

**Cognome e nome:** \_\_\_\_\_ **Matricola:** \_\_\_\_\_ **Posto:** \_\_\_\_\_

**REGOLE.** Il presente esame scritto deve essere svolto in un tempo massimo di 90 minuti dalla sua presentazione.

Non è consentita la consultazione di libri o appunti in forma cartacea o elettronica, né l'uso di palmari e telefoni cellulari.

Per superare l'esame il candidato deve riportare generalità e matricola negli spazi indicati, acquisire almeno 1,5 punti nel Quesito 1 e un totale di almeno 18 punti su tutti i quesiti, inserendo le proprie risposte interamente su questi fogli.

Per la convalida e registrazione del voto finale il docente si riserva di proporre al singolo candidato una prova orale.

**Quesito 1 (punti 4): 1 punto per risposta giusta, diminuzione di 0,33 punti per risposta sbagliata, 0 punti per risposta vuota**

**[1.A]** Indicare quale tra le seguenti affermazioni è corretta. Un semaforo contatore può:

1. assumere valori interi arbitrari
2. assicurare l'accesso in mutua esclusione ad una risorsa condivisa
3. servire alla realizzazione di una struttura *monitor*
4. consentire l'accesso simultaneo di più processi ad una risorsa condivisa, fino al limite fissato dal valore di inizializzazione del contatore.

**[1.B]** Quale tra le seguenti politiche di ordinamento, in generale minimizza il tempo medio di attesa dei processi:

1. FCFS
2. Round-Robin con valutazione dell'attributo di priorità dei processi
3. Round-Robin senza valutazione dell'attributo di priorità dei processi
4. Shortest Job First.

**[1.C]:** Quale tra i seguenti costituisce un criterio valido di valutazione di una politica di ordinamento di processi:

1. la capacità di trattare anche processi di lunga durata
2. il numero di processi completati per unità di tempo
3. il numero di processi in esecuzione per unità di tempo
4. il numero di processi in attesa di essere eseguiti.

**[1.D]:** Data un disco ampio 4 GB, con blocchi ampi 4 KB, e contenente 128 K *file*, l'ampiezza in B della FAT dipende da:

1. il numero di file in essa rappresentati
2. l'ampiezza dei blocchi
3. l'ampiezza del disco in blocchi e l'ampiezza degli indici di blocco
4. l'ampiezza del disco.

**Quesito 2 – (6 punti):**

Si consideri un sistema dotato di memoria virtuale, con memoria fisica (RAM) divisa in 8 *page frame* condivisa da 4 processi contemporaneamente attivi: A, B, C e D. Si supponga che la RAM sia inizialmente vuota e che ogni riferimento a una pagina non presente in RAM comporti il suo caricamento in RAM (se serve, con rimpiazzo di un'altra pagina già presente). Si supponga inoltre che:

- all'istante 15 venga riferita la pagina logica 2 del processo C (caricandola nel *page frame* 0);
- all'istante 27 venga riferita la pagina logica 8 del processo D (caricandola nel *page frame* 3);
- all'istante 33 venga riferita la pagina logica 3 del processo A (caricandola nel *page frame* 5);
- all'istante 40 venga riferita la pagina logica 1 del processo A (caricandola nel *page frame* 6);
- all'istante 48 venga riferita la pagina logica 0 del processo A (caricandola nel *page frame* 7);
- all'istante 80 venga riferita la pagina logica 5 del processo C (caricandola nel *page frame* 2);
- all'istante 90 venga riferita la pagina logica 9 del processo C (caricandola nel *page frame* 1);
- all'istante 92 venga riferita la pagina logica 3 del processo B (caricandola nel *page frame* 4);
- all'istante 95 venga eseguito un *refresh* dei bit R (riferita) di tutte le pagine in memoria;
- all'istante 96 C riferisce la pagina logica 2;
- all'istante 97 A riferisce la pagina logica 3;
- all'istante 98 A riferisce la pagina logica 0;
- all'istante 100, C riferisce la pagina logica 5;
- all'istante 105, A riferisce la pagina logica 9;

Se utile, si assuma che nessuna pagina sia modificata durante tutta l'esecuzione.

Si supponga che il sistema utilizzi un algoritmo di sostituzione *second chance* (locale)

Si supponga inoltre che a supporto di tale algoritmo vi sia una o più liste ordinate delle pagine accedute ad un dato istante e che tale/tali lista/liste siano composti di elementi del tipo: C2(15, 1)

dove C2 indica che si tratta della pagina logica 2 del processo C, mentre (15, 1) indica che tale pagina è stata caricata in memoria all'istante 15 e che il suo bit 'riferita' è uguale a 1.

Assumendo che non accada più nulla di significativo fino all'istante 110 si scriva/scrivano:

Cognome e nome: \_\_\_\_\_ Matricola: \_\_\_\_\_ Posto: \_\_\_\_\_  
 [2.A] la/le lista/liste delle pagine all'istante 85

[2.B] la/le lista/liste delle pagine all'istante 101

[2.C] la/le lista/liste delle pagine all'istante 106

**Quesito 3 – (7 punti):**

Process A {     $V(mutex);$  $y = y + 2x;$     $P(SemA);$     }	Process B {   $P(semB);$     $x = x + 1;$  $V(mutex);$  $V(SemB);$   $Print(x);$  }	Process C {   $V(semA)$     $x = y * 3;$     $V(SemC);$     $Print(y);$  }
--	--	--

Sincronizzazione di 3 Processi con Semafori.

Si considerino tre processi (A, B e C) i quali devono eseguire alcune operazioni sulle variabili x e y e poi stamparne il risultato finale.

A questo proposito, si consideri la seguente sequenza di avvenimenti:

- I processi A e C devono essere in attesa di essere risvegliati
- Il processo B sveglia il processo A, che a sua volta sveglia il processo C.
- I tre processi accedono concorrentemente (ma in mutua esclusione) alle variabili condivise x e y, poi B e C si bloccano in attesa che A abbia terminato la computazione su x e y (B e C avranno eseguito la loro computazione su x e y prima di bloccarsi)
- Terminata la computazione, è A stesso a svegliare B che a sua volta sveglia C.
- B stampa il valore di x e C stampa il valore di y (senza un particolare ordine prestabilito).

**[3.A]** Assumendo che i semafori SemA, SemB e SemC siano inizializzati a 0, che il semaforo Mutex sia inizializzato a 1, e che inizialmente si abbia  $x = 2$  e  $y = 1$ , si correggano e/o completino i frammenti di codice a lato.

**[3.B]** Si discuta inoltre i possibili valori finali delle variabili x e y.

Cognome e nome: \_\_\_\_\_ Matricola: \_\_\_\_\_ Posto: \_\_\_\_\_

**Quesito 4– (6 punti):**

Si consideri la seguente serie di riferimenti a pagine di memoria: 1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6.

Si considerino le seguenti politiche di rimpiazzo LRU ed Optimal.Quanti *page fault* avvengono considerando una RAM con solo 4 *page frame* ed inizialmente vuota?Si completino inoltre le tabelle mostrando ad ogni istante il contenuto dei 4 *page frame* di cui è composta la RAM (non è necessario che lo studente mantenga un preciso ordine delle pagine virtuali nelle *page frame*).Politica di rimpiazzo **LRU**; totale *page fault*? \_\_\_\_\_

r1	r2	r3	r4	r2	r1	r5	r6	r2	r1	r2	r3	r7	r6	r3	r2	r1	r2	r3	r6

Politica di rimpiazzo **Optimal**; totale *page fault*? \_\_\_\_\_

r1	r2	r3	r4	r2	r1	r5	r6	r2	r1	r2	r3	r7	r6	r3	r2	r1	r2	r3	r6

**Quesito 5 – (3 punti):**Si consideri il caso di un *file system* NTFS; si discuta brevemente se e perché la struttura di *overhead* per memorizzare un *file* risenta della contiguità con cui i blocchi di tale *file* sono disposti sul disco fisso.**Quesito 6 – (6 punti):**

Un dispositivo elettronico (GPS) calcola la posizione dal segnale ricevuto dai satelliti del servizio GPS ed invia su base regolare tale posizione ad un elaboratore elettronico utilizzando una linea seriale, tramite un collegamento Bluetooth, con protocollo UART - 8N1. Per limitazioni della tecnologia utilizzata lato ricevente si ipotizzi che la velocità supportata sia di 19200 baud (la comunicazione avverrà pertanto a tale velocità).

Si consideri la posizione descritta da 3 stringhe alfanumeriche testuali, dette "NMEA sentences", costituite ciascuna al più da 82 caratteri, compresi i ritorni a capo. Nota: le stringhe NMEA possibili sono molte, ma il progettista ha scelto di considerarne 3 per propria scelta progettuale, essendo interessato solo a determinati parametri.

Il dispositivo di cui sopra può funzionare aggiornando la posizione in tre modalità differenti:

- a)
- una
- volta al secondo;      b)
- cinque
- volte al secondo;      c)
- dieci
- volte al secondo.

**[5.A]** valutare se la linea di trasmissione dati (la sua velocità) è adeguata a ciascuna delle tre modalità.**[5.B]** valutare il *gap* temporale fra una spedizione e l'altra durante la comunicazione fra il dispositivo e l'elaboratore nella modalità "a".

Cognome e nome: \_\_\_\_\_ Matricola: \_\_\_\_\_ Posto: \_\_\_\_\_

## Soluzione

### Soluzione al Quesito 1

[1.A]: risposta 4

[1.B]: risposta 4

[1.C]: risposta 2

[1.D]: risposta 3

### Soluzione al Quesito 2

[2.A]: Ci sono tre liste:

A3(33,1) - A1(40,1) - A0(48,1)

C2(15,1) - C5(80,1)

D8(27,1)

[2.B]: Le liste son diventate quattro:

A3(33,1) - A1(40,0) - A0(48,1)

B3(92,0)

C2(15,1) - C5(80,1) - C9(90,0)

D8(27,0)

[2.C]: cambia solo la lista del processo A

A0(48,1) - A3(33,0) - A9(105,1)

B3(92,0)

C2(15,1) - C5(80,1) - C9(90,0)

D8(27,0)

### Soluzione al Quesito 3

[3.A] Una versione corretta dei tre processi è quella che segue.

Process A {	Process B {	Process C {
P(semB);	V(semB);	P(semA);
V(semA);		
P(mutex);	P(mutex);	P(mutex);
y = y + 2*x	x = x + 1	x = y*3;
V(mutex);	V(mutex);	V(mutex);
V(semA);	P(semA);	P(semC);
	V(semC);	
	Print(x);	Print(y);
}	}	}

[3.B] I tre processi potrebbero entrare e uscire dalla sezione critica limitata da P(mutex) e V(mutex) in un ordine qualsiasi dunque per determinare il valore finale di x e y occorre provare tutte le combinazioni.

- Ordine A, B, C: risultato x = 15 e y = 5
- Ordine A, C, B: risultato x = 16 e y = 5
- Ordine B, A, C: risultato x = 21 e y = 7
- Ordine B, C, A: risultato x = 3 e y = 7
- Ordine C, A, B: risultato x = 4 e y = 7
- Ordine C, B, A: risultato x = 4 e y = 9

Cognome e nome: \_\_\_\_\_ Matricola: \_\_\_\_\_ Posto: \_\_\_\_\_

**Soluzione al Quesito 4**Politica di rimpiazzo **LRU**; totale *page fault*? **10** (quelli in grassetto)

r1	r2	r3	r4	r2	r1	r5	r6	r2	r1	r2	r3	r7	r6	r3	r2	r1	r2	r3	r6
<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	2	1	<b>5</b>	<b>6</b>	2	1	2	<b>3</b>	<b>7</b>	<b>6</b>	3	2	<b>1</b>	2	3	<b>6</b>
	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3
		1	2	3	4	2	1	5	6	6	1	2	3	7	6	3	3	1	2
			1	1	3	4	2	1	5	5	6	1	2	2	7	6	6	6	1

Politica di rimpiazzo **Optimal**; totale *page fault*? **8** (quelli in grassetto)

r1	r2	r3	r4	r2	r1	r5	r6	r2	r1	r2	r3	r7	r6	r3	r2	r1	r2	r3	r6
<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	4	4	<b>5</b>	<b>6</b>	6	6	6	6	<b>7</b>	7	7	7	<b>1</b>	1	1	1
	1	2	3	3	3	3	3	3	3	3	3	6	6	6	6	6	6	6	6
		1	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3
			1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2

**Soluzione al Quesito 5**

Sì ne risente in quanto l'attributo "dati" di NTFS contiene tante coppie valori quante sono le sequenze contigue su disco. Di conseguenza un file enorme contiguamente scritto su disco richiederà unicamente una coppia valori, mentre un file piccolo composto da N blocchi nessuno contiguo a nessun altro richiederà in MFT un numero N di coppie valori.

**Soluzione al Quesito 6**

[5.A] Ad ogni rilevamento di posizione i dati da trasferire sono al più  $82 \times 3 = 246$  byte.

Con il protocollo 8N1 ciò significa  $246 \times (1 + 8 + 1) \text{ bit} = 2460 \text{ bit}$ .

Nel caso a), ciò corrisponde a  $2460 \times 1 = 2460$  baud necessari  $< 19200$ , e il collegamento seriale è adeguato.

Nel caso b), ciò corrisponde a  $2460 \times 5 = 12300$  baud necessari  $< 19200$ , e il collegamento seriale è ancora adeguato.

Nel caso c), ciò corrisponde a  $2460 \times 10 = 24600$  baud necessari  $> 19200$ , e il collegamento seriale non è più adeguato.

[5.B] La comunicazione avverrà dunque a 19.200 baud (attenzione, non a 2460!), e il tempo per trasmettere 2460 bit è di 128 ms. Pertanto fra una comunicazione e l'altra il gap sarà di  $1000 \text{ ms} - 128 \text{ ms} = 872 \text{ ms}$  (circa)