

PROGRAMMAZIONE DI SISTEMA

Esercizi proposti e svolti di teoria
Tratti da compiti di esame

Gestione Memoria 2 (esercizi relativi a cap 10)

1. (1/7/2016) Si consideri la seguente sequenza di riferimenti in memoria nel caso di un programma di 1000 parole: 261, 409, 985, 311, 584, 746, 632, 323, 470, 915, 858. Si determini la stringa dei riferimenti a pagine, supponendo che la loro dimensione sia di 200 parole. Si utilizzi un algoritmo di sostituzione pagine di tipo second-chance set (con limite di 3 frame disponibili). Si supponga che i riferimenti inizino a programma già avviato, con i 3 frame già allocati alle pagine n. 4, 3, e 1, e la coda FIFO contenente, nell'ordine, $4_0, 3_1, 1_0$ (Il pedice rappresenta il reference bit). Determinare quali e quanti page fault (accessi a pagine non presenti nel resident set) si verificheranno (è richiesta la visualizzazione del resident set dopo ogni riferimento). Si supponga che il bit di riferimento di una pagina venga inizializzato a 0 in corrispondenza a un page-fault.

R:	<p>La stringa di riferimento è: 1,2,4,1,2,3,3,1,2,4,4 (Attenzione, la numerazione parte da 0, non da 1: es. $261/200 = 1$)</p> <table><tr><td></td><td>1</td><td>2</td><td>4</td><td>1</td><td>2</td><td>3</td><td>3</td><td>1</td><td>2</td><td>4</td><td>4</td></tr><tr><td>4_0</td><td>4_0</td><td>2_0</td><td>4_0</td><td>4_0</td><td>4_0</td><td>3_0</td><td>3_1</td><td>3_1</td><td>3_1</td><td>3_0</td><td>3_0</td></tr><tr><td>3_1</td><td>3_1</td><td>3_1</td><td>3_0</td><td>3_0</td><td>2_0</td><td>2_0</td><td>2_0</td><td>2_0</td><td>2_1</td><td>4_0</td><td>4_1</td></tr><tr><td>1_0</td><td>1_1</td><td>1_1</td><td>1_0</td><td>1_1</td><td>1_1</td><td>1_0</td><td>1_0</td><td>1_1</td><td>1_1</td><td>1_0</td><td>1_0</td></tr><tr><td></td><td></td><td>PF</td><td>PF</td><td></td><td>PF</td><td>PF</td><td></td><td></td><td></td><td>PF</td><td></td></tr></table> <p>In totale ci sono 5 Page Fault.</p>												1	2	4	1	2	3	3	1	2	4	4	4_0	4_0	2_0	4_0	4_0	4_0	3_0	3_1	3_1	3_1	3_0	3_0	3_1	3_1	3_1	3_0	3_0	2_0	2_0	2_0	2_0	2_1	4_0	4_1	1_0	1_1	1_1	1_0	1_1	1_1	1_0	1_0	1_1	1_1	1_0	1_0			PF	PF		PF	PF				PF	
	1	2	4	1	2	3	3	1	2	4	4																																																												
4_0	4_0	2_0	4_0	4_0	4_0	3_0	3_1	3_1	3_1	3_0	3_0																																																												
3_1	3_1	3_1	3_0	3_0	2_0	2_0	2_0	2_0	2_1	4_0	4_1																																																												
1_0	1_1	1_1	1_0	1_1	1_1	1_0	1_0	1_1	1_1	1_0	1_0																																																												
		PF	PF		PF	PF				PF																																																													

2. (5/7/2019)

D. Si consideri la seguente sequenza di riferimenti in memoria nel caso di un programma in cui, per ogni accesso (indirizzi in esadecimale, si indirizza il Byte), si indica se si tratta di lettura (R) o scrittura (W): **R 3F5**, R 364, **W 4D3**, W 47E, R 4C8, W 2D1, R 465, W 2A0, **R 3BA**, W 4E6, R 480, R 294, R 0B8, R 14E.

Supponendo che sia indirizzi fisici che logici siano su 12 bit, che si usi paginazione con pagine di dimensione 128 Byte e che il massimo indirizzo utilizzabile dal programma sia C10, si dica quante pagine sono presenti nello spazio di indirizzamento del programma e se ne calcoli la frammentazione interna.

R. Numero totale di pagine indirizzabili (incluse quelle fuori dai riferimenti proposti): $C10 = 1100\ 0, 001\ 0000$. Il massimo indice di pagina 2^*C (esadecimale) = 2^*12 (decimale) = 24 => In totale lo spazio indirizzabile comprende 25 pagine. Metodo alternativo: $C10_H + 1 = 3089_{10}$, $NP = \lceil 3089/128 \rceil = 25$

Frammentazione interna: L'ultima pagina è occupata fino all'offset 0010000 (16) => fr. Int. = $128-17 = 111$ Byte. Alternativa: $128 - 3089\%128 = 128-17 = 111$ (Byte)

D. Si determini la stringa dei riferimenti a pagine (Si consiglia di passare da esadecimale a binario, per determinare correttamente il numero di pagina e, se necessario, il displacement/offset). Si utilizzi un algoritmo di sostituzione pagine di tipo LRU (Least Recently Used). Si assuma che siano disponibili 3 frame, agli indirizzi fisici (espressi in esadecimale) 780, A00, B00. Si richiede la visualizzazione (dopo ogni accesso) del resident set (i frame fisici contenenti pagine logiche). Determinare quali e quanti page fault (accessi a pagine non presenti nel resident set) si verificheranno. Si dica infine a quali indirizzi fisici vengono effettuati gli accessi (tra quelli sopra elencati) R 3F5, W 4D3, R 3BA

R. Indirizzi logici (p,d) (pagina,displacement). Una pagina contiene 128 Byte, quindi per d servono 7 bit, per p 5 bit. Riferimenti in binario: R (0011 1,111 0101), R (0011 0,110 0011), W (0100 1,101 0011). Stringa dei riferimenti (basta p non serve d): 7 (2*3+1), 6 (2*3+0), 9, 8, 9, 5, 8, 5, 7, 9, 9, 5, 1, 2 (si noti che p, dato dai 5 bit più significativi dell'indirizzo logico, può essere velocemente calcolato come il doppio della prima cifra esadecimale + il bit più significativo della seconda cifra esadecimale).

Riferimenti	7	6	9	8	9	5	8	5	7	9	9	5	1	2
Resident Set	780	7	7	8		8			8	9			9	2
	A00		6	6		5			5	5			5	5
	B00			9	9		9		7	7			1	1
Page Fault	*	*	*	*		*			*	*			*	*

Indicare nella prima riga la stringa dei riferimenti a pagine (rappresentate a scelta in esadecimale o decimale), nelle tre successive (che rappresentano i 3 frame del resident set), le pagine allocate nei corrispondenti frame. La free frame list sia inizialmente (780, A00, B00). Nell'ultima riga si indichi la presenza o meno di un Page Fault

Numero totale di page fault: 9

Indirizzi logici: R 3F5: (0011 1,111 0101) W 4D3: (0100 1,101 0011) R 3BA: (0011 1,011 1010)
 Indirizzi fisici: R (0111 1,111 0101) = 7F5, W (1011 0,101 0011) = B53, R (1011 0,011 1010) = B3A

Attenzione: La pagina logica 7 viene posta in due frame diversi in due momenti diversi

3. (11/9/2018) Si consideri la seguente sequenza di riferimenti in memoria nel caso di un programma di 4K parole in cui, per ogni accesso (indirizzi in esadecimale), si indica se si tratta di lettura (R) o scrittura (W): W 3A1, R 3F5, R A64, W BD3, W 57E, R A08, R B85, W 3A0, R A1A, W A36, R B20, R 734, R AB8, R C4E, W B64.

Si determini la stringa dei riferimenti a pagine, supponendo che la loro dimensione sia di 512 parole. Si utilizzi un algoritmo di sostituzione pagine di tipo Enhanced Second-Chance, per il quale, al bit di riferimento (da inizializzare a 0 in corrispondenza al primo accesso a una nuova pagina dopo il relativo page fault), si unisce il bit di modifica (modify bit). Si assuma che una pagina venga sempre modificata in corrispondenza a una scrittura (write), che siano disponibili 3 frame e che l'algoritmo operi con il criterio seguente: dato il puntatore alla pagina corrente (secondo la strategia FIFO) si fa un primo giro, senza modificare il reference bit, sulle pagine per localizzare la vittima (l'ordine di priorità è (reference,modify): (0,0), (0,1), (1,0), (1,1)); una volta determinata la vittima, si fa un secondo giro per azzerare i reference bit delle pagine "salvate" (comprese tra la posizione di partenza e la vittima).

Determinare quali e quanti page fault (accessi a pagine non presenti nel resident set) si verificheranno. Si richiede la visualizzazione (dopo ogni accesso) del resident set, indicando per ogni frame i bit di riferimento e modifica. Si numerino le pagine a partire da 0.

Utilizzare, per questa domanda, lo schema seguente per svolgere l'esercizio, indicando nella prima riga la stringa dei riferimenti a pagine (rappresentate a scelta in esadecimale o decimale), nella seconda Read o Write, nelle tre successive (che rappresentano i 3 frame del resident set), le pagine allocate nei corrispondenti frame, indicando per ognuna i bit (reference,modify). Indicare inoltre (sottolineandola, circolettandola o ponendo una freccia), quale pagina si trova in testa al FIFO. Nell'ultima riga si indichi la presenza o meno di un Page Fault.

R. Si noti che ogni cifra esadecimale rappresenta 4 bit. Due cifre sono 8 bit. La divisione per 512 (2^9) si ottiene quindi eliminando le due cifre esadecimali meno significative e dividendo ulteriormente per 2: in pratica si divide per 2 la prima delle 3 cifre esadecimali.

Riferimenti	1	1	5	5	2	5	5	1	5	5	5	3	5	6	5
Read/write	W	R	R	W	W	R	R	W	R	W	R	R	R	R	W
Resident Set	1 ₀₁	1 ₁₁	1 ₁₁	1 ₁₁	1 ₁₁	1 ₁₁	1 ₁₁	1 ₁₁	1 ₁₁	1 ₁₁	1 ₁₁	1 ₀₁	1 ₀₁	1 ₀₁	1 ₀₁
			5 ₀₀	5 ₁₁	5 ₁₁	5 ₁₁	5 ₁₁	5 ₁₁	5 ₁₁	5 ₁₁	5 ₁₁	5 ₀₁	5 ₁₁	5 ₀₁	5 ₁₁
					2 ₀₁	2 ₀₁	2 ₀₁	2 ₀₁	2 ₀₁	2 ₀₁	2 ₀₁	3 ₀₀	3 ₀₀	6 ₀₀	6 ₀₀
Page Fault	X		X		X							X		X	

Numero totale di page fault:5.....