

ESERCITAZIONE

Data la seguente tabella delle pagine all'istante iniziale e la seguente lista di istruzioni, determinare la tabella delle pagine dopo ogni istruzione, usando l'algoritmo LRU gestito con uno stack.

(Blocco logico (BL.) \equiv Pagina)

Blocco logico	FRAME	DIRTY BIT	*T
0	15	1	7
1			
2			
3			
TOP \rightarrow 4	7	0	0
5			
6			
7	1	1	9
8			
9	5	1	NIL

ISTRUZIONI
1) Read (3)
2) Write (4)
3) Write (3)
4) Read (3)
5) Read (8)

BL.

La precedente tabella delle pagine non rappresenta l'ordine dello stack all'istante iniziale, ma, l'ordine esatto deve essere ricavato, partendo dall'elemento puntato da TOP e seguendo l'ordine degli elementi puntati da *T, fino a NIL (ultimo elemento) il risultato è:

BL.	Fr	DB
4	7	0
0	15	1
7	1	1
9	5	1

Attenzione a non confondere il DIRTY BIT con il BIT-REF.

Qui il DIRTY BIT è necessario a sapere se una pagina deve essere deallocata su disco o direttamente sovrascritta in

quanto ogni blocco logico può essere letto (Read: DB resta a 0)
o scritto (Write: DB va a 1)

SVOLGIMENTO:

1) Read(3) \Rightarrow BL. 3 Assente

Il BL. 3 viene caricato in memoria allo stack e prende il posto del BL. 9 che si trova in fondo (quindi è il meno recente)

BL.	Fz	DB
3	5	0
4	7	0
0	15	1
7	1	1

BL. 9 è stato riscritto su disco prima di essere deallocato, perché il suo DB era a 1

2) Write(4) \Rightarrow BL. 4 presente

BL.	Fz	DB
4	7	1
3	5	0
0	15	1
7	1	1

BL. 4 viene portato in memoria allo stack e il suo DB viene messo a 1

3) Write(3) \Rightarrow BL. 3 presente

BL.	Fz	DB
3	5	1
4	7	1
0	15	1
7	1	1

4) Read(3) \Rightarrow BL 3 presente

BL	Fz	DB
3	5	1
4	7	1
0	15	1
7	1	1

5) Read(8) \Rightarrow BL 8 Assente

BL	Fz	DB
8	1	0
3	5	1
4	7	1
0	15	1

□