

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

Università degli Studi di Padova - Facoltà di Scienze MM.FF.NN. - Corso di Laurea in Informatica

**Regole dell'esame**

Il presente esame scritto deve essere svolto in forma individuale in un tempo massimo di 2 ore 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. La correzione e la sessione orale avverrà in data e ora comunicate dal docente durante la prova scritta; i risultati saranno esposti sul sito del docente entro il giorno precedente gli orali. Per superare l'esame il candidato deve 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. Riportare generalità e matricola negli spazi indicati. 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] In un confronto prestazionale tra *hard link* (HL) e *symbolic link* (SL):

1. gli HL sono da preferire perchè velocizzano gli accessi ai *file*
2. gli SL sono da preferire perchè assicurano la singolarità dell'associazione tra *directory* e *i-node*
3. i due sono sostanzialmente indistinguibili
4. gli SL hanno prestazioni superiori perchè impiegano meno spazio nella *directory*.

[1.B]: Sia dato un sistema di memoria con indirizzi virtuali suddivisi in 4 campi: *a, b, c, d*, i primi 3 dei quali siano utilizzati per indirizzare tre livelli gerarchici di tabelle delle pagine e il quarto campo rappresenti l'*offset* entro la pagina selezionata. Indicare dall'ampiezza di quali campi dipende il numero di pagine indirizzate nel sistema:

1. da quella di tutti e quattro i campi
2. da quella del campo *d*
3. da quella del campo *a* e *d*
4. da quelle dei campi *a, b, c*.

[1.C]: Un semaforo binario può:

1. assumere solo valori discreti
2. gestire solo l'accesso a due risorse condivise
3. gestire solo le richieste di accesso provenienti da due processi
4. assumere solo i valori 0 e 1, con essi denotando "risorsa occupata" e "risorsa libera".

[1.D] Quale tra le seguenti affermazioni, fatte osservando un grafo di allocazione delle risorse, è certamente vera in generale:

1. se vi sono percorsi chiusi allora vi è situazione di stallo
2. se non vi sono percorsi chiusi allora non vi è situazione di stallo
3. se in un percorso chiuso rilevato si trovano solo risorse a molteplicità unaria, occorre analizzare il caso per decidere
4. nessuna delle precedenti tre possibili risposte.

**RISPOSTE AL QUESITO 1:**

**A** \_\_\_\_\_ **B** \_\_\_\_\_ **C** \_\_\_\_\_ **D** \_\_\_\_\_

**Quesito 2 – (6 punti):** Cinque processi *batch*, identificati dalle lettere A, B, C, D, E rispettivamente, arrivano all'elaboratore agli istanti 0, 1, 2, 6, 7 rispettivamente. Tali processi hanno un tempo di esecuzione stimato di 3, 7, 2, 3, 1 unità di tempo rispettivamente e con priorità 3, 5, 2, 4, 1 rispettivamente (dove 5 è la massima priorità e 0 è la minima). Per ognuna delle seguenti politiche di ordinamento:

1. Round Robin (divisione di tempo, con priorità, senza preilascio, e con quanto di tempo di ampiezza 2)
2. Fair Priority Scheduling

Per evitare attesa infinita la politica di *Fair Priority Scheduling* prevede che, a seguito di due unità di tempo consecutive di esecuzione, la priorità del processo in esecuzione scenda di un punto.

(Esempio 1. Se il processo è in esecuzione per 4 unità di tempo consecutive, la priorità di tale processo scende di 1 punto dopo le prime due unità temporali e di 1 altro punto dopo le ultime due unità temporali.)

(Esempio 2. Se un processo è in esecuzione per 3 unità di tempo consecutivamente, la priorità di tale processo scende di 1 punto dopo le prime due unità temporali e basta; l'altra unità temporale di esecuzione non concorre in alcun modo, nemmeno successivamente, a far decrementare la priorità del processo in considerazione.)

Infine, la priorità di un processo non risale mai e non può scendere sotto lo zero.

Determinare, trascurando i ritardi dovuti allo scambio di contesto: (i) il tempo medio di risposta; (ii) il tempo medio di attesa; (iii) il tempo medio di *turn around*.



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

**Quesito 3 – (10 punti):**

**Lo studente risolve, a scelta, UNO SOLO fra i quesiti 3.1 e 3.2**

**Quesito 3.1 (alternativa):**

Il problema dei “lettori e scrittori” è un classico problema di sincronizzazione tra più processi (*reader* e *writer*) che accedono concorrentemente a una risorsa condivisa (un database).

Lo studente descriva brevemente il problema e poi utilizzi i semafori per scrivere una procedura *Reader* e una procedura *Writer* che cerchino di leggere/scrivere dal/sul database. Tali procedure dovranno poter essere eseguite concorrentemente evitando il *deadlock* del sistema.

Le funzioni `read_data_base()` e `write_data_base()` possono essere assunte come già realizzate pronte ad essere richiamate nel codice di *Reader* e *Writer*.

Nota: lo studente si ricordi di dichiarare e inizializzare i valori delle variabili semaforo usate nella sua soluzione.

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

**Quesito 3.2 (alternativa):**

Un sistema di controllo di una cisterna misura la quantità d'acqua contenuta nella stessa misurandone l'altezza ogni secondo facendo uso di un apposito sensore. Grazie a questa misura una centralina prende le dovute decisioni di svuotamento o riempimento. Il sensore comunica con la centralina per mezzo di una linea seriale asincrona; il sensore invia ogni misura corredandola di alcuni dati come segue:

2 byte per il proprio identificativo univoco;

4 byte per contenere un timestamp;

2 byte per la misura vera e propria;

1 byte contenente un dato ausiliario (crc).

Considerando che:

la misura viene effettuata e spedita alla centralina ogni secondo; l'invio consiste nella spedizione da parte del sensore alla centralina del pacchetto di dati come sopra descritto; la linea seriale asincrona è configurata per funzionare ad una velocità di 1200 baud, "8N1" (8 bit di dati, nessuna parità, un solo bit di stop);

**a)** Calcolare la percentuale di utilizzo della linea (tale percentuale è il rapporto fra la banda realmente usata e la velocità massima permessa).

Considerando ora che:

la centralina deve loggare le misure su una memoria persistente per un intero mese (30 gg nominali per i calcoli); la centralina bufferizza i dati ricevuti e li scrive tal quali (=in formato grezzo) all'interno di un file binario, registrandolo su base oraria (ogni ora cambia file);

i files vengono registrati su un filesystem basato su i-node, blocchi da 1K, assumendo i-node ampi 128 B, record da 32 bit, i-node principale contenente 12 indici di blocco e 1 indice di I, II e III indizione ciascuno.

**b)** (4 punti) Calcolare il rapporto inflattivo nel caso in cui i files siano lasciati come files singoli contenenti i dati orari (pertanto in tutto:  $24 \times 30 = 720$  files)

**c)** (3 punti) diversamente dal caso b) si consideri il caso in cui, alla fine dell'anno, tali files vengano accorpati in un unico file binario (accodando pertanto i singoli files in uno solo).

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

**Quesito 4 (6 punti):**

In una chiavetta USB da 2 GB con filesystem FAT32 e blocchi da 4K vengono registrati 100 files da 1000 blocchi ciascuno.

Considerando che:

il tempo di lettura di un blocco è 100 microsecondi;

il tempo medio di accesso (tempo per raggiungere qualsiasi blocco da un altro qualsiasi, purché non contiguo altrimenti il tempo è nullo) è di 5 millisecondi;

si presuma che la FAT sia precaricata in memoria (si trascurino pertanto i tempi di accesso alla stessa).

Calcolare il tempo necessario a leggere tutti i files, nelle ipotesi

**a)** di minima frammentazione;

**b)** di massima frammentazione;

Si calcoli inoltre

**c)** il tempo di scanning dello spazio libero, nel caso a)

**Quesito 5 – (4 punti):**

Un sistema di allocazione della memoria ha le seguenti pagine libere, in questo ordine:  
8KB, 15KB, 3KB, 11KB, 5KB, 7KB, 20KB, 25KB.

Si considerino tre richieste di allocazione  
che arrivano, una di seguito all'altra, nel seguente ordine:

A) 11K;

B) 4K;

C) 13K.

Indicare a quali pagine vengono assegnate le tre richieste sequenziali A, B e C considerando le politiche *First Fit*, *Next Fit*, *Best Fit* e *Worst Fit*.

Nota: si assuma che, qualora un blocco libero venga assegnato a seguito di una richiesta di dimensione inferiore, il blocco libero sia comunque interamente assegnato.

	A)	B)	C)
<i>First Fit</i>			
<i>Next Fit</i>			
<i>Best Fit</i>			
<i>Worst Fit</i>			

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

**Soluzione****Soluzione al Quesito 1**

[1.A]: risposta 1

[1.B]: risposta 4

[1.C]: risposta 4

[1.D]: risposta 2

**Soluzione al Quesito 2**

a) • RR (divisione di tempo, senza priorità e con quanto di tempo di ampiezza 2)

processo A	AAAAAAAAAAAA	LEGENDA DEI SIMBOLI
processo B	-BBBBBBBB	- non ancora arrivato
processo C	--cccccccccc	x (minuscolo) attesa
processo D	-----dddDDD	X (maiuscolo) esecuzione
processo E	-----eeeeeeeeE	. coda vuota
CPU	AABBBBBBDDDACCE	
coda	.baaaadddaacee. ..ccccaaacce... .....cccee..... .....ee.....	

processo	risposta	tempo di attesa	turn-around
A	0	10	10 + 3 = 13
B	1	1	1 + 7 = 8
C	11	11	11 + 2 = 13
D	3	3	3 + 3 = 6
E	8	8	8 + 1 = 9
<b>medie</b>	<b>4,60</b>	<b>6,60</b>	<b>9,80</b>

b) • Priority Scheduling (con valori di priorità espliciti e con prerilascio)

processo A	AAAAAAAAAA	LEGENDA DEI SIMBOLI
processo B	-BBBBBbbbbbb	- non ancora arrivato
processo C	--cccccccccc	x (minuscolo) attesa
processo D	-----DDD	X (maiuscolo) esecuzione
processo E	-----eeeeeeeeE	. coda vuota
CPU	ABBBBDDDAABBCCE	
coda	.aaaaaaabbccee. ..cccbbbccee... .....cccee..... .....ee.....	

Nota:

- All'istante 3, il valore di priorità di B passa da 5 a 4.
- All'istante 5, il valore di priorità di B passa da 4 a 3 (e viene dunque prerilasciato all'ingresso di D).
- All'istante 8, il valore di priorità di D passa da 4 a 3.

processo	risposta	tempo di attesa	turn-around
A	0	8	8 + 3 = 11
B	0	5	5 + 7 = 12
C	11	11	11 + 2 = 13
D	0	0	0 + 3 = 3
E	8	8	8 + 1 = 9
<b>medie</b>	<b>3,80</b>	<b>6,40</b>	<b>9,60</b>

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

**Soluzione al Quesito 3.1**

Si veda slide del corso sull'argomento, con esercizio svolto.

**Soluzione al Quesito 3.2**

**a)** Per ogni byte bisogna inviare 10 bit: 1 bit di start (sempre presente), 8 bit di dati (v. "8" N1), nessun bit di parità (v. 8" N"1), un (solo) bit di stop (v. 8N"1").

Ogni misura consiste di  $2+4+2+1 = 9$  byte = 90 bit da spedire. Ad una misura al secondo è pertanto necessaria una velocità di almeno 90 baud.

La linea è ben dimensionata essendo da 1200 baud; la percentuale di utilizzo è  $90 / 1200 = 7,5 \%$

**b)** Un file contenente un'ora di dati grezzi occupa  $9 \times 3600 = 32400$  byte;

$32400 \text{ byte} / 1024 = 31,6k = 32$  blocchi da 1K;

Pertanto ogni file per la sua mappatura richiede l'i-node principale impegnando i 12 indici di blocco diretti e solo l'i-node di I indirezione dato che sono necessari 20 blocchi < 32 blocchi max. ( $32 = 128 \text{ B}$  ampiezza dell'i-node /  $4 \text{ B}$  ampiezza del record).

Il rapporto inflattivo globale è definito dal rapporto fra lo spazio richiesto dagli i-node di mappatura e lo spazio utilizzato dai dati:

$$\frac{2 \text{ i-node/file} \times 720 \text{ file} \times 128 \text{ B / i-node}}{32 \text{ blocchi / file} \times 1 \text{ K / blocco} \times 720 \text{ file}} = 0,78\%$$

**c)** il file globale occupa  $24 \times 30 \times 32400 \text{ B} = 23.328.000 \text{ B} = 22782$  blocchi da 1 K.

Pertanto per descriverlo sono necessari:

1 i-node principale con i 12 blocchi diretti: rimangono 22770 blocchi;

1 i-node I indirezione con i suoi 32 blocchi: rimangono 22738 blocchi;

1 i-node II indirezione più 32 i-node da esso mappati, per un totale di 1024 blocchi: rimangono 21714 blocchi;

1 i-node III indirezione più 21 i-node II livello completi ( $21 \times 32 \times 32 = 21504$  blocchi): rimangono 210 blocchi;

si utilizza il 22° i-node della III indirezione, 6 suoi i-node completi (puntano 32 blocchi) e uno parziale (punta 18 blocchi):  $6 \times 32 + 18 = 210$  blocchi

In totale pertanto si sono utilizzati  $1 + 1 + (1+32) + (1+22+21 \times 32 + 1 \times 7) = 737$  i-node

Il rapporto inflattivo è del 4 per mille:  $737 \times 128 \text{ B} /$  dimensione del file.

Esso è circa metà del precedente.

**Soluzione al Quesito 4**

**a)** nel caso di minima frammentazione, tutti i blocchi dei files sono contigui e disposti uno di seguito all'altro. Si tratta perciò di leggere  $100 \times 1000 = 100.000$  blocchi  $\times 100$  microsecondi = 10 secondi.

**b)** nel caso di massima frammentazione, dato che ci sono in tutto  $2^{31} / 2^{12}$  blocchi = 524.288 blocchi, è possibile considerare che fra ogni blocco di dati ce ne sia uno libero. Pertanto i files sono tutti "polverizzati". Ad ogni lettura va sommato quindi un "salto" (perché non si tratta di blocchi contigui):

tempo totale = 10 secondi (dal punto precedente) +  $(100000-1) \times 5 \text{ ms} = 8$  minuti e 30 secondi.

**c)** i blocchi liberi (escludendo lo spazio per la FAT stessa) sono  $524288 - 100000 = 424288$ , tutti contigui; pertanto si tratta di circa 42 secondi.

**Soluzione al Quesito 5**

	A)	B)	C)
<i>First Fit</i>	15KB	8KB	20KB
<i>Next Fit</i>	15KB	11KB	20KB
<i>Best Fit</i>	11KB	5KB	15KB
<i>Worst Fit</i>	25KB	20KB	15KB