2       | 5120 | 2       | 7168 |
| 3    | 9728 | 3       | 1536 | 3       | 3584 | 3       | 5632 | 3       | 7680 |

Si determini l'indirizzo fisico corrispondente all'indirizzo logico  $< 0, 3, 408 >$

[A] 4731

[B] 1944

[C] 7803

[D] 6267

**domanda N. 7:**

- o Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo First In First Out (FIFO), che assegna ad ogni processo 3 frame di 512 Byte e che richiede 2 Byte per la memorizzazione di un intero. Ponendo attenzione solo agli array A, B e C, e ricordando che

il linguaggio C memorizza gli array 2-dimensional per righe, si determini il numero di page faults del seguente frammento di codice C. ...

```
int A[512], B[2][256], C[2][512];
...
N=256;
for (i=0; i<2; i++){
 for (j=0; j<N; j++){
 C[i][2*j] = A[i*N+j] + B[i][j];
 }
}
```

[A] 11

[B] 12

[C] 8

[D] 10

**domanda N. 8:**

- Sia dato un disco con 2000 tracce. Una operazione di lettura di un blocco richiede 3 ms per lo spostamento tra due tracce adiacenti, 5 ms per la latenza rotazionale e 10 ms per il trasferimento di un blocco di dati. Si calcoli il tempo necessario per leggere un file costituito da 45 blocchi nel caso in cui questi ultimi siano distanti mediamente 15 tracce l'uno dall'altro.

[A] 2.7 sec

[B] 12.15 sec

[C] 2.04 sec

[D] 8.44 sec

**domanda N. 9:**

- Si consideri un disco che opera a 12000 rpm (giri al minuto), dove ogni traccia è composta da 160 settori di 1024 byte ognuno. Si determini la velocità di trasferimento necessaria per poter leggere o scrivere una traccia intera durante una rotazione del disco.

[A] circa 8000 KB/sec

[B] circa 32000 KB/sec

[C] circa 16000 KB/sec

[D] circa 9600 KB/sec

domanda N. 10:

Qual'e' la massima dimensione della partizione del disco per un sistema operativo MS-DOS che utilizza il file system FAT-16 con blocchi del disco da 32KB?

- [A] 1 GB
- [B] 4 GB
- [C] 512 MB
- [D] 2 GB

domanda N. 11:

In un sistema operativo multimediale sono in esecuzione tre processi periodici *A*, *B* e *C* con le seguenti caratteristiche:

|          | Periodo | Burst |
|----------|---------|-------|
| <i>A</i> | 40      | 10    |
| <i>B</i> | 30      | 5     |
| <i>C</i> | 50      | 25    |

dove i tempi sono espressi in ms. Si determini quale processo è in esecuzione all'istante t=135, schedulando i processi con l'algoritmo Earliest Deadline First (EDF).

- [A] il processo *B*
- [B] il processo *C*
- [C] il processo *A*
- [D] la CPU è inattiva

domanda N. 12:

Un piccolo server di una stazione di tv on demand usa uno schema di schedulazione del disco statica round per video con 25 frame/sec e 20 KB per frame. I flussi provengono da un disco con le seguenti caratteristiche:

- tempo medio di seek di 2 ms
- velocità di trasferimento di 420 Mbit/sec
- Latenza rotazionale trascurabile

Quanti flussi può supportare il server, supponendo che i frame sono memorizzati uno per blocco in tracce consecutive?

- [A] 11 flussi

[B] 10 flussi

[C] 16 flussi

[D] 18 flussi

**domanda N. 13:**

- Si consideri un sistema operativo con un algoritmo di paginazione simile a quello di Windows XP con dimensione massima del working set di 3 frame, ed un tempo di permanenza tra le pagine in attesa di al più 3 unità di tempo. Si determini il numero di page faults e di soft faults per la seguente stringa di riferimento:

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

[A] 9 page faults e 7 soft faults

[B] 10 page faults e 7 soft faults

[C] 9 page faults e 8 soft faults

[D] 10 page faults e 8 soft faults

**domanda N. 14:**

- Un piccolo sistema distribuito con file system AFS-3 (Andrew File System 3a versione) è composto da un file server S a cui accedono due client C1 e C2. A partire dall'istante  $t=0$ , il client C1 accede in lettura ad un file *test* memorizzato su S ad intervalli di tempo regolari di 27 minuti, mentre C2 vi accede in scrittura ogni 35 minuti. Si determini quante volte il file *test* viene trasferito da S a C1 nell'arco di 4 ore. Si consideri trascurabile il tempo di trasferimento del file dal server ai client.

[A] 1

[B] 6

[C] 7

[D] 9

**domanda N. 15:**

- Un sistema operativo impegna l'algoritmo del banchiere per controllare l'allocazione di 7 unità nastro. I tre processi  $P_1$ ,  $P_2$  e  $P_3$  ne hanno necessità di al più 7, 3 e 5 rispettivamente per completare la loro esecuzione. Quante unità il sistema operativo può allocare in modo sicuro al processo  $P_1$ , se l'attuale stato di allocazione è il seguente: 1, 2 e 1 unità nastro sono allocate ai processi  $P_1$ ,  $P_2$  e  $P_3$  rispettivamente.

- [A] 0
- [B] 2
- [C] 3
- [D] 1

⌚ domanda N. 16:

Si consideri un file formato da 100 record allocato su disco in forma contigua dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se l'ultimo record letto è il record 10 quanti accessi in lettura e in scrittura sono necessari per cancellare il 60-mo record?

- [A] 50 accessi in lettura e 50 in scrittura
- [B] 40 accessi in lettura e 40 in scrittura
- [C] 40 accessi in lettura e 1 in scrittura.
- [D] 50 accessi in lettura e 1 in scrittura

=====  
==== CORSO DI SISTEMI OPERTIVI -9CFU (gr. 1) =====

===== compito N. 11 del 220610 =====

SVOLTO DA \_\_\_\_\_ MATRICOLA \_\_\_\_\_

|            |   |
|------------|---|
| domanda 1  | D |
| domanda 2  | C |
| domanda 3  | C |
| domanda 4  | C |
| domanda 5  | A |
| domanda 6  | B |
| domanda 7  | B |
| domanda 8  | A |
| domanda 9  | B |
| domanda 10 | D |
| domanda 11 | C |
| domanda 12 | C |
| domanda 13 | A |
| domanda 14 | C |
| domanda 15 | D |
| domanda 16 | B |

- 1.5 punti per ogni risposta esatta
- 0 punti per ogni risposta non data
- 0.5 punti di penalita' per ogni risposta errata

FIRMA \_\_\_\_\_

===== CORSO DI SISTEMI OPERATIVI -9CFU (gr. 1) =====  
 ===== compito N. 18 del 220610 =====

**domanda N. 1:****Come fare domanda 10?**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 60    |
| $P_2$ | 0     | 120   |
| $P_3$ | 0     | 100   |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU espresso in ms. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 20$  ms, si determini quanto deve durare al piu' il cambio di contesto se si vuole che il tempo medio di completamento dei tre processi sia al piu' 290 ms.

- [A] 6.13 ms 
- [B] 5.41 ms
- [C] 4.29 ms
- [D] 3.35 ms

**domanda N. 2:**

Si determini il numero di processi generati (compreso quello mandato in esecuzione dalla linea di comando) dal seguente codice C, compilato e fatto eseguire su un sistema con il sistema operativo Unix:

```
main(){
int pid, i;
for (i = 0; i <= 3; i + +){
 pid=fork();
 if (pid == 0 && i > 1) exit(0);
}
exit(0);
}
```

- [A] 12 
- [B] 9
- [C] 8

[D] 10

**domanda N. 3:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 3     |
| $P_2$ | 1     | 5     |
| $P_3$ | 3     | 7     |
| $P_4$ | 8     | 5     |
| $P_5$ | 10    | 6     |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 2$ , si determini il tempo medio di esecuzione.

[A] 13.6

[B] 14 

[C] 14.2

[D] 13.8

**domanda N. 4:**

Siano dati tre processi che condividono una variabile  $x$  e i cui pseudo codici sono di seguito riportati:

| $P_1$        | $P_2$             | $P_3$        |
|--------------|-------------------|--------------|
| $wait(S)$    | $wait(R)$         | $wait(T)$    |
| $x = 2+x;$   | $if (x < 0) then$ | $x = x+1;$   |
| $signal(T);$ | $x = x-1;$        | $signal(R);$ |
| $wait(S);$   | $signal(S);$      | $wait(T);$   |
| $print\ x;$  | $else$            | $x = x+1;$   |
|              | $x = x+1;$        | $signal(S);$ |
|              | $signal(T);$      |              |
|              | $endif$           |              |

Si determini l'output del processo  $P_1$  supponendo che inizialmente  $x = -1$  e i tre semafori binari siano inizializzati rispettivamente  $S = 1$ ,  $R = 0$ ,  $T = 0$ .

[A] 4 

[B] 3

[C] 5

**domanda N. 5:**

Si consideri un sistema di paginazione con tabella delle pagine in memoria centrale. Supponendo che un riferimento alla memoria centrale richieda 180 ns e che il tempo di accesso ai registri TLB richieda 25 ns, si determini la percentuale di successo nell'accesso ai TLB (hit ratio dei TLB) per ottenere un tempo medio di accesso di 232 ns.

- [A] 82%
- [B] 85% ←**
- [C] 91%
- [D] 88%

**domanda N. 6:**

Si consideri un sistema con paginazione su due livelli. Per un processo che richiede 16 pagine, di seguito si riportano la tabella delle pagine di primo livello ( $P'$ ) e le 4 tabelle di secondo livello ( $P''_0$ ,  $P''_1$ ,  $P''_2$  e  $P''_3$ ), le quali hanno rispettivamente indirizzo iniziale 8192, 8704, 9216 e 9728:

| $P'$ |      | $P''_0$ |      | $P''_1$ |      | $P''_2$ |      | $P''_3$ |      |
|------|------|---------|------|---------|------|---------|------|---------|------|
| 0    | 8192 | 0       | 0    | 0       | 2048 | 0       | 4096 | 0       | 6144 |
| 1    | 8704 | 1       | 512  | 1       | 2560 | 1       | 4608 | 1       | 6656 |
| 2    | 9216 | 2       | 1024 | 2       | 3072 | 2       | 5120 | 2       | 7168 |
| 3    | 9728 | 3       | 1536 | 3       | 3584 | 3       | 5632 | 3       | 7680 |

Si determini l'indirizzo fisico corrispondente all'indirizzo logico  $< 1, 2, 341 >$

- [A] 3413 ←**
- [B] 1877
- [C] 6997
- [D] 5461

**domanda N. 7:**

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo First In First Out (FIFO), che assegna ad ogni processo 3 frame di 512 Byte e che richiede 2 Byte per la memorizzazione di un intero. Ponendo attenzione solo agli array A, B e C, e ricordando che il linguaggio C memorizza gli array 2-dimensionali per righe, si determini il numero di page faults del seguente frammento di codice C. ...

```

int A[2][512], B[512], C[512];
...
N=512;
for (i=0; i<2; i++){
 for (j=0; j<N; j++){
 B[j/2 + i*256] = A[i][j] + C[j];
 }
}

```

- [A] 10
- [B] 12
- [C] 8
- [D] 11

**domanda N. 8:**

Sia dato un disco con 200 tracce e velocità di seek di 1 traccia per ms. All'istante  $t=0$  il sistema operativo sta servendo una richiesta sulla traccia 100 e in coda ci sono richieste per le tracce (10; 30; 55; 80; 92; 130). Successivamente arrivano altre richieste all'istante  $t=40$  per la traccia 50 e all'istante  $t=100$  per la traccia 15. Si calcoli il tempo di ricerca complessivo per servire tutte le richieste secondo la politica LOOK, iniziando in ordine discendente e trascurando la latenza rotazionale e il tempo di trasferimento.

- [A] 285
- [B] 325
- [C] 110
- [D] 215

**domanda N. 9:**

Si consideri un disco che opera a 12000 rpm (giri al minuto), dove ogni traccia è composta da 80 settori di 512 bytes ognuno. Si determini la velocità di trasferimento necessaria per poter leggere o scrivere una traccia intera durante una rotazione del disco.

- [A] circa 9600 KB/sec
- [B] circa 19200 KB/sec
- [C] circa 8000 KB/sec
- [D] circa 16000 KB/sec

**domanda N. 10:**

Sia dato un file system di 8 MB che gestisca i file mediante allocazione tabellare (ad es. una File Allocation Table). Si determini la dimensione della tabella nel caso in cui i blocchi del disco siano di 512 byte.

- [A] 13 KB
- [B] 32 KB
- [C] 28 KB
- [D] 12 KB

**domanda N. 11:**

In un sistema operativo multimediale sono in esecuzione tre processi periodici *A*, *B* e *C* con le seguenti caratteristiche:

|          | <i>Periodo</i> | <i>Burst</i> |
|----------|----------------|--------------|
| <i>A</i> | 30             | 10           |
| <i>B</i> | 20             | 10           |
| <i>C</i> | 40             | 5            |

dove i tempi sono espressi in ms. Si determini l'istante in cui inizia il primo periodo di inattività della CPU schedulando i processi con l'algoritmo Earliest Deadline First (EDF).

- [A] 120
- [B] 125
- [C] 130
- [D] 115

**domanda N. 12:**

Un piccolo server di una stazione di tv on demand usa uno schema di schedulazione del disco statica round per video con 30 frame/sec e 20 KB per frame. I flussi provengono da un disco con le seguenti caratteristiche:

- tempo medio di seek di 1 ms
- velocità di trasferimento di 380 Mbit/sec
- Latenza rotazionale media di 1 ms

Quanti flussi può supportare il server supponendo che i frame sono memorizzati uno per blocco in tracce consecutive?

- [A] 11 flussi
- [B] 9 flussi
- [C] 13 flussi** 
- [D] 15 flussi

**domanda N. 13:**

Sia dato un sistema operativo che utilizza un algoritmo di paginazione simile a quello di Linux 2.4 che assegna rispettivamente 3 frame in memoria per le pagine attive e 3 frame per le pagine inattive. Si determini il numero di page faults per la seguente stringa di riferimenti:

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

- [A] 12 page faults
- [B] 10 page faults** 
- [C] 9 page faults
- [D] 11 page faults

**domanda N. 14:**

Un piccolo sistema distribuito è composto da due nodi su cui sono in esecuzione due processi (denominati P1 e P2) che si sincronizzano per l'accesso ad una regione critica secondo l'algoritmo di Agrawala e Ricart. Il processo P1, a partire dall'istante  $t=0$ , esegue in sequenza per 3 volte una porzione di codice (da eseguirsi in mutua esclusione) che richiede 5 sec ed una porzione di codice (senza vincoli particolari) che richiede 3 sec. Il processo P2 invece, a partire da  $t=0$ , esegue in sequenza per 3 volte una porzione di codice (senza vincoli particolari) che richiede 4 sec ed una porzione di codice (da eseguirsi in mutua esclusione) che richiede 4 sec. Supponendo che il tempo necessario per mandare un messaggio da un nodo all'altro sia di 1 sec, determinare l'istante in cui il processo P1 termina la sua esecuzione.

- [A] 36
- [B] 32
- [C] 30
- [D] 24

**domanda N. 15:**

Un sistema operativo impegna l'algoritmo del banchiere per controllare l'allocazione di 7 unità nastro. I tre processi  $P_1$ ,  $P_2$  e  $P_3$  ne hanno necessità di al più 7, 3 e 5 rispettivamente per

completare la loro esecuzione. Quante unità il sistema operativo può allocare in modo sicuro al processo  $P_1$ , se l'attuale stato di allocazione è il seguente: 2, 1 e 1 unità nastro sono allocate ai processi  $P_1$ ,  $P_2$  e  $P_3$  rispettivamente.

- [A] 3
- [B] 0
- [C] 1
- [D] 2

**domanda N. 16:**

Si consideri un file formato da 100 record allocato su disco in forma indicizzata dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se non è stato fatto ancora nessun accesso al file, quanti accessi in lettura e in scrittura sono necessari per cancellare il 50-mo record?

- [A] 50 accessi in lettura e 1 in scrittura
- [B] 1 accesso in lettura e 50 in scrittura
- [C] 50 accessi in lettura e 50 in scrittura
- [D] 1 accesso in lettura e 1 in scrittura

=====  
== CORSO DI SISTEMI OPERATIVI -9CFU (gr. 1) ==  
===== compito N. 20 del 220610 =====  
=====

**domanda N. 1:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 100   |
| $P_2$ | 0     | 80    |
| $P_3$ | 0     | 90    |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU espresso in ms. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 20$  ms, si determini quanto deve durare al piu' il cambio di contesto se si vuole che il tempo medio di completamento dei tre processi sia al piu' 280 ms.

- [A] 2.57 ms
- [B] 0.78 ms
- [C] 1.22 ms
- [D] 3.35 ms

**domanda N. 2:**

Si determini il numero di processi generati (compreso quello mandato in esecuzione dalla linea di comando) dal seguente codice C, compilato e fatto eseguire su un sistema con il sistema operativo Unix:

```
main(){
int pid, i;
for (i = 0; i <= 3; i ++){
 pid=fork();
 if (pid == 0 && i == 1) exit(0);
}
exit(0);
}
```

- [A] 6
- [B] 8
- [C] 9

**domanda N. 3:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 8     |
| $P_2$ | 0     | 2     |
| $P_3$ | 3     | 4     |
| $P_4$ | 5     | 6     |
| $P_5$ | 9     | 2     |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 2$ , si determini il tempo medio di esecuzione.

[A] 11,4

[B] 11,8

[C] 11,2

[D] 11,6

**domanda N. 4:**

Siano dati tre processi che condividono una variabile  $x$  e i cui pseudo codici sono di seguito riportati:

| $P_1$        | $P_2$             | $P_3$             |
|--------------|-------------------|-------------------|
| $wait(R)$    | $wait(S)$         | $wait(T)$         |
| $x = x - 5;$ | $x = x + 2;$      | $x = x + 3;$      |
| $signal(T);$ | $if(x >= 5) then$ | $if(x == 1) then$ |
| $wait(R);$   | $signal(T);$      | $signal(R);$      |
| $signal(S);$ | $else$            | $endif$           |
|              | $signal(R);$      | $wait(T);$        |
|              | $endif$           |                   |
|              | $wait(S);$        |                   |
|              | $signal(T);$      |                   |

Supponendo che i tre semafori binari  $R$ ,  $S$  e  $T$  siano inizializzati rispettivamente a  $R = 0$ ,  $S = 1$  e  $T = 0$ , si determini quale valore iniziale della variabile  $x$ , tra quelli proposti nelle risposte, non comporti una situazione di stallo per i processi.

[A]  $x = 4$ [B]  $x = 1$

[]  $x = 2$

[]  $x = 3$

**domanda N. 5:**

Sia data una memoria di 32MByte gestita mediante paginazione con frame di 512 KByte e si supponga che il sistema operativo richieda 6 MByte. Si calcoli la dimensione della memoria effettivamente ancora utilizzabile dopo l'allocazione di 5 processi rispettivamente di 2530, 1840, 785, 3421, 2013 KByte.

[] 19456 KByte

[] 17408 KByte

[] 21540 KByte

[] 15360 KByte 

**domanda N. 6:**

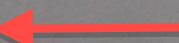
Si consideri un sistema con paginazione su due livelli. Per un processo che richiede 16 pagine, ognuna di 512 byte, di seguito si riportano la tabella delle pagine di primo livello ( $P'$ ) e le 4 tabelle di secondo livello ( $P''_0$ ,  $P''_1$ ,  $P''_2$  e  $P''_3$ ), le quali hanno rispettivamente indirizzo fisico iniziale 0, 512, 1024 e 1536:

| $P'$ |      | $P''_0$ |       | $P''_1$ |       | $P''_2$ |       | $P''_3$ |       |
|------|------|---------|-------|---------|-------|---------|-------|---------|-------|
| 0    | 0    | 0       | 8192  | 0       | 10752 | 0       | 5632  | 0       | 9216  |
| 1    | 512  | 1       | 11776 | 1       | 7680  | 1       | 10240 | 1       | 3584  |
| 2    | 1024 | 2       | 5120  | 2       | 3072  | 2       | 13824 | 2       | 12800 |
| 3    | 1536 | 3       | 14336 | 3       | 6144  | 3       | 11264 | 3       | 14848 |

Si determini quale indirizzo fisico, tra quelli proposti, non appartiene allo spazio di indirizzamento del processo.

[] 14338

[] 3074

[] 7666 

[] 11266

**domanda N. 7:**

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo First In First Out (FIFO), che assegna ad ogni processo 3 frame di 512 Byte e che richiede 2 Byte per

la memorizzazione di un intero. Ponendo attenzione solo agli array A, B e C, e ricordando che il linguaggio C memorizza gli array 2-dimensionali per righe, si determini il numero di page faults del seguente frammento di codice C. ...

```
int A[2][512], B[2][256], C[512];
...
N=256;
for (i=0; i<2; i++){
 for (j=0; j<N; j++){
 C[i*N+j] = A[i][2*j] + B[i][j];
 } }
```

- [A] 12
- [B] 8
- [C] 10
- [D] 11

**domanda N. 8:**

Sia dato un disco con 200 tracce e velocità di seek di 1 traccia per ms. All'istante  $t=0$  il sistema operativo sta servendo una richiesta sulla traccia 70 e in coda ci sono richieste per le tracce (55; 62; 65; 95; 97; 105). Successivamente arrivano altre richieste all'istante  $t=7$  per la traccia 71 e all'istante  $t=50$  per la traccia 63. Si calcoli il tempo di ricerca complessivo in ms per servire tutte le richieste secondo la politica Shortest Seek Time First (SSTF), trascurando la latenza rotazionale e il tempo di trasferimento.

- [A] 121
- [B] 107
- [C] 97
- [D] 65

**domanda N. 9:**

Si consideri un disco che opera a 12000 rpm (giri al minuto), dove ogni traccia è composta da 160 settori di 512 bytes ognuno. Si determini la velocità di trasferimento necessaria per poter leggere o scrivere una traccia intera durante una rotazione del disco.

- [A] circa 9600 KB/sec
- [B] circa 8000 KB/sec
- [C] circa 19200 KB/sec

**domanda N. 10:**

Qual'e' la massima dimensione della partizione del disco per un sistema operativo MS-DOS che utilizza il file system FAT-16 con blocchi del disco da 16KB?

[A] 2 GB

[B] 4 GB

[C] 512 MB

[D] 1 GB



**domanda N. 11:**

In un sistema operativo multimediale sono in esecuzione tre processi periodici A, B e C con le seguenti caratteristiche:

|   | Periodo | Burst |
|---|---------|-------|
| A | 30      | 10    |
| B | 20      | 10    |
| C | 40      | 5     |

dove i tempi sono espressi in ms. Si determini quale processo è in esecuzione all'istante  $t = 125$  ms schedulando i processi con l'algoritmo Earliest Deadline First (EDF).

[A] il processo A

[B] il processo C

[C] la CPU è inattiva

[D] il processo B

**domanda N. 12:**

Sia dato un sistema multimediale con algoritmo di scheduling RMS, dove già sono in esecuzione tre processi periodici A, B e C con caratteristiche

- A periodo = 50ms e burst di CPU = 10ms
- B periodo = 40ms e burst di CPU = 10ms
- C periodo = 25ms e burst di CPU = 5ms

SI determini quale può essere al più il burst di CPU di un quarto processo D il cui periodo è di 40 ms

- [A] 7.12 ms
- [B] 3.72 ms
- [C] 4.28 ms
- [D] 5.2 ms



**domanda N. 13:**

Si consideri un sistema operativo con un algoritmo di paginazione simile a quello di Windows XP con dimensione massima del working set di 3 frame, ed un tempo di permanenza tra le pagine in attesa di al piú 3 unitá di tempo. Si determini il numero di page faults e di soft faults per la seguente stringa di riferimento:

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

- [A] 9 page faults e 7 soft faults
- [B] 10 page faults e 7 soft faults
- [C] 9 page faults e 8 soft faults
- [D] 10 page faults e 8 soft faults

**domanda N. 14:**

Un piccolo sistema distribuito e' composto da due nodi su cui sono in esecuzione due processi (denominati P1 e P2) che si sincronizzano per l'accesso ad una regione critica secondo l'algoritmo di Agrawala e Ricart. Il processo P1, a partire dall'istante  $t=0$ , esegue in sequenza per 3 volte una porzione di codice (da eseguirsi in mutua esclusione) che richiede 5 sec ed una porzione di codice (senza vincoli particolari) che richiede 3 sec. Il processo P2 invece, a partire da  $t=0$ , esegue in sequenza per 3 volte una porzione di codice (senza vincoli particolari) che richiede 4 sec ed una porzione di codice (da eseguirsi in mutua esclusione) che richiede 4 sec. Supponendo che il tempo necessario per mandare un messaggio da un nodo all'altro sia di 1 sec, determinare l'istante in cui il processo P2 termina la sua esecuzione.

- [A] 32
- [B] 36
- [C] 34
- [D] 30

**domanda N. 15:**

Un sistema operativo impegna l'algoritmo del banchiere per controllare l'allocazione di 7 unitá

nastro. I tre processi  $P_1$ ,  $P_2$  e  $P_3$  ne hanno necessità di al più 7, 3 e 5 rispettivamente per completare la loro esecuzione. Quante unità il sistema operativo può allocare in modo sicuro al processo  $P_1$ , se l'attuale stato di allocazione è il seguente: 1, 2 e 1 unità nastro sono allocate ai processi  $P_1$ ,  $P_2$  e  $P_3$  rispettivamente.

- [A] 1
- [B] 0
- [C] 2
- [D] 3

**domanda N. 16:**

Si consideri un file formato da 100 record allocato su disco in forma concatenata dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se l'ultimo record letto e' il record 35, quanti accessi in lettura e in scrittura sono necessari per cancellare il 60-mo record ?

- [A] 1 accesso in lettura e 1 in scrittura
- [B] 1 accesso in lettura e 25 in scrittura
- [C] 25 accessi in lettura e 1 in scrittura.
- [D] 25 accessi in lettura e 25 in scrittura.

=====  
==== CORSO DI SISTEMI OPERTIVI mod.A (gr. 1) ====  
===== compito N. 38 del 31-03-2010=====

**domanda N. 1:**

Si determini la veridicità delle seguenti affermazioni:

- a) Per motivi di sicurezza, le seguenti istruzioni dovrebbero essere sempre eseguite in modalità sistema: lettura del clock di sistema, disabilitazione delle interruzioni, switch dalla modalità utente a quella di sistema, accesso ai dispositivi di I/O.
- b) La principale difficoltà nello sviluppo di un sistema operativo real time è garantire che i processi terminino la loro esecuzione in un fissato intervallo di tempo. In tal senso un ruolo chiave è svolto dallo scheduler.

- [A] a) falsa ; b) vera  
[B] a) vera ; b) falsa  
[C] entrambe vere  
[D] entrambe false

**domanda N. 2:**

Si determini la veridicità delle seguenti affermazioni:

- a) L'insieme delle informazioni che caratterizzano un processo in esecuzione e che sono memorizzate nel Process Control Block è chiamato *contesto*. Per tale motivo la procedura che sospende un processo in esecuzione per mandarne un altro è detta *cambio di contesto* (context switch).
- b) Il tempo medio di esecuzione di un insieme di processi di un sistema operativo time sharing diminuisce sempre quando aumenta la frequenza dei context switch, in quanto i processi in esecuzione si avvicendano più rapidamente nell'uso della CPU.

- [A] entrambe vere  
 [B] a) vera ; b) falsa  
[C] entrambe false  
[D] a) falsa ; b) vera

**domanda N. 3:**

Si determini il numero di processi generati (compreso quello mandato in esecuzione dalla linea di comando) dal seguente codice C, compilato e fatto eseguire su un sistema con il sistema operativo Unix:

```
main(){
 int pid, i;
 for (i = 0; i <= 3; i++){
 pid=fork();
 if (pid == 0 && i > 1) exit(0);
 }
 exit(0);
}
```

[A] 10

~~B~~ 12

[C] 9

[D] 8

**domanda N. 4:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ | $P$ |
|-------|-------|-------|-----|
| $P_1$ | 1     | 7     | 2   |
| $P_2$ | 2     | 5     | 4   |
| $P_3$ | 6     | 4     | 5   |
| $P_4$ | 7     | 7     | 1   |
| $P_5$ | 10    | 2     | 3   |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti,  $T_C$  è il tempo di utilizzo della CPU e  $P$  è la priorità (priorità massima = 5). Utilizzando l'algoritmo di scheduling a priorità con prelazione, si determini il tempo medio di esecuzione.

[A] 11

[B] 10.4

[C] 11.2

~~D~~ 10.6

**domanda N. 5:**

Sia dato il seguente insieme di processi:

|       | $T_A$ | $T_C$ |
|-------|-------|-------|
| $P_1$ | 0     | 5     |
| $P_2$ | 3     | 6     |
| $P_3$ | 5     | 7     |
| $P_4$ | 8     | 6     |
| $P_5$ | 16    | 3     |

dove  $T_A$  è il tempo di arrivo nella coda dei processi pronti e  $T_C$  è il tempo di utilizzo della CPU. Utilizzando l'algoritmo di scheduling Round Robin con quanto di tempo  $q = 2$ , si determini il tempo medio di esecuzione.

- [A] 13.6
- [B] 14
- [C] 14.2
- [D] 13.8

**domanda N. 6:**

Siano dati tre processi che condividono una variabile  $x$  e i cui pseudo codici sono di seguito riportati:

| $P_1$        | $P_2$             | $P_3$        |
|--------------|-------------------|--------------|
| $wait(R)$    | $wait(S)$         | $wait(T)$    |
| $x = -x;$    | $x = x-1;$        | $x = x+1;$   |
| $signal(S);$ | $if (x > 0) then$ | $signal(R);$ |
| $wait(R);$   | $x = x-1;$        | $wait(T);$   |
| $x = -x;$    | $signal(T);$      | $print x;$   |
| $signal(T);$ | $else$            |              |
|              | $signal(R);$      |              |
|              | $endif$           |              |

Si determini l'output del processo  $P_3$  supponendo che inizialmente  $x = 1$  e i tre semafori binari siano inizializzati rispettivamente  $S = 0$ ,  $R = 0$ ,  $T = 1$ .

- [A] 5
- [B] 6
- [C] 3
- [D] 4

**domanda N. 7:**

Si determini la veridicità delle seguenti affermazioni:

- a) Nei moderni sistemi operativi le tecniche di gestione della memoria centrale sono poco importanti a causa del basso costo economico delle stesse.
- b) Andando dai livelli di memoria più bassi (dispositivi secondari e terziari) a quelli più alti (registri e cache), il costo economico per bit aumenta.
- [A] a) vera ; b) falsa
- [B] entrambe false
- a) falsa ; b) vera
- [D] entrambe vere

**domanda N. 8:**

Sia data una memoria di 32MByte gestita mediante paginazione con frame di 512 KByte e si supponga che il sistema operativo richieda 7 MByte. Si calcoli la dimensione della memoria effettivamente ancora utilizzabile dopo l'allocazione di 5 processi rispettivamente di 2343, 1724, 489, 1270, 2110 KByte.

- [A] 21504 KByte
- 16384 KByte
- [C] 15360 KByte
- [D] 19456 KByte

**domanda N. 9:**

Si consideri un sistema con paginazione su due livelli. Per un processo che richiede 16 pagine, ognuna di 512 byte, di seguito si riportano la tabella delle pagine di primo livello ( $P'$ ) e le 4 tabelle di secondo livello ( $P''_0, P''_1, P''_2$  e  $P''_3$ ), le quali hanno rispettivamente indirizzo fisico iniziale 0, 512, 1024 e 1536:

| $P'$ |      | $P''_0$ |       | $P''_1$ |       | $P''_2$ |       | $P''_3$ |       |
|------|------|---------|-------|---------|-------|---------|-------|---------|-------|
| 0    | 0    | 0       | 8192  | 0       | 10752 | 0       | 5632  | 0       | 9216  |
| 1    | 512  | 1       | 11776 | 1       | 7680  | 1       | 10240 | 1       | 3584  |
| 2    | 1024 | 2       | 5120  | 2       | 3072  | 2       | 13824 | 2       | 12800 |
| 3    | 1536 | 3       | 14336 | 3       | 6144  | 3       | 11264 | 3       | 14848 |

Si determini quale indirizzo fisico, tra quelli proposti, non appartiene allo spazio di indirizzamento del processo.

- [A] 13826
- 5118

[C] 10241

[D] 5634

**domanda N. 10:**

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo First In First Out (FIFO), che assegna ad ogni processo 3 frame di 512 Byte e che richiede 2 Byte per la memorizzazione di un intero. Ponendo attenzione solo agli array A, B e C, e ricordando che il linguaggio C memorizza gli array 2-dimensional per righe, si determini il numero di page faults del seguente frammento di codice C. ...

```
int A[2][512], B[256], C[512];
...
N=256;
for (i=0; i<2; i++){
 for (j=0; j<N; j++){
 C[i*N+j] = A[i][2*j] + B[j];
 }
}
```

[A] 11

~~☒~~ 8

[C] 12

[D] 10

**domanda N. 11:**

Si consideri un sistema operativo con paginazione su richiesta basato sull'algoritmo Least Recently Used (LRU), che assegna ad ogni processo 5 frame. Se determini il numero di page faults per la sequenza di riferimenti alle pagine:

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

[A] 10

[B] 7

~~☒~~ 9

[D] 8

**domanda N. 12:**

Sia dato un disco con 200 tracce e velocità di seek di 1 traccia per ms. All'istante t=0 il

sistema operativo sta servendo una richiesta sulla traccia 70 e in coda ci sono richieste per le tracce (55; 62; 65; 95; 97; 105). Successivamente arrivano altre richieste all'istante  $t=7$  per la traccia 71 e all'istante  $t=50$  per la traccia 63. Si calcoli il tempo di ricerca complessivo in ms per servire tutte le richieste secondo la politica Shortest Seek Time First (SSTF), trascurando la latenza rotazionale e il tempo di trasferimento.

[A] 97

[B] 65

[C] 121

[D] 107

**domanda N. 13:**

Si determini la veridicità delle seguenti affermazioni:

- a) Un problema frequente nell'algoritmo First Come First Served (FCFS) di scheduling del disco è la posticipazione indefinita di alcune richieste di servizio.
- b) La latenza rotazionale è identica per ogni accesso al disco.

[A] entrambe false

[B] a) vera ; b) falsa

[X] a) falsa ; b) vera

[D] entrambe vere

**domanda N. 14:**

Si calcoli la probabilità di perdere i dati con un sistema RAID level 1 con 6 dischi, se il Mean Time to Failure (MTF) è di 1300 giorni e il Mean Time to Repair (MTR) è di 16.9 giorni.

[X] 0.00003

[B] 0.00009

[C] 0.00001

[D] 0.00011

**domanda N. 15:**

Si determini la veridicità delle seguenti affermazioni:

- a) Per motivi di efficienza, la lista dei blocchi liberi è generalmente ordinata secondo l'indice del blocco.
- b) I principali compiti di un file system sono: gestire le strutture logiche del file system (directory e file), organizzare i file in blocchi e gestirne l'allocazione, verificare l'integrità dei dati, gestire lo spazio libero e verificare che i dati nei file non contengano errori.

- [A] entrambe false
- [B] a) vera ; b) falsa
- [C] entrambe vere
- [D] a) falsa ; b) vera

**domanda N. 16:**

Sia dato un file system di 2 MB che gestisca i file mediante allocazione tabellare (ad es. una File Allocation Table). Si determini la dimensione della tabella nel caso in cui i blocchi del disco siano di 512 byte.

- [A] 48 KB
- [B] 32 KB
- [C] 6 KB
- [D] 12 KB

**domanda N. 17:**

Si determini la veridicità delle seguenti affermazioni:

- a) Si consideri un file formato da 100 record allocato su disco in forma contigua dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se l'ultimo record letto è il record 25 allora per cancellare il 70-mo record sono necessari 30 accessi in lettura e 30 in scrittura.
- b) Si consideri un file formato da 100 record allocato su disco in forma concatenata dove i record del file sono memorizzati uno per blocco e il file descriptor è già in memoria centrale. Se l'ultimo record letto è il record 35, allora per cancellare il 75-mo record sono necessari 40 accessi in lettura e 1 in scrittura.

- [A] a) falsa ; b) vera
- [B] entrambe vere
- [C] a) vera ; b) falsa
- [D] entrambe false

**45.** Si supponga che due processi  $P_0$  e  $P_1$  abbiano accesso a due semafori  $S$  e  $Q$  impostati al valore 1. Siano  $P_0$  e  $P_1$  strutturati nel modo seguente:

$P_0 \rightarrow \text{wait}(S); \text{wait}(Q); \dots; \text{signal}(S); \text{signal}(Q);$   
 $P_1 \rightarrow \text{wait}(Q); \text{wait}(S); \dots; \text{signal}(Q); \text{signal}(S);$

- A se  $P_0$  esegue  $\text{wait}(S)$  e  $P_1$  esegue  $\text{wait}(Q)$ ,  $P_0$  e  $P_1$  sono in stallo.
- B se  $P_1$  esegue  $\text{wait}(Q)$  e  $\text{wait}(S)$ ,  $P_0$  è in stallo.
- C se  $P_0$  esegue  $\text{wait}(S)$  e  $\text{wait}(Q)$ ,  $P_1$  è in stallo.
- D se  $P_0$  esegue  $\text{wait}(S)$  e  $\text{wait}(Q)$ ,  $P_1$  esegue  $\text{wait}(Q)$  e  $\text{wait}(S)$ .
- E se  $P_1$  esegue  $\text{wait}(Q)$  e  $\text{wait}(S)$ ,  $P_0$  e  $P_1$  sono in stallo.

**46.** In un algoritmo con Grafo di Assegnazione delle Risorse quale tra le seguenti affermazioni risulta essere vera:

- A Un arco di reclamo  $P_i \rightarrow R_j$  indica che il processo  $P_i$  sicuramente richiederà la risorsa  $R_j$ .
- B Ogni tipo di risorsa NON può contenere un numero di risorse maggiori del numero dei processi richiedenti.
- C Le risorse devono essere reclamate a priori nel sistema.
- D Se  $P_i$  rilascia  $R_j$  l'arco di assegnazione  $R_j \rightarrow P_i$  può diventare un arco di reclamo.
- E Le risorse possono essere reclamate a priori nel sistema.

**47.** Un codice è definito essere RILOCABILE se:

- A È generabile solo da linguaggi ad alto livello
- B È stata ritardata l'associazione finale degli indirizzi alla fase di caricamento.
- C Quando richiede maggiore memoria di quella prevista.
- D Fa riferimento esclusivamente ad indirizzi fisici.
- E È possibile riproporlo per una nuova esecuzione.

**8.** Una soluzione del problema della *Sezione Critica* deve soddisfare i seguenti tre criteri:

A Indipendenza; Progresso; Attesa limitata  
B Mutua Esclusione; Progresso; Indipendenza  
C Mutua Esclusione; Progresso; Attesa limitata

D Variabilità; Mutua esclusione; Attesa limitata  
E Mutua Esclusione; Indipendenza; Attesa limitata

• *Un processo leggero o Thread*

- Può condividere dati soltanto se essi sono posizionati su Hard Disk.
- Può condividere dati solo se essi non superano il limite massimo di 5k bytes.
- Può condividere dati soltanto attraverso dei files.
- Può condividere dati soltanto attraverso dei files.
- Può condividere direttamente uno spazio logico di indirizzi.

**1.** Si supponga di dover gestire un sistema di memoria segmentata.

Se ad un certo istante si ha la seguente configurazione:

- \* *Valori del segmento n.3 nella Tabella dei segmenti:* BASE = 5320; LIMITE = 324
  - \* *Valore di un indirizzo logico:* NUMERO SEGMENTO = 5; SCOSTAMENTO = 234
- possiamo allora dire che:

- [A] Si punta all'elemento il cui indirizzo fisico è 5878
- [B] Si punta all'elemento il cui indirizzo fisico è 5554
- [C] Si punta all'elemento il cui indirizzo fisico è 5410
- [D] Si punta all'elemento il cui indirizzo logico è 5554
- [E] si invia un segnale di eccezione al sistema operativo

**2.** Nella tecnica di base della paginazione della memoria, in che modo viene tradotto indirizzo logico in un indirizzo fisico? si assume:

```
p="numero_di_pagina",
d="scostamento_di_pagina",
T[i] = "indirizzo_di_base_della_pagina_i".
```

- [A]  $d+T[p]$
- [B]  $p \cdot T[d]$
- [C]  $d \cdot T[p]$
- [D]  $p+T[d]$
- [E]  $(p-1)+T[d]$

**3.** Si presuma che un sistema di elaborazione S disponga di due istruzioni atomiche ed  $I_1$ .

- [A] Nell'insieme di istruzioni di S solo  $I_1$  può prelazionare  $I_2$  (e viceversa).
- [B]  $I_1$  ed  $I_2$  devono garantire la mutua esclusione di esecuzione del codice del programma nel quale sono definite.
- [C]  $I_1$  ed  $I_2$  vengono eseguite sequenzialmente soltanto se non contengono una propria sezione critica.
- [D]  $I_1$  ed  $I_2$  possono essere eseguite in modo sequenziale arbitrario.
- [E]  $I_1$  ed  $I_2$  vengono eseguite sequenzialmente soltanto se contengono una propria sezione critica.

**4.** In un Sistema di Elaborazione, l'utilizzo delle risorse mediante un algoritmo che evita le situazioni di stallo, può essere inferiore rispetto a quello che si avrebbe in assenza di un algoritmo per evitare le situazioni di stallo.

- [A] VERO.
- [B] Vero solo se il numero delle risorse è minore dal numero dei processi che ne fanno richiesta.
- [C] Vero solo se il numero delle risorse è maggiore dal numero dei processi che ne fanno richiesta.
- [D] Falso solo se il numero delle risorse è maggiore dal numero dei processi che ne fanno richiesta.
- [E] FALSO.

11. Qual è tra le seguenti una definizione corretta di un semaforo ad attesa passiva?

- A void wait(semaforo S) {S.valore -; if (S.valore<0) {sottrai questo processo a S.L. block();}}
- B void wait(semaforo S) {S.valore -; if (S.valore<0) {block(); aggiungi questo processo a S.L.}}
- C void wait(semaforo S) {S.valore -; if (S.valore>0) {aggiungi questo processo a S.L. block();}}
- D void wait(semaforo S) {S.valore -; if (S.valore>0) {block(); aggiungi questo processo a S.L.}}
- E void wait(semaforo S) {S.valore -; if (S.valore<0) {aggiungi questo processo a S.L. block();}}

12. Sia K un File-system assegnato. Qual è dei seguenti è un Percorso Relativo?

- A /usr/local/bin
- B /windows/win32
- C /usr/local
- D /root/spell/..../first
- E /var/spool/mail

13. Tra le seguenti strutture di File-System, quale realizza correttamente il modello di Sintificazione?

- |                                                                                                                                                              |                                                                                                                                                              |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <input type="checkbox"/> A Programmi d'applicazione<br>File-System logico<br>File-System di base<br>Modulo di organizzazione dei file<br>Controllo dell' I/O | <input type="checkbox"/> B Programmi d'applicazione<br>Modulo di organizzazione dei file<br>File-System logico<br>File-System di base<br>Controllo dell' I/O |
| <input type="checkbox"/> C Programmi d'applicazione<br>Modulo di organizzazione dei file<br>File-System logico<br>Controllo dell' I/O<br>File-System di base | <input type="checkbox"/> D Programmi d'applicazione<br>File-System logico<br>Modulo di organizzazione dei file<br>File-System di base<br>Controllo dell' I/O |
| <input type="checkbox"/> E Programmi d'applicazione<br>File-System logico<br>Modulo di organizzazione dei file<br>Controllo dell' I/O<br>File-System di base |                                                                                                                                                              |

14. Qual è strategia viene principalmente adottata per superare il limite dell'Attesa di un processo?

- A Mediante una wait() inviata ai processi relativi allo stesso semaforo.
- B Variando la priorità di esecuzione dei processi correntemente in esecuzione.
- C Bloccando il processo fino alla terminazione di tutti i processi attualmente in esecuzione.
- D Posendo il processo in una coda di attesa associata al semaforo.
- E Inserendo il processo in uno stack di attesa associata al semaforo.

15. In quale delle seguenti circostanze sono prelazionabili le Istruzioni Atomiche?

- A Quando l'interruzione è causata da exception.
- B Se viene impiegato almeno un semaforo.
- C In nessuna circostanza.
- D Se il processo che effettua la prelazione è l'ultimo elemento dello stack di sistema.
- E Quando l'interruzione è causata da interrupt.

**25.** Nei messaggi scambiati tra processi comunicanti con code a capacità ZERO, si ha:

- A Il ricevente e il trasmettente devono fermarsi se le rispettive CPU sono al 100% del carico di lavoro.
- B Il ricevente deve fermarsi se la CPU è al 100% del carico di lavoro.
- C Il trasmettente deve fermarsi se la CPU è al 100% del carico di lavoro.
- D Il trasmettente deve fermarsi finché il ricevente prende in consegna il messaggio.
- E Il ricevente deve fermarsi se lunghezza messaggio ricevuto è diversa da lunghezza messaggio inviato.

**26.** Sia assegnato un disco *A* con blocchi da 512 byte e un disco *B* da 1024 per blocco. Si supponga che si debba memorizzare un file *F* da 1949 byte: quale dei due dischi *A* e *B* genererà maggiore frammentazione interna?

- A dipende dalla frammentazione esterna di *F*
- B Il disco *A*
- C È la stessa per entrambi i dischi *A* e *B*
- D Il disco *B*
- E dipende dallo specifico algoritmo di paginazione della memoria utilizzato

**27.** In quale circostanza si può permettere che *wait* e *signal* possano essere eseguite in modo non atomico?

- A Se la CPU è particolarmente veloce.
- B Se si presume che l'attesa sia breve.
- C Se la CPU è particolarmente lenta.
- D In nessuna circostanza.
- E In situazioni di carico elevato.

**28.** In quale dei seguenti casi è possibile applicare lo scheduling della CPU?

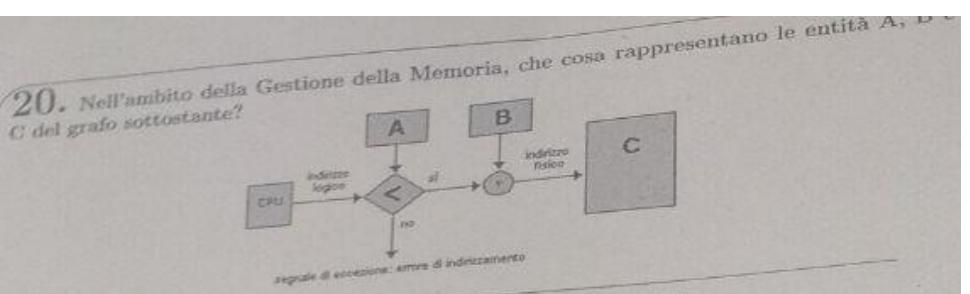
- A quando un processo dallo stato di attesa passa allo stato di esecuzione
- B quando un processo dallo stato di esecuzione o di attesa passa allo stato di pronto
- C quando un processo passa allo stato di terminazione
- D quando un processo dallo stato di esecuzione passa allo stato di attesa
- E quando un processo dallo stato di esecuzione o di pronto passa allo stato di attesa

**29.** Si supponga di utilizzare l'algoritmo SSTF per lo Scheduling del disco e che la testina sia inizialmente al cilindro 100. Si supponga di dover accedere ad una sequenza di blocchi dislocati rispettivamente nei cilindri 45, 22, 110: quale sarà la (misurata in cilindri) percorsa dalla testina?

- A 67
- B 76
- C 89
- D 98
- E 110

**30.** Nei dischi in *Configurazione Raid*, quale tra le caratteristiche di seguito elencate risulta errata?

- +  A Ridondanza di batterie di dischi.
- B Miglioramento dell'affidabilità della memoria secondaria.
- C Aumento di capacità di elaborazione della CPU.
- D Copiatura speculare (in configurazione mirroring).
- E Aumento di capacità di trasferimento dei dati.



- A C="Memoria"; A="Registro di Rilocazione"; B="Registro di Limite".
- B C="Memoria"; B="Registro di Rilocazione"; A="Registro di Limite".
- C A="Registro di Rilocazione"; C="Registro di Limite"; B="Memoria".
- D B="Registro di Rilocazione"; A="Memoria"; C="Registro di Limite".
- E C="Registro di Rilocazione"; B="Memoria"; A="Registro di Limite".

- 21.** Se il nome di un assegnato file F è "prova.exe", allora possiamo affermare che:
- A E' probabile che sia un file in formato Excel.
  - B E' certamente un file con i permessi di esecuzione.
  - C E' probabile che sia un file compilabile.
  - D E' sicuramente un file eseguibile.
  - E E' sicuramente un file eseguibile.

- 22.** Quali informazioni, tra le altre, contiene un PCB di un processo?

- A Contatore di programma, metodi di accesso alla RAM e alla memoria virtuale.
- B Stato del processo, contatore di programma, Registri di CPU.
- C Contatore di programma, metodi per segmentare la memoria, metodi per il DMA.
- D Stato del processo, Numero Dispositivi fisici connessi al sistema, Registri di CPU.
- E Registri di CPU, prime linee del codice sorgente del processo, Stato del processo.

- 23.** Si supponga di utilizzare l'algoritmo FCFS per lo *Scheduling del disco*, e che la testina sia inizialmente al cilindro N.100. Se occorre accedere ad una sequenza ordinata di blocchi dislocati rispettivamente nei cilindri 45, 22, 110 allora quale sarà la distanza totale (misurata in cilindri) percorsa dalla testina?

- A 155
- B 188
- C 177
- D 144
- E 166

- 24.** Qual è la caratteristica di maggior rilievo del *Metodo delle Pagine Invertite*?

- A Riduzione della quantità di memoria necessaria per memorizzare ogni tabella delle pagine.
- B Semplifica il tempo necessario per scambiare il contenuto di due pagine.
- C Notevole semplificazione per la gestione dei Threads.
- D Riduzione del tempo di ricerca delle singole pagine.
- E Migliora il tempo di ripristino da situazioni di stallo.

5.  B  
 C  
 D  
 E

6. che

1. Nell'ambito di uno scheduling CPU a code multiple, quale tra i seguenti schemi ha maggior senso?

- A FCFS per processi in primo piano e R.R. per processi in background.
- B R.R. per processi in primo piano e SJF per processi in background.
- C SJF per processi in primo piano e SJF con prelazione per processi in background.
- D R.R. per processi in primo piano e FCFS per processi in background.
- E SJF con prelazione per processi in primo piano e SJF per processi in background.

2. Si supponga di aver sviluppato un algoritmo di sincronizzazione per soli due Processi Pi e Pj; si assuma che la variabile turno (che vale i oppure j) individua il processo (Pi oppure Pj) che ha permesso di entrare nella propria sezione critica. A quali requisiti soddisfa l'algoritmo indicato?

```
do {
 while (turno != i);
 ...sezione critica...
 turno = j;
 ...sezione non critica...
} while (1);
```

A Progresso  
Mutua esclusione  
Indipendenza

B Attesa Limitata  
Progresso

C Mutua esclusione  
Indipendenza  
Conflitto

D Attesa Limitata  
Progresso  
Mutua esclusione

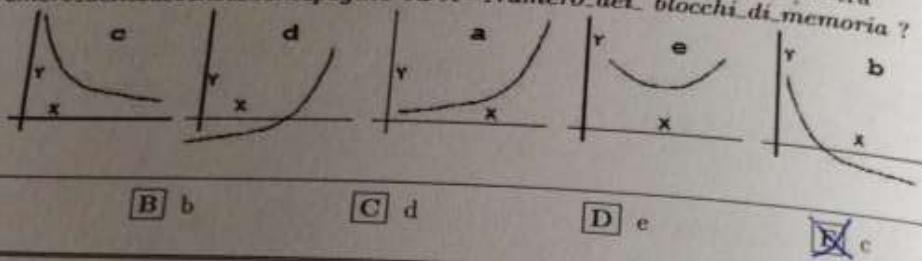
E Mutua esclusione  
Attesa limitata

3. Qual è la caratteristica fondamentale del codice ECC ?

A Dà la possibilità di specificare la grandezza dei files (espresso in bytes).

- D Si mappa la Memoria Centrale in modo da compattare lo spazio libero per predisporlo per l'I/O.  
 E Si tratta del trasferimento dati full-duplex verso i dispositivi di I/O.

9. Quale dei seguenti grafici descrive la relazione generale che intercorre tra  $Y = \text{Numero delle assenze delle pagine}$  ed  $X = \text{Numero dei blocchi di memoria}$ ?



- A a       B b       C d       D e       E c

10. In un File-System Unix-Like, cosa si intende per **LINK**?

- A Un collegamento ad una specifica pagina web.  
 B Un puntatore ad un file eseguibile.  
 C Un puntatore ad un file di sistema.  
 D Un puntatore ad un altro file o directory.  
 E Un collegamento tra l'i-node ed il boot-block.

2

14. In quale modo

- A Con la duplicazione  
 C Mediante directori  
 E Aumentando la fi-

15. Che relazione

- A E3 = "Indirizzi Logici"  
 B E2 = E1 - E3

16. Nella reali-

- A L'impiego di liste  
 C La tecnica dell'  
 E La tecnica delle

P<sub>1</sub> 0 8  
P<sub>2</sub> 1 4  
P<sub>3</sub> 2 9  
P<sub>4</sub> 3 5

P<sub>1</sub>

**11.** Con l'Algoritmo del Fornaio si serve prima:

- A Il cliente con il numero progressivo più basso che richiede la minor quantità di risorsa
- B Il cliente che richiede la minor quantità di risorsa
- C Il cliente con il minor valor medio ottenuto sommando il proprio numero progressivo con il proprio nome
- D Il cliente con il numero progressivo più basso e, a parità, si serve quello con nome minore
- E Il cliente con il nome minore e, a parità, si serve quello con numero progressivo più basso

**12.** Si supponga di voler adottare uno schema di Gestione gerarchica della memoria mediante la Paginazione a due livelli. Si supponga che lo spazio degli indirizzi logici sia a 28 bit e che la grandezza di una pagina sia di un byte. Qual è il numero massimo degli elementi della Tabella esterna?

- A  $2^{\text{exp}12}$
- B 1k
- C  $2^{\text{exp}20}$
- D  $2^{\text{exp}16}$
- E  $2^{\text{exp}18}$

**13.** Quando un processo è in esecuzione nella propria sezione critica:

- A Non si deve consentire a nessun altro processo di effettuare la lettura di un file
- B Può essere consentita l'esecuzione di altri processi nella propria sezione critica
- C Non si deve consentire a nessun altro processo di poter effettuare stampe
- D Può essere consentita l'esecuzione di altri processi nella propria sezione critica, solo se non richiedono ulteriore memoria di esecuzione.
- E Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica

**14.** In quale modo è possibile realizzare efficacemente un File-System Condiviso?

- A Con la duplicazione delle risorse condivise.
- C Mediante directory a singolo livello.
- E Aumentando la frequenza del processore.
- B Mediante dei link.
- D Riducendo il grado di multi-programmazione.

A 2exp12

B 1K

C 2exp20

D 2exp16

E 2exp18

**13. Quando un processo è in esecuzione nella propria sezione critica:**

- A Non si deve consentire a nessun altro processo di effettuare la lettura di un file
- B Può essere consentita l'esecuzione di altri processi nella propria sezione critica
- C Non si deve consentire a nessun altro processo di poter effettuare stampe
- D Può essere consentita l'esecuzione di altri processi nella propria sezione critica, solo se non richiedono ulteriore memoria di esecuzione.
- E Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica

**14. In quale modo è possibile realizzare efficacemente un File-System Condiviso?**

- A Con la duplicazione delle risorse condivise.
- B Mediante dei link.
- C Mediante directory a singolo livello.
- D Riducendo il grado di multi-programmazione.
- E Aumentando la frequenza del processore.

**15. Che relazione lega le 3 entità E1="Indirizzi Fisici", E2="Registro di Rilocazione", E3="Indirizzi Logici"?**

A E2=E1-E3



E3=E2+E1



E2=E3+E1+1



E2=E3+E1-1



E2=E1+E3

**16. Nella realizzazione delle directory mediante la tecnica dell'Hash qual è, tra le seguenti, una tecnica alternativa per la risoluzione del problema delle collisioni?**

- A L'impiego di liste concatenate.
- B Tecnica dell'assegnazione concatenata.
- C La tecnica dell'indirizzamento indiretto.
- D Tecnica dell'assegnazione contigua.
- E La tecnica delle directory a singolo livello.

**39.** Quale delle seguenti è una caratteristica dell'interazione uomo-sistema?

- A) È l'applicazione principale di comunicare dati numerici ai dati numerici.
- B) Si tratta della più comune forma di gestire le comunicazioni dei processi.
- C) Si intende la capacità di comunicare tra utente e Sistema Operativo.
- D) Si tratta dell'insieme che permette di effettuare gli interfacce del Sistema Operativo.
- E) Si intende la capacità di comunicare tra il Sistema Operativo e le periferiche su cui esiste.

**40.** Nel successivo ad paginazione si come serve il file di validità nella tabella delle pagine?

- A) Definisce la condizione iniziale per il salvataggio dei valori della tabella in uno stack.
- B) Definisce una condizione necessaria per l'inizio del caricamento di una pagina.
- C) Definisce una condizione sufficiente per il caricamento correttivo delle pagine indicate in tabella.
- D) Definisce la validità dei valori che sono memorizzati nelle pagine indicate.
- E) Definisce la validità del numero di elementi presenti nella tabella.

**41.** Qual è la caratteristica principale della Lettura Anticipata (Read-Ahead) di un file da disco?

- A) Vengono caricate in memoria centrale anche le pagine successive a quella espressamente richiesta.
- B) Viene assegnata priorità maggiore alle letture e priorità inferiore alle scritture.
- C) Il file viene preventivamente e completamente letto dal Sistema Operativo per verificare la correttezza del codice in esso contenuto.
- D) Si leggono e si mettono nella cache la pagina richiesta e parecchie pagine successive.
- E) Viene assegnata priorità maggiore alle scritture e priorità inferiore alle letture.

**42.** Cosa è possibile fare quando si esaurisce l'Area di Avvicendamento?

- A) Terminazione forzata di processi e/o arresto del sistema.
- B) Si riduce il quanto di tempo assegnato ad ogni singolo processo.
- C) Si eseguono solo fork associate a processi appartenenti alla coda con priorità massima.
- D) Si cambia criterio di gestione della memoria centrale.
- E) Si effettua una Garbage Collection dei files.

**43.** In un file-System distribuito, quando si preferisce il metodo di scrittura differita per l'aggiornamento della cache?

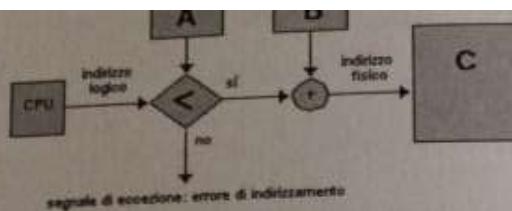
- A) File aperti per lunghi periodi e modificati spesso.
- B) File aperti per lunghi periodi e modificati raramente.
- C) File aperti per brevi periodi ma non modificabili.
- D) File aperti per lunghi periodi ma non modificabili.
- E) File aperti per brevi periodi e modificati raramente.

**44.** Quale delle seguenti strategie di gestione della memoria non è soggetta all'*Anomalia di Belady*?

- |                                                  |                                                         |
|--------------------------------------------------|---------------------------------------------------------|
| <input type="checkbox"/> A Solo sostituzione LRU | <input type="checkbox"/> B Sostituzione Ottimale ed LRU |
| <input type="checkbox"/> C SJF ed LRU            | <input type="checkbox"/> D Solo Sostituzione Ottimale   |
| <input type="checkbox"/> E LRU e FIFO            |                                                         |

**45.** Per uno *Stato Sicuro* si definisce la *Sequenza Sicura* dei processi  $\langle P_1, P_2, \dots, P_n \rangle$  se:

- A Per ogni  $P_i$  le richieste che  $P_i$  può ancora fare si possono soddisfare impiegando le risorse attualmente disponibili più le risorse possedute da tutti i  $P_j$  con  $j < (i+1)$ .
- B Per ogni  $P_i$  le richieste che  $P_i$  può ancora fare si possono soddisfare impiegando le risorse attualmente disponibili.
- C Per ogni  $P_i$  le richieste che  $P_i$  può ancora fare si possono soddisfare impiegando le risorse attualmente disponibili più le risorse possedute da tutti i  $P_j$  con  $j < i$ .
- D Per ogni  $P_i$  le richieste che  $P_i$  può ancora fare si possono soddisfare impiegando le risorse possedute da tutti i  $P_j$  con  $j < i$ .
- E Per ogni  $P_i$  le richieste che  $P_i$  può ancora fare si possono soddisfare impiegando le risorse attual-



- A C="Registro di Rilocazione"; B="Memoria"; A="Registro di Limite".  
 B B="Registro di Rilocazione"; A="Memoria"; C="Registro di Limite".  
 C="Memoria"; B="Registro di Rilocazione"; A="Registro di Limite".  
 D C="Memoria"; A="Registro di Rilocazione"; B="Registro di Limite".  
 E A="Registro di Rilocazione"; C="Registro di Limite"; B="Memoria".

**49.** Sia K un File-System strutturato con directory ROOT (al livello 1) ed altri 4 livelli. Se per ogni livello si ha che ( file + sottodirectory )  $\leq 3$ , qual è il numero massimo di files contenuti in K?

- A 162       B 27       C 729       D 81       E 243

**50.** Tra le seguenti strategie di instradamento, quali assicurano un arrivo ordinato dei dati trasmessi?

- A Instradamento Virtuale.  
 B Instradamento Dinamico e Virtuale.  
 C Nessuna strategia può garantire al 100% un ordine di arrivo.  
 D Instradamento Dinamico e Fisso.  
 E Instradamento Fisso e Virtuale.

FINE

- B La Frammentazione Esterna è relativa ai file se sono in memoria.
- C La Frammentazione Interna è la somma tra memoria assegnata e memoria non assegnata.
- D La Frammentazione Interna è la differenza tra memoria assegnata e quella richiesta. Nella Frammentazione Esterna lo spazio richiesto non è contiguo.
- E La Frammentazione Interna è relativa ai processi interni alla memoria centrale e la Frammentazione Esterna è relativa ai processi esterni alla memoria centrale.

#### 7. Cosa caratterizza l'*Anomalia di Belady*?

- A Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può diminuire con il diminuire del numero dei blocchi di memoria assegnati.
- B Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine resta invariata diminuendo il numero dei blocchi di memoria assegnati.
- C Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può aumentare con il diminuire del numero dei blocchi di memoria assegnati.
- D Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può diminuire con l'aumentare del numero dei blocchi di memoria assegnati.
- E Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può aumentare con l'aumentare del numero dei blocchi di memoria assegnati.

#### 8. Qual è la limitazione principale dell'algoritmo OPT nella sostituzione delle pagine?

- A Richiede conoscenza istantanea del carico di CPU
- B Complessità dell'algoritmo troppo elevata in termini di tempo
- C Richiede la conoscenza futura della successione dei riferimenti
- D I riferimenti alle pagine sono strutturati in una linked-list ad accesso sequenziale
- E Complessità dell'algoritmo troppo elevata in termini di spazio

#### 9. Qual è la differenza fondamentale tra comunicazione *Diretta* ed *Indiretta* dei processi?

- A Nella indiretta i messaggi vengono senza l'ausilio della CPU.
- B Nella indiretta i messaggi si inviano ai processi.

17. Sono presenti 4 istanti di arrivo della CPU, si presentino i processi  $P_1$ ,  $P_2$ ,  $P_3$  e  $P_4$  con durate di esecuzione 3, 2, 1, 4 millisecondi. Se essi vengono serviti mediante SJF con Prelazione, allora quanti millisecondi di tempo complessivo (considerando cioè dall'istante iniziale 0) occorrerà attendere per ottenere il completamento del processo  $P_3$ ?

A 10

B 12

C 8

D 9

E 11

18. Si supponga che in un sistema ogni tipo di risorsa abbia più istanze. Quali delle seguenti affermazioni è falsa?

- A L'arco di richiesta della risorsa punta all'intera classe delle risorse.
- B Un ciclo nel grafo di assegnazione delle risorse non implica necessariamente uno stallo.
- C Migliori prestazioni possibili rispetto sistemi ad una istanza per ogni risorsa.
- D Possono verificarsi cicli nel grafo di assegnazione delle risorse.
- E Un ciclo nel grafo di assegnazione delle risorse implica necessariamente uno stallo.

19. Quali sono le principali categorie in cui risulta possibile classificare le Chia-Sistemi?

- A Controllo processi; Gestione file; Gestione dei dispositivi; Gestione delle informazioni; Comunicazioni.
- B Gestione dei dispositivi; Gestione delle informazioni; Comunicazioni; Gestione FCFS; Controllo processi.
- C Comunicazioni; Gestione delle informazioni; Word processing; Ricorsione; Gestione dei dispositivi.
- D Gestione file; Ricorsione; Comunicazioni; Gestione delle informazioni; Controllo processi.

**21.** Ogni elemento della Tabella di Stato dei Dispositivi è:

- [A] Tipo di dispositivo, grandezza del buffer, indirizzo.  
 [B] Tipo di dispositivo, indirizzo, stato.  
 [C] Stato, id del processo richiedente, tempo di richiesta.  
 [D] Grandezza buffer, stato, indirizzo.  
 [E] Tempo di richiesta, tipo di dispositivo.

**22.** Tra i seguenti elementi, quali sono quelli usualmente utilizzati per il Passaggio Parametri al Sistema Operativo?

|                                 |                                |                             |                                                                              |                                 |
|---------------------------------|--------------------------------|-----------------------------|------------------------------------------------------------------------------|---------------------------------|
| A<br>registri<br>driver<br>pile | B<br>pile<br>blocchi<br>driver | C<br>pcb<br>blocchi<br>pile | <i>No</i> <input checked="" type="checkbox"/><br>registri<br>blocchi<br>pile | E<br>registri<br>blocchi<br>pcb |
|---------------------------------|--------------------------------|-----------------------------|------------------------------------------------------------------------------|---------------------------------|

**23.** In quale dei seguenti casi è possibile applicare lo scheduling della CPU?

- [A] quando un processo dallo stato di attesa passa allo stato di esecuzione  
 [B] quando un processo dallo stato di esecuzione o di pronto passa allo stato di attesa  
 [C] quando un processo dallo stato di esecuzione o di attesa passa allo stato di pronto  
 [D] quando un processo passa allo stato di terminazione  
 [E] quando un processo dallo stato di esecuzione passa allo stato di attesa

**4.** Si supponga di dover gestire un sistema di memoria segmentata.  
 ad un certo istante si ha la seguente configurazione:  
*Valori del segmento n. 7 nella Tabella dei Segmenti: BASE = 8888; LIMITE = 555.*

*Valore di un indirizzo logico: NUMERO SEGMENTO = 12345678.*

*Siamo allora dire che...*

**29.**

C del

**25.** Quale tra le seguenti definisce in modo corretto la struttura generale di un processo Produttore?

- A do { ...; produce un elemento in appena\_prodotto ...; wait(mutex); wait(piene); ...; inserisci appena\_prodotto in vettore; ... signal(vuote); signal(mutex); } while (1);
- B do { ...; produce un elemento in appena\_prodotto ...; wait(vuote); wait(piene); ...; inserisci appena\_prodotto in vettore; ... signal(mutex); signal(mutex); } while (1);
- C do { ...; produce un elemento in appena\_prodotto ...; wait(vuote); wait(mutex); ... ; inserisci appena\_prodotto in vettore; ... signal(piene); signal(mutex); } while (1);
- D do { ...; produce un elemento in appena\_prodotto ...; wait(mutex); wait(vuote); ...; inserisci appena\_prodotto in vettore; ... signal(mutex); signal(piene); } while (1);
- E do { ...; produce un elemento in appena\_prodotto ...; wait(vuote); wait(mutex); ...; inserisci appena\_prodotto in vettore; ... signal(mutex); signal(piene); } while (1);

**26.** Un codice è definito essere **RILOCABILE** se:

- A Fa riferimento esclusivamente ad indirizzi fisici.
- B E' generabile solo da linguaggi ad alto livello
- C E' stata ritardata l'associazione finale degli indirizzi alla fase di caricamento.
- D Quando richiede maggiore memoria di quella prevista.
- E E' possibile riproporlo per una nuova esecuzione.

**27.** In caso di stallo, quali fattori devono essere considerati per la **Selezione della Vittima**?

- A 1) Quantità e tipo di risorse impiegate

A

B

C

D

E

F

G

H

- A Fa riferimento esclusivamente ad un solo processo.
- B E' generabile solo da linguaggi ad alto livello.
- C E' stata ritardata l'associazione finale degli indirizzi alla fase di caricamento.
- D Quando richiede maggiore memoria di quella prevista.
- E' possibile riproporlo per una nuova esecuzione.

**27.** In caso di stallo, quali fattori devono essere considerati per la Selezione della Vittima?

- A 1) Quantità e tipo di risorse impiegate. 2) Numero di eventi di page-fault causati dai singoli processi. 3) Tempo già trascorso e tempo ancora necessario per ogni processo.
- B 1) Tempo già trascorso e tempo ancora necessario per ogni processo. 2) Grafo di assegnazione delle risorse. 3) Grado di multiprogrammazione.
- C 1) Priorità dei processi. 2) Tempo già trascorso e tempo ancora necessario per ogni processo. 3) Quantità e tipo di risorse impiegate.
- D 1) Numero di eventi di page-fault causati dai singoli processi. 2) Tempo già trascorso e tempo ancora necessario per ogni processo. 3) Processi che fanno uso intensivo della tecnica della ricorsione.
- E 1) Grafo di assegnazione delle risorse. 2) Tempo già trascorso e tempo ancora necessario per ogni processo. 3) Quantità e tipo di risorse impiegate.

**28.** Un processo leggero o *Thread*

- A Può condividere direttamente uno spazio logico di indirizzi.
- B Deve condividere dati soltanto attraverso dei files.
- C Può condividere dati solo se essi non superano il limite massimo di 5k bytes.
- D Può condividere dati soltanto se essi sono posizionati su Hard Disk.
- E Può condividere dati soltanto attraverso dei files.

**30.** Una scelta di requisiti:

- A Indipendenza.
- C Variabilità.
- E Mutua Esclusività.

**31.** In quale caso si verifica la mutua esclusività?

- A Se si verificano le condizioni di prelazione.
- B Se si verificano le condizioni di esclusività.
- C Se si verificano le condizioni di esclusività.
- D Se si verificano le condizioni di possesso.
- E Se si verificano le condizioni di esclusività.

**32.** Si definiscono:

- A A="Registro di Rilocazione"; C="Registro di Limite"; B="Memoria".  
 B B="Registro di Rilocazione"; A="Memoria"; C="Registro di Limite".  
 C C="Registro di Rilocazione"; B="Memoria"; A="Registro di Limite".  
 D C="Memoria"; A="Registro di Rilocazione"; B="Registro di Limite".  
 E C="Memoria"; B="Registro di Rilocazione"; A="Registro di Limite".

**30.** Una soluzione del problema della *Sezione Critica* deve soddisfare i seguenti tre requisiti:

- A Indipendenza; Progresso; Attesa limitata       B Mutua Esclusione; Progresso; Attesa limitata  
 C Variabilità; Mutua esclusione; Attesa limitata       D Mutua Esclusione; Progresso; Indipendenza  
 E Mutua Esclusione; Indipendenza; Attesa limitata

**31.** In quale circostanza può verificarsi lo *Stallo dei Processi*:

- A Se si verificano contemporaneamente le condizioni di:  
Mutua prelazione; Possesso e attesa; Impossibilità di esclusione; Attesa circolare.  
 B Se si verificano contemporaneamente le condizioni di:  
Mutua esclusione; Possesso e attesa; Attesa circolare.  
 C Se si verificano almeno due delle condizioni di:  
Mutua esclusione; Possesso e attesa; Impossibilità di prelazione; Attesa circolare.  
 D Se si verificano almeno una delle condizioni di:  
Possesso e attesa; Impossibilità di prelazione; Attesa circolare; Mutua esclusione.  
 E Se si verificano contemporaneamente le condizioni di:  
Mutua esclusione; Possesso e attesa; Impossibilità di prelazione; Attesa circolare.

**32.** Si indichi la sequenza di "Statements" di seguito elencati (A, B, C, D, E, F) che definiscono il *Corpo* dell'istruzione Swap(boolean &a, boolean &b) {...Corpo...}

A = "boolean temp = b"  
B = "..."

- A) Mutua prelazione; Possesso e attesa; impossibilità di prelazione; Attesa circolare.
- B) Se si verificano contemporaneamente le condizioni di:  
Mutua esclusione; Possesso e attesa; Attesa circolare.
- C) Se si verificano almeno due delle condizioni di:  
Mutua esclusione; Possesso e attesa; Impossibilità di prelazione; Attesa circolare.
- D) Se si verificano almeno una delle condizioni di:  
Possesso e attesa; Impossibilità di prelazione; Attesa circolare; Mutua esclusione.
- E) Se si verificano contemporaneamente le condizioni di:  
Mutua esclusione; Possesso e attesa; Impossibilità di prelazione; Attesa circolare.

**32.** Si indichi la sequenza di "Statements" di seguito elencati (A, B, C, D, E, F) che definiscono il *Corpo* dell'istruzione Swap(boolean &a, boolean &b) {...*Corpo...*}

A = "boolean temp = b"  
B = "a = b"  
C = "boolean temp = a"  
D = "b = temp"  
E = "a = temp"  
F = "b = a"

- 
- A) A; B; D;       B) C; B; E;       C) C; B; D;       D) E; B; C;       E) F; A; E;

di *signal()*

- C semafori ad attesa attiva  
 E se il numero di *signal()* è maggiore del numero di *wait()*

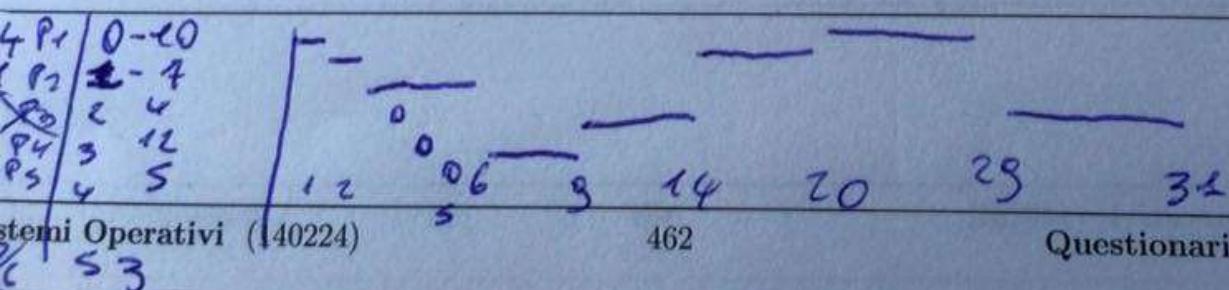
- D semafori binari ad attesa attiva

### 37. Nella Memoria Virtuale, mediante quale meccanismo il paginatore segnala una *Page Fault Trap* ?

- A Mediante il bit di parità associato ad ogni pagina caricata in memoria.  
 B Controllando il bit di validità nella memoria logica.  
 C Controllando il bit di validità nella tabella delle pagine.  
 D Controllando il bit di validità nel registro della CPU.  
 E Controllando che il puntatore alla pagina richiesta sia settato a "null".

### 38. Quando un processo è in esecuzione nella propria sezione critica:

- A Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica  
 B Non si deve consentire a nessun altro processo di effettuare la lettura di un file  
 C Può essere consentita l'esecuzione di altri processi nella propria sezione critica, solo se non richiedono ulteriore memoria di esecuzione.  
 D Può essere consentita l'esecuzione di altri processi nella propria sezione critica  
 E Non si deve consentire a nessun altro processo di poter effettuare stampe



**39.** Quando il sistema entra in una *Situazione di Stallo*, si possono applicare le seguenti *Strategie di ripristino automatico*:

- A Reinizializzazione dei driver delle periferiche; prelazione sulle risorse relative alla memoria secondaria in possesso di uno o più processi in stallo.
- C Prelazione sulla Memoria in possesso di uno o più processi in stallo; terminazione dei processi che hanno fatto maggior uso della memoria.
- E Prelazione sulla CPU in possesso di uno o più processi in stallo; terminazione dei processi che hanno fatto maggior uso della CPU.

**B** Terminazione di un solo processo per l'interruzione dell'attesa circolare; prelazione di tutte le risorse in possesso di uno o più processi in stallo.

**D** Terminazione di uno o più processi per interrompere l'attesa circolare; prelazione su alcune risorse in possesso di uno o più processi in stallo.

**40.** Qual è la principale caratteristica di un'istruzione *TestAndSet*?

- A Esecuzione atomica anche in presenza di più unità di elaborazione.
- C Esecuzione parallela su diverse unità di elaborazione.
- E Attesa attiva su più unità di elaborazione.

**B** Esecuzione atomica per processi a prevalenza di elaborazione.

**D** Attesa attiva su più unità di elaborazione.

**41.** Sia assegnato il seguente Frammento di Codice:

```
#include <stdio.h>
void main(int argc, char *argv[])
{
 int pid;
 pid = fork();
 if (pid < 0) {fprintf(stderr, "CIAO"); exit(-1)}
 else if (pid == 0) {execvp("/bin/ls", "ls", NULL);}
 else {wait(NULL);
 printf("HELLO"); exit(0); }
```

**A** il processo padre stampa CIAO; il processo figlio esegue un comando di "ls".

**B** il processo padre stampa CIAO; il processo figlio stampa HELLO.

**C** il processo padre stampa HELLO; il processo figlio esegue un comando di "ls".

**D** il processo padre stampa HELLO; il processo figlio stampa CIAO.

**E** il processo figlio stampa HELLO; il processo padre esegue un comando di "ls".

**42.** Nei sistemi con scambio di messaggi, in quale circostanza viene generalmente utilizzato l' *Automatic Buffering*?

- A Code con capacità zero.
- C Code con capacità zero, limitata e illimitata.
- E Code con capacità illimitata.

**B** Code con capacità limitata (o zero).

**D** Code con capacità limitata e illimitata.

**43.** Quale strategia adotta l'*Algoritmo del Fornaio* se due processi ricevono lo stesso "numero"?

- A Si seleziona il processo da terminare, valutando le risorse richieste dai processi.
- B Si serve per primo l'ultimo processo entrato in coda.
- C Si serve per prima il processo con il nome "minore".
- D Si esegue una system call random che determina un ordine casuale.
- E Si serve per primo il processo con minor tempo di elaborazione restante.

**44.** Una differenza rilevante tra i *Thread a livello utente* ed i *Thread a livello nucleo* è:

- A I thread a livello nucleo sono interpretati mentre i thread a livello utente sono compilati.
- B I thread a livello utente sono sempre meno compatti dei thread a livello nucleo.
- C I thread a livello nucleo sono compilati mentre i thread a livello utente sono interpretati.
- D I thread a livello utente sono generalmente più veloci dei thread a livello nucleo.
- E I thread del nucleo sono generalmente più veloci dei thread a livello utente.

**45.** Quale dei seguenti componenti non è inutile per il funzionamento un sistema di elaborazione?

- |                                             |                                                       |
|---------------------------------------------|-------------------------------------------------------|
| <input type="checkbox"/> A Hard Disk        | <input type="checkbox"/> B Cache                      |
| <input type="checkbox"/> C Memoria centrale | <input checked="" type="checkbox"/> D Unità di backup |
| <input type="checkbox"/> E Disco RAM        |                                                       |

**46.** In un sistema A di elaborazione concorrente, sia S un semaforo (con S inizializzato a 1) e sia P un processo che contenga al proprio interno il seguente Enunciato: *Codice*

**49.**

Nelle può rappre

- A Quando
- B Se viene
- C Mai se
- D Se le
- E Se il

**50.**

Prelazion oltre defi

**PRIORI**

[P1; 0; 2] Qual è l

A 9

a 1) e sia P un processo  
for i=1 to 8 if (i % 2 > 0) {wait(S); print "i"}  
Si supponga che, in A, P sia l'unico processo che fa riferimento ad S. In output P stampera i seguenti valori:

A 1, 2, 3, 4, 5,

B 2

C 1

D 2, 4, 6, 8

E 1, 3, 5, 7

47. Quale dispositivo del Sistema Operativo controlla il *Grado di Multiprogrammazione*?

- A Lo scheduler a breve termine (in modo automatico).
- B Lo scheduler a medio termine mediante l'utilizzo degli i-node.
- C Lo scheduler a breve termine mediante la gestione dei PCB.
- D Lo scheduler a lungo termine.
- E Lo scheduler a breve termine mediante una *SystemCall*.

48. In uno schema *PRODUTTORE/CONSUMATORE*, sia V un vettore circolare condiviso di grandezza *DIM\_VETTORE*. Qual è il test per la verifica di "Vettore pieno"?

- A inserisci % (*DIM\_VETTORE* - 1) == preleva
- B (*inserisci*+1) % *DIM\_VETTORE* == preleva
- C (*preleva*-1) % *DIM\_VETTORE* == inserisci
- D (*preleva*+1) % *DIM\_VETTORE* == inserisci
- E (*inserisci*-1) % *DIM\_VETTORE* == preleva

**49.** Nello Scheduling della CPU, in quale circostanza la *varianza del Tempo di Risposta* può rappresentare un concreto svantaggio?

- A Quando è rilevante il fattore di prevedibilità.
- B Se viene completata prima l'esecuzione di un processo con minor priorità.
- C Mai se il tempo di risposta complessivo ottenuto è il migliore possibile.
- D Se le richieste delle risorse da parte dei processi supera le risorse disponibili.
- E Se il tempo di attesa per lo sblocco di una periferica supera il tempo previsto.

**50.** Si supponga di effettuare lo scheduling della CPU mediante uno schema a *Prelazione Esterna* (che per valori maggiori di priorità associa priorità maggiori). Sia inoltre definito l'insieme dei processi del tipo [*PROCESSO*; *ISTANTE\_ARRIVO*; *DURATA*; *PRIORITA'*]:

[P1; 0; 2; 2] - [P2; 1; 6; 3] - [P3; 2; 3; 4]

Qual è l'istante in cui il processo P2 sarà completamente eseguito?

- A 9
- B 10
- C 13
- D 12
- E 11

FINE

0 2 2  
1 6 3

—  
—

**1. Quale dispositivo del Sistema Operativo controlla il Grado di Multiprogrammazione?**

- A - Lo scheduler a breve termine (in modo automatico).
- B - Lo scheduler a medio termine mediante l'utilizzo degli i-node
- C - Lo scheduler a breve termine mediante la gestione dei PCB
- D - Lo scheduler a lungo termine
- E - Lo scheduler a breve termine mediante una System Call

**2. In uno schema PRODUTTORE/CONSUMATORE, sia V un vettore circolare condiviso di grandezza**

**DIM\_VETTORE. Qual'è il test per la verifica di "Vettore pieno" ?**

- A - Inserisci % (DIM\_VETTORE-1) == preleva
- B - (inserisci +1) % DIM\_VETTORE == preleva
- C - (preleva-1) % DIM\_VETTORE == inserisci
- D - (preleva+1) % DIM\_VETTORE == inserisci
- E - (inserisci-1) % DIM\_VETTORE == preleva

**3. Nella memoria Virtuale, mediante quale meccanismo il paginatore segnala una Page Fault Trap?**

- A - Mediante il bit di parità associato ad ogni pagina caricata in memoria
- B - Controllando il bit di validità nella memoria logica
- C - Controllando il bit di validità nella tabella delle pagine
- D - Controllando il bit di validità nel registro della CPU
- E - Controllando che il puntatore alla pagina richiesta sia settato a "null"

**4. Quando un processo è in esecuzione nella propria *sezione critica***

- A - Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica
- B - Non si deve consentire a nessun altro processo di effettuare la lettura di un file
- C - Può essere consentita l'esecuzione di altri processi nella propria sezione critica, solo se non richiedono ulteriore memoria di esecuzione
- D - Può essere consentita l'esecuzione di altri processi nella propria sezione critica
- E - Non si deve consentire a nessun altro processo di poter effettuare stampe

**5. Una soluzione del problema della *Sezione Critica* deve soddisfare i seguenti tre requisiti:**

- A - Indipendenza; Progresso; Attesa limitata
- B - Mutua Esclusione; Progresso, Attesa limitata
- C - Variabilità; Mutua esclusione, Attesa limitata
- D - Mutua Esclusione; Progresso; Indipendenza
- E - Mutua Esclusione; Indipendenza; Attesa limitata

**6. In Quale circostanza può verificarsi lo *Stallo dei Processi***

- A - Se si verificano contemporaneamente le condizioni di:
  - Mutua prelazione; Possesso e attesa; Impossibilità di esclusione; Attesa circolare
- B - Se si verificano contemporaneamente le condizioni di:
  - Mutua esclusione; Possesso e attesa; Attesa circolare.
- C - Se si verificano almeno due delle condizioni di:
  - Mutua esclusione; Possesso e attesa; Impossibilità di prelazione; Attesa circolare
- D - Se si verificano almeno una delle condizioni di:
  - Mutua esclusione; Possesso e attesa; Impossibilità di prelazione; Attesa circolare.
- E - Se si verificano contemporaneamente le condizioni di:
  - Mutua esclusione; Possesso e attesa; Impossibilità di prelazione; Attesa circolare.

**7. Si indichi la sequenza di “Statements” di seguito elencati (A, B, C, D, E, F) che definiscono il Corpo dell’istruzione Swap(boolean &a, boolean &b) {...Corpo...}**

A = “boolean temp = b”

B = “a = b”

C = “boolean temp = a”

D = “b = temp”

E = “a = temp”

F = “b = a”

A - A; B; D;

B - C; B; E;

C - C; B; D;

D - E; B; C;

E - F; A; E;

**8. Cosa caratterizza l’Anomalia di Belady?**

A - Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può diminuire con il diminuire del numero dei blocchi di memoria assegnati.

B - Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine resta invariata diminuendo il numero dei blocchi di memoria assegnati

C - Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può aumentare con il diminuire del numero dei blocchi di memoria assegnati.

D - Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può diminuire con l’aumentare del numero dei blocchi di memoria assegnati

E - Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può aumentare con l’aumentare del numero dei blocchi di memoria assegnati.

**9. Qual’è la limitazione principale dell’algoritmo OPT nella sostituzione delle pagine**

A - Richiede conoscenza istantanea del carico di CPU

B - Complessità dell’algoritmo troppo elevata in termini di tempo

C - Richiede la conoscenza futura della successione dei riferimenti

D - I Riferimenti alle pagine sono strutturati in una linked-list ad accesso sequenziale

E - Complessità dell’algoritmo troppo elevata in termini di spazio

**10. Quale tra le seguenti definisce in modo corretto la struttura generale di un processo Produttore?**

A - do { ...; produce un elemento in appena\_prodotto ... ; wait (mutex); wait(piene); ... ; inserisci appena\_prodotto in vettore; .. signal (vuote); signal(mutex); } while (1);

B - do { ...; produce un elemento in appena\_prodotto ... ; wait(vuote); wait(piene); ... ; inserisci appena\_prodotto in vettore; signal(mutex); signal(mutex); } while (1);

C - do { ...; produce un elemento in appena\_prodotto ... ; wait(vuote); wait(mutex); ... ; inserisci appena\_prodotto in vettore; ... signal(piene); signal(mutex); } while (1);

D - do { ...; produce un elemento in appena\_prodotto ... ; wait(mutex); wait(vuote); ... ; inserisci appena\_prodotto in vettore; ... signal(mutex); signal(piene); } while(1);

E - do { ...; produce un elemento in appena\_prodotto ... ; wait(vuote); wait(mutex); ... ; inserisci appena\_prodotto in vettore; ... signal(mutex); signal(piene); } while (1);

**11. Un codice è definito essere RILOCABILE se:**

- A - fa riferimento esclusivamente ad indirizzi fisici
- B - E' generabile solo da linguaggi ad alto livello
- C - E' stata ritardata l'associazione finale degli indirizzi alla fase di caricamento
- D - Quando richiede maggiore memoria di quella prevista
- E - E' possibile riproporlo per una nuova esecuzione

**12. In caso di stallo, quali fattori devono essere considerati per la Selezione della Vittima?**

- A - 1) Quantità e tipo di risorse impiegate. 2) Numero di eventi di page-fault causati dai singoli processi.
  - 3) Tempo già trascorso e tempo ancora necessario per ogni processo
- B - 1) Tempo già trascorso e tempo ancora necessario per ogni processo.
  - 2) Grafo di assegnazione delle risorse. 3) Grado di multiprogrammazione
- C - 1) Priorità dei processi. 2) Tempo già trascorso e tempo ancora necessario per ogni processo.
  - 3) Quantità e tipo di risorse impiegate
- D - 1) Numero di eventi di page-fault causati dai singoli processi.
  - 2) Tempo già trascorso e tempo ancora necessario per ogni processo.
  - 3) Processi che fanno uso intensivo della tecnica della ricorsione
- E - 1) Grafo di assegnazione delle risorse. 2) Tempo già trascorso e tempo ancora necessario per ogni processo.
  - 3) Quantità e tipo di risorse impiegate.

**13. Un processo leggero o Thread**

- A - Può condividere direttamente uno spazio logico di indirizzi
- B - Deve condividere dati soltanto attraverso dei files
- C - Può condividere dati solo se essi non superano il limite massimo di 5k bytes
- D - Può condividere dati soltanto se essi sono posizionati su Hard Disk
- E - Può condividere dati soltanto attraverso dei files

**14. Quando il sistema entra in una Situazione di Stallo, si possono applicare le seguenti Strategie di ripristino automatico:**

- A - Reinizializzazione dei driver delle periferiche; prelazione sulle risorse relative alla memoria secondaria in possesso di uno dei processi in stallo.
- B - Terminazione di un solo processo per l'interruzione dell'attesa circolare; prelazione di tutte le risorse in possesso di uno o più processi in stallo
- C - Prelazione sulla Memoria in possesso di uno o più processi in stallo; terminazione dei processi che hanno fatto maggior uso di memoria
- D - Terminazione di uno o più processi per interrompere l'attesa circolare; prelazione su alcune risorse in possesso di uno o più processi in stallo
- E - Prelazione sulla CPU in possesso di uno o più processi in stallo; terminazione dei processi che hanno fatto maggior uso della CPU

**15. Nei sistemi con scambio di messaggi, in quale circostanza viene generalmente utilizzato l'Automatic Buffering?**

- A - Code con capacità zero
- B - Code con capacità limitata (o zero)
- C - Code con capacità zero, limitata e illimitata
- D - Code con capacità limitata e illimitata
- E - Code con capacità illimitata

**16. Sia assegnato il seguente Frammento di Codice:**

```
#include <stdio.h>
void main(int argc, char *argv[])
{
 int pid;
 pid = fork();
 if (pid < 0) {fprintf(stderr, "CIAO"); exit(-1)}
 else if (pid == 0) {execlp("/bin/ls/", "ls", NULL);}
 else {wait(NULL);
 printf("HELLO"); exit(0); }
}
```

- A - Il processo padre stampa CIAO; il processo figlio esegue un comando di “ls”
- B - Il processo padre stampa CIAO; il processo figlio stampa HELLO.
- C - Il processo padre stampa HELLO; il processo figlio esegue un comando di “ls”
- D - Il processo padre stampa HELLO; il processo figlio stampa CIAO.
- E - Il processo figlio stampa HELLO; il processo padre esegue un comando di “ls”

**17. Ogni elemento della Tabella di Stato dei dispositivi contiene**

- A - Tipo di dispositivo, grandezza del buffer, indirizzo.
- B - Tipo di dispositivo, indirizzo, stato
- C - Stato, id del processo richiedente, tempo di richiesta
- D - Grandezza buffer, stato, indirizzo.
- E - Tempo di richiesta, tipo di dispositivo.

**18. Tra i seguenti elementi, quali sono quelli usualmente utilizzati per il Passaggio di Parametri al Sistema Operativo?**

- A - Registri Driver Pile
- B - Pile, Blocchi, Driver
- C - PCB, Blocchi, Pile
- D - Registri, Blocchi, Pile
- E - Registri, Blocchi, PCB

**19. In quale dei seguenti casi è possibile applicare lo scheduling della CPU?**

- A - quando un processo dallo stato di attesa passa allo stato di esecuzione
- B - quando un processo dallo stato di esecuzione o di pronto passa allo stato di attesa
- C - quando un processo dallo stato di esecuzione o di attesa passa allo stato di pronto
- D - quando un processo passa allo stato di terminazione
- E - quando un processo dallo stato di esecuzione passa allo stato di attesa

**20. Si supponga di dover gestire un sistema di memoria segmentata.**

**Ad un certo istante si ha la seguente configurazione:**

A - Valori del segmento n.7 nella Tabella dei Segmenti: BASE = 8888; LIMITE = 555

...

**21. Nello scheduling della CPU, in quale circostanza la varianza del Tempo di Risposta può rappresentare un concreto svantaggio?**

- A - Quando è rilevante il fattore di prevedibilità
- B - Se viene completata prima l'esecuzione di un processo con minor priorità
- C - Mai se il tempo di risposta complessivo ottenuto è il migliore possibile
- D - Se le richieste delle risorse da parte dei processi supera le risorse disponibili.
- E - Se il tempo di attesa per lo sblocco di un a periferica supera il tempo previsto.

**22. Si supponga di effettuare lo scheduling della CPU mediante uno schema a *Prelazione Esterna* (che per valori maggiori di priorità associa priorità maggiori). Sia inoltre definito l'insieme dei processi del tipo [PROCESSO; ISTANTE\_ARRIVO; DURATA; PRIORITA']:**

[P1; 0; 2; 2] - [P2; 1; 6; 3] - [P3; 2; 3; 4]

Qual'è l'istante in cui il processo P2 sarà completamente eseguito?

- A - 9
  - B - 10
  - C - 13
  - D - 12
  - E - 11
- 
- 

**23. Si supponga che in un sistema ogni tipo di risorsa abbia più istanze. Quali delle seguenti affermazioni è falsa?**

- A - L'arco di richiesta della risorsa punta all'intera classe delle risorse
- B - Un ciclo nel grafo di assegnazione delle risorse non implica necessariamente uno stallo
- C - Migliori prestazioni possibili rispetto sistemi ad una istanza per ogni risorsa
- D - Possono verificarsi cicli nel grafo di assegnazione delle risorse.
- E - Un ciclo nel grafo di assegnazione delle risorse implica necessariamente uno stallo

**24. Quali sono le principali categorie in cui risulta possibile classificare le Chiamate di Sistema?**

- A - Controllo processi; Gestione file; Gestione dei dispositivi; Gestione delle informazioni; Comunicazioni
- B - Gestione dei dispositivi; Gestione delle informazioni; Comunicazioni; Gestione FCFS; Controllo Processi
- C - Comunicazioni; gestione delle informazioni; Word processing; Ricorsione; Gestione dei dispositivi
- D - Gestione file; Ricorsione; Comunicazioni; Gestione delle informazioni; Controllo processi

...

**25. Quale strategia adotta l'Algoritmo del Fornaio se due processi ricevono lo stesso "numero" ?**

- A - Si seleziona il processo da terminare, valutando le risorse richieste dai processi
- B - Si serve per primo l'ultimo processo entrato in coda.
- C - Si serve per prima il processo con il nome "minore".
- D - Si esegue una system call random che determina un ordine casuale.
- E - Si serve per primo il processo con minor tempo di elaborazione restante

- 26. Una differenza rilevante tra i Thread a livello utente e i Thread a livello nucleo è**
- A - I thread a livello nucleo sono interpretati mentre i thread a livello utente sono compilati
  - B - I thread a livello utente sono sempre meno compatti dei thread a livello nucleo
  - C - I thread a livello nucleo sono compilati mentre i thread a livello utente sono interpretati
  - D - I thread a livello utente sono generalmente più veloci dei thread a livello nucleo.
  - E - I thread del nucleo sono generalmente più veloci dei Thread a livello utente.

- 27. Quale dei seguenti componenti non è inutile per il funzionamento di un sistema di elaborazione**

- A - Hard disk
- B - Cache
- C - Memoria Centrale
- D - Unità di Backup
- E - Disco RAM

- 28. Quale tra quelle di seguito elencate, caratterizza nel modo migliore la definizione di un interprete di comandi (Shell)?**

- A - E' l'applicativo principale di conversione dei comandi in files binari eseguibili
- B - E' un'interfaccia di comunicazione tra il Sistema Operativo e le periferiche in esso definite
- C - E' il modulo software che permette di effettuare gli upgrade del Sistema Operativo
- D - E' un'interfaccia tra utente e Sistema Operativo
- E - E' il modulo software che permette di gestire lo scheduling dei processi

- 29. Nel meccanismo di paginazione a cosa serve il bit di validità nella tabella delle pagine?**

- A - Definisce la condizione iniziale per il salvataggio dei valori della tabella in uno stack
- B - Definisce una condizione necessaria per l'inizio del caricamento di una pagina
- C - Definisce una condizione sufficiente per il caricamento immediato delle pagine indicate in tabella.
- D - Definisce la validità dei valori che sono memorizzati nelle pagine indicate.
- E - Definisce la validità del numero di elementi presenti nella tabella.

- 30. Qual'è la caratteristica principale della Lettura Anticipata (Read-Ahead) di un file da disco?**

- A - Vengono caricate in memoria centrale anche le pagine successive a quella espressamente richiesta.
- B - Viene assegnata priorità maggiore alle letture e priorità inferiore alle scritture
- C - Il file viene preventivamente e completamente letto dal Sistema Operativo per verificare la correttezza del codice in esso contenuto
- D - Si leggono e si mettono nella cache la pagina richiesta e parecchie pagine successive
- E - Viene assegnata priorità maggiore alle scritture e priorità inferiore alle letture

- 31. Cosa è possibile fare quando si esaurisce l'Area di Avvicendamento?**

- A - Terminazione forzata di processi e/o arresto del sistema
- B - Si riduce il quanto di tempo assegnato ad un singolo processo
- C - Si eseguono solo fork associate a processi appartenenti alla cosa con priorità massima
- D - Si cambia criterio di gestione della memoria centrale.
- E - Si effettua una Garbage Collection dei files

- 32. In Un file-System distribuito, quando si preferisce il metodo di scrittura differita per l'aggiornamento della cache?**

- A - File aperti per lunghi periodi e modificati spesso
- B - File aperti per lunghi periodi e modificati raramente
- C - File aperti per brevi periodi ma non modificabili
- D - File aperti per lunghi periodi ma non modificabili.
- E - File aperti per brevi periodi e modificati raramente

**33. Nell'ambito di uno scheduling CPU a code multiple, quale tra i seguenti schemi ha maggior senso?**

- A - FCFS per processi in primo piano e R.R. per processi in background.
- B - R.R. per processi in primo piano e SJF per processi in background.
- C - SJF per processi in primo piano e SJF con prelazione per processi in background
- D - R.R. per processi in primo piano e FCFS per processi in background
- E - SJF con prelazione per processi in primo piano e SJF per processi in Background

**34. Si supponga di aver sviluppato un algoritmo di sincronizzazione per soli due Processi Pi e Pj; si assuma che la variabile turno (che vale i oppure j) individua il processo (pi oppure Pj) che ha permesso di entrare nella propria sezione critica. A quali requisiti soddisfa l'algoritmo indicato?**

```
do {
 while (turno != i)
 ... sezione critica ...
 turno = j
 ... sezione critica ...
} while (1);
```

- A - Progresso, Mutua esclusione, Indipendenza
- B - Attesa Limitata, Progresso
- C - Mutua esclusione, Indipendenza, Conflitto
- D - Attesa Limitata, Progresso, Mutua esclusione
- E - Mutua esclusione, Attesa limitata

**35. Che relazione lega le 3 entità E1="indirizzi Fisici", E2="registro di Rilocazione", E3="indirizzi Logici"?**

- A - E2 = E1-E3
- B - E3 = E2+E1
- C - E2 = E3+E1+1
- D - E2 = E3+E1-1
- E - E2 = E1+E3

**36. Quando un processo è in esecuzione nella propria sezione critica**

- A - Non si deve consentire a nessun altro processo di effettuare la lettura di un file
- B - Può essere consentita l'esecuzione di altri processi nella propria sezione critica
- C - Non si deve consentire a nessun altro processo di poter effettuare stampe
- D - Può essere consentita l'esecuzione di altri processi nella propria sezione critica, solo se non richiedono ulteriore memoria di esecuzione.
- E - Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica.

**37. In quale modo è possibile realizzare efficacemente un File-System Condiviso?**

- A - Con la duplicazione delle risorse condivise.
- B - Mediante dei link
- C - Mediante directory a singolo livello
- D - Riducendo il grado di multi-programmazione
- E - Aumentando la frequenza del processore

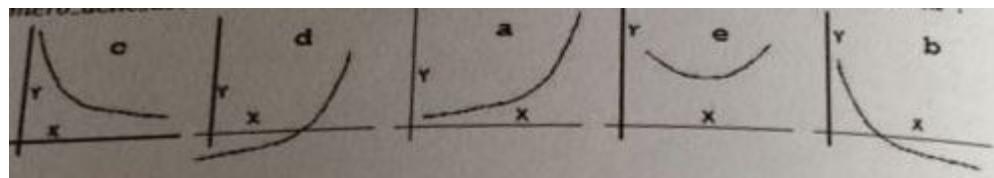
**38. Nella realizzazione delle directory mediante la tecnica dell'Hash qual'è tra le seguenti, una tecnica alternativa per la risoluzione del problema delle collisioni?**

- A - L'impiego di liste concatenate
- B - Tecnica dell'assegnazione concatenata
- C - La tecnica dell'indirizzamento indiretto
- D - Tecnica dell'assegnazione contigua
- E - La tecnica delle directory a singolo livello.

**39. Quale dei seguenti grafici descrive la relazione generale che intercorre tra**

Y = Numero delle assenza delle pagine

X = Numero dei blocchi di memoria ?



- A - a
- B - b
- C - d
- D - e
- E - c

**40. Sia K un File-System strutturato con directory ROOT (al livello1) ed altri 4 livelli.**

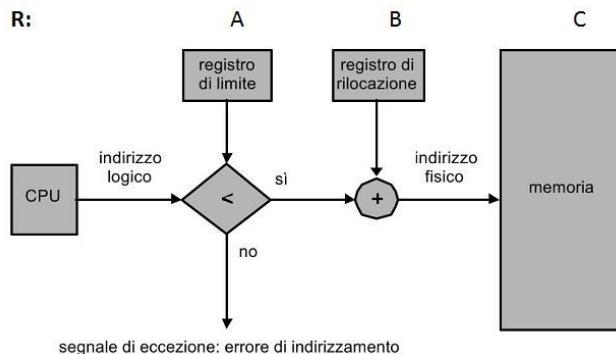
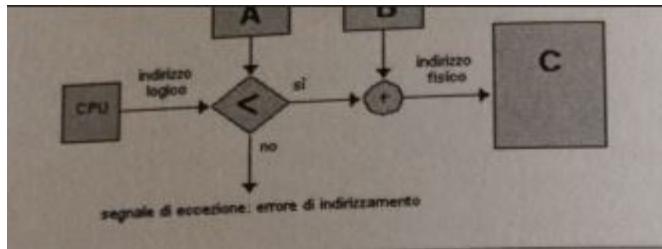
**Se per ogni livello si ha che (file+Sottodirectory) <=3, qual'è il numero massimo di files contenuti in K ?**

- A - 162
- B - 27
- C - 729
- D - 81
- E - 243

**41. Tra le seguenti strategie di instradamento, quali assicurano un arrivo ordinato dei dati trasmessi?**

- A - Instradamento Virtuale.
- B - Instradamento Dinamico e Virtuale
- C - Nessuna strategia può garantire al 100% un ordine di arrivo
- D - Instradamento Dinamico e Fisso
- E - Instradamento Fisso e Virtuale

**42. Nell'ambito della Gestione della Memoria, che cosa rappresentano le entità A, B e C del grafo sottostante?**



A - C = "Registro di Rilocazione"; B="Memoria"; A="Registro di Limite".

B - B = "Registro di Rilocazione"; A="Memoria"; C="Registro di Limite".

C - C = "Memoria"; B="Registro di Rilocazione"; A="Registro di Limite".

D - C = "Memoria"; A="Registro di Rilocazione"; B="Registro di Limite".

E - A = "Registro di Rilocazione"; C="Registro di Limite"; B= " Memoria".

**43. Con l'Algoritmo del Fornaio si serve prima**

A - Il cliente con il numero progressivo più basso che richiede la minor quantità di risorse

B - Il cliente che richiede la minor quantità di risorse

C - Il cliente con il minor valore medio ottenuto sommando il proprio numero progressivo con il proprio nome

D - Il cliente con il numero progressivo più basso e, a parità, si serve quello con nome minore

E - Il cliente con il nome minore e, a parità, si serve quello con numero progressivo più basso

**44. Si supponga di voler adottare uno schema di Gestione gerarchica della memoria mediante la Paginazione a due livelli. Si supponga che lo spazio degli indirizzi logici sia a 28 bit e che la grandezza di una pagina si di un byte. Qual'è il numero massimo degli elementi della Tabella esterna?**

A -  $2^{exp12}$

B - 1k

C -  $2^{exp20}$

D -  $2^{exp16}$

E -  $2^{exp18}$

**45. Quando un processo è in esecuzione nella propria sezione critica:**

- A - Non si deve consentire a nessun altro processo di effettuare la lettura di un file
- B - Può essere consentita l'esecuzione di altri processi nella propria sezione critica
- C - Non si deve consentire a nessun altro processo di poter effettuare stampe
- D - Può essere consentita l'esecuzione di altri processi nella propria sezione critica, solo se non richiedono ulteriore memoria di esecuzione.
- E - Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica.

**46. In quale modo è possibile realizzare efficacemente un FileSystem Condiviso?**

- A - Con la duplicazione delle risorse condivise
- B - Mediante dei link
- C - Mediante directory a singolo livello
- D - Riducendo il grado di multi-programmazione
- E - Aumentando la frequenza del processore

**47. Quale delle seguenti strategie di gestione della memoria non è soggetta all'Anomalia di Belady?**

- A - Solo sostituzione LRU
- B - Sostituzione Ottimale ed LRU
- C - SJF ed LRU
- D - Solo Sostituzione Ottimale
- E - LRU e FIFO

**48. Per uno Stato Sicuro si definisce la Sequenza Sicura dei processi <P1, P2,...,Pn> se:**

- A - Per ogni Pi le richieste che Pi può ancora fare si possono soddisfare impiegando le risorse attualmente disponibili più le risorse possedute da tutti i Pj con j < (i+1)
- B - Per ogni Pi le richieste che Pi può ancora fare si possono soddisfare impiegando le risorse attualmente disponibili.
- C - Per ogni Pi le richieste che Pi può ancora fare si possono soddisfare impiegando le risorse attualmente disponibili più le risorse possedute da tutti i Pj con j < 1
- D - Per ogni Pi le richieste che Pi può ancora fare si possono soddisfare impiegando le risorse possedute da tutti i Pj con j < i
- E - Per ogni Pi le richieste che Pi può ancora fare si possono soddisfare impiegando le risorse attualmente disponibili.

**49. Nello Scheduling della CPU, in quale circostanza la varianza del Tempo di Risposta può rappresentare un concreto svantaggio?**

- A - Quando è rilevante il fattore di prevedibilità
- B - Se viene completata prima l'esecuzione di un processo con minor priorità
- C - Mai se il tempo di risposta complessivo ottenuto è il migliore possibile
- D - Se le richieste delle risorse da parte dei processi supera le risorse disponibili
- E - Se il tempo di attesa per lo sblocco di una periferica supera il tempo previsto

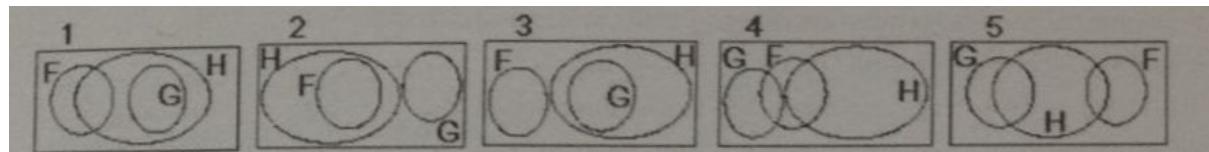
**50. Si supponga di effettuare lo scheduling della CPU mediante uno schema a Prelazione Esterna (che per valori maggiori di priorità associa priorità maggiori). Sia inoltre definito l'insieme dei processi del tipo [PROCESSO; ISTANTE\_ARRIVO; DURATA; PRIORITA']:**

[P1; 0;2;2] - [P2; 1;6;3] - [P3; 2;3;4]

**Qual'è l'istante in cui il processo p2 sarà completamente eseguito?**

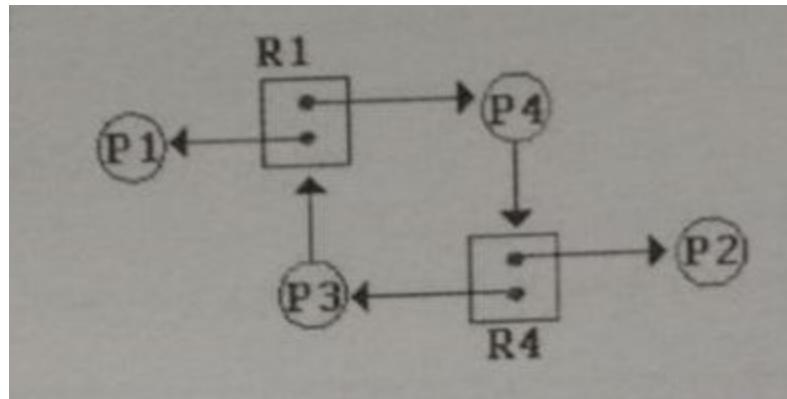
- A - 9
  - B - 10
  - C - 13
  - D - 12
  - E - 11
- 
- 

**51. Nello Stallo Processi, se F = Stallo, G = Stato\_Sicuro, H = Stato\_non\_Sicuro, allora quale tra le seguenti figure rappresenta la corretta relazione tra gli insiemi F, G, H ?**



- A - 1
- B - 2
- C - 3
- D - 4
- E - 5

**52. Considerando il seguente grafo di Assegnazione delle Risorse, determinare la situazione istantanea che esso rappresenta:**



- A - Ciclo con stallo sicuro
- B - Stallo ciclico
- C - Stallo indefinito.
- D - Condizione sufficiente di stallo
- E - Ciclo senza stallo.

**53. In che modo opera una TLB (Translation Look-aside Buffer)?**

- A - Scambia il valore dei registri tra 2 processi concorrenti
- B - Scambia i contenuti delle pagine di 2 processi
- C - Ricerca binaria di una chiave
- D - Permette lo scambio di posizione di 3 processi
- E - Ricerca contemporanea (di un valore) tra tutte le chiavi

**54. In genere, in quale tra le fasi di 1) Compilazione 2) Caricamento 3) Esecuzione è possibile creare una associazione tra istruzioni e dati con indirizzi di memoria?**

- A - 1,3
- B - 2
- C - 1,2
- D - 1,2,3
- E - 2,3

**55. Si supponga di tracciare un Diagramma di Transizione degli Stati di un Processo. In quale unica circostanza si consente ad un processo di passare dallo stato PRONTO a quello di TERMINATO?**

- A - mai
- B - Quando si effettua un RESET del sistema
- C - Le risorse richieste dal processo sono state rimosse dal sistema
- D - Si è superata la grandezza massima degli elementi contenibili nella cosa
- E - Quando il tempo di attesa supera un valore prefissato

**56. Si supponga che nello scheduling della CPU, al momento zero, si presenti la serie ordinata di processi P1, P2, P3, P4, con rispettive durate di sequenza, 6, 8, 7, 3 unità di tempo. Se essi vengono serviti mediante SJF senza prelazione quale è il tempo medio di attesa per ogni processo?**

- A - 8
- B - 9,5
- C - 7
- D - 9
- E - 10

**57. In uno schema PRODUTTORE/CONSUMATORE, sia V un vettore circolare condiviso di grandezza DIM\_VETTORE. Qual'è il test per la verifica di "Vettore pieno" ?**

- A - (preleva-1)%DIM\_VETTORE == inserisci
- B - (preleva+1)%DIM\_VETTORE ==inserisci
- C - (inserisci-1)% DIM\_VETTORE == preleva
- D - inserisci%(DIM\_VETTORE-1) == preleva
- E - (inserisci+1)%DIM\_VETTORE == preleva

**58. Quali sono i dispositivi di memoria direttamente accessibili dalla CPU?**

- A - Memoria Secondaria, Memoria Centrale
- B - Cache dispositivi i/o, Memoria Centrale
- C - Memoria Centrale, Registri CPU
- D - Registri CPU, Dischi, Memoria Centrale
- E - Registri CPU, Memoria Secondaria, Cache

**59. Si supponga che un controller di un Hard Disk gestisca i blocchi difettosi mediante la tecnica del SECTOR SLIPPING (traslazione dei settori). Si supponga che si sia verificato un errore nel blocco logico 17 e che il primo settore di riserva disponibile sia quello successivo al settore 202. Quale nuova posizione occuperà il settore 100 dopo l'applicazione della correzione?**

- A - 101
- B - 201
- C - 203
- D- 99
- E - 219

**60. Cosa contiene un i-node in un sistema UFS (Unix file system)?**

- A - Le strutture delle directory
- B - I blocchi di controllo delle partizioni
- C - I Descrittori dei files
- D - I riferimenti ai blocchi di riserva
- E - I blocchi di controllo dell'avviamento

**61. Lo schema di nominazione di un DFS mediante un NFS permette  
DFS (file-system distribuito); NFS (Network File System)**

- A - Eliminare ricorsivamente elementi obsoleti
- B - Copiare files più velocemente
- C - Unire le directory remote alle directory locali
- D - Ottenere prestazioni elevate ... traffico elevato
- E - Unire le periferiche remote alle periferiche locali

**62. Nel passaggio dei parametri al Sistema Operativo, si preferisce il ...Blocco anziché Passaggio tramite Registri se:**

- A - Il numero dei registri è maggiore del numero dei Blocchi.
- B - Il numero dei parametri è maggiore del numero dei registri
- C - Il numero dei Blocchi è inferiore al numero dei parametri
- D - Il numero dei parametri è inferiore al numero dei Blocchi
- E - Il numero dei registri è maggiore del numero dei parametri

**63. Quale caratteristica accomuna le due tecniche CLV e CAV dei dischi ottici**

- A - Sono entrambe gestibili mediante opportune System Call
- B - Permettono la ridefinizione del numero di tracce contenute in ogni cilindro
- C - Migliori tempi di Backup dei dati
- D - Mantengono costante la quantità di dati che scorre sotto le testine
- E - Minimizzano i tempi di accesso ai dati

**64. Qual'è lo scopo principale del contatore delle aperture di un file?**

- A - Poter stabilire se ha raggiunto un valore max consentito.
- B - Poter stabilire la frequenza di accesso ai file
- C - Poter stabilire quando il file non è più in uso
- D - Poter stabilire il numero di chiamate possibili dei file
- E - Poter stabilire la relativa priorità in caso di Swap

**65. Qual'è la peculiarità dell'algoritmo LRU di sostituzione delle pagine nella Gestione della memoria?**

- A - Ad ogni pagina associa l'indirizzo della memoria centrale in cui è stata memorizzata per l'ultima volta
- B - Ad ogni pagina associa l'indirizzo della tavola di tutte le pagine del processo
- C - Ad ogni pagina associa l'istante in cui è stata usata per la prima volta
- D - Sostituisce la pagina entrata per prima in memoria centrale
- E - Ad ogni pagina associa l'istante in cui è stata usata per l'ultima volta

**66. Si supponga che nello scheduling della CPU, si presentino i processi P1, P2, P3, P4 con rispettivi istanti di arrivo 0,1,2,3 e con rispettive durate di sequenza 7,2,5,1 millisecondi. Se essi vengono serviti mediante SJF con Prelazione, allora quanti millisecondi di tempo complessivo (considerando cioè dall'istante iniziale 0) occorrerà attendere per ottenere il completamento del processo P3?**

- A - 8
  - B - 9
  - C - 12
  - D - 11
  - E - 10
- 
- 

**67. Tra le seguenti strategie di instradamento, quali assicurano un arrivo ordinato dei dati trasmessi?**

- A - Nessuna strategia può garantire al 100% un ordine di arrivo
- B - Instradamento Dinamico e Fisso
- C - Instradamento Dinamico e Virtuale
- D - Instradamento Fisso e Virtuale
- E - Instradamento Virtuale

**68. Quale è la principale differenza tra A= “Disco RAM” e B= “Cache di un Disco”?**

- A - A è una virtualizzazione di B
- B - B è totalmente controllato dall’utente; A è sotto il controllo dei Sistema Operativo
- C - A è totalmente controllato dall’utente; B è sotto il controllo dei Sistema Operativo
- D - Se il numero attuale dei Threads è elevato allora il Sistema Operativo scarica i dati di A a B
- E - Si tratta solo di una differenza “fisica” ma non “logica”

**69. In che modo calcolo Tattesa (Tempo Attesa di elaborazione) di un processo P?**

- A - Tattesa = TempoFineProcesso - TempoArrivoProcesso + DurataProcesso
- B - Tattesa = DurataProcesso - TempoArrivoProcesso - TempoFineProcesso
- C - Tattesa = TempoFineProcesso - TempoArrivoProcesso - DurataProcesso
- D - Tattesa = TempoFineProcesso + TempoArrivoProcesso - DurataProcesso
- E - Tattesa = TempoArrivoProcesso + TempoFineProcesso + DurataProecss

**70. Nell’ambito dello stallo dei processi, quale, tra le seguenti, costituisce una strategia alternativa per il non verificarsi delle condizioni di “Possesso e Attesa”?**

- A - Un processo può richiedere risorse solo se non ne possiede.
- B - Modifica della tabella delle priorità dei processi
- C - Aumento della frequenza di intervento di un algoritmo di rilevamento stallo
- D - Un processo può attendere che si liberino solo risorse condivise.
- E - Un processo può attendere che si liberino solo risorse non condivise

**71. Qual’è la differenza principale tra (1) “Batterie di sistemi ASIMMETRICHE” e (2) ”Batterie di sistemi SIMMETRICHE”?**

- A - In (2) c’è un maggior livello di condivisione di risorse
- B - In (2) la comunicazione è basata su protocolli seriali, in (1) su protocolli paralleli
- C - In (2) un calcolatore resta in attesa attiva
- D - In (1) un calcolatore resta in attesa attiva
- E - In (1) la comunicazione è basata su protocolli seriali, in (2) su protocolli paralleli

**72. Generalmente in quali delle seguenti circostanze il valore del semaforo può essere negativo?**

- A - Semafori ad attesa passiva
- B - Se il numero di Signal() è maggiore del numero di wait()
- C - Semafori binari ad attesa attiva
- D - se il numero di wait() è maggiore al numero di Signal()
- E - Semafori ad attesa attiva

**73. Quali sono le principali categorie in cui risulta possibile classificare le Chiamate di Sistema?**

- A - Controllo processi; gestione file; Gestione dei dispositivi; Gestione delle informazioni; Comunicazioni
- B - Gestione file, Ricorsione; Comunicazioni; Gestione delle informazioni; Controllo processi
- C - Comunicazioni; gestione delle informazioni; Word Processing, Ricorsione, Gestione dei Dispositivi;
- D - Comunicazioni, Controllo processi, Word Processing, Gestione dei dispositivi, Gestione FCFS, su disco
- E - Gestione dei dispositivi; Gestione delle Informazioni; Comunicazioni; Gestione FCS su disco; Controllo Processi

**74. Nella Gestione della memoria, quale, tra i seguenti, rappresenta uno svantaggio nell'uso della Tabella delle Pagine Invertite?**

- A - Il processo di inversione potrebbe richiedere un eccessivo tempo di elaborazione
- B - La disposizione degli elementi della tabella non è organizzata per indirizzi virtuali
- C - Sensibile aumento del numero degli swap di pagina necessari
- D - Maggior spreco di memoria
- E - La condivisione della memoria è più difficile da realizzare

**75. Nello scheduling della CPU, si supponga che i processi P1, P2, P3, P4 si presentino con rispettivi istanti di arrivo 0,1,2,3 e con rispettive durate di sequenza 8, 4, 9, 5 millisecondi. Se essi vengono serviti mediante SJF con Prelazione qual'è il tempo medio di attesa per ogni processo?**

- A - 6
- B - 9
- C - 8
- D - 6,5
- E - 7,5

**76. Lo schema di ricezione asincrona è caratterizzato dal fatto che:**

- A - il ricevente riceve un messaggio solo se il canale di comunicazione è di capacità illimitata
- B - il canale viene impostato a capacità zero
- C - il ricevente si blocca nell' attesa di un messaggio
- D - il ricevente riceve un messaggio valido oppure nullo
- E - il ricevente riceve messaggi di grandezza assegnata

**77. Nello scheduling della CPU, si supponga che al momento zero si presenti la serie ordinata di processi P1, P2, P3, P4, con rispettive durate di sequenza 23, 8, 11, 2 unità di tempo. Se essi vengono serviti mediante FCFS qual'è il tempo medio di attesa per ogni processo?**

- A - 11
- B - 16
- C - 20
- D - 14
- E - 24

**78. A cosa serve un Orologio Virtuale?**

- A - A garantire l'allineamento degli orologi fisici presenti nel sistema.
- B - Per la corretta gestione dei processi che richiedono memoria più di quella disponibile
- C - Per gestire un numero maggiore di temporizzatori rispetto quelli fisici
- D - Per la corretta gestione dei processi virtuali
- E - Per ottimizzare i tempi medi di esecuzione dei processi

**79. Cosa generalmente accade quando si verifica una Interruzione o Eccezione ?**

- A - Si pone a 0 il bit di modo
- B - Si pone a 0 il bit di modo solo se si tratta di un evento di eccezione
- C - Si passa dal modo di Sistema al modo di Utente
- D - Si pone ad 1 il bit di modo
- E - Si pone a -1 il bit di modo

**80. Qual'è la caratteristica principale dell'algoritmo Round Robin**

- A - E' simile all'FCFS ma con capacità di swap
- B - E' simile all'FCFS ma con capacità di prelazione
- C - E' simile all'FCFS ma si considera l'ordine inverso della sequenza FCFS
- D - E' simile all'SJF ma senza capacità di prelazione
- E - Rappresenta il caso generale degli algoritmi di scheduling

**81. Si supponga di utilizzare l'algoritmo FCFS per lo Scheduling del Disco, e che la testina sia inizialmente al cilindro N.100. Se occorre accedere ad una sequenza ordinata di blocchi dislocati rispettivamente nei cilindri 45,22,110 allora quale sarà la distanza totale (misurata in cilindri) percorsa dalla testina?**

- A - 155
- B - 177
- C - 144
- D - 166
- E - 188

**82. Si supponga che nello scheduling della CPU, al momento zero, si presenti la serie ordinata di processi P1, P2, P3, P4, con rispettive durate di sequenza 6,8,7,3 unità di tempo. Se essi vengono serviti mediante SJF senza prelazione qual'è il tempo medio di attesa per ogni processo?**

- A - 9,5
- B - 10
- C - 9
- D - 8
- E - 7

**83. Quali sono gli elementi di base che il Sistema Operativo deve conoscere per realizzare correttamente un montaggio di File System?**

- A - Il file-system da montare, il numero di dischi già montati, velocità della CPU
- B - La memoria disponibile, il dispositivo, il file-system da montare
- C - Il punto di montaggio, il dispositivo, il file-system da montare
- D - Il numero di partizioni presenti sia nel file system ospite sia nel file system ospitante
- E - La memoria disponibile, il file-system da montare, elementi nel filesystem da montare

**84. Cosa caratterizza la chiamata di sistema Exec dopo una Fork?**

- A - Esecuzione concorrente del processo padre e del processo figlio
- B - Esecuzione parallela del processo padre e del processo figlio
- C - Sostituzione del Pid del processo generato
- D - Sostituzione dello spazio di memoria del processo con un nuovo programma
- E - Esecuzione seriale del processo padre e poi del processo figlio

**85. Lo schema di Ricezione Asincrona è caratterizzato dal fatto che:**

- A - Il ricevente riceve un messaggio solo se il canale di comunicazioni è di capacità illimitata
- B - Il canale viene impostato con capacità Zero
- C - Il ricevente si blocca nell'attesa dell'arrivo di un messaggio
- D - Il ricevente riceve un messaggio valido oppure nullo
- E - Il ricevente riceve messaggi di grandezza assegnata

**86. Quale, tra le seguenti, costituisce una differenza caratteristica tra un Sistema Multiprogrammato (1) a Lotti e (2) a Partizione del Tempo?**

- A - (2) è realizzato mediante memoria virtuale e (1) no
- B - (1) è applicato a sistemi monoprocessoressi e (2) no
- C - (1) è interattivo e (2) no
- D - (1) è realizzato mediante memoria virtuale e (2) no
- E - (2) è interattivo e (1) no

**87. La situazione in cui più processi accedono e modificano gli stessi dati in modo concorrente ed i risultati dipendono dall'ordine degli accessi viene definita:**

- A - ...
- B - Interleaved
- C - Race Condition
- D - Sezione Critica
- E - Dead Lock

**88. Generalmente, qual'è il formato della struttura relativa al File System dell'Area di Avvicendamento (Swapping)**

- A - Fat32 oppure NTFS
- B - EXT2 Oppure NTFS
- C - FAT oppure FAT16 Oppure FAT32
- D - Nessuna
- E - NTFS Oppure RAISER

**89. In genere, in quale delle seguenti circostanze si hanno migliori prestazioni preferendo i “Processi con prevalenza di I/O” rispetto i “processi con prevalenza di Elaborazione” ?**

- A - Quando esistono molti dispositivi di I/O
- B - In generale è preferibile quando la memoria centrale è superiore allo spazio disponibile sui dischi
- C - Quando si verificano troppo frequentemente gli eventi di swap su disco
- D - In generale è preferibile una combinazione dei due tipi menzionati.
- E - Quando i dispositivi fisici sono prevalentemente di tipologia di accesso asincrono

**90. In cosa consiste la Tecnica del Memory Mapped I/O ?**

- A - Si fanno corrispondere registri di dispositivi a intervalli dello spazio di indirizzi della CPU
- B - Si tratta del trasferimento dati full-duplex verso i dispositivi di I/O
- C - Si associa la memoria cache della CPU con dispositivi asincroni di I/O
- D - Si mappa la memoria in modo da far corrispondere segmenti di “parte alta” con segmenti di “parte bassa”
- E - Si mappa la Memoria centrale in modo da compattare lo spazio libero per predisporlo per l’ I/O

**91. Una differenza rilevante tra i Thread a livello utente ed i Thread a livello nucleo è:**

- A - I thread del nucleo sono generalmente più veloci dei thread a livello utente
- B - I thread a livello utente sono generalmente più veloci dei thread a livello nucleo
- C - I thread a livello utente sono sempre meno compatti dei thread a livello nucleo
- D - I thread a livello nucleo sono interpretati mentre i thread a livello utente sono compilati
- E - I thread a livello nucleo sono compilati mentre i thread a livello utente sono interpretati

**92. Quale delle seguenti Gerarchie di Memoria (dei tempi di accesso) è quella esatta?**

- A - Registri, Memoria Centrale, Cache, Disco Ram, Dischi magnetici, Dischi ottici, Nastri magnetici
- B - Registri, Cache, Dischi magnetici, Disco Ram, Memoria centrale, Dischi ottici, Nastri magnetici
- C - Cache, Registri, Memoria centrale, Disco Ram, Dischi magnetici, Dischi ottici, Nastri magnetici
- D - Registri, Cache, Memoria centrale, Disco Ram, Dischi magnetici, Dischi ottici, Nastri magnetici
- E - Registri, Cache, Memoria centrale, Disco Ram, Dischi magnetici, Nastri magnetici, Dischi ottici

**93. Si supponga di utilizzare l'algoritmo SSTF per lo scheduling del disco D e che D abbia 50 cilindri e che la testina all'istante T0 sia posizionata al cilindro N.25. Se la sequenza delle richieste dei cilindri è 1,2,4,19,20,32,44 allora quante richieste saranno state soddisfatte quando la testina avrà già percorso una distanza di 28 cilindri ? (SSTF = Shortest seek first, conosciuto anche come Shortest Seek / Service Time First;)**

- A - 3
- B - 2
- C - 5
- D - 6
- E - 4

**94. Da cosa è caratterizzato un Sistema di Elaborazione Fault-Tolerant?**

- A - Capacità di continuare il servizio anche senza l'ausilio dei gruppi di continuità
- B - Capacità di continuare il servizio in proporzione ai dispositivi correttamente funzionanti
- C - Capacità di ignorare interrupt di sistema se viene richiesta grande quantità di elaborazione
- D - Capacità di segnalare un errore di elaborazione
- E - Capacità di scambiare rapidamente i dati tra registri della CPU ed unità di Backup

**95. Che cosa rappresenta il meccanismo dell'RPC?**

- A - Garantisce la funzionalità dell'Ftp tra un Client ed un Server della rete
- B - Una astrazione della chiamata di procedura impiegata tra sistemi collegati in rete
- C - Definisce l'interazione di un Client con un Web Server
- D - Costituisce una generalizzazione di un Socket
- E - Ottimizza il tempo di scambio dati tra due PC della rete

**96. Qual'è la differenza tra "I/O Programmato (PIO)" ed "I/O guidato dalle Interruzioni" ?**

- A - Nell' "I/O Programmato" le operazioni di I/O sono effettuate solo al termine di ogni processo mentre nell' "I/O guidato da interruzioni" l' "I/O viene eseguito al termine di tutte le interruzioni.
- B - Nell' "I/O Programmato" la CPU verifica mediante polling la disponibilità del dispositivo; nell' "I/O guidato da interruzioni" si attende mediante l'interruzione la disponibilità di un dispositivo.
- C - Si usa l' "I/O guidato da interruzioni" solo per i processi mentre si usa l' "I/O Programmato" per i threads
- D - Nell' "I/O Programmato" la CPU verifica se le interruzioni provengono da un dispositivo fisico, nell' "I/O guidato da interruzioni", si verifica se occorrono interruzioni a qualche dispositivo.
- E - Nell' "I/O Programmato" la CPU richiede la disponibilità di un dispositivo mentre nell' "I/O guidato da interruzioni" la CPU interrompe il dispositivo.

**97. Qual'è un obiettivo principale della Multiprogrammazione ?**

- A - Ottimizzare il numero di files che possono essere memorizzati nell'Hard Disk
- B - Aumentare l'eterogeneità dei processi caricati in memoria centrale
- C - Ottimizzare il numero di periferiche collegate al sistema mediante i canali SCSI, SATA, EIDE
- D - Consentire di aumentare l'utilizzo della CPU, organizzando i lavori in modo da mantenerla in continua attività
- E - Garantire che il numero di Threads non superi il numero dei processi correttamente nella memoria centrale.

**98. In un File-System Unix-Like, cosa si intende per LINK**

- A - Un collegamento ad una specifica pagina web
- B - Un puntatore ad un file eseguibile
- C - Un puntatore ad un file di sistema
- D - Un puntatore ad un altro file o directory
- E - Un collegamento tra l'i-node ed il boot-block

**99. Qual'è la principale caratteristica di un'istruzione TestAndSet?**

- A - Esecuzione atomica anche in presenza di più unità di elaborazione
- B - Esecuzione atomica per processi a prevalenza di elaborazione
- C - Esecuzione parallela su diverse unità di elaborazione
- D - Attesa attiva su più unità di elaborazione
- E - Attesa passiva su più unità di elaborazione

**100. Qual'è la differenza fondamentale tra comunicazione Diretta ed Indiretta dei processi**

- A - Nella indiretta i messaggi vengono senza ...
- B - Nella indiretta i messaggi si inviano ai processi ...

**101. Generalmente, qual è il formato della struttura relativa al File-System dell'Area di Avvicendamento (Swapping)?**

- A- Nessuna

**102. Qual'è la caratteristica fondamentale del codice ECC ?**

- A - Possedere informazioni per la correzione degli errori generati dall'algoritmo di deframmentazione
- B - Possedere informazioni per la verifica di un errore
- C - Possedere informazioni per la verifica della grandezza dei files (espresso in bytes)
- D - Possedere informazioni per la verifica della posizione (traccia e settore) di memorizzazione
- E - Possedere informazioni per la correzione di un errore

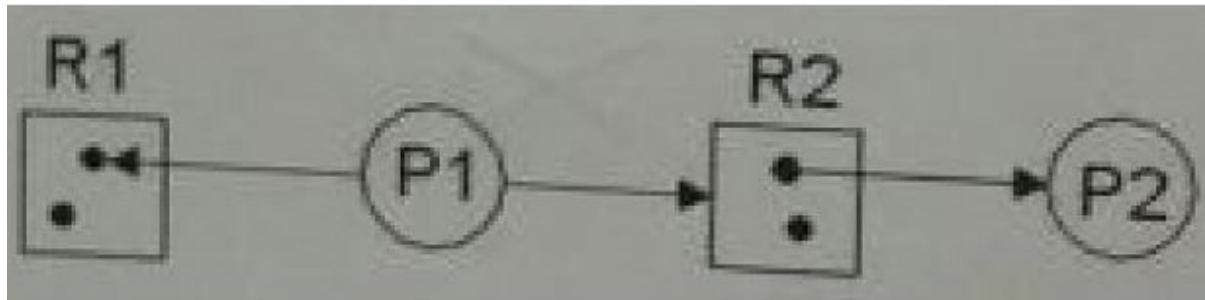
**103. Sia assegnato un disco A con blocchi da 512 byte e un disco B da 1024 byte per blocco. Si supponga che si debba memorizzare un file F da 1949 byte: quale dei due dischi A e B genererà maggiore frammentazione interna?**

- A - Il disco A

**104. Un Sistema Distribuito può essere definito come un insieme d'unità d'elaborazione che:**

- A - Non condividono la memoria o il clock
- B - Condividono il clock ma non la memoria
- C - Condividono la memoria di massa
- D - Condividono la memoria e non il di disco
- E - Condividono memoria paginata.

**105. Cosa caratterizza fortemente il grafo di assegnazione delle risorse?**



- A - Errore di richiesta
- B - Insufficienza di risorse
- C - Compatibilità di P1 e P2
- D - Abbondanza di risorse
- E - Errore di Assegnazione

**106. Da cosa sono caratterizzati i Driver dei Dispositivi?**

- A - Gestiscono il controllo degli I/O mediante segnali di interruzione
- B - Gestiscono il controllo degli I/O tra i dispositivi e al CPU
- C - Gestiscono il controller mediante meccanismi di condivisione
- D - Gestiscono il controllo degli I/O tra dispositivi di memoria secondaria
- E - Gestiscono il controllo degli I/O mediante segnali di exception

**107. Perché la tecnica Copy-on-Write è molto usata dai processi legati alle Fork?**

- A - Processi padri e figli condividono molti dati.
- B - I tempi di accesso alla memoria centrale sono più rapidi degli accessi al disco
- C - La tecnica copy-on-write NON è largamente utilizzata nei processi fork
- D - Il processo di copia seguito da una fork + un processo maggiore priorità
- E - Perché così migliora la corrispettiva tecnica speculare di Copy-on-read

**108. Si supponga di utilizzare l'algoritmo C-SCAN per lo scheduling del disco D di 150 cilindri, testina al cilindro 66 e direzione del braccetto verso numerazione crescente dei cilindri. Se l'insieme delle richiede dei cilindri è 1,2,4,19,41,79,138,44 allora quante richieste dovranno ancora essere soddisfatte quando la testina avrà già percorso una distanza di 275 cilindri**

- A - 1
- B - 4
- C - 3
- D - 0
- E - 2

**109. In un S.O. Unix-like, quali sono i valori di ritorno di una System Call fork()?**

- A - pid >0 nel figlio e Zero nel padre
- B - Pid1>0 nel padre e pid2>0 nel figlio
- C - Zero nel figlio e pid>0 nel padre
- D - pid padre = pidfiglio+1
- E - pid=0 nel padre e Zero nel figlio

**110. In che modo un utente può eseguire istruzioni privilegiate?**

- A - Aumentando la priorità del processo durante la sua esecuzione
- B - In nessun caso
- C - Mediante la chiamata di una System Call
- D - Aumentandola priorità del processi prima che esso venga eseguito
- E - Utilizzando le risorse critiche del sistema

**111. In che modo alcune architetture di calcolatori ottimizzano il tempo di cambio contesto dei processi?**

- A - Limitando il numero di processi in esecuzione
- B - Limitando il numero delle fork dei processi
- C - Aumentando la frequenza di lavoro della CPU
- D - Mettendo a disposizione maggiore memoria RAM
- E - Mettendo a disposizione più gruppi di registri

**112. Nella gestione della memoria si consideri la tecnica di swapping tra i processi P1 e P2. Si supponga che, al completamento di P2, si debba di nuovo ricaricare in RAM il processo P1. Quale spazio occuperà P1?**

- A - Dipende dalle grandezze di P1 e P2
- B - Sicuramente lo stesso spazio che P1 occupava in precedenza
- C - Lo stesso spazio che P2 occupava in precedenza
- D - Dipende dal tipo di associazione indirizzi (logici/fisici) adottato
- E - Dipende dalla quantità di overlay necessary

## CORRETTORE

- 1 D - Lo scheduler a lungo termine
- 2 B - (inserisci +1) % DIM\_VETTORE == preleva
- 3 C - Controllando il bit di validità nella tabella delle pagine
- 4 A - Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica (Vedi libro pag. 217)
- 5 B - Mutua Esclusione; Progresso, Attesa limitata
- 6 E - Se si verificano contemporaneamente le condizioni di:  
Mutua esclusione; Possesso e attesa; Impossibilità di prelazione; Attesa circolare.
- 7 C - C; B; D
- 8 E - Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può aumentare con l'aumentare del numero dei blocchi di memoria assegnati.
- 9 C - Richiede la conoscenza futura della successione dei riferimenti
- 10 E - do { ...; produce un elemento in appena\_prodotto ... ; wait(vuote); wait(mutex); ... ; inserisci appena\_prodotto in vettore; ... signal(mutex); signal(piene); } while (1);
- 11 C - E' stata ritardata l'associazione finale degli indirizzi alla fase di caricamento
- 12 C - 1) Priorità dei processi. 2) Tempo già trascorso e tempo ancora necessario per ogni processo.  
3) Quantità e tipo di risorse impiegate.
- 13 A - Può condividere direttamente uno spazio logico di indirizzi
- 14 D - Terminazione di uno o più processi per interrompere l'attesa circolare; prelazione su alcune risorse in possesso di uno o più processi in stallo
- 15 D - Code con capacità limitata e illimitata
- 16 C - Il processo padre stampa HELLO; il processo figlio esegue un comando di "ls"
- 17 B - Tipo di dispositivo, indirizzo, stato
- 18 D - Registri, Blocchi, Pile
- 19 C - quando un processo dallo stato di esecuzione o di attesa passa allo stato di pronto
- 20 ---
- 21 A - Quando è rilevante il fattore di prevedibilità
- 22 B - 10 (P2 finisce a 10, P1 finisce ad 11, P3 finisce al 5)
- 23 E - Un ciclo nel grafo di assegnazione delle risorse implica necessariamente uno stallo
- 24 A - Controllo processi; Gestione file; Gestione dei dispositivi; Gestione delle informazioni; Comunicazioni
- 25 C - Si serve per prima il processo con il nome "minore".
- 26 D - I thread a livello utente sono generalmente più veloci dei thread a livello nucleo
- 27 C - Memoria Centrale
- 28 D - E' un'interfaccia tra utente e Sistema Operativo
- 29 D - Definisce la validità dei valori che sono memorizzati nelle pagine indicate.
- 30 D - Si leggono e si mettono nella cache la pagina richiesta e parecchie pagine successive
- 31 A - Terminazione forzata di processi e/o arresto del sistema
- 32 A - File aperti per lunghi periodi e modificati spesso
- 33 D - R.R. per processi in primo piano e FCFS per processi in background
- 34 E - Mutua esclusione, Attesa limitata
- 35 A - E2 = E1-E3 (vedere diagramma della domanda 42 per capire)
- 36 E - Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica.
- 37 B - \*Mediante dei link (rispondo per esclusione, perché le altre non hanno senso)
- 38 A - L'impiego di liste concatenate
- 39 E - La Prima figura a sinistra
- 40 D - 81
- 41 E - Instradamento Fisso e Virtuale
- 42 C - C = "Memoria"; B="Registro di Rilocazione"; A="Registro di Limite". [ Corretta ]
- 43 D - Il cliente con il numero progressivo più basso e, a parità, si serve quello con nome minore

- 44 /\* NON SO \*/ ((indirizzi-log2pagina)/2)
- 45 E - Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica.
- 46 B - \*Mediante dei link
- 47 B - Sostituzione Ottimale ed LRU
- 48 D - Per ogni Pi le richieste che Pi può ancora fare si possono soddisfare impiegando le risorse possedute da tutti i Pj con  $j < i$  (vedi capitolo 7, pag. 280 del libro)
- 49 A - Quando è rilevante il fattore di prevedibilità
- 50 B - 10
- 51 B - 2
- 52 E - Ciclo senza stallo
- 53 E - Ricerca contemporanea (di un valore) tra tutte le chiavi
- 54 D - 1,2,3
- 55 A - mai
- 56 C - 7 ( per calcolare il tempo di attesa medio si sommano i vari tempi d'attesa e si dividono per il numero di processi quindi costruendo il grafico abbiamo che P4 attende 0, P1 attende 3, P3 attende 9 e P2 attende 16 =>  $=> (0+3+9+16)/4 = 28/4=7$  )
- 57 E - (inserisci+1)%DIM\_VETTORE == preleva
- 58 C - Memoria Centrale, Registri CPU
- 59 A - 101
- 60 C - I Descrittori dei files
- 61 C - Unire le directory remote alle directory locali
- 62 B - Il numero dei parametri è maggiore del numero dei registri
- 63 D - Mantengono costante la quantità di dati che scorre sotto le testine
- 64 C - Poter stabilire quando il file non è più in uso
- 65 E - Ad ogni pagina associa l'istante in cui è stata usata per l'ultima volta
- 66 B - 9
- 67 D - Instradamento Fisso e Virtuale
- 68 C - A è totalmente controllato dall'utente; B è sotto il controllo del Sistema Operativo
- 69 C - Tattesa = TempoFineProcesso - TempoArrivoProcesso - DurataProcesso
- 70 A - Un processo può richiedere risorse solo se non ne possiede.
- 71 D - In (1) un calcolatore resta in attesa attiva
- 72 D - se il numero di wait() è maggiore al numero di Signal()
- 73 A - Controllo processi; gestione file; Gestione dei dispositivi; Gestione delle informazioni; Comunicazioni
- 74 B - La disposizione degli elementi della tabella non è organizzata per indirizzi virtuali
- 75 D - 6,5
- 76 D - il ricevente riceve un messaggio valido oppure nullo
- 77 E - 24
- 78 C - Per gestire un numero maggiore di temporizzatori rispetto quelli fisici
- 79 A - Si pone a 0 il bit di modo
- 80 B - E' simile all'FCFS ma con capacità di prelazione
- 81 D - 166
- 82 E - 7
- 83 C - Il punto di montaggio, il dispositivo, il file-system da montare
- 84 D - Sostituzione dello spazio di memoria del processo con un nuovo programma
- 85 D - Il ricevente riceve un messaggio valido oppure nullo
- 86 E - (2) è interattivo e (1) no
- 87 C - Race Condition
- 88 D - Nessuna
- 89 D - In generale è preferibile una combinazione dei due tipi menzionati.
- 90 A - Si fanno corrispondere registri di dispositivi a intervalli dello spazio di indirizzi della CPU
- 91 B - I thread a livello utente sono generalmente più veloci dei thread a livello nucleo

92 D - Registri, Cache, Memoria centrale, Disco Ram, Dischi magnetici, Dischi ottici, Nastri magnetici  
93 A - 3

94 B - Capacità di continuare il servizio in proporzione ai dispositivi correttamente funzionanti

95 B - Una astrazione della chiamata di procedura impiegata tra sistemi collegati in rete

96 B - Nell' "I/O Programmato" la CPU verifica mediante polling la disponibilità del dispositivo; nell' "I/O guidato da interruzioni" si attende mediante l'interruzione la disponibilità di un dispositivo.

97 D - Consentire di aumentare l'utilizzo della CPU, organizzando i lavori in modo da mantenerla in continua attività

98 D - Un puntatore ad un altro file o directory

99 A - Esecuzione atomica anche in presenza di più unità di elaborazione

100

101 A - Nessuna

102 E - Possedere informazioni per la correzione di un errore

103 A - Il disco A (1-Dim\_blocchi (F%DBlocco)) / Dblocco)). Si sceglie il blocco che da valore più alto.

104 A - Non condividono la memoria o il clock

105 A - Errore di richiesta

106 A - Gestiscono il controllo degli I/O mediante segnali di interruzione

107 A - Processi padri e figli condividono molti dati

108 A - 1

109 C - Zero nel figlio e pid>0 nel padre.

110 C - Mediante la chiamata di una System Call

111 C - Aumentando la frequenza di lavoro della CPU

112 D - Dipende dal tipo di associazione indirizzi (logici/fisici) adottato

## CORRETTORE CON SPIEGAZIONI E/O RIFERIMENTI AL LIBRO

1 D - Lo scheduler a lungo termine (Vedi Pag 104 Capitolo 3 ed8)

(Lo scheduler a lungo termine si esegue con una frequenza molto inferiore, lo scheduler a lungo termine controlla il grado di multiprogrammazione cioè il numero di processi in memoria. Quello a breve termine seleziona, tra quelli pronti per l'esecuzione. ).

2 B - (inserisci +1) % DIM\_VETTORE == preleva

3 C - Controllando il bit di validità nella tabella delle pagine

(chi ha fatto ADE, lo sa già')

4 A - Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica

(Vedi libro pag. 217 Capitolo 6 ed8)

(quando un processo è in esecuzione nella propria sezione critica, non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica)

5 B - Mutua Esclusione; Progresso, Attesa limitata

(Vedi libro pag. 217 Capitolo 6 ed8)

6 E - Se si verificano contemporaneamente le condizioni di:

Mutua esclusione; Possesso e attesa; Impossibilità di prelazione; Attesa circolare.

(Pagina 271, capitolo 7 ed8)

Il Dead Lock: Condizioni necessarie se si verificano

Muta Esclusione: Non possono essere seguiti processi se un processo è nella sua sezione critica

Possesso e Attesa: Un processo in possesso di almeno una risorsa attende di acquisirne un'altra già in possesso di altri processi

Impossibilità di prelazione: non esiste un diritto di prelazione sulle risorse. Una risorsa può essere rilasciata dal processo che la possiede solo in attesa di aver terminato il proprio compito.

Attesa Circolare: Deve esistere un insieme di processi | P0 attende risorsa da P1, P1 attende da P2, ... , Pn attende da P0

)

7 C - C; B; D (mi rifiuto di dare una spiegazione a questa boiata!!!)

8 E - Con alcuni algoritmi di sostituzione delle pagine la frequenza di assenza delle pagine può aumentare con l'aumentare del numero dei blocchi di memoria assegnati.

(Anomalia di belady riguarda lo scheduling FIFO, in base a questa anomalia, più blocchi di memoria si possiede maggiore è l'assenza delle pagine. OPT e LRU ne sono immuni)

9 C - Richiede la conoscenza futura della successione dei riferimenti

(OPT rimuove dalla lista dei blocchi di memoria quello che userò più lontano nel futuro, quindi è ovvio che debba conoscere prossimamente)

10 E - do { ...; produce un elemento in appena\_prodotto ... ; wait(vuote); wait(mutex); ... ; inserisci appena\_prodotto in vettore; ... signal(mutex); signal(piene); } while (1);

11 C - E' stata ritardata l'associazione finale degli indirizzi alla fase di caricamento

Associazione di istruzioni e dati alla memoria

Compilazione: se la locazione di memoria è conosciuta a priori possono essere generati indirizzi assoluti.

La ricompilazione è necessaria quando la locazione di partenza cambia.

Caricamento: se la locazione di memoria non è conosciuta a priori si genera codice rilocabile (al variare dell'indirizzo iniziale).

Esecuzione: se il processo può essere spostato, l'associazione viene ritardata al momento dell'esecuzione.

Necessario hardware specializzato (es: registri base e limite).

12 C - 1) Priorità dei processi. 2) Tempo già trascorso e tempo ancora necessario per ogni processo.

3) Quantità e tipo di risorse impiegate

Leggere anche (è spiegato chiaramente) <http://www2.units.it/mumolo/stallo.pdf>

Una volta che lo stallo è rilevato:

- La selezione dei processi vittima secondo i seguenti Criteri di selezione:

Priorità, Tempo di calcolo, Tipo di risorse occupate;

terminazione dei processi: totale o incrementale

(capitolo 7 pag 290 ed8)

Occorre stabilire quali risorse e quali processi si devono sottoporre a prelazione.

Come per la terminazione è necessario stabilire ordine di prelazione per ridurre i costi.

-Numero di risorse occupate.

-Per quanto tempo il processo ha elaborato e per quanto tempo

ancora il processo proseguirà prima di completare l'operazione pianificata

13 A - Può condividere direttamente uno spazio logico di indirizzi

14 D - Terminazione di uno o più processi per interrompere l'attesa circolare; prelazione su alcune risorse in possesso di uno o più processi in stallo

15 D - Code con capacità limitata e illimitata

(Capitolo 3 pag. 117 ed8)

Se la comunicazione è diretta o indiretta i messaggi scambiati tra i processi risiedono in code temporanee. Esistono 3 modi per realizzare queste code:

-capacità 0 (sistema a scambio di messaggio senza memorizzazione transitoria)

-capacità limitata (automatic buffering)

-capacità illimitata (automatic buffering)

16 C - Il processo padre stampa HELLO; il processo figlio esegue un comando di "ls"

17 B - Tipo di dispositivo, indirizzo, stato

(Pag 548, capitolo 13)

Gli elementi della tabella indicano, il tipo, l'indirizzo e lo stato del dispositivo.

I kernel che mettono a disposizione I/O sincrono asincrono gestiscono più richieste I/O contemporaneamente a questo fine alcuni sistemi annessono una tabella dello stato dei dispositivi alla coda dei processi in attesa.

18 D - Registri, Blocchi, Pile

19 C - quando un processo dallo stato di esecuzione o di attesa passa allo stato di pronto

(pagina 175, capitolo 5)

20 ---

21 A - Quando è rilevante il fattore di prevedibilità (Rif pag 179)

22 B - 10 (P2 finisce a 10, P1 finisce ad 11, P3 finisce al 5)

23 E - Un ciclo nel grafo di assegnazione delle risorse implica necessariamente uno stallo

24 A - Controllo processi; Gestione file; Gestione dei dispositivi; Gestione delle informazioni; Comunicazioni

25 C - Si serve per prima il processo con il nome "minore".

26 D - I thread a livello utente sono generalmente più veloci dei thread a livello nucleo

(The kernel-level threads are slow and inefficient. For instance, threads operations are hundreds of times slower than that of user-level threads).

27 C - Memoria Centrale

28 D - E' un'interfaccia tra utente e Sistema Operativo

29 D - Definisce la validità dei valori che sono memorizzati nelle pagine indicate.

30 D - Si leggono e si mettono nella cache la pagina richiesta e parecchie pagine successive

31 A - Terminazione forzata di processi e/o arresto del sistema

32 A - File aperti per lunghi periodi e modificati spesso

33 D - R.R. per processi in primo piano e FCFS per processi in background

34 E - Mutua esclusione, Attesa limitata

35 A - E2 = E1-E3 (vedere diagramma della domanda 42 per capire)

36 E - Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica.

37 B - \*Mediante dei link

C - Mediante directory a singolo livello (vedi capitolo 17 del libro e dammi conferma)

38 A - L'impiego di liste concatenate

39 E - La Prima figura a sinistra

A - La terza figura (all'aumentare dei blocchi di memoria aumenta il numero delle assenze delle pagine)

40 D - 81

41 E - Instradamento Fisso e Virtuale

(Pag 661, Cap 16)

I tre schemi di instradamento più diffusi sono fisso, virtuale e dinamico.

Fisso o Virtuale, assicura che i messaggi da A->B arrivino nell'ordine in cui sono stati trasmessi  
Con Instademetno dinamico, possono arrivare disordinatamente.

42 C - C = "Memoria"; B="Registro di Rilocazione"; A="Registro di Limite". [ Corretta ]

43 D - Il cliente con il numero progressivo più basso e, a parità, si serve quello con nome minore

44 /\* NON SO \*/ ((indirizzi-log2pagina)/2)

45 E - Non si deve consentire a nessun altro processo di essere in esecuzione nella propria sezione critica.

46 B - \*Mediante dei link

47 B - Sostituzione Ottimale ed LRU

48 D - Per ogni Pi le richieste che Pi può ancora fare si possono soddisfare impiegando le risorse possedute da tutti i Pj con j<=i (vedi capitolo 7, pag. 280 del libro)

(uno stato si dice sicuro se il sistema è in grado di assegnare risorse a ciascun processo in un certo ordine impedendo il verificarsi di uno stallo.

In altre parole uno stato si dice sicuro se esiste una SEQUENZA SICURA:

Una Sequenza di sicura P1, P2, Pn è SICURA per lo stato di assegnazione attuale

PER OGNI Pi, le richieste che Pi può ancora fare si possono soddisfare impiegando le risorse attualmente disponibili + le risorse possedute da tutti i Pj con j<=i).

49 A - Quando è rilevante il fattore di prevedibilità

50 B - 10 /\* DA CALCOLARE \*/

51 B - 2

52 E - Ciclo senza stallo

53 E - Ricerca contemporanea (di un valore) tra tutte le chiavi

54 D - 1,2,3

55 A - mai

56 C - 7 ( per calcolare il tempo di attesa medio si sommano i vari tempi d'attesa e si dividono per il numero di processi quindi costruendo il grafico abbiamo che P4 attende 0, P1 attende 3, P3 attende 9 e P2 attende 16 => => (0+3+9+16)/4 = 28/4=7 )

57 E - (inserisci+1)%DIM\_VETTORE == preleva

58 C - Memoria Centrale, Registri CPU

59 A - 101

60 C - I Descrittori dei files Un inode è un record nella tabella di un hard disk, che contiene informazioni sul file o sulla directory tra cui dimensioni, proprietari, socket, pipe, device node e così via. Tutte informazioni, o per meglio dire meta-informationi, il cui numero corrisponde con il numero di file e directory contenuti nell'hard disk del sistema). Aggiungo, che quando in C usiamo la Open a basso livello, iniziamo il FileDescriptor

```
if ((fd = open(filename, O_WRONLY | O_CREAT | O_TRUNC,
S_IRUSR | S_IWUSR | S_IRGRP | S_IROTH)) == -1)
```

61 C - Unire le directory remote alle directory locali

62 B - Il numero dei parametri è maggiore del numero dei registri

63 D - Mantengono costante la quantità di dati che scorre sotto le testine

64 C - Poter stabilire quando il file non è più in uso

65 E - Ad ogni pagina associa l'istante in cui è stata usata per l'ultima volta

66 B - 9

67 D - Instradamento Fisso e Virtuale

68 C - A è totalmente controllato dall'utente; B è sotto il controllo del Sistema Operativo

69 C - Tattesa = TempoFineProcesso - TempoArrivoProcesso - DurataProcesso

70 A - Un processo può richiedere risorse solo se non ne possiede.

71 D - In (1) un calcolatore resta in attesa attiva

72 D - se il numero di wait() è maggiore al numero di Signal()

73 A - Controllo processi; gestione file; Gestione dei dispositivi; Gestione delle informazioni; Comunicazioni

74 B - La disposizione degli elementi della tabella non è organizzata per indirizzi virtuali

(sorge solo un dubbio... leggendo questo. Credo però che la risposta A sia ambigua)

Usando questo schema (tabella delle pagine inverita)

-la tabella delle pagine contiene una entry per ogni pagina reale in memoria

**-ogni entry consiste dell'indirizzo virtuale (esclude risposta B)** della pagina a dell'identificatore del processo che possiede quella pagina

**-diminuisce la memoria necessaria** per memorizzare la tabella delle pagine, MA **aumenta il tempo per cercare la tabella quando viene fatto un riferimento ad una pagina**

-Si può usare una tabella di hash per limitare la ricerca a poche entry della tabella delle pagine

75 D - 6,5

76 D - Il ricevente riceve un messaggio valido oppure nullo

77 E - 24

78 C - Per gestire un numero maggiore di temporizzatori rispetto quelli fisici

79 A - Si pone a 0 il bit di modo

80 B - È simile all'FCFS ma con capacità di prelazione

81 D - 166

82 E - 7

83 C - Il punto di montaggio, il dispositivo, il file-system da montare

84 D - Sostituzione dello spazio di memoria del processo con un nuovo programma

85 D - Il ricevente riceve un messaggio valido oppure nullo

86 E - (2) è interattivo e (1) no

87 C - Race Condition

88 D - Nessuna

89 D - In generale è preferibile una combinazione dei due tipi menzionati.

90 A - Si fanno corrispondere registri di dispositivi a intervalli dello spazio di indirizzi della CPU

91 B - I thread a livello utente sono generalmente più veloci dei thread a livello nucleo.

(The kernel-level threads are slow and inefficient. For instance, threads operations are hundreds of times slower than that of user-level threads).

92 D - Registri, Cache, Memoria centrale, Disco Ram, Dischi magnetici, Dischi ottici, Nastri magnetici

93 A - 3

94 B - Capacità di continuare il servizio in proporzione ai dispositivi correttamente funzionanti

95 B - Una astrazione della chiamata di procedura impiegata tra sistemi collegati in rete

96 B - Nell' "I/O Programmato" la CPU verifica mediante polling la disponibilità del dispositivo; nell' "I/O guidato da interruzioni" si attende mediante l'interruzione la disponibilità di un dispositivo.

97 D - Consentire di aumentare l'utilizzo della CPU, organizzando i lavori in modo da mantenerla in continua attività

98 D - Un puntatore ad un altro file o directory

99 A - Esecuzione atomica anche in presenza di più unità di elaborazione

100

101 A - Nessuna

102 E - Possedere informazioni per la correzione di un errore

103 A - Il disco A (1-Dim\_blocco (F%DBlocco)) / Dblocco). Si sceglie il blocco che ha valore più alto.

104 A - Non condividono la memoria o il clock

105 A - Errore di richiesta

106 A - Gestiscono il controllo degli I/O mediante segnali di interruzione

107 A - Processi padri e figli condividono molti dati.

108 A - 1 (66->149 + 149->0 + 0->41 si arriva a 273 Cilindri. Al 275 sarà a metà strada tra il 41 e il 44 che dovrà ancora fare)

109 C - Zero nel figlio e pid>0 nel padre

110 C - Mediante la chiamata di una System Call

111 C - Aumentando la frequenza di lavoro della CPU

112 D - Dipende dal tipo di associazione indirizzi (logici/fisici) adottato

Nome e cognome \_\_\_\_\_ Matricola \_\_\_\_\_

La durata della prova è di 2 ore. Scrivere in stampatello maiuscolo. È obbligatorio restituire il testo e tutti i fogli forniti, anche se non si consegna il compito.

1. Considerati CPU-burst time (in ms) del set di processi descritto in tabella, e considerato che l'ordine di arrivo dei processi è P1, P2, P3, P4, P5, tutti all'istante 0.

| Process | Burst Time |
|---------|------------|
| P1      | 10         |
| P2      | 1          |
| P3      | 2          |
| P4      | 1          |
| P5      | 5          |

a) Disegnare 2 diagrammi di Gantt che illustrano l'esecuzione di questi processi usando gli algoritmi di scheduling SJF e RR (quanto = 3 ms). Indicare l'istante di tempo di ciascuna commutazione fra un processo e l'altro.

b) Calcolare il tempo di attesa per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo d'attesa e come l'avete applicato per calcolare i risultati esposti)

|    | SJF | RR |
|----|-----|----|
| P1 |     |    |
| P2 |     |    |
| P3 |     |    |
| P4 |     |    |
| P5 |     |    |

c) Calcolare il tempo di turnaround per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo di turnaround e come l'avete applicato per calcolare i risultati esposti)

|    | SJF | RR |
|----|-----|----|
| P1 |     |    |
| P2 |     |    |
| P3 |     |    |
| P4 |     |    |
| P5 |     |    |

**2. Tenuto conto che:**

- il pid del processo padre è **10** e quello del figlio è **11**.

Cosa stamperà il seguente programma? Giustificare in estrema sintesi la risposta

```
int main()
{
 pid_t pid;
 int status, var = 5;
 printf("Sono il processo padre.\n");

 pid = fork();

 if(pid > 0)
 {
 var++;
 printf("var = %d\n", var);
 wait(&status);
 }

 printf("var = %d\n", var);
 printf("pid = %d\n", getpid());
 return 0;
}
```

**3. Dato il frammento di programma che segue:**

- a) Quanti processi e quanti thread genera il frammento di programma seguente?
- b) Evidenziare, se ci sono, le corse critiche. Come si possono evitare?
- c) Cosa stamperà il programma?

Giustificare in estrema sintesi le risposte

```
#define NUM_THREADS 4

int shared = 0;

void *func(void* param)
{
 for (int i = 0; i < 100; ++i) {
 shared += 1;
 }
 return 0;
}

int main()
{
 pthread_t threads[NUM_THREADS];

 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_create(&threads[i], NULL, func, NULL);
 }

 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_join(threads[i], NULL);
 }
 printf("%d\n", shared);

 return 0;
}
```

**4. Nel contesto dei metodi per evitare i deadlock cosa si intende per "stato sicuro"?**

- a) Dati tre processi P<sub>0</sub>, P<sub>1</sub>, P<sub>2</sub> e tre tipi di risorsa A (5 istanze), B (4 istanze), C (2 istanze) con le seguenti matrici di allocazione, max e disponibile, verificare se il sistema è in uno stato sicuro

| Allocazione    |   |   | Max |   |   | Disponibile |   |   |   |
|----------------|---|---|-----|---|---|-------------|---|---|---|
|                | A | B | C   | A | B | C           | A | B | C |
| P <sub>0</sub> | 2 | 0 | 1   | 3 | 2 | 1           | 1 | 2 | 0 |
| P <sub>1</sub> | 1 | 1 | 1   | 4 | 1 | 1           |   |   |   |
| P <sub>2</sub> | 1 | 1 | 0   | 5 | 3 | 1           |   |   |   |

- b) Se P<sub>2</sub> richiede (0, 1, 0) cosa succede?

**5. Si consideri una memoria paginata, dato un processo con una tabella delle pagine in memoria**

- a) Se ogni accesso in memoria richiede 60 ns qual è il tempo per accedere ad un indirizzo della memoria paginata?  
b) Aggiungendo una TLB con hit ratio del 60% e stimando un tempo di accesso di 3 ns qual è il EAT (Effective Access Time)?  
c) Descrivere succintamente cos'è una TLB e qual è la sua funzione

**6. Si consideri un processo che genera la seguente stringa dei riferimenti alle pagine virtuali:**

1, 2, 0, 2, 1, 0, 1

- a) Se il processo ha 2 frame gestiti con LRU quanti page fault vengono generati  
b) Quanti page fault vengono generati con l'algoritmo OPT

**7. Descrivere in maniera sintetica ma esaustiva il concetto di working set WS all'istante t con intervallo  $\Delta$  WS(t,  $\Delta$ ) e la sua funzione.**

- a) Data la stringa dei riferimenti 1, 3, 3, 7, 1, 1, 9, 10, 7 scrivere il contenuto di WS(9,5)

**8. Assumendo 2000 cilindri e testina su 1100 (con richiesta precedente a 801)**

- a) Data la coda di richieste 100, 50, 1500, 1700, 400 come vengono servite le richieste con l'algoritmo C-LOOK  
b) Qual è la distanza in cilindri percorsi della testina con l'algoritmo C-LOOK

**9. Domande (più di una risposta può essere corretta)**

- a) **Quale delle seguenti affermazioni sul copy\_on\_write è vera**  
a. È una tecnica che per la sincronizzazione tra processi evitando corse critiche  
b. È un metodo che permette al processo figlio di condividere inizialmente le stesse pagine del processo padre  
c. È un metodo per velocizzare la scrittura sul file system  
d. È una tecnica che consente di minimizzare il numero di pagine allocate per un nuovo processo
- b) **Quale tra queste affermazioni su thread e processi è vera**  
a. Due thread possono condividere lo stesso PID e PPID  
b. Un thread può esistere all'interno di più processi  
c. Quando all'interno di un processo viene creato un thread, questo riceve una copia dello spazio di indirizzamento del processo che lo ha creato  
d. Due thread di due processi diversi possono determinare una corsa critica su una variabile globale
- c) **Quale tra queste affermazioni sull'allocazione dei file è vera**  
a. L'allocazione linkata di un file presenta problemi di frammentazione esterna

- b. L'allocazione contigua soffre problemi di frammentazione esterna
  - c. L'allocazione indicizzata presenta problemi di frammentazione esterna ma risolve il problema della frammentazione interna
  - d. L'allocazione indicizzata favorisce l'accesso random al file
- d) **Quale tra queste affermazioni sui meccanismi di sincronizzazione è vera**
- a. Lo spin-lock determina un context switch perché sospende un processo
  - b. Le istruzioni `test_and_set()` e `compare_and_swap()` non sono interrompibili
  - c. Il semaforo binario permette l'accesso esclusivo ad una risorsa condivisa
  - d. Il mutex generalizza il concetto di semaforo contatore

Nome e cognome \_\_\_\_\_

Matricola \_\_\_\_\_

La durata della prova è di 2 ore. Scrivere in stampatello maluscolo. È obbligatorio restituire il testo e tutti i fogli forniti, anche se non si consegna il compito.

1. Considerati CPU-burst time (in ms) del set di processi descritto in tabella, e considerato che l'ordine di arrivo dei processi è P1, P2, P3, P4, P5.

| Processo | Burst Time | Priorità | Tempo di arrivo |
|----------|------------|----------|-----------------|
| P1       | 7          | 1        | 0               |
| P2       | 3          | 2        | 2               |
| P3       | 6          | 2        | 4               |
| P4       | 4          | 3        | 5               |
| P5       | 5          | 3        | 6               |

a) Disegnare 2 diagrammi di Gantt che illustrino l'esecuzione di questi processi usando gli algoritmi di scheduling RR (quanto = 3 ms) e Priorità con RR (quanto = 3 ms). Indicare l'istante di tempo di ciascuna commutazione fra un processo e l'altro

b) Calcolare il tempo di attesa per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo d'attesa e come l'avete applicato per calcolare i risultati esposti)

|    | RR | Priorità (con RR) |
|----|----|-------------------|
| P1 |    |                   |
| P2 |    |                   |
| P3 |    |                   |
| P4 |    |                   |
| P5 |    |                   |

c) Calcolare il tempo di risposta per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo di risposta e come l'avete applicato per calcolare i risultati esposti)

|    | RR | Priorità (con RR) |
|----|----|-------------------|
| P1 |    |                   |
| P2 |    |                   |
| P3 |    |                   |
| P4 |    |                   |
| P5 |    |                   |

2. Tenuto conto che:

- o il pid del processo padre è 9 e quello del figlio è 10.

Cosa stamperà il seguente programma? Giustificare in estrema sintesi la risposta

```

int var = 7;

int main()
{
 pid_t pid;
 int status = 0;
 printf("Sono il processo padre.\n");

 pid = fork();

 status++;
 if(pid == 0)
 var = var + status;

 var = var + pid;
 printf("var = %d \n", var);
 printf("pid = %d \n", getpid());

 return 0;
}

```

**3. Dato il frammento di programma che segue:**

- Quanti processi e quanti thread genera il frammento di programma seguente?
- Evidenziare, se ci sono, le corse critiche. Come si possono evitare?
- Cosa stamperà il programma?

**Giustificare in estrema sintesi le risposte**

```

#define NUM_THREADS 2

int var = 0;

void *func(void* param)
{
 var += 10;
 return 0;
}

int main()
{
 pthread_t threads[NUM_THREADS];

 fork();

 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_create(&threads[i], NULL, func, NULL);
 }

 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_join(threads[i], NULL);
 }
 printf("%d\n", var);

 return 0;
}

```

**4. Nel contesto dei metodi per evitare i deadlock:**

- Dati tre thread T0, T1, T2 e tre tipi di risorsa A (6 istanze), B (5 istanze), C (3 istanze) con le seguenti matrici di Allocazione e Max:

|    | Allocazione |   |   | Max |   |   |
|----|-------------|---|---|-----|---|---|
|    | A           | B | C | A   | B | C |
| T0 | 2           | 2 | 2 | 3   | 3 | 3 |
| T1 | 2           | 1 | 0 | 2   | 2 | 1 |

- b) calcolare il vettore Disponibile e la matrice Bisogno  
 c) Verificare se il sistema è in uno stato sicuro (spiegare i passaggi)  
 d) Può T<sub>1</sub> richiedere (0, 0, 1)? Se viene assegnato cosa succede? Può richiederlo T<sub>2</sub>? Se viene assegnato cosa succede?
5. Si consideri una memoria paginata:  
 Si assumano indirizzi logici a 64 bit, pagine da 32 KB, 4 GB di memoria fisica.  
 a) Quante entry per una tabella delle pagine di un solo livello?  
 b) Quante entry per una tabella delle pagine di secondo livello assumendo 2 KB per la pagina di primo livello?  
 c) Quanti entry per una tabella delle pagine invertita?
6. Si consideri un processo che genera la seguente stringa dei riferimenti alle pagine virtuali:  
 3, 0, 1, 3, 4, 2, 1, 4  
 a) Se il processo ha 3 frame gestiti con LRU quanti page fault vengono generati?  
 b) Quanti page fault vengono generati con l'algoritmo FIFO?
7. Assumendo 3000 cilindri e testina su 1500 (con richiesta precedente a 700)  
 a) Data la coda di richieste 10, 2500, 1400, 1200, 20 come vengono servite le richieste con l'algoritmo SSF (Shortest Seek First)  
 b) Qual è la distanza in cilindri percorsi della testina con l'algoritmo SSF?
8. Spiegare brevemente cos'è un monitor e una variabile di condizione
9. Domande (più di una risposta può essere corretta)
- a) Quale delle seguenti affermazioni è vera (giustificare brevemente le risposte selezionate)
    - a. Un Sistema Operativo a struttura monolitica è più efficiente di uno a microkernel
    - b. Un Sistema Operativo a microkernel è più efficiente di uno a struttura monolitica
    - c. Un Sistema Operativo stratificato ha migliori prestazioni di un sistema monolitico
    - d. Un Sistema Operativo a microkernel facilita l'estensione del sistema
  - b) Quale tra queste affermazioni sullo scheduling dei processi è vera (giustificare brevemente le risposte selezionate)
    - a. Lo scheduling FCFS può causare starvation
    - b. Lo scheduling RR non può portare a starvation
    - c. L'algoritmo SJF può causare starvation
    - d. Lo scheduling priorità con prelazione può causare starvation
  - c) Quale tra queste affermazioni sui thread è vera (giustificare brevemente le risposte selezionate)
    - a. Ad ogni thread user-level corrisponde sempre un solo thread kernel-level
    - b. Ad ogni thread user-level possono corrispondere più thread kernel-level
    - c. A molti thread user-level può corrispondere un solo thread kernel-level
    - d. Ad un thread kernel-level corrisponde un solo thread hardware
  - d) Quale tra queste affermazioni è vera (giustificare brevemente le risposte selezionate)
    - a. Il modello del working set si basa sull'assunzione di località
    - b. Il cambio di località non ha alcun effetto sul page fault rate
    - c. Il modello del working set serve per uscire dal trashing
    - d. Il modello working set può supportare il prepagina

Nome e cognome

Matricola

La durata della prova è di 2 ore. Scrivere in stampatello maiuscolo. È obbligatorio restituire il testo e tutti i fogli forniti, anche se non si consegna il compito.

1. Considerati CPU-burst time (in ms) del set di processi descritto in tabella, e considerato che l'ordine di arrivo dei processi è P1, P2, P3, P4, P5.

| Processo | Burst Time | Priorità | Tempo di arrivo |
|----------|------------|----------|-----------------|
| P1       | 5          | 3        | 0               |
| P2       | 3          | 1        | 2               |
| P3       | 4          | 2        | 4               |
| P4       | 2          | 4        | 5               |
| P5       | 2          | 1        | 6               |

a) Disegnare 2 diagrammi di Gantt che illustrino l'esecuzione di questi processi usando gli algoritmi di scheduling Priorità con prelazione e RR (quanto = 3 ms). Indicare l'istante di tempo di ciascuna commutazione fra un processo e l'altro.

b) Calcolare il tempo di attesa per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo d'attesa e come l'avete applicato per calcolare i risultati esposti)

|    | RR           | Priorità (con RR) |
|----|--------------|-------------------|
| P1 | $3 - 0 = 3$  | $3 - 0 = 3$       |
| P2 | $3 - 2 = 1$  | $2 - 2 = 0$       |
| P3 | $12 - 4 - 8$ | $7 - 4 = 3$       |
| P4 | $11 - 5 = 6$ | $14 - 5 = 9$      |
| P5 | $13 - 6 = 7$ | $6 - 6 = 0$       |

c) Calcolare il tempo di turnaround per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo di risposta e come l'avete applicato per calcolare i risultati esposti)

|    | RR           | Priorità     |
|----|--------------|--------------|
| P1 | $3 + 5 = 8$  | $3 + 5 = 14$ |
| P2 | $1 + 3 = 4$  | $0 + 3 = 3$  |
| P3 | $8 + 4 = 12$ | $3 + 4 = 7$  |
| P4 | $6 + 2 = 8$  | $7 + 2 = 11$ |
| P5 | $7 + 2 = 9$  | $0 + 2 = 2$  |

2. Tenuto conto che:

- o il pid del processo padre è 9 e quello del primo figlio è 10 e quello del secondo figlio è 11.

Cosa stamperà il seguente programma? Giustificare in estrema sintesi la risposta

```
int var = 5;

int main()
{
 pid_t pid1, pid2;
 printf("Sono il processo padre.\n");
 pid1 = fork();
 if(pid1 == 0)
 var = var + 1;
 else {
 pid2 = fork();
 var = var + pid1 + pid2;
 }
 printf("pid = %d var = %d \n", var, getpid());
 return 0;
}
```

**3. Dato il frammento di programma che segue:**

- a) Quanti processi e quanti thread genera il frammento di programma seguente?
- b) Evidenziare, se ci sono, le corse critiche.
- c) Cosa stamperà il programma?

**Giustificare in estrema sintesi le risposte**

```
#define NUM_THREADS 3

int var1 = 10, var2 = 0;
void *func1(void* param)
{
 var1 -= 1;
 return 0;
}

void *func2(void* param)
{
 var2 += 1;
 return 0;
}

int main()
{
 pthread_t threads[NUM_THREADS];

 pthread_create(&threads[0], NULL, func1, NULL); VAR1=-3
 pthread_create(&threads[1], NULL, func2, NULL); VAR2=+1
 pthread_join(threads[0], NULL);
 pthread_join(threads[1], NULL);

 fork();

 pthread_create(&threads[1], NULL, func2, NULL); VAR1=8
 pthread_create(&threads[2], NULL, func1, NULL); VAR2=2
```

```
pthread_join(threads[1], NULL);
pthread_join(threads[2], NULL);

printf("%d %d\n", var1, var2);
return 0;
}
```

**4. Nel contesto dei metodi per evitare i deadlock:**

- a) Dati tre thread T0, T1, T2 e tre tipi di risorsa A (4 istanze), B (4 istanze), C (4 istanze) con le seguenti matrici di Allocazione e Max:

|    | Allocazione |   |   | Max |   |   |
|----|-------------|---|---|-----|---|---|
|    | A           | B | C | A   | B | C |
| T0 | 0           | 2 | 1 | 2   | 3 | 3 |
| T1 | 2           | 1 | 2 | 3   | 2 | 2 |
| T2 | 1           | 0 | 0 | 2   | 2 | 2 |

- b) Calcolare il vettore Disponibile e la matrice Bisogno  
c) Verificare se il sistema è in uno stato sicuro (spiegare i passaggi)  
d) Può T2 richiedere subito (1, 0, 0)? Se viene assegnato cosa succede?

**5. Si consideri una memoria paginata:**

Si assumano uno spazio di indirizzi logici di 4096 pagine con pagine di 2 KB mappate su memoria fisica da 512 frame.

- a) Quante sono i bit dell'indirizzo logico?  
b) Quante sono i bit dell'indirizzo fisico?  
c) Quanti entry per una tabella delle pagine invertita?

**6. Si consideri un processo che genera la seguente stringa dei riferimenti alle pagine virtuali:**

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

- a) Se il processo ha 3 frame gestiti con LRU quanti page fault vengono generati?  
b) Quanti page fault vengono generati con l'algoritmo OPT?

**7. Assumendo 3500 cilindri e testina su 1800 (con richiesta precedente a 500)**

- a) Data la coda di richieste 10, 1500, 1450, 3200, 20, 250 come vengono servite le richieste con SCAN  
b) Qual è la distanza in cilindri percorsi della testina con l'algoritmo SCAN?

**8. Spiegare brevemente vantaggi e svantaggi dei metodi di allocazione dei file concatenata e indicizzata.**

**9. Cos'è un read-write lock? Discutere brevemente il problema dei lettori e scrittori (illustrandone una soluzione).**

**10. Nell'ambito dello scheduling dei processi spiegare brevemente il problema dell'inversione di priorità.**

Nome e cognome \_\_\_\_\_

Matricola \_\_\_\_\_

La durata della prova è di 2 ore. Scrivere in stampatello maiuscolo. È obbligatorio restituire il testo e tutti i fogli forniti, anche se non si consegna il compito.

1. Considerati CPU-burst time (in ms) del set di processi descritto in tabella, e considerato che l'ordine di arrivo dei processi è P1, P2, P3, P4, P5.

| Processo | Burst Time | Priorità | Tempo di arrivo |
|----------|------------|----------|-----------------|
| P1       | 5          | 2        | 0               |
| P2       | 4          | 3        | 2               |
| P3       | 4          | 1        | 4               |
| P4       | 5          | 3        | 6               |
| P5       | 4          | 2        | 8               |

- a) Disegnare 2 diagrammi di Gantt che illustrino l'esecuzione di questi processi usando gli algoritmi di priorità con prelazione e SJF. Indicare l'istante di tempo di ciascuna commutazione fra un processo e l'altro
- b) Calcolare il tempo di attesa per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo d'attesa e come l'avete applicato per calcolare i risultati esposti)

|    | Priorità (prel)   | SJF           |
|----|-------------------|---------------|
| P1 | (9 - 0 - 5 =) 4   | (0 - 0 =) 0   |
| P2 | (17 - 2 - 4 =) 11 | (5 - 2 =) 3   |
| P3 | (8 - 4 - 4 =) 0   | (9 - 4 =) 5   |
| P4 | (22 - 6 - 5 =) 11 | (17 - 6 =) 11 |
| P5 | (13 - 8 - 4 =) 1  | (13 - 8 =) 5  |

- c) Calcolare il tempo di risposta per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo di risposta e come l'avete applicato per calcolare i risultati esposti)

|    | Priorità (prel) | SJF           |
|----|-----------------|---------------|
| P1 | (0 - 0 =) 0     | (0 - 0 =) 0   |
| P2 | (13 - 2 =) 11   | (5 - 2 =) 3   |
| P3 | (4 - 4 =) 0     | (9 - 4 =) 5   |
| P4 | (17 - 6 =) 11   | (14 - 6 =) 11 |
| P5 | (9 - 8 =) 1     | (13 - 8 =) 5  |

2. Tenuto conto che:

- o il pid del processo padre è 5 e quello del primo figlio è 6 e del secondo è 7.

Cosa stamperà il seguente programma? Giustificare in estrema sintesi la risposta

```
int var = 5;
```

```

int main()
{
 pid_t pid1, pid2;
 int status = 2;
 printf("Sono il processo padre.\n");

 pid1 = fork();

 status++;
 if(pid1 == 0) {
 pid2 = fork();
 var = var + pid1 + pid2;
 }

 var = var + status;
 printf("var = %d \n", var);
 printf("pid = %d \n", getpid());

 return 0;
}

```

**3. Dato il frammento di programma che segue:**

- Quanti processi e quanti thread genera il frammento di programma seguente?
- Evidenziare, se ci sono, le corse critiche. Come si possono evitare?
- Cosa stamperà il programma?

**Giustificare in estrema sintesi le risposte**

```

#define NUM_THREADS 2

int var = 0;

void *func1(void* param)
{
 var += 1;
 return 0;
}

void *func2(void* param)
{
 var -= 1;
 return 0;
}

int main()
{
 pthread_t threads1[NUM_THREADS];
 pthread_t threads2[NUM_THREADS];

 fork();

 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_create(&threads1[i], NULL, func1, NULL);
 pthread_create(&threads2[i], NULL, func2, NULL);
 }

 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_join(threads1[i], NULL);
 pthread_join(threads2[i], NULL);
 }
 printf("%d\n", var);

 return 0;
}

```

**4. Nel contesto dei metodi per evitare i deadlock:**

- a) Dati tre thread T<sub>0</sub>, T<sub>1</sub>, T<sub>2</sub> e tre tipi di risorsa A (7 istanze), B (4 istanze), C (3 istanze) con le seguenti matrici di Allocazione e Max:

|                | Allocazione |   |   | Max |   |   |
|----------------|-------------|---|---|-----|---|---|
|                | A           | B | C | A   | B | C |
| T <sub>0</sub> | 2           | 2 | 2 | 4   | 4 | 3 |
| T <sub>1</sub> | 1           | 1 | 1 | 2   | 2 | 1 |
| T <sub>2</sub> | 3           | 0 | 0 | 6   | 3 | 3 |

- b) calcolare il vettore Disponibile e la matrice Bisogno  
c) Verificare se il sistema è in uno stato sicuro (spiegare i passaggi)  
d) Può T<sub>1</sub> richiedere (1, 0, 0)? Se viene assegnato cosa succede?

**5. Si consideri una memoria paginata, dato un processo con una tabella delle pagine in memoria**

- a) Se ogni accesso in memoria richiede 40 ns qual è il tempo per accedere ad un indirizzo della memoria paginata?  
b) Aggiungendo una TLB con hit ratio del 80% e stimando un tempo di accesso di 2 ns qual è il EAT (Effective Access Time)?  
c) Descrivere succintamente cos'è una TLB e qual è la sua funzione

**6. Si consideri un processo che genera la seguente stringa dei riferimenti alle pagine virtuali:**

3, 1, 0, 2, 4, 2, 1, 3

- a) Se il processo ha 3 frame gestiti con LRU quanti page fault vengono generati?  
b) Quanti page fault vengono generati con l'algoritmo OPT?

**7. Assumendo 4000 cilindri e testina su 1600 (con richiesta precedente a 800)**

- a) Data la coda di richieste 1000, 1500, 400, 3000, 20 come vengono servite le richieste con l'algoritmo SSF (Shortest Seek First)  
b) Qual è la distanza in cilindri percorsi della testina con l'algoritmo SSF?

**8. Descrivere in maniera sintetica ma esaustiva il concetto di working set WS all'istante t con intervallo Δ WS(t, Δ) e la sua funzione.**

- a) Data la stringa dei riferimenti 5, 4, 1, 6, 2, 2, 7, 10, 7 scrivere il contenuto di WS(9,4)

**9. Spiegare brevemente la problematica del thrashing.**

**10. Domande (più di una risposta può essere corretta), giustificare brevemente le risposte selezionate.**

- a) **Quale delle seguenti affermazioni è vera**
- a. Un Sistema Operativo stratificato è più efficiente di uno a struttura monolitica
  - b. Un Sistema Operativo monolitico è più facilmente estendibile di un sistema stratificato
  - c. Un Sistema Operativo a struttura monolitica è più efficiente di uno a microkernel
  - d. Un Sistema Operativo a microkernel semplifica l'estensione del sistema
- b) **Quale tra queste affermazioni su thread e processi è vera**
- a. Due thread possono determinare una corsa critica solo su variabili globali
  - b. Ogni thread è associato ad un solo PID
  - c. Due thread non possono condividere lo stesso PID ma possono avere lo stesso PPID
  - d. Quando viene creato un thread, questo riceve una copia dello spazio di indirizzamento del processo che lo ha creato

- c) Quale tra queste affermazioni sullo scheduling dei processi è vera
- Lo scheduling RR può portare a starvation
  - Lo scheduling FCFS può causare starvation
  - Lo scheduling priorità con prelazione può causare starvation
  - L'algoritmo SJF non può causare starvation
- d) Quale tra queste affermazioni è vera
- Il modello del working set previene lo starvation
  - Il modello working set può supportare il prepaging
  - Il modello del working set può portare al trashing
  - Il cambio di località ha effetto sul page fault rate
- e) Quale delle seguenti affermazioni sul copy\_on\_write è vera
- È una tecnica per sincronizzare i processi evitando corse critiche
  - È un metodo che velocizza la creazione dei processi
  - È una tecnica per ridurre i tempi di scrittura sul file system
  - È una tecnica che consente di minimizzare il numero di pagine allocate per un nuovo processo
- f) Quale tra queste affermazioni sui meccanismi di sincronizzazione è vera
- Il semaforo contatore è un caso particolare di semaforo binario
  - Lo spin-lock determina un context switch perché sospende un processo
  - L'istruzione test\_and\_set() serve per implementare la mutua esclusione
  - L'uso dei semafori rende impossibile il deadlock
- g) Quale delle seguenti affermazioni sugli algoritmi di sostituzione di pagina è vera
- Permettono di gestire in modo efficiente l'accesso al file system
  - Permettono di implementare il demand paging in modo efficiente
  - Permettono l'allocazione efficiente delle pagine di memoria
  - Permettono di risolvere l'anomalia di Belady
- h) Quale tra queste affermazioni sull'allocazione dei file è vera
- L'allocazione linkata di un file presenta problemi di frammentazione esterna
  - L'allocazione contigua soffre di problemi di frammentazione esterna
  - L'allocazione indicizzata non presenta problemi di frammentazione esterna
  - L'allocazione indicizzata favorisce l'accesso random al file

Nome e cognome [REDACTED]

Matricola [REDACTED]

La durata della prova è di 2 ore. Scrivere in stampatello maiuscolo. È obbligatorio restituire il testo e tutti i fogli forniti, anche se non si consegna il compito.

1. Considerati CPU-burst time (in ms) del set di processi descritto in tabella, e considerato che l'ordine di arrivo dei processi è P1, P2, P3, P4, P5.

| Processo | Burst Time | Priorità | Tempo di arrivo |
|----------|------------|----------|-----------------|
| P1       | 3          | 3        | 0               |
| P2       | 6          | 1        | 1               |
| P3       | 2          | 3        | 3               |
| P4       | 3          | 2        | 4               |
| P5       | 6          | 1        | 5               |

- a) Disegnare 2 diagrammi di Gantt che illustrino l'esecuzione di questi processi usando gli algoritmi di priorità e round robin (quanto = 3). Indicare l'istante di tempo di ciascuna commutazione fra un processo e l'altro
- b) Calcolare il tempo di attesa per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo d'attesa e come l'avete applicato per calcolare i risultati esposti)

|    | Priorità | RR |
|----|----------|----|
| P1 |          |    |
| P2 |          |    |
| P3 |          |    |
| P4 |          |    |
| P5 |          |    |

- c) Calcolare il tempo di risposta per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo di risposta e come l'avete applicato per calcolare i risultati esposti)

|    | Priorità | RR |
|----|----------|----|
| P1 |          |    |
| P2 |          |    |
| P3 |          |    |
| P4 |          |    |
| P5 |          |    |

2. Tenuto conto che:

- o il pid del processo padre è 8 e quello del primo figlio è 9 quello del secondo figlio è 10.

Cosa stamperebbe il seguente programma? Giustificare in estrema sintesi la risposta

```
int var = 1;
```

```

int main()
{
 pid_t pid1, pid2;
 int status = 0;

 pid1 = fork();

 status += var;

 if(pid1 == 0)
 var = var + status;
 else
 var = status;

 pid2 = fork();

 var = var + pid1 + pid2;

 printf("var = %d \n", var);
 printf("pid = %d \n", getpid());
}

return 0;
}

```

**3. Dato il frammento di programma che segue:**

- a) Quanti processi e quanti thread genera il frammento di programma seguente?
- b) Evidenziare, se ci sono, le corse critiche. Come si possono evitare?
- c) Cosa stamperà il programma?

**Giustificare in estrema sintesi le risposte**

```

#define NUM_THREADS 2

int var = 10;

void *func1(void* param)
{
 var += 2;
 return 0;
}

void *func2(void* param)
{
 var -= 2 ;
 return 0;
}

int main()
{
 pthread_t threads1[NUM_THREADS];
 pthread_t threads2[NUM_THREADS];

 fork();

 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_create(&threads1[i], NULL, func1, NULL);
 pthread_join(threads1[i], NULL);
 }

 fork();
}

```

```

for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_create(&threads2[i], NULL, func2, NULL);
 pthread_join(threads2[i], NULL);
}

printf("%d\n", var);

return 0;
}

```

**4. Nel contesto dei metodi per evitare i deadlock:**

- a) Dati tre thread T<sub>0</sub>, T<sub>1</sub>, T<sub>2</sub> e tre tipi di risorsa A (4 istanze), B (3 istanze), C (5 istanze) con le seguenti matrici di Allocazione e Max:

| Allocazione    |   |   | Max |   |   |   |   |
|----------------|---|---|-----|---|---|---|---|
|                | A | B | C   | A | B | C |   |
| T <sub>0</sub> | 2 | 1 | 2   | . | 3 | 2 | 4 |
| T <sub>1</sub> | 1 | 0 | 1   | . | 1 | 2 | 3 |
| T <sub>2</sub> | 1 | 1 | 0   | . | 1 | 2 | 2 |

- b) calcolare il vettore Disponibile e la matrice Bisogno  
 c) Verificare se il sistema è in uno stato sicuro (spiegare i passaggi)  
 d) Può T<sub>1</sub> richiedere (0, 1, 0)? Se viene assegnato cosa succede?

**5. Si consideri una memoria paginata:**

Si assumano indirizzi logici a 32 bit e pagine da 16 KB

- a) Quante entry occorrono per una tabella delle pagine di un solo livello?  
 b) Quante entry per una tabella delle pagine di secondo livello assumendo 4 KB per la pagina di primo livello?

**6. Si consideri un processo che genera la seguente stringa dei riferimenti alle pagine virtuali:**  
 1, 4, 5, 2, 4, 5, 1, 5, 4, 1, 3, 5

- a) Se il processo ha 4 frame gestiti con LRU quanti page fault vengono generati?  
 b) Quanti page fault vengono generati con l'algoritmo FIFO?  
 c) Spiegare brevemente cos'è la stringa dei riferimenti e a cosa servono gli algoritmi di sostituzione di pagina

**7. Assumendo 3000 cilindri e testina su 100 (con richiesta precedente a 200)**

- a) Data la coda di richieste 2000, 1200, 300, 200, 500 come vengono servite le richieste con l'algoritmo SCAN  
 b) Qual è la distanza in cilindri percorsi della testina con l'algoritmo SSTF?  
 c) Spiegare brevemente per quali dispositivi sono utili questi algoritmi e perché

**8. Descrivere schematicamente il layout di memoria di un processo**

**9. Spiegare brevemente la differenza tra indirizzo fisico ed indirizzo virtuale**

**10. Descrivere brevemente cos'è un monitor e una variabile di condizione**

La durata della prova è di 2 ore. Scrivere in stampatello maiuscolo. È obbligatorio restituire il testo e tutti i fogli forniti, anche se non si consegna il compito.

- 1. Considerati CPU-burst time (in ms) del set di processi descritto in tabella, e considerato che l'ordine di arrivo dei processi è P1, P2, P3, P4, P5.**

| Processo | Burst Time | Priorità | Tempo di arrivo |
|----------|------------|----------|-----------------|
| P1       | 5          | 3        | 0               |
| P2       | 5          | 1        | 2               |
| P3       | 4          | 1        | 3               |
| P4       | 3          | 4        | 5               |
| P5       | 4          | 2        | 7               |

a) Disegnare 2 diagrammi di Gantt che illustrino l'esecuzione di questi processi usando gli algoritmi di scheduling Priorità con prelazione e SJF. Indicare l'istante di tempo di ciascuna commutazione fra un processo e l'altro

b) Calcolare il tempo di attesa per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo d'attesa e come l'avete applicato per calcolare i risultati esposti)

|    | Priorità (prelazione) | SJF |
|----|-----------------------|-----|
| P1 | 13                    | 11  |
| P2 | 6                     | 16  |
| P3 | 3                     | 6   |
| P4 | 18                    | 5   |
| P5 | 11                    | 10  |

c) Calcolare il tempo di risposta per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo di risposta e come l'avete applicato per calcolare i risultati esposti)

|    | Priorità (prelazione) | SJF |
|----|-----------------------|-----|
| P1 | 0                     | 0   |
| P2 | 0                     | 14  |
| P3 | 0                     | 0   |
| P4 | 13                    | 0   |
| P5 | 4                     | 3   |

**2. Tenuto conto che:**

- o il pid del processo padre è 9 e quello del primo figlio è 10 e quello del secondo figlio è 11 quello del figlio del primo figlio è 12

Cosa stamperà il seguente programma? Giustificare in estrema sintesi la risposta

```
int var = 50;

int main()
{
 pid_t pid1, pid2;
 printf("Sono il processo padre.\n");
 pid1 = fork();
 if(pid1 == 0)
 var = var + 20;
 pid2 = fork();
 else
 pid2 = fork();
 var = var + pid1 + pid2;
 printf("pid = %d var = %d\n", getpid(), var);
 return 0;
}
```

**3. Dato il frammento di programma che segue:**

- a) Quanti processi e quanti thread genera il frammento di programma seguente?
- b) Evidenziare, se ci sono, le corse critiche.
- c) Cosa stamperà il programma?

**Giustificare in estrema sintesi le risposte**

```
#define NUM_THREADS 3

int var1 = 10, var2 = 0;
void *func1(void* param)
{
 var1 -= 5;
 return 0;
}

void *func2(void* param)
{
 var2 = 5 + var1 + var2;
 return 0;
}

int main()
{
 pthread_t threads[NUM_THREADS];
 fork();

 pthread_create(&threads[0], NULL, func1, NULL);
 pthread_create(&threads[1], NULL, func2, NULL);
 pthread_join(threads[0], NULL);
 pthread_join(threads[1], NULL);

 pthread_create(&threads[2], NULL, func2, NULL);
 printf("%d %d\n", var1, var2);
```

```
 return 0;
}
```

**4. Nel contesto dei metodi per evitare i deadlock:**

- a) Dati tre thread T0, T1, T2 e tre tipi di risorsa A (5 istanze), B (5 istanze), C (5 istanze) con le seguenti matrici di Allocazione e Max:

|                | Allocazione |   |   | Max |   |   |
|----------------|-------------|---|---|-----|---|---|
|                | A           | B | C | A   | B | C |
| T <sub>0</sub> | 0           | 3 | 1 | 4   | 4 | 3 |
| T <sub>1</sub> | 2           | 1 | 2 | 3   | 2 | 4 |
| T <sub>2</sub> | 1           | 0 | 0 | 3   | 4 | 2 |

- b) Calcolare il vettore Disponibile e la matrice Bisogno  
c) Verificare se il sistema è in uno stato sicuro (spiegare i passaggi)  
d) Può T2 richiedere subito (0, 0, 1)? Se viene assegnato cosa succede?

**5. Si consideri una memoria paginata:**

Si assuma una memoria virtuale paginata di 4096 indirizzi logici e pagine di 2 KB mappate su memoria fisica da 512 frame.

- a) Quante sono i bit dell'indirizzo logico?  
b) Quante sono i bit dell'indirizzo fisico?  
c) Quanti entry per una tabella delle pagine?

**6. Si consideri un processo che genera la seguente stringa dei riferimenti alle pagine virtuali:  
5, 1, 3, 4, 1, 2, 1, 4**

- a) Se il processo ha 3 frame gestiti con LRU quanti page fault vengono generati?  
b) Quanti page fault vengono generati con l'algoritmo FIFO?

**7. Assumendo 3500 cilindri e testina su 1800**

- a) Data la coda di richieste 10, 1500, 1450, 3200, 20, 250 come vengono servite le richieste con shortest seek time first (SSTF)  
b) Qual è la distanza in cilindri percorsi della testina con l'algoritmo (SSTF)?

**8. Spiegare brevemente vantaggi e svantaggi dei metodi di allocazione dei file concatenata e indicizzata.**

**9. Descrivere schematicamente come viene servita un'interruzione e cos'è un vettore delle interruzioni.**

**10. Descrivere schematicamente il layout di memoria di un processo.**

Nome e cognome \_\_\_\_\_ Matricola \_\_\_\_\_

La durata della prova è di 2 ore. Scrivere in stampatello maiuscolo. E' obbligatorio restituire il testo e tutti i fogli forniti, anche se non si consegna il compito.

**1. Considerati CPU-burst time (in ms) del set di processi descritto in tabella, e considerato che l'ordine di arrivo dei processi è P1, P2, P3, P4, P5.**

| Processo | Burst Time | Priorità | Tempo di arrivo |
|----------|------------|----------|-----------------|
| P1       | 7          | 2        | 0               |
| P2       | 5          | 1        | 2               |
| P3       | 3          | 2        | 4               |
| P4       | 3          | 4        | 5               |
| P5       | 5          | 3        | 6               |

a) Disegnare 2 diagrammi di Gantt che illustrino l'esecuzione di questi processi usando gli algoritmi di scheduling Priorità e prelazione e RR (quanto = 4 ms). Indicare l'istante di tempo di ciascuna commutazione fra un processo e l'altro

b) Calcolare il tempo di attesa per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo d'attesa e come l'avete applicato per calcolare i risultati esposti)

|    | RR (q=4) | Priority Preemptive |
|----|----------|---------------------|
| P1 |          |                     |
| P2 |          |                     |
| P3 |          |                     |
| P4 |          |                     |
| P5 |          |                     |

c) Calcolare il tempo di turnaround per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo di risposta e come l'avete applicato per calcolare i risultati esposti)

|    | RR (q=4) | Priority Preemptive |
|----|----------|---------------------|
| P1 |          |                     |
| P2 |          |                     |
| P3 |          |                     |
| P4 |          |                     |
| P5 |          |                     |

**2. Tenuto conto che:**

- il pid del processo padre è **100** e quello del suo primo figlio è **101**, quello del secondo figlio è **102**, quello del figlio del primo figlio è **103**.

Cosa stamparerà il seguente programma? Giustificare in estrema sintesi la risposta

```

int var = 10;

int main()
{
 pid_t pid1, pid2;

 printf("Sono il processo padre.\n");

 pid1 = fork();

 if(pid1 == 0)
 var = var + 5;
 pid2 = fork();
 else {
 pid2 = fork();
 }
 var = var + pid1 + pid2;

 printf("pid = %d var = %d \n", var, getpid());

 return 0;
}

```

**3. Dato il frammento di programma che segue:**

- a) Quanti processi e quanti thread genera il frammento di programma seguente?
- b) Cosa stamperà il programma?
- c) Evidenziare, se ci sono, le corse critiche indicando come evitarle

**Giustificare in estrema sintesi le risposte**

```

int var1 = 5, var2 = 50;

void *func1(void* param)
{
 var1 += 10;
 var2 -= 15;
 return 0;
}

void *func2(void* param)
{
 var2 += 1;
 return 0;
}

int main()
{
 pthread_t threads[NUM_THREADS];
 int pid;

 pid = fork();
 if (pid == 0) {
 pthread_create(&threads[0], NULL, func1, NULL);
 pthread_create(&threads[1], NULL, func2, NULL);
 pthread_join(threads[0], NULL);
 pthread_join(threads[1], NULL);
 }
}

```

```

else {
 pthread_create(&threads[2], NULL, func1, NULL);
 pthread_join(threads[2], NULL);
}

printf("%d %d\n", var1, var2);

return 0;
}

```

**4. Nel contesto dei metodi per evitare i deadlock:**

- a) Dati tre thread T0, T1, T2 e tre tipi di risorsa A (5 istanze), B (5 istanze), C (4 istanze) con le seguenti matrici di Allocazione e Max:

| Allocazione |   |   | Max |   |   |
|-------------|---|---|-----|---|---|
|             | A | B | C   | A | B |
| T0          | 1 | 2 | 1   | 6 | 5 |
| T1          | 2 | 1 | 2   | 3 | 2 |
| T2          | 1 | 1 | 0   | 5 | 4 |

- b) Calcolare il vettore Disponibile e la matrice Bisogno  
 c) Verificare se il sistema è in uno stato sicuro (spiegare i passaggi)  
 d) Può T0 richiedere subito (0, 1, 0)? Se viene assegnato lo stato rimane sicuro?  
 e) Spiegare in modo succinto cosa significa stato sicuro.

**5. Si consideri una memoria paginata:**

Si assumano uno spazio di indirizzi logici di 4096 pagine con pagine di 4 KB mappate su memoria fisica di 1024 frame.

- a) Quante sono i bit dell'indirizzo logico?  
 b) Quante sono i bit dell'indirizzo fisico?  
 c) Quanti entry per una tabella delle pagine?  
 d) Quante entry ha una tabella delle pagine di secondo livello se quella di primo è di 4 entry?

**6. Si consideri un processo che genera la seguente stringa dei riferimenti alle pagine virtuali:**

7, 7,3, 3, 4,7, 1, 4, 1, 4,7,5,3

- a) Se il processo ha 4 frame gestiti con LRU quanti page fault vengono generati?  
 b) Quanti page fault vengono generati con l'algoritmo OPT?

**7. Assumendo 5000 cilindri e testina su 3800 (con richiesta precedente a 3500)**

- a) Data la coda di richieste 1000, 2500, 4450, 4150, 2000, 250 come vengono servite le richieste con C-LOOK  
 b) Qual è la distanza in cilindri percorsi della testina con l'algoritmo C-LOOK?

**8. Cosa si intende per thrashing? Descrivere schematicamente la problematica.**

**9. Cos'è un'istruzione compare-and-swap? Spiegarne funzionamento e funzione.**

**10. Descrivere schematicamente le strutture dati principali utilizzate dal sistema operativo per la gestione del file system.**

Nome e cognome \_\_\_\_\_ Matricola \_\_\_\_\_

La durata della prova è di 2 ore. Scrivere in stampatello maiuscolo. E' obbligatorio restituire il testo e tutti i fogli forniti, anche se non si consegna il compito.

- 1. Considerati CPU-burst time (in ms) del set di processi descritto in tabella, e considerato che l'ordine di arrivo dei processi è P1, P2, P3, P4, P5.**

| Processo | Burst Time | Tempo di arrivo |
|----------|------------|-----------------|
| P1       | 6          | 0               |
| P2       | 5          | 1               |
| P3       | 4          | 3               |
| P4       | 6          | 5               |
| P5       | 6          | 7               |

- a) Disegnare 2 diagrammi di Gantt che illustrino l'esecuzione di questi processi usando gli algoritmi di SJF e RR (quanto 3 mm). Indicare l'istante di tempo di ciascuna commutazione fra un processo e l'altro
- b) Calcolare il tempo di attesa per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo d'attesa e come l'avete applicato per calcolare i risultati esposti)

|    | SJF | RR |
|----|-----|----|
| P1 |     |    |
| P2 |     |    |
| P3 |     |    |
| P4 |     |    |
| P5 |     |    |

- c) Calcolare il tempo di risposta per ciascun processo e per ciascun algoritmo di scheduling. Descrivere il procedimento usato (in estrema sintesi definire il tempo di risposta e come l'avete applicato per calcolare i risultati esposti)

|    | SJF | RR |
|----|-----|----|
| P1 |     |    |
| P2 |     |    |
| P3 |     |    |
| P4 |     |    |
| P5 |     |    |

**2. Tenuto conto che:**

- o il pid del processo padre è 8 e quello del primo figlio è 9 quello del secondo figlio è 10.

Cosa stamperà il seguente programma? Giustificare in estrema sintesi la risposta

```
int var = 20;
```

```

int main()
{
 pid_t pid1, pid2;
 int status = 5;

 pid1 = fork();

 status++;
 if(pid1 == 0){
 var = var + status;
 }
 else {
 pid2 = fork();
 if (pid2 != 0) var = var + pid1 + pid2 + status;
 }
 printf("var = %d \n", var);
 printf("pid = %d \n",getpid());

 return 0;
}

```

**3. Dato il frammento di programma che segue:**

- a) Quanti processi e quanti thread genera il frammento di programma seguente?
- b) Evidenziare, se ci sono, le corse critiche. Come si possono evitare?
- c) Cosa stamperà il programma?

**Giustificare in estrema sintesi le risposte**

```

#define NUM_THREADS 3

int var1 = 30, var2 = 3;

void *func1(void* param)
{
 var2 = var2 + 10;
 return 0;
}

void *func2(void* param)
{
 var1 = var1 + var2 - 20;
 return 0;
}

int main()
{
 pthread_t threads[NUM_THREADS];

 fork();
 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_create(&threads[i], NULL, func1, NULL);
 }
 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_join(threads[i], NULL);
 }
 for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_create(&threads[i], NULL, func2, NULL);
 }
}

```

```

for (int i = 0; i < NUM_THREADS; ++i) {
 pthread_join(threads[i], NULL);
}

printf("%d\n", var1);

return 0;
}

```

**4. Nel contesto dei metodi per evitare i deadlock:**

- a) Dati tre thread T<sub>0</sub>, T<sub>1</sub>, T<sub>2</sub> e tre tipi di risorsa A (5 istanze), B (5 istanze), C (5 istanze) con le seguenti matrici di Allocazione e Max:

|                | Allocazione |   |   | Max |   |   |
|----------------|-------------|---|---|-----|---|---|
|                | A           | B | C | A   | B | C |
| T <sub>0</sub> | 1           | 1 | 1 | 5   | 4 | 4 |
| T <sub>1</sub> | 1           | 0 | 1 | 4   | 3 | 3 |
| T <sub>2</sub> | 0           | 2 | 0 | 3   | 4 | 3 |

- b) calcolare il vettore Disponibile e la matrice Bisogno  
 c) Verificare se il sistema è in uno stato sicuro (spiegare i passaggi)  
 d) Può T<sub>0</sub> richiedere (0, 1, 0)? Se viene assegnato cosa succede?

**5. Si consideri una memoria paginata:**

Si assuma uno spazio di indirizzi logici di 2048 pagine con pagine da 4 KB mappate su memoria fisica di 512 frame

- a) Quanti bit occorrono per l'indirizzo logico?  
 b) Quanti bit occorrono per l'indirizzo fisico?

**6. Si consideri un processo che genera la seguente stringa dei riferimenti alle pagine virtuali:**

4, 2, 3, 4, 4, 2, 2, 4, 3, 2, 2

- a) Se il processo ha 2 frame gestiti con LRU quanti page fault vengono generati?  
 b) Quanti page fault vengono generati con l'algoritmo OPT?

**7. Assumendo 5000 cilindri e testina su 1000 (con richiesta precedente a 1100)**

- a) Data la coda di richieste 300, 1800, 700, 4800, 1200, 800 come vengono servite le richieste con l'algoritmo LOOK  
 b) Qual è la distanza in cilindri percorsi della testina con l'algoritmo LOOK?  
 c) Spiegare brevemente quando è utile lo scheduling del disco

**8. Descrivere brevemente cos'è un monitor e una variabile di condizione**

**9. Descrivere il ruolo del Translation Look-Aside Buffer (TLB) nella gestione del paging.**

**10. Descrivere il problema del trashing e discutendo le strategie per prevenirlo.**