

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 90 min 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.

I punti assegnati a ciascun esercizio sono indicativi, il docente si riserva di modificarli (lievemente) in sede di correzione. Per la convalida e registrazione del voto finale il docente si riserva di proporre al singolo candidato una prova orale.

Quesito 1 (7 punti): 0,5 punti per risposta giusta, diminuzione di 0,25 punti per ogni sbaglio, 0 punti per risposta vuota

DOMANDA	Vero/Falso
In un sistema di memoria a paginazione, il <i>Translation Lookaside Buffer</i> (TLB) velocizza la traduzione di indirizzi fisici in indirizzi virtuali	
La gestione della memoria con segmentazione può ridurre il consumo di memoria, in quanto consente a più processi di condividere blocchi di codice e di dati	
Nella gestione della memoria con paginazione, il fenomeno della frammentazione interna è tanto meno rilevante quanto più la lunghezza media dei programmi è grande rispetto alla dimensione della pagina	
Il meccanismo dei semafori consente forme più generali di sincronizzazione tra processi rispetto alla mutua esclusione	
l'attesa indefinita (<i>starvation</i>) è un caso particolare dello stallo	
In riferimento allo <i>scheduling</i> di processi, la politica FIFO non causa mai meno <i>page fault</i> della politica LRU	
Una <i>system call</i> generata da un processo utente viene gestita in modalità utente	
Lo <i>switch</i> di un processo utente avviene sempre contestualmente a 2 <i>mode switch</i> (utente->kernel, kernel->utente)	
Un <i>interrupt</i> viene gestito in modalità <i>kernel</i>	
Il <i>process switch</i> può avvenire sia in modalità <i>kernel</i> che in modalità utente	
Il termine <i>copy-on-write</i> indica il caso in cui ad ogni modifica di una variabile in memoria RAM si procede immediatamente con l'aggiornamento anche della sua copia su partizione di disco	
Ogni <i>interrupt</i> può essere associato ad un processo che ha richiesto una operazione di I/O	
Molti <i>page fault</i> su un processo non modificano le prestazioni degli altri processi	
Un processo per lanciare un nuovo processo deve fare una <i>system call</i>	

Quesito 2 – (6 punti):

Si consideri l'algoritmo AGING di *page replacement* con contatore (o stimatore) di 3 bit, e una memoria di 4 *frames* contenenti rispettivamente le pagine 1 2 3 e 4 di un certo processo. Si supponga che subito dopo uno *sweep* (aggiornamento del contatore) all'istante t_0 i contatori siano inizializzati come segue:

contatore pagina 1: **110** contatore pagina 2: **111** contatore pagina 3: **101** contatore pagina 4: **100**

All'istante t_1 avviene uno *sweep*. Tra t_0 e t_1 è stata eseguita la seguente sequenza di accessi a memoria, nell'ordine: pagina 4, pagina 2; pagina 4, pagina 2.

[2.A] Che valore avranno i contatori dopo lo *sweep* in t_1 ?

contatore pagina 1: _____ contatore pagina 2: _____ contatore pagina 3: _____ contatore pagina 4: _____

[2.B] Supponendo invece che subito dopo t_0 fosse avvenuto un *page fault*, di quale pagina avrebbe causato la sostituzione? Perché?

Cognome e nome: _____ Matricola: _____ Posto: _____

Quesito 3 – (8 punti):

Sia data una partizione di disco ampia 256 GB organizzata in blocchi dati di ampiezza 1 KB. Sotto queste ipotesi si determini l'ampiezza massima di file ottenibile per l'architettura di file system ext2fs nel caso pessimo di contiguità nulla, assumendo i-node ampi 512 B, i-node principale contenente 13 indici di blocco e 1 indice di I, II e III indizione ciascuno. Si determini poi il rapporto inflattivo che ne risulta, ossia l'onere proporzionale dovuto alla memorizzazione delle strutture di rappresentazione rispetto a quella dei dati veri e propri.

Effettuati tali calcoli si discuta se e con quale rapporto inflattivo le architetture FAT e NTFS rispettivamente possano rappresentare file di tale ampiezza nella partizione data, sotto le medesime ipotesi di contiguità nulla. Per l'architettura NTFS si assumano record ampi 512 B, 208 B riservati all'attributo dati nel record principale e 400 B nei record di estensione.

Cognome e nome: _____ Matricola: _____ Posto: _____

Quesito 4 – (2 punti):

Si consideri la politica di *scheduling Round Robin* di quanto q e si supponga che in un sistema ci siano N processi interattivi tutti con lo stesso comportamento. Ciascuna interazione dà luogo ad un CPU *burst* che richiede la CPU per un tempo c . Se $c < q$ quanto tempo aspetta al più l'utente prima che la CPU finisca di elaborare l'interazione?

Quesito 5 – (2 punti):

Si elenchino, senza spiegarle, le condizioni relative al verificarsi dello stallo (*deadlock*).

Quesito 6 – (5 punti):

Supponiamo di avere 3 processi che condividono una variabile x e che i loro pseudo-codici siano i seguenti:

P1:	P2:	P3:
<pre> P (SemA) P (Mutex) x=x-2 V (Mutex) V (SemC) P (SemA) P (Mutex) x=x-1 V (Mutex) V (SemC) </pre>	<pre> P (SemB) P (Mutex) x=x+2 V (Mutex) V (SemC) P (SemB) </pre>	<pre> P (SemC) P (Mutex) if (x<0) then V (SemB) V (Mutex) P (SemC) print (x) </pre>

Si assuma che il valore iniziale di x sia 0 e che i tre semafori abbiano i seguenti valori iniziali:

SemA = 1, SemB = 0, SemC = 0, Mutex = 1.

Si discuta l'ordine di esecuzione dei vari processi (e delle loro istruzioni) e si determini l'output `print (x)` del processo **P3**

Cognome e nome: _____ Matricola: _____ Posto: _____

Soluzione**Soluzione al Quesito 1**

DOMANDA	Vero/Falso
In un sistema di memoria a paginazione, il <i>Translation Lookaside Buffer</i> (TLB) velocizza la traduzione di indirizzi fisici in indirizzi virtuali	F
La gestione della memoria con segmentazione può ridurre il consumo di memoria, in quanto consente a più processi di condividere blocchi di codice e di dati	V
Nella gestione della memoria con paginazione, il fenomeno della frammentazione interna è tanto meno rilevante quanto più la lunghezza media dei programmi è grande rispetto alla dimensione della pagina	V
Il meccanismo dei semafori consente forme più generali di sincronizzazione tra processi rispetto alla mutua esclusione	V
L'attesa indefinita (<i>starvation</i>) è un caso particolare dello stallo	F
In riferimento allo <i>scheduling</i> di processi, la politica FIFO non causa mai meno <i>page fault</i> della politica LRU	F
Una <i>system call</i> generata da un processo utente viene gestita in modalità utente	F
Lo <i>switch</i> di un processo utente avviene sempre contestualmente a 2 <i>mode switch</i> (utente->kernel, kernel->utente)	V
Un interrupt viene gestito in modalità <i>kernel</i>	V
Il <i>process switch</i> può avvenire sia in modalità <i>kernel</i> che in modalità utente	F
Il termine <i>copy-on-write</i> indica il caso in cui ad ogni modifica di una variabile in memoria RAM si procede immediatamente con l'aggiornamento anche della sua copia su partizione di disco	F
Ogni interrupt può essere associato ad un processo che ha richiesto una operazione di I/O	F
Molti <i>page fault</i> su un processo non modificano le prestazioni degli altri processi	F
Un processo per lanciare un nuovo processo deve fare una <i>system call</i>	V

Soluzione al Quesito 2[2.A] contatore pagina 1: 011 contatore pagina 2: 111 contatore pagina 3: 010 contatore pagina 4: 110

[2.B] Sostituirebbe la pagina 4 perché ha il valore di contatore più basso fra tutti.

Soluzione al Quesito 3

In questa soluzione useremo la notazione informatica tradizionale, con prefissi che denotano potenze di 2.

Essendo la memoria secondaria ampia 256 GB e i blocchi dati ampi 1 KB, è immediato calcolare

che sono necessari: $\left\lceil \frac{256GB}{1KB} \right\rceil = 256 M = 2^8 \times 2^{20} = 2^{28}$ indici, la cui rappresentazione binaria banalmente richiede 28 bit.

Stante l'ovvio vincolo che la dimensione dell'indice debba essere un multiplo di un "ottetto" (8 bit), otteniamo la dimensione di 32 bit (4 B).

File system di tipo ext2fs:

Sotto queste ipotesi, il file di massima dimensione rappresentabile dall'architettura ext2fs fissata dal quesito sarà composto da:

- 13 blocchi, risultanti dall'utilizzo dei corrispondenti indici diretti presenti nell'i-node principale, al costo di 1 i-node, pari a 512 B
- $\left\lceil \frac{512B}{4B} \right\rceil = 128$ blocchi, risultanti dall'utilizzo dell'intero i-node secondario denotato dall'indice di I indirezione presente nell'i-node principale, al costo di 1 i-node, pari a 512 B
- $128^2 = 2^{14} = 16 K$ blocchi, risultanti dall'utilizzo dell'indice di II indirezione, al costo di $1 + 128 = 129$ i-node, pari a: $129 \times 512B = (4.096 + 128)B = (2^{16} + 512) B = 66.048 B$
- $128^3 = 2^{21} = 2 M$ blocchi, risultanti dall'utilizzo dell'indice di III indirezione, al costo di $1 + 128 + 128^2 = 16.513$ i-node, pari a: $16.513 \times 512 B = 8.454.656 B$

corrispondenti a $13 + 128 + 16.384 + 2.097.152 = 2.113.677$ blocchi ampi 1 KB, al costo complessivo di $1 + 1 + 129 + 16.513 = 16.644$ i-nodei-node ampi 128 B, per un rapporto inflattivo di: $\frac{16.644 \times 512 B}{2.113.677 \times 1 KB} = \frac{16.644}{2.113.677 \times 2} = 0,39\%$.

Cognome e nome: _____ **Matricola:** _____ **Posto:** _____

Vediamo ora di determinare se e in che modo le architetture di file system FAT e NTFS siano in grado di rappresentare file di tale ampiezza sotto le ipotesi fissate dal quesito.

File system di tipo FAT:

La struttura FAT, che rappresenta la vista dell'intera partizione in termini di blocchi dati, sarà composta da $\left\lceil \frac{256GB}{1KB} \right\rceil = 256 \text{ M}$

celle ampie 4 B, una per indice di blocco: di queste, il file che dobbiamo rappresentare ne occuperà 2.113.677, per un rapporto inflattivo — calcolato considerando che l'architettura FAT concettualmente usa l'intera struttura per ogni singolo file — pari a:

$$\frac{256 \text{ M} \times 4 \text{ B}}{2.113.677 \text{ KB}} = \frac{1 \text{ GB}}{2.113.677 \text{ KB}} = \frac{1.048.576 \text{ KB}}{2.113.677 \text{ KB}} = 49,61\%, \text{ un onere piuttosto imponente.}$$

Nota: anche il caso 28 bit (anziché 32 bit = 4B è stata considerata corretta per quanto discusso in aula sulla FAT-32).

File system di tipo NTFS:

Dei 208 B riservati all'attributo dati nel record principale, $2 \times 4 \text{ B} = 8 \text{ B}$ saranno riservati alla coppia {base, indice}, mentre i rimanenti $208 - 8 = 200 \text{ B}$ potranno essere utilizzati per denotare le sequenze contigue che, sotto le ipotesi di contiguità nulla fissate del quesito, sono tutte ampie 1 blocco. Poiché ciascuna sequenza di tipo {inizio, fine} richiede 8 B, il record principale

potrà ospitare: $\left\lfloor \frac{200B}{8B} \right\rfloor = 25$, mentre un singolo *record* di estensione dispone di 400 B per la memorizzazione di $\left\lfloor \frac{400B}{8B} \right\rfloor = 50$

ulteriori sequenze. Ne segue che, per rappresentare un file dell'ampiezza data, l'architettura NTFS necessiterà, in prima approssimazione, di: $1 + \left\lceil \frac{2.113.677 - 25 \text{ blocchi}}{50 \text{ blocchi/record}} \right\rceil = 1 + \left\lceil \frac{2.113.652}{50} \right\rceil = 1 + 42.274 = 42.275 \text{ record}.$

In conclusione, un rapporto inflattivo pari a: $\frac{42.275 \times 512B}{2.113.677 \text{ KB}} = 1 \text{ \%}.$

Soluzione al Quesito 4

L'utente aspetta $(N-1)c + c = Nc$

Soluzione al Quesito 5

Le quattro condizioni sono mutua esclusione, assenza di preilascio, accumulo di risorse, attesa circolare

Soluzione al Quesito 6

Riguardo all'ordine di esecuzione, prima parte P1, l'unico dei tre processi che non si blocca subito a un semaforo, ed esegue

P (SemA)

P (Mutex)

x=x-2 \\ il valore di x diventa dunque -2

V (Mutex)

V (SemC)

A questo punto, P1 si blocca su P (SemA) ma ha già risvegliato P3 tramite V (SemC); P3 quindi esegue

P (SemC)

P (Mutex)

if (x<0) then V (SemB) \\ essendo x == -2 la condizione è soddisfatta

V (Mutex)

A questo punto P3 si blocca su P (SemC) ma ha già risvegliato P2 tramite V (SemB); P2 quindi esegue

P (SemB)

P (Mutex)

x=x+2 \\ il valore di x torna dunque 0

V (Mutex)

V (SemC)

A questo punto P2 si blocca su P (SemB) ma ha già risvegliato P3 tramite V (SemC); P3 quindi esegue

P (SemC)

print (x)

L'ultima istruzione, print (x), stampa il valore finale di x che risulta quindi essere 0