

Cognome e nome: _____ Matricola: _____ Posto: _____

Università degli Studi di Padova - Corso di Laurea in Informatica

Regole dell'esame

Il presente esame scritto deve essere svolto in forma individuale in un tempo massimo di 45 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 avverrà in data e ora comunicate dal docente; i risultati saranno esposti sul sito del docente.

Quesito 0: Riportare generalità e matricola negli spazi indicati in alto e inserisca le proprie risposte interamente su questi fogli.

Quesito 1: (1 pt per ogni risposta corretta; -0,5 pt per ogni risposta sbagliata) (nessun minimo punteggio richiesto)

DOMANDA	Vero/Falso
In un sistema di memoria a paginazione, il <i>Translation Lookaside Buffer</i> (TLB) velocizza la traduzione di indirizzi logici in indirizzi fisici	
La gestione della memoria con paginazione consente a più processi di condividere pagine contenenti codice o dati	
<code>chmod</code> è un comando utilizzabile nei sistemi GNU/Linux per modificare i permessi di file e directory	
GNU/Linux tende a scrivere i file su disco come blocchi sequenziali contigui	
Con <code>ext2fs</code> è possibile che il file system scriva il contenuto (i dati) di file di piccola dimensione (es. <1KB) direttamente dentro l'i-node principale	
<code>rmdir</code> è un comando POSIX per eliminare directory	

Quesito 2:

Si consideri un sistema che utilizza la paginazione per gestire la memoria.

Si discutano sinteticamente vantaggi e svantaggi nell'adottare pagine di dimensione ampia oppure di piccola.

A) Pagine di dimensione ampia (vantaggi e svantaggi):

B) Pagine di dimensione piccola (vantaggi e svantaggi):

Come visto in classe, il valore ottimo di dimensione di una pagina può essere definito matematicamente.

C) Data la funzione $f(\pi) = (\sigma / \pi) \times \varepsilon + \pi / 2$ che definisce matematicamente lo spreco di memoria in maniera dipendente dalla dimensione π di una pagina, si indichi a quale tipo di spreco si riferisce ciascuno dei due addendi.

D) si determini quindi il valore ottimo di π che minimizza lo spreco di memoria (si mostrino i calcoli, ovviamente!).

Cognome e nome: _____ Matricola: _____ Posto: _____

Quesito 3:

[3.A] La dimensione massima di un file ottenibile con file system ext2fs dipende dalla contiguità con cui sono scritti i blocchi del file su disco? *Sì / No* ?

[3.B] La dimensione massima di un file ottenibile con file system FAT dipende dalla contiguità con cui sono scritti i blocchi del file su disco? *Sì / No* ?

[3.C] La dimensione massima di un file ottenibile con file system NTFS dipende dalla contiguità con cui sono scritti i blocchi del file su disco? *Sì / No* ?

[3.D] Sia data una partizione di disco ampia 64 GB organizzata in blocchi dati di ampiezza 1 KB. In caso serva, si consideri l'ipotesi di contiguità nulla di un file (ciascun blocco si trova su disco in posizione non adiacente al blocco precedente e a quello successivo nella composizione del file). Si consideri inoltre il minimo numero di bit per un indirizzo data la dimensione della partizione e l'ovvio vincolo che debba essere un multiplo di 8.

Si determini l'ampiezza massima di file ottenibile per l'architettura di file system ext2fs assumendo i-node ampi 128 B, i-node principale contenente 12 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 della struttura di rappresentazione rispetto a quella dei dati veri e propri.

Cognome e nome: _____ Matricola: _____ Posto: _____

Quesito 4:

Si consideri un sistema con paginazione della memoria virtuale, pagine di 128 Bytes e la seguente *page table* dove la riga più in alto corrisponde alla entry 0 e quella più in basso corrisponde alla entry 7

In/Out	Frame
In	00101
Out	01011
In	00001
Out	11010
In	00011
Out	10101
Out	11111
In	10101

Dire se i seguenti indirizzi logici genereranno un *page fault*. In caso negativo, scrivere l'indirizzo fisico corrispondente. (In caso la tabella delle pagine non sia sufficiente a rispondere in alcuni casi, lo si dichiara)

a) 0000001101001

b) 0000010010110

c) 0000100000101

Quesito 5:

Si consideri un sistema dotato di memoria virtuale, con memoria fisica divisa in 4 page frame. Il tempo di caricamento, tempo di ultimo accesso e i bit di R (*Referred*) e M (*Modified*) per ogni pagina sono come mostrato nella tabella sottostante.

pagina	tempo caricamento	ultimo riferimento	R	M
0	132	286	1	0
1	236	251	0	1
2	154	267	0	0
3	91	301	1	1

Si supponga che il sistema abbia bisogno di caricare in memoria una nuova pagina. Giustificando (molto brevemente) la risposta, quale di quelle in tabella sarà rimpiazzata se si adotta una politica:

a) NRU

b) FIFO

c) LRU

d) second chance

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 logici in indirizzi fisici	V
La gestione della memoria con paginazione consente a più processi di condividere pagine contenenti codice o dati	F
<code>chmod</code> è un comando utilizzabile nei sistemi GNU/Linux per modificare i permessi di file e directory	V
GNU/Linux tende a scrivere i file su disco come blocchi sequenziali contigui	V
Con ext2fs è possibile che il file system scriva il contenuto (i dati) di file di piccola dimensione (es. <1KB) direttamente dentro l'i-node principale	F
<code>rmdir</code> è un comando POSIX per eliminare directory vuote	V

Soluzione al Quesito 2**A) Pagine ampie**

- Maggiore rischio di **frammentazione interna** ma tabella delle pagine più piccola
 - In media ogni processo lascia inutilizzata metà del suo ultimo *page frame*

B) Pagine piccole

- Maggiore ampiezza della tabella delle pagine ma minor frammentazione interna

C) Spreco per processo come $f(\pi) = (\sigma / \pi) \times \varepsilon + \pi / 2$

- Dimensione della tabella delle pagine + frammentazione interna di una pagina e in particolare:
 - σ byte dimensione media di un processo
 - π byte dimensione media di una pagina
 - ε byte per riga in tabella delle pagine

D) Occorre calcolare la derivata prima: $-\sigma \varepsilon / \pi^2 + 1/2$

- Ponendo uguale a zero si ha che il minimo di $f(\pi)$ si ha per $\pi = \sqrt{2 \sigma \varepsilon}$
(in generale, per essere sicuri che si tratti del minimo e non del massimo occorrerebbe procedere al calcolo della derivata seconda, ma sappiamo già che si tratta del minimo)

Soluzione al Quesito 3**[3.A]** No**[3.B]** No**[3.C]** Sì**[3.D]** In questa soluzione useremo la notazione informatica tradizionale, con prefissi che denotano potenze di 2.

Essendo la memoria secondaria ampia 64 GB e i blocchi dati ampi 1 KB, è immediato calcolare

che sono necessari: $\left\lceil \frac{64GB}{1KB} \right\rceil = 64 \text{ M} = 2^6 \times 2^{20} = 2^{26}$ indici, la cui rappresentazione binaria banalmente richiede 26 *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).

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

- 12 blocchi, risultanti dall'utilizzo dei corrispondenti indici diretti presenti nell'i-node principale, al costo di 1 i-node, pari a 128 B

Cognome e nome: _____ Matricola: _____ Posto: _____

- $\left\lfloor \frac{128B}{4B} \right\rfloor = 32$ 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 128 B
- $32^2 = 2^{10} = 1$ K blocchi, risultanti dall'utilizzo dell'indice di II indirezione, al costo di $1 + 32 = 33$ i-node, pari a: $33 \times 128B = (4.096 + 128)B = 4$ KB + 128 B
- $32^3 = 2^{15} = 32$ K blocchi, risultanti dall'utilizzo dell'indice di III indirezione, al costo di $1 + 32 + 32^2 = 1.057$ i-node, pari a: 1.057×128 B

La dimensione massima dei dati di un file è dunque $12 + 32 + 1.024 + 32.768 = 33.836$ blocchi ampi 1 KB, al costo complessivo di $1 + 1 + 33 + 33 + 32^2 = 1.092$ i-node ampi 128 B, per un rapporto inflattivo di: $\frac{1.092 \times 128 B}{33.836 \times 1 KB} = \frac{1.092}{33.836 \times 8} \approx 0,40\%$.

Soluzione al Quesito 4

Vista la dimensione di pagina (2^7 Bytes) allora gli ultimi 7 bit sono l'offset all'interno della pagina mentre i precedenti indicano la pagina.

Dunque potremmo riscrivere gli indirizzi logici separando le due parti

a) 000000 1101001

b) 000001 0010110

c) 000010 0000101

Usando i primi sette bit come selettori nella tabella delle pagine otteniamo che

a) pagina 0 - pagina valida, (sostituisco prima parte con quanto dentro la tabella delle pagine) diventa 001011101001

b) pagina 1 è out e dunque *page fault*

c) pagina 2 - pagina valida, (sostituisco prima parte con quanto dentro la tabella delle pagine) diventa 000010000101

Soluzione al Quesito 5

- NRU rimuove ovvero la pagina 2 perché è l'unica che abbia $R = 0$ e $M = 0$.

- FIFO rimuove la prima pagina che è stata caricata, ovvero la pagina 3.

- LRU rimuove la pagina 1 perché è quella riferita meno di recente.

- second chance rimuove la pagina più vecchia tra quelle con $R = 0$, ovvero la pagina 2.