



Sistemi Operativi
Prova Scritta

Nome e Cognome:

Flavio Righi

Tempo a disposizione: 90 minuti / Documentazione: 1 foglio di appunti A5 manoscritto.

Vero o falso? [12 punti]

Domanda 1 Con il comando TRIM posso aumentare lo spazio libero a disposizione dell'utente

1/1

☐

Vero

☒

Falso

Domanda 2 In un harddisk tradizionale il numero di settori per ogni traccia è sempre uguale

0/1

☒

Vero

☒

Falso

Domanda 3 Il sistema operativo implementa una componente chiamata TLB per velocizzare la traduzione degli indirizzi virtuali

0/1

☒

Vero

☒

Falso

Domanda 4 La dimensione delle pagine è legata alla quantità di memoria fisica disponibile

0/1

☒

Vero

☒

Falso

Domanda 5 Il bit R nella tabella delle pagine viene messo a 1 dalla TLB

1/1

☐

Vero

☒

Falso

Domanda 6 Il journaling di tipo *ordered* garantisce la consistenza dei dati utente

1/1

☐

Vero

☒

Falso

Domanda 7 La tecnica dell'interleaving riduce il tempo di *seek*

0/1

☒

Vero

☒

Falso

Domanda 8 Il wear-out di una memoria SSD è aggravato se si utilizza un filesystem con journaling

0/1

☒

Vero

☒

Falso

Domanda 9 Ogni chiamata alla procedura **malloc** alloca una nuova pagina nella tabella del processo

0/1

☒

Vero

☒

Falso

Domanda 10 Il Copy-on-Write viene usato quando la dimensione dello spazio di indirizzamento virtuale è più grande dello spazio di indirizzamento fisico

1/1

☐

Vero

☒

Falso



Domanda 11 Le celle di memoria basate su tecnologia TLC possono distinguere tre diversi stati di carica

1/1



Vero



Falso

Domanda 12 Le tabelle delle pagine a più livelli richiedono tipicamente meno spazio rispetto a una tabella unica

1/1



Vero



Falso

Gestione della memoria [38 punti]

► **Domanda** Supponiamo di avere un sistema con 5 frame di pagina che utilizza l'**algoritmo della seconda opportunità**. Quando le pagine vengono caricate in memoria hanno il bit $R=0$. Supponiamo che il bit R venga messo a 0 solo dall'algoritmo. Vengono effettuati i seguenti riferimenti alle pagine:

9 1 8 7 0 2 11 7 13 13 1 7 1
PF PF PF PF PF PF PF PF PF PF PF PF

- Qual è il **numero totale di page fault**? [7 punti]

7/7



5



6



7



8



9

- Quali pagine si troveranno in memoria dopo l'ultimo riferimento? [3 punti]

3/3



[0, 5, 6, 7, 8]



[3, 6, 4, 2, 8]



[2, 11, 13, 7, 1]



[5, 2, 6, 4, 7]

► **Domanda** Supponiamo di avere un sistema che utilizza l'**algoritmo NRU**. Sono stati effettuati diversi accessi alle pagine, e la situazione attuale nella tabella delle pagine è:

Pagina	Caricata al tempo	R	M
0	145ms	1	1
1	200ms	0	0
2	80ms	0	1
3	32ms	1	0

- Qual è l'ordine di rimpiazzamento delle pagine (da quella che verrebbe rimossa per prima a quella che verrebbe rimossa per ultima)? [3 punti]

3/3



[3,2,0,1]



[1,2,3,0]



[0,1,2,3]



[1,3,2,0]

► **Domanda** Supponiamo di avere un sistema con 4 frame che utilizza l'**algoritmo LRU in hardware (con una matrice)**. Vengono effettuati i seguenti accessi ai frame:

2 2 1 0 1 3 2 0 1 3



- Qual è il contenuto finale della matrice ? [5 punti]

☐

0	1	1	1
0	0	1	1
0	0	0	1
0	0	0	0



0	0	1	0
1	0	1	0
0	0	0	0
1	1	1	1



0	1	1	1
0	0	1	1
0	0	0	1
0	1	0	0

- Quale frame verrebbe liberato per primo ? [2 punti]

☐ 0☐ 1☒ 2☐ 3

► **Domanda** Supponiamo di avere un sistema con 4 pagine in memoria che utilizza l'algoritmo **NFU (modificato con invecchiamento)**. All'inizio i contatori (di **lunghezza 4 bit**) sono tutti a **zero**. Al primo ciclo di clock i bit R sono 1101 (1 per la pagina 0, 1 per la pagina 1, 0 per la pagina 2 e 1 per la pagina 3). Successivamente i valori sono 0001, 0010, 0011, 0100, 1101, 0110 e infine 1011.

- Qual è il valore dei contatori dopo l'ultimo ciclo di clock ? [8 punti]



1	0	1	1
1	0	0	1
1	1	0	0
1	0	1	0



1	0	1	0
0	1	1	1
1	1	0	0
1	0	1	0



1	0	1	1
1	0	0	1
1	0	1	0
1	0	1	0

► **Domanda** Su computer che utilizza la segmentazione 8086 (Real Mode) scrivo il valore (esadecimale) 0xa2 nel registro selettore; successivamente accedo all'indirizzo virtuale (esadecimale) 0x400. Quale indirizzo fisico sto accedendo? [5 punti]

☒ 3616☒ 1696☐ 1024



► **Domanda** Quando faccio uso di memoria condivisa la gestione dei puntatori risulta più difficile. Per quale motivo? Quali alternative posso adottare? [5 punti]

☐ w ☐ p ☒ c

5/5

I puntatori potrebbero far riferimento a indirizzi di memoria uguali per tutti ma differenti dal pto di vista del singolo processo.
La soluzione è riferirsi ad un'offset rispetto all'inizio della struttura anziché a dei ptr.

Virtualizzazione [10 punti]

► **Domanda** Quali sono le differenze principali tra system level virtualization e application level virtualization? [5 punti]

☐ w ☒ p ☐ c

2.5/5

La prima simula un'intera ambiente di lavoro HW ospitando 1 o più sistemi operativi completi senza che essi si rendano conto di essere virtualizzati. (VMware - VirtualBox ecc.)

La seconda contiene delle ~~istruzioni~~ applicazioni le quali prima di arrivare al S.O vengono in qualche modo tradotte e/o manipolate? ad esempio la Java VM.



► **Domanda** A quale scopo sono state introdotte le estensioni per la virtualizzazione nei processori moderni? Fai degli esempi. [5 punti]

☐ w ☒ p ☐ c

2.5/5

Per permettere ai sistemi operativi guest di accedere a risorse normalmente protette e senza che l'Hypervisor debba mettere in atto meccanismi di ~~emulazione~~ emulazione delle ~~richieste~~ richieste per mantenendo l'hardware e il sistema operativo Host sicuro. In questo modo le performance del S.O. guest è maggiore.
 Ad esempio dando accesso diretto ~~alla~~ al MMU per tradurre le pagine e quindi senza ~~che~~ che l'Hypervisor debba intercettare ogni Page Fault o richieste di traduzione di indirizzi virtuali.

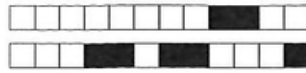
Filesystem [20 punti]

► **Domanda** Supponiamo di utilizzare un filesystem senza journaling. Cosa succederebbe se la bitmap degli blocchi liberi andasse persa in seguito ad un crash del sistema? Esiste un modo per ripristinare queste informazioni? Se sì, come funzionerebbe? [5 punti]

☐ w ☒ p ☐ c

2.5/5

Se il sistema è ext4 basato su inode, è possibile scorrere la tabella dei puntatori e per ogni spazio di ^{gli o blocchi} memoria indirizzato o meno ricreare la bitmap.
 Allo stesso modo per gli altri Filesystem facendo scorrere tutti i ptr al disco, i restanti saranno spazi di memoria liberi.



► **Domanda** Su un filesystem che **ext3** con *journaling* di tipo *full data*, cosa succederebbe se il sistema andasse in crash mentre viene scritto il *journal*? [5 punti] ☐ w ☒ p ☐ c

2.5/5

Full data tiene log sia di metadati che dei dati stessi, il comportamento in caso di ~~crash~~ crash dipende dal punto di avvenimento della scrittura del journal, ~~solo~~ solo dopo aver scritto il journal i dati vengono messi su disco. Se abbiamo loggato solo i metadati i dati vengono persi, se abbiamo loggato sia metadati che dati ma non è stato effettuato il commit al riavvio verrà ripetuta la procedura recuperando i dati del Journal.

→ cosa intendi?
scrittura su disco
dei dati

► **Domanda** Consideriamo un sistema che implementa un algoritmo di schedulazione del braccio di un disco con 10 cilindri (cilindri da 1 a 10). La coda delle richieste dell'algoritmo di schedulazione è limitata a un massimo di 3 cilindri (cioè decide sulla base di tre richieste future al massimo): non appena una richiesta viene soddisfatta viene tolta dalla coda e un'altra entra nella coda. Le richieste che arrivano al kernel coinvolgono, nell'ordine, i seguenti cilindri:

9 8 5 4 5 3 7 6 1 10

- Qual è l'ordine con cui queste richieste verranno servite se la testina si trova inizialmente sul cilindro 7 e l'algoritmo utilizzato è SSTF (*Shortest Seek Time First*)? Nota: in caso di pari distanza viene sempre scelto il cilindro con indice più piccolo [5 punti]

5/5

☒ [8,9,5,5,4,3,1,6,7,10] ☐ [8,9,4,5,5,6,7,10,1,3] ☐ [9,8,5,4,5,3,7,6,1,10]

- Qual è l'ordine con cui queste richieste verranno servite se la testina si trova inizialmente sul cilindro 3 e l'algoritmo utilizzato è C-SCAN (Circular SCAN), e il braccio si muove correntemente verso i cilindri con indice più grande? [5 punti]

5/5

☐ [8,9,4,5,5,6,7,10,1,3] ☐ [4,5,5,6,7,8,9,10,1,3] ☒ [5,8,9,3,4,5,6,7,10,1]