

Sistemi Operativi e Laboratorio, Prova del 31/5/2018

Nome: _____ Cognome: _____ Matricola: _____ fila: ___ posto: ___ corso: _____
 aula: _____

Esercizio 1 (5 punti)

In un file system simile a UNIX i blocchi del disco hanno ampiezza di 1 Kbyte e i puntatori ai blocchi sono a 16 bit. Gli i-node contengono, oltre agli altri attributi, 5 puntatori diretti e 3 puntatori indiretti (indiretto singolo, doppio e triplo). Il primo blocco del disco ha indice 0.

Si consideri il file aperto in scrittura individuato dal file descriptor *fd*, ed inizialmente vuoto, sul quale viene effettuata un'operazione di scrittura:

```
write(fd, &buf, 10.000)
```

con cui si scrivono 10.000 caratteri (ciascuno di un byte) sul file a partire dalla posizione 0. Si suppone che l'operazione termini con successo.

Nell'ipotesi che, al momento nel quale venga richiesta l'operazione di scrittura, i blocchi liberi nel disco siano (nell'ordine): 1000, 1001, 1002, 1003, 1004, 900, 901, 902, 903, 1100, 1101, 1102, 1103, ... e che i blocchi vengano allocati ai file attingendo a questa lista in ordine crescente, si chiede:

1. Su quali blocchi fisici è allocato il file dopo l'operazione di scrittura
2. Quale è il contenuto dell'i-node del file
3. Quali sono i contenuti degli eventuali blocchi indice (blocchi che contengono puntatori a blocchi)
4. Quanti byte del file sono contenuti nell'ultimo blocco fisico allocato al file

Soluzione

1. Il file è allocato sui blocchi 1000, 1001, 1002, 1003, 1004, 901, 902, 903, 1100, 1101
2. L'i-nodo del file è:

Puntatore	0	1	2	3	4	5	6	7
Valore del puntatore	1000	1001	1002	1003	1004	900	-	-

3. Il file ha bisogno di un solo blocco indice, il 900, che contiene:

Blocco 900:

Indice di elemento nel blocco	0	1	2	3	4	5
Valore del puntatore	901	902	903	1100	1101	-	-

4. L'ultimo blocco dati (il 1101), contiene $((10000) \bmod 1024)$ byte = 784 byte

Soluzione

1. Il file è allocato sui blocchi: _____
2. L'i-nodo del file è: _____

Puntatore	0	1	2	3	4	5	6	7
Valore del puntatore								

3. Il file ha bisogno di _____ blocchi indice: _____, che contengono:

Blocco _____ :

Indice di elemento nel blocco	0	1	2	3	4	5	
Valore del puntatore							

Blocco _____ :

Indice di elemento nel blocco	0	1	2	3	4	5	
Valore del puntatore							

Blocco _____ :

Sistemi Operativi e Laboratorio, Prova del 31/5/2018

Indice di elemento nel blocco	0	1	2	3	4	5	
Valore del puntatore							

4. L'ultimo blocco dati contiene _____ byte

Esercizio 2 (5 punti)

Un sistema operativo schedula i thread (implementati a livello del nucleo) con la tecnica delle MFQ (Multilevel Feedback Queue). La politica di scheduling prevede il prerilascio, che avviene immediatamente dopo l'evento che lo provoca, senza attendere l'esaurimento del quanto di tempo corrente. Quando un thread va in esecuzione gli viene assegnato un intero quanto di tempo, indipendentemente dal tempo consumato nel precedente turno di esecuzione. Ai thread è associato un livello di priorità compreso tra 0 e 7 (7 corrisponde alla massima priorità) e un quanto di tempo dipendente dalla priorità. In particolare, alle priorità da 0 a 3 (estremi inclusi) è associato un quanto di tempo di 10 ms, mentre alle priorità da 4 a 7 (estremi inclusi) è associato un quanto di tempo di 2 ms.

La MFQ è gestita con seguente politica:

- Alla generazione un thread acquisisce priorità 4.
- Se un thread esaurisce tutto il suo quanto di tempo la sua priorità si decrementa di 1.
- Se un thread va in stato di attesa prima della fine del quanto di tempo o se subisce un prerilascio la sua priorità viene incrementata di 1.

Nota: in ogni caso la priorità è limitata tra 0 e 7 e il decremento o l'incremento di priorità viene applicato immediatamente, quando si verifica l'evento che lo causa.

Al tempo T sono presenti il processo A (che al tempo T passa in esecuzione) e i processi B, C e D (che al tempo T sono in stato di pronto). Le priorità dei processi sono indicate negli schemi di soluzione tra parentesi.

Si chiede quale è il processo in esecuzione e la composizione delle code se si verificano (**in alternativa**) le seguenti sequenze di eventi (nella tabella accanto ad ogni processo è indicata la sua priorità tra parentesi):

Sequenza 1 (nota: i cambi di contesto dovuti all'esaurimento del quanto di tempo non sono mostrati in questa tabella. Vanno considerati per la soluzione ma in questo esercizio non è richiesto di mostrare cosa avviene in tal caso)

Tempo	Evento	In esecuzione	Code pronti	In attesa
T	A passa in esecuzione	A(7)	C(3), D(2)	B(4)
T+5	il thread in esecuzione si sospende			
T+6	il thread in esecuzione riattiva il thread B			
T+9	Il thread in esecuzione termina			
T+20	viene riattivato il thread A			

Sequenza 2 (nota: i cambi di contesto dovuti all'esaurimento del quanto di tempo non sono mostrati in questa tabella. Vanno considerati per la soluzione ma in questo esercizio non è richiesto di mostrare cosa avviene in tal caso)

Tempo	Evento	In esecuzione	Code pronti	In attesa
T	A passa in esecuzione	A(5)	C(3), D(2)	B(5)
T+12	il thread in esecuzione si sospende			
T+15	viene riattivato il thread B			
T+28	il thread in esecuzione si sospende			
T+29	viene riattivato il thread A			

Soluzione

Sequenza 1

Sistemi Operativi e Laboratorio, Prova del 31/5/2018

Tempo	Evento	In esecuzione	Code pronti	In attesa
T	A passa in esecuzione	A(7)	C(3), D(2)	B(4)
T+2	Scade quanto di tempo	A(6)	C(3), D(2)	B(4)
T+4	Scade quanto di tempo	A(5)	C(3), D(2)	B(4)
T+5	il thread in esecuzione si sospende	C(3)	D(2)	A(6), B(4)
T+6	il thread in esecuzione riattiva il thread B	B(4)	C(4), D(2)	A(6)
T+8	Scade il quanto di tempo	C(4)	B(3), D(2)	A(6)
T+9	Il thread in esecuzione termina	B(3)	D(2)	A(6)
T+19	scade il quanto di tempo	D(2)	B(2)	A(6)
T+20	viene riattivato il thread A	A(6)	D(3), B(2)	

Sequenza 2

Tempo	Evento	In esecuzione	Code pronti	In attesa
T	A passa in esecuzione	A(5)	C(3), D(2)	B(5)
T+2	scade il quanto di tempo	A(4)	C(3), D(2)	B(5)
T+4	scade il quanto di tempo	C(3)	A(3), D(2)	B(5)
T+12	il thread in esecuzione si sospende	A(3)	D(2)	B(5), C(4)
T+15	viene riattivato il thread B	B(5)	A(4), D(2)	C(4)
T+17	scade il quanto di tempo	A(4)	B(4), D(2)	C(4)
T+19	scade il quanto di tempo	B(4)	A(3), D(2)	C(4)
T+21	scade il quanto di tempo	A(3)	B(3), D(2)	C(4)
T+28	il thread in esecuzione si sospende	B(3)	D(2)	C(4), A(4)
T+29	viene riattivato il thread A	B(4)	A(4), D(2)	C(4)

Esercizio 3 (5 punti)

Si consideri un sistema che gestisce la memoria con paginazione a domanda, applicando una politica di controllo dinamico del Working Set. Per ogni processo sono definiti i seguenti dati:

- l'intero *MaxBlocchi*: massimo numero di blocchi disponibili per il caricamento del Working Set, che viene ridefinito periodicamente dal demone WorkingSetManager;
- l'intero *PagineResidenti*, uguale al numero di pagine attualmente caricate in memoria e variabile nel tempo;
- la tabella delle pagine di ogni processo, con campi *Blocco*, *R* (bit di pagina riferita) e *DP* (Distanza Passata). In particolare, il valore di *R* è quello usuale (0 se la pagina non è stata riferita, 1 altrimenti). Invece il valore di *DP*

Sistemi Operativi e Laboratorio, Prova del 31/5/2018

conta il numero di applicazioni consecutive dell'algoritmo di sostituzione durante le quali la pagina era marcata come non riferita ($R=0$).

Quando un processo avanza, per ogni pagina riferita:

- se il riferimento determina Page Fault, la pagina riferita viene caricata nel blocco libero individuato dal primo elemento della lista dei blocchi liberi in memoria (che si suppone sempre non vuota), viene aggiornata conseguentemente la tabella delle pagine (inizializzando anche i valori di DP e R) e viene incrementata la variabile $PagineResidenti$;

Il demone WorkingSetManager, che interviene periodicamente, esegue le seguenti operazioni per ogni processo:

Fase 1) per ogni pagina residente in memoria aggiorna la distanza passata con il seguente algoritmo:

- se $R=0$ assegna $DP=DP+1$;
- altrimenti assegna $R=0$ e $DP=0$

Fase 2) per ogni processo:

- se $PagineResidenti > MaxBlocchi$:
 - scarica dalla memoria principale $PagineResidenti - MaxBlocchi$, selezionandole in ordine decrescente del valore di DP ;
 - se per l'ultima pagina scaricata si ha $DP < 5$, incrementa $MaxBlocchi$;
- se invece $PagineResidenti < MaxBlocchi$, decrementa $MaxBlocchi$.

Al tempo $T1$, subito dopo un intervento di WorkingSetManager, per il processo P si ha $MaxBlocchi = 8$ e la tabella delle pagine ha la seguente configurazione:

Pagina	Blocco	R	DP
0	-	-	-
1	-	-	-
2	-	-	-
3	20	0	9
4	-	-	-
5	-	-	-
6	-	-	-
7	10	0	4
8	-	-	-
9	-	-	-
10	8	0	5
11	6	0	2
12	15	0	7

Inoltre, sempre al tempo $T1$, la lista dei blocchi liberi in memoria sia: $\rightarrow 50 \rightarrow 51 \rightarrow 52 \rightarrow 53 \rightarrow 54 \rightarrow \dots$,

Tra il tempo $T1$ e il tempo $T2$ avanza il processo P , poi al tempo $T2$ viene eseguito nuovamente il demone WorkingSetManager.

Si chiede di mostrare il valore di $PagineResidenti$, di $MaxBlocchi$ e la tabella delle pagine del processo P prima dell'esecuzione del WorkingSetManager al tempo $T2$, al termine della prima fase di esecuzione del

WorkingSetManager (al tempo $T3$) e al termine dell'esecuzione del WorkingSetManager (al tempo $T4$), nelle seguenti ipotesi (**da considerare in alternativa**):

- 1) Durante la sua esecuzione il processo P riferisce nell'ordine le seguenti pagine: **8, 10, 0, 3, 8, 11, 0, 11, 3**

T2: Tabella delle Pagine

Pagina	Blocco	R	DP
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			

T3: Tabella delle Pagine

Pagina	Blocco	R	DP
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			

T4: Tabella delle Pagine

Pagina	Blocco	R	DP
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			

Pagine Residenti= _____ ; Pagine Residenti= _____ ; Pagine Residenti= _____ ;

Sistemi Operativi e Laboratorio, Prova del 31/5/2018

MaxBlocchi= _____ ; MaxBlocchi= _____ ; MaxBlocchi= _____ ;

2) Durante la sua esecuzione il processo P e riferisce nell'ordine le seguenti pagine: **6, 8, 3, 7, 0, 3, 8, 9, 4**

T2: Tabella delle Pagine

Pagina	Blocco	R	DP
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			

Pagine Residenti= _____ ;
MaxBlocchi= _____ ;

T3: Tabella delle Pagine

Pagina	Blocco	R	DP
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			

Pagine Residenti= _____ ;
MaxBlocchi= _____ ;

T4: Tabella delle Pagine

Pagina	Blocco	R	DP
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			

Pagine Residenti= _____ ;
MaxBlocchi= _____ ;

Soluzione

1) Durante la sua esecuzione il processo P e riferisce nell'ordine le seguenti pagine: **8, 10, 0, 3, 8, 11, 0, 11, 3**

T2: Tabella delle Pagine

Pagina	Blocco	R	DP
0	51	1	0
1			
2			
3	20	1	9
4			
5			
6			
7	10	0	4
8	50	1	0
9			
10	8	1	5
11	6	1	2
12	15	0	7

Pagine Residenti= 7; MaxBlocchi=8;

T3: Tabella delle Pagine

Pagina	Blocco	R	DP
0	51	0	0
1			
2			
3	20	0	0
4			
5			
6			
7	10	0	5
8	50	0	0
9			
10	8	0	0
11	6	0	0
12	15	0	8

Pagine Residenti= 7; MaxBlocchi=8;

T4: Tabella delle Pagine

Pagina	Blocco	R	DP
0	51	0	0
1			
2			
3	20	0	0
4			
5			
6			
7	10	0	5
8	50	0	0
9			
10	8	0	0
11	6	0	0
12	15	0	8

Pagine Residenti=7; MaxBlocchi=7

2) Durante la sua esecuzione il processo P e riferisce nell'ordine le seguenti pagine: **6, 8, 3, 7, 0, 3, 8, 9, 4**

T2: Tabella delle Pagine

Pagina	Blocco	R	DP
0	52	1	0
1			
2			
3	20	1	9
4	54	1	0
5			
6	50	1	0
7	10	1	4
8	51	1	0
9	53	1	0
10	8	0	5
11	6	0	2
12	15	0	7

Pagine Residenti=10; MaxBlocchi=8;

T3: Tabella delle Pagine

Pagina	Blocco	R	DP
0	52	0	0
1			
2			
3	20	0	0
4	54	0	0
5			
6	50	0	0
7	10	0	0
8	51	0	0
9	53	0	0
10	8	0	6
11	6	0	3
12	15	0	8

Pagine Residenti= 10; MaxBlocchi=8;

T4: Tabella delle Pagine

Pagina	Blocco	R	DP
0	52	0	0
1			
2			
3	20	0	0
4	54	0	0
5			
6	50	0	0
7	10	0	0
8	51	0	0
9	53	0	0
10			
11	6	0	3
12			

Pagine Residenti=8; MaxBlocchi=8

Sistemi Operativi e Laboratorio, Prova del 31/5/2018

Esercizio 4 (5 punti)

In un disco della capacità di 128 MBytes è installato un file system FAT-16 (con indirizzi a 16 bit) con blocchi di 2 Kbyte. La copia permanente della FAT è allocata sul disco a partire dal blocco fisico 2. Gli elementi della FAT sono in corrispondenza biunivoca con i blocchi del disco, compresi i blocchi 0 e 1 (riservati al codice di boot e ad altri dati del file system) e quelli, a partire dal blocco 2 occupati dalla FAT. I blocchi successivi all'ultimo blocco occupato dalla FAT sono i blocchi dati. Pertanto, i valori legittimi dei puntatori contenuti nella FAT sono quelli non minori di $2 + \text{LunghezzaFAT}$, dove LunghezzaFAT è espressa in blocchi, purché non eccedano la capacità del disco. Nel file system il file `estate2018` contenuto nella directory `vacanze` occupa 9 blocchi logici allocati come segue:

Blocco logico:	0	1	2	3	4	5	6	7	8
Blocco fisico:	400	331	901	449	120	637	199	1002	588

Ad un dato tempo, quando la directory `vacanze` è caricata in memoria, il sistema operativo riceve una richiesta di lettura dei caratteri del file `estate2018` compresi tra 5000 e 9499 (estremi inclusi).

Si chiede:

1. La lunghezza della FAT, espressa in numero di elementi, byte e in numero di blocchi.
2. Il minimo e il massimo indice dei blocchi dati
3. Quali blocchi logici del file `estate2018` sono interessati alla lettura
4. Quali blocchi fisici del file `estate2018` devono essere letti
5. Quali bytes di ogni blocco sono interessati alla lettura (indicando il byte iniziale e il byte finale)
6. Quali elementi della FAT devono essere letti.

Soluzione

1. Lunghezza della FAT: _____ elementi
Lunghezza della FAT: _____ bytes
Lunghezza della FAT: _____ blocchi
2. Il primo blocco dati ha indice _____
L'ultimo blocco dati ha indice _____
3. Blocco logico del file `estate2018` che contiene il byte 5000: _____
Blocco logico del file `estate2018` che contiene il byte 9499: _____
Blocchi logici del file `estate2018` interessati alla lettura: _____
4. Blocchi fisici del file `estate2018` che devono essere letti: _____
5. Del blocco _____ si leggono i bytes da _____ a _____
Del blocco _____ si leggono i bytes da _____ a _____
Del blocco _____ si leggono i bytes da _____ a _____
Del blocco _____ si leggono i bytes da _____ a _____
Del blocco _____ si leggono i bytes da _____ a _____
6. Vengono letti gli elementi della FAT: _____

Soluzione

7. Lunghezza della FAT: $128\text{M}/2\text{K}$ elementi = **64K elementi** ($2^{27}/2^{11} = 2^{16}$ elementi)
Lunghezza della FAT in bytes: $64\text{Kelementi} * 2 = \mathbf{128\text{KB}}$
Lunghezza della FAT in blocchi: $128\text{KB}/2\text{KB} = \mathbf{64\text{ blocchi}}$
8. Il primo blocco dati ha indice $2 + 64 = \mathbf{66}$
L'ultimo blocco dati ha indice $128\text{MB}/2\text{KB} - 1 = 64\text{K} - 1 = \mathbf{65.535}$
9. Blocco logico del file `estate2018` che contiene il byte 5000: $5000 \text{ div } 2048 = \mathbf{2}$;
Blocco logico del file `estate2018` che contiene il byte 9499: $9499 \text{ div } 2048 = \mathbf{4}$;
Blocchi logici del file `estate2018` interessati alla lettura: **2, 3, 4**
10. Blocchi fisici del file `estate2018` che devono essere letti: **901, 449, 120**
11. Del blocco 2 si leggono i bytes da $5000 \text{ mod } 2048 = \mathbf{904}$ a 2047
Del blocco 3 si leggono i bytes **da 0 a 2047**
Del blocco 4 si leggono i bytes **da 0 a 9499 mod 2048 = 1306**

Sistemi Operativi e Laboratorio, Prova del 31/5/2018

12. Considerato che il puntatore al primo blocco del file (il 400) è contenuto nella directory *vacanze*, vanno letti gli elementi della FAT: **400, 331,901, 449**. Nota: il puntatore del blocco 120 è contenuto nell'elemento della FAT 449.

Esercizio 5 (4 punti)

Si consideri un sistema che gestisce la memoria con paginazione e tabelle delle pagine a due livelli. Gli indirizzi logici hanno lunghezza di 32 bit e le pagine logiche e fisiche hanno ampiezza di 1 kByte. La tabella di primo livello comprende 2^{10} elementi.

Gli elementi di ogni tabella di primo o secondo livello hanno una lunghezza di 4 byte. Nella tabella 10 bit sono riservati agli indicatori (pagina caricata, riferita, modificata, ecc.) mentre i rimanenti 22 bit codificano un indice di blocco fisico.

Si chiede:

1. la lunghezza del campo offset, in numero di bit;
2. il numero di elementi contenuti in ogni tabella di secondo livello;
3. lo spazio occupato in memoria da ogni tabella di secondo livello (numero di byte);
4. la massima dimensione della memoria fisica, espressa in numero di blocchi e in numero di byte.
5. l'indice della prima e dell'ultima pagina dello spazio virtuale indirizzate dalla tabella di secondo livello numero 5
6. la tabella di secondo livello necessaria per tradurre l'indirizzo logico 20982784 (supponendo che la MMU nella sua cache non abbia le informazioni necessarie per tradurre questi indirizzi).

Soluzione

1. lunghezza del campo offset: _____ bit
2. La tabella di secondo livello contiene _____ elementi
3. Lo spazio occupato in memoria da ogni tabella di secondo livello è: _____
4. La massima dimensione della memoria fisica è _____
5. la tabella delle pagine numero 5 indicizza le pagine da _____ a _____
6. la tabella di secondo livello necessaria per tradurre l'indirizzo logico 20982784 è la _____

Soluzione

1. lunghezza del campo offset: **10 bit**
2. per la codifica degli indici di blocco nell'indirizzo logico sono disponibili 22 bit, di cui 10 sono indici della tabella di primo livello e 10 sono riservati all'offset. Pertanto la tabella di secondo livello contiene **$2^{12} = 4096$ elementi**
3. spazio occupato in memoria da ogni tabella di secondo livello: $2^{12} * 4 = 2^{14}$ byte \rightarrow **16k Byte**
4. gli indici dei blocchi di memoria fisica sono codificati con 22 bit e l'offset è codificato con 10 bit. Pertanto, la massima dimensione dell'indirizzo fisico è di 32 bit e la massima dimensione della memoria fisica è di 2^{22} blocchi di 2^{10} bytes ciascuno, per cui la massima dimensione della memoria fisica è: $2^{22} * 2^{10} = 2^{32}$ byte = **4 GB**
5. la tabella delle pagine numero 5 indicizza le pagine dello spazio virtuale da $4.096 * 5 = 20.480$ a $20.480 + 4.096 - 1 = 24.575$
6. l'indirizzo logico 20982784 è contenuto nella pagina dello spazio virtuale numero $20982784 \div 2^{10} = 20.491$. Quindi, in base al punto precedente, la tabella di secondo livello necessaria per tradurre l'indirizzo logico 20982784 è la **5**

Esercizio 6 (3 punti)

Un disco RAID di livello 4 è composto da 4 dischi fisici, numerati da 0 a 3. Le strip corrispondono a blocchi. Il disco di indice 3 è ridondante e il suo blocco di indice i contiene la parità dei blocchi di indice i dei dischi non ridondanti, cioè dei dischi 0, 1 e 2.

Sistemi Operativi e Laboratorio, Prova del 31/5/2018

Al tempo t i blocchi di indice 9 dei dischi fisici 0,1,2 hanno i contenuti mostrati in tabella.

Disco 0	0	0	1	0	1	0	1	1
Disco 1	1	1	0	1	1	1	0	1
Disco 2	1	0	0	1	1	0	1	0

A tempo t arriva un comando di scrittura nel blocco 9 del disco 1 del valore: 0 1 0 0 0 1 0.

Successivamente, al tempo $t+1$ arriva un comando di lettura del blocco 9 del disco 0 che però risulta illeggibile perché danneggiato. Si chiede:

- 1) Quale è il contenuto del blocco 9 del disco 3 **prima** dell'operazione di scrittura;
- 2) Quali sono i dischi fisici sui quali si deve operare per eseguire la scrittura e di quali operazioni si tratta;
- 3) Quali di queste operazioni sono eseguibili in parallelo e quali vanno eseguite sequenzialmente?
- 4) Il contenuto del blocco 9 dei dischi che vengono che vengono modificati.
- 5) Quali operazioni bisogna fare al tempo $t+1$ per recuperare il contenuto del blocco 9 del disco 0?
(in particolare, quali blocchi è necessario leggere e che operazione viene eseguita per ricalcolare il contenuto del blocco danneggiato)

Soluzione

- 1) Contenuto del blocco 9 del disco 3 prima dell'operazione di scrittura:

Disco 3								
---------	--	--	--	--	--	--	--	--

- 2) Si devono eseguire le seguenti operazioni:

- disco/i: _____ operazione: _____ per quale motivo: _____
- disco/i: _____ operazione: _____ per quale motivo: _____
- disco/i: _____ operazione: _____ per quale motivo: _____
- disco/i: _____ operazione: _____ per quale motivo: _____
- disco/i: _____ operazione: _____ per quale motivo: _____

- 3) Sono eseguibili in parallelo le operazioni: _____

Vanno eseguite sequenzialmente le operazioni: _____

- 4) Queste operazioni modificano il blocco 9 dei dischi _____ e i nuovi contenuti sono i seguenti:

Disco 0								
Disco 1								
Disco 2								
Disco 3								

- 5) Blocchi da leggere: _____

blocchi da scrivere: _____

Operazione da eseguire: _____

Soluzione

- 1) Contenuto del blocco 9 del disco 3 prima dell'operazione di scrittura:

Disco 3	0	1	1	0	1	1	0	0
---------	---	---	---	---	---	---	---	---

- 2) Si devono eseguire le seguenti operazioni:

- lettura dei vecchi valori sul blocco 9 dei dischi 1 e 3 (per ricalcolare la parità)
- scrittura sul blocco 9 del disco 1 (per modificare il contenuto)
- scrittura sul blocco 9 del disco 3 (per modificare la parità)

- 3) Sono eseguibili in parallelo le due operazioni di lettura, poi sono eseguibili in parallelo le due operazioni di scrittura

- 4) Queste operazioni modificano il blocco 9 dei dischi 1 e 3 e i nuovi contenuti sono i seguenti:

Disco 0	0	0	1	0	1	0	1	1
Disco 1	0	1	0	0	0	0	1	0

Sistemi Operativi e Laboratorio, Prova del 31/5/2018

Disco 2	1	0	0	1	1	0	1	0
Disco 3	1	1	1	1	0	0	1	1

- 5) Bisogna leggere tutti i blocchi numero 9 da tutti i dischi funzionanti (1,2 e 3), e quindi eseguire un'XOR bit a bit di tutti i blocchi letti.

Esercizio 7 (3 punti)

Un sistema operativo simile a UNIX gestisce la memoria con paginazione a domanda mediante il processo *PageDaemon* (con parametri *Lotsfree* e *MinFree*), che applica l'algoritmo di sostituzione *Second Chance*. Ad ogni intervento, il processo *PageDaemon* si comporta nel modo seguente:

- se $\text{NumeroDiBlocchiLiberi} \geq \text{MinFree}$ e $\text{NumeroDiBlocchiLiberi} < \text{LotsFree}$, applica ripetutamente l'algoritmo *Second Chance* sui blocchi che contengono pagine caricate in memoria, e ogni volta scarica la pagina da questo selezionata come vittima, finché $\text{NumeroDiBlocchiLiberi} = \text{LotsFree}$.
- se $\text{NumeroDiBlocchiLiberi} < \text{MinFree}$ esegue lo swapout di uno o più processi, finché diviene $\text{NumeroDiBlocchiLiberi} \geq \text{LotsFree}$. Per lo swapout dei processi si seleziona di volta in volta quello che ha il maggior numero di pagine caricate in memoria principale, e in caso di parità si segue l'ordine alfabetico.

Gli elementi della *CoreMap* hanno i campi *Proc* (processo a cui è assegnato il blocco; il campo è vuoto se il blocco è libero); *Pag* (pagina del processo caricata nel blocco), *Rif* (bit di pagina riferita, utilizzato da *Second Chance*). Al tempo t (quando sono presenti i processi A, B, C, D, la *Core Map* ha la configurazione mostrata in tabella (non sono mostrati i blocchi assegnati al sistema operativo) e il puntatore dell'algoritmo di *SecondChance* è posizionato sul blocco 7, viene attivato il processo il *PageDaemon*.

								↓																
Proc		D	C	C	C	B	A	C	B	C	B	B	A		D	D	D	B	C	B		A	D	A
Pag		0	5	10	12	7	0	1	0	2	9	6	3		1	2	6	2	7	3		7	11	2
Rif		1	0	0	1	1	0	1	0	0	1	1	1		1	1	1	1	1	0		1	1	0
Blocco	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23

Core Map al tempo t

Si chiede quali sono le pagine scaricate dal processo *PageDaemon* e la configurazione della *CoreMap* nelle seguenti ipotesi (**alternative**):

- a) *PageDaemon* ha parametri $\text{Lotsfree}=7$ e $\text{MinFree}=3$;
 b) *PageDaemon* ha parametri $\text{Lotsfree}=7$ e $\text{MinFree}=4$.

Soluzione

Ipotesi a) *PageDaemon* ha parametri $\text{Lotsfree}=7$ e $\text{MinFree}=3$:

- Vengono scaricate le pagine: _____
- La configurazione finale della *CoreMap* è la seguente:

Proc																								
Pag																								
Rif																								
Blocco	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23

Ipotesi b) *PageDaemon* ha parametri $\text{Lotsfree}=7$ e $\text{MinFree}=4$:

- Vengono scaricate le pagine: _____
- La configurazione finale della *CoreMap* è la seguente:

Proc																								
Pag																								
Rif																								
Blocco	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23

Soluzione

Ipotesi a)

