

**COMPITO DI SISTEMI INFORMATIVI**  
**18 gennaio 2016 (Tot. 16) Tempo: 2h, Compito B**

**Esercizio 1 (punti 3)**

Si consideri il seguente log:

- |                      |                      |
|----------------------|----------------------|
| 1. D(T1,O1,B1)       | 15. C(T4)            |
| 2. I(T1,O2,A2)       | 16. U(T3,O6,B16,A16) |
| 3. B(T2)             | 17. I(T3,O7,A17)     |
| 4. U(T1,O2,B4,A4)    | 18. B(T5)            |
| 5. B(T3)             | 19. D(T5,O6,B19)     |
| 6. D(T2,O2,B6)       | 20. U(T3,O7,B20,A20) |
| 7. U(T3,O3,B7,A7)    | 21. C(T3)            |
| 8. U(T2,O4,B8,A8)    | 22. U(T2,O8,B22,A22) |
| 9. C(T1)             | 23. B(T6)            |
| 10. U(T3,O4,B10,A10) | 24. I(T5,O8,A24)     |
| 11. B(T4)            | 25. D(T6,O7,B25)     |
| 12. I(T4,O5,A12)     | 26. D(T5,O8,B26)     |
| 13. CK(T2,T3,T4)     | 27. U(T2,O9,B27,A27) |
| 14. I(T2,O6,A14)     |                      |

si mostrino le operazioni di recovery da effettuare supponendo che il guasto avvenga subito dopo l'ultimo record del log.

**Esercizio 2 (punti 4)**

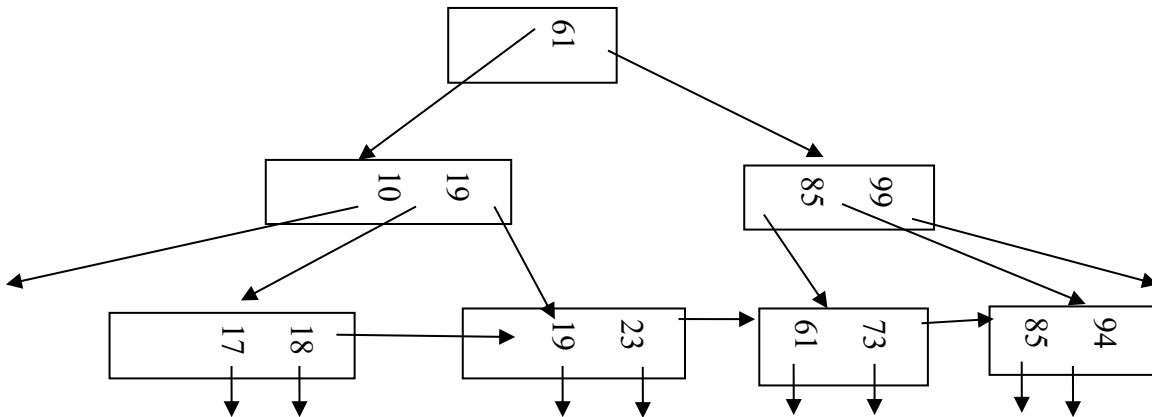
Dato il seguente schedule:

w2(x) r1(y) r2(x) w3(y) r4(y) w5(y) r5(z) w3(y)

si indichi se rispetta il two-phase locking. Nel caso lo rispetti, si indichi una sequenza di acquisizione e rilascio di lock compatibile con lo schedule e che rispetti il two-phase locking. Nel caso non lo rispetti, si indichi almeno uno dei conflitti.

### Esercizio 3 (punti 3)

Sia dato il seguente B+ tree di ordine 4.



Si mostri l'albero risultante dalla cancellazione della chiave 61

### Esercizio 4 (punti 6)

Siano date le seguenti relazioni

Assicuratore(CodAss, Nome, Cognome, Via, Città, CAP, Filiale, Telefono)

Cliente(CodCli, Nome, Cognome, Via, Città, CAP, Filiale, Telefono, CodAss, Portafoglio)

la query

```
SELECT A.*, C.*
```

```
FROM Assicuratore AS A, Cliente AS C
```

```
WHERE A.CodAss = C.CodAss AND C.Portafoglio <=40000
```

e i parametri:

buffer di memoria centrale disponibili:  $M=200.000$

dimensione del buffer:  $B=20.000$  bytes

numero di tuple:  $T(\text{Cliente})= 5.000.000$ ,  $T(\text{Assicuratore})= 500.000$ ,

dimensione delle tuple:  $S(\text{Cliente})=10.000$ ,  $S(\text{Assicuratore})=10.000$ ,  $S(\text{CodAss})=50$

numero di valori  $V(\text{Cliente}, \text{CodAss})=400.000$

$\text{Min}(\text{Cliente}, \text{Portafoglio})=1.000$   $\text{Max}(\text{Cliente}, \text{Portafoglio})=100.000$

Si calcoli il costo minimo di ciascuna delle seguenti sequenze:

$(\sigma_{\text{Portafoglio} \leq 40000} \text{ Cliente}) \triangleright \triangleleft \text{ Assicuratore}$

$\sigma_{\text{Portafoglio} \leq 40000} (\text{ Cliente } \triangleright \triangleleft \text{ Assicuratore})$

Si indichi quale delle sequenze ha costo inferiore.

Si supponga che i record siano impaccati nei buffer e che tutto lo spazio nei buffer sia occupato dai record.

Si supponga di avere un indice primario su  $\text{Cliente.CodAss}$  e uno secondario su  $\text{Cliente.Portafoglio}$ .

## SOLUZIONE

### Esercizio 1

1. D(T1,O1,B1)
2. I(T1,O2,A2)
3. B(T2)
4. U(T1,O2,B4,A4)
5. B(T3)
6. D(T2,O2,B6)
7. U(T3,O3,B7,A7)
8. U(T2,O4,B8,A8)
9. C(T1)
10. U(T3,O4,B10,A10)
11. B(T4)
12. I(T4,O5,A12)
13. CK(T2,T3,T4)
14. I(T2,O6,A14)
15. C(T4)
16. U(T3,O6,B16,A16)
17. I(T3,O7,A17)
18. B(T5)
19. D(T5,O6,B19)
20. U(T3,O7,B20,A20)
21. C(T3)
22. U(T2,O8,B22,A22)
23. B(T6)
24. I(T5,O8,A24)
25. D(T6,O7,B25)
26. D(T5,O8,B26)
27. U(T2,O9,B27,A27)

- 13 UNDO={T2,T3,T4 } REDO={}  
15 UNDO={T2,T3 } REDO={T4}  
18 UNDO={T2,T3,T5 } REDO={T4}  
21 UNDO={T2,T5 } REDO={ T3,T4}  
23 UNDO={ T2,T5,T6 } REDO={ T3,T4}

#### UNDO

- 27 O9=B27  
26 I(O8,B26)  
25 I(O7,B25)  
24 D(O8)  
22 O8=B22  
19 I(O6,B19)  
14 D(O6)  
8 O4=B8  
6 I(O2,B6)

#### REDO

- 7 O3=A7  
10 O4=A10  
12 I(O5,A12)  
16 O6=A16  
17 I(O7,A17)  
20 O7=A20

**Esercizio 2**

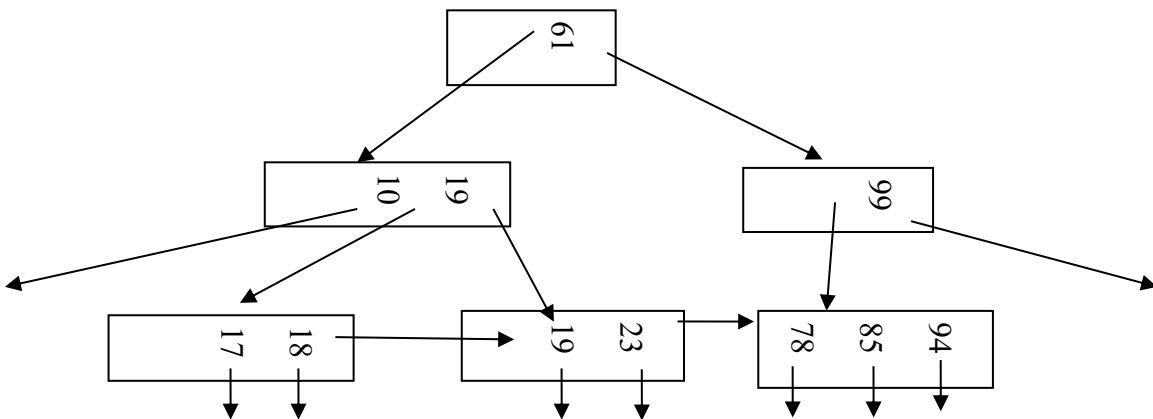
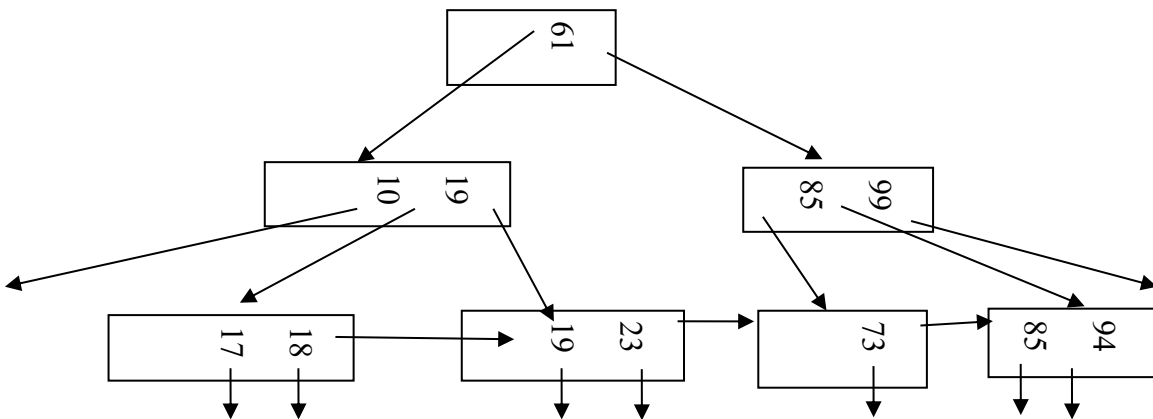
w2(x) r1(y) r2(x) w3(y) r4(y) w5(y) r5(z) w3(y)

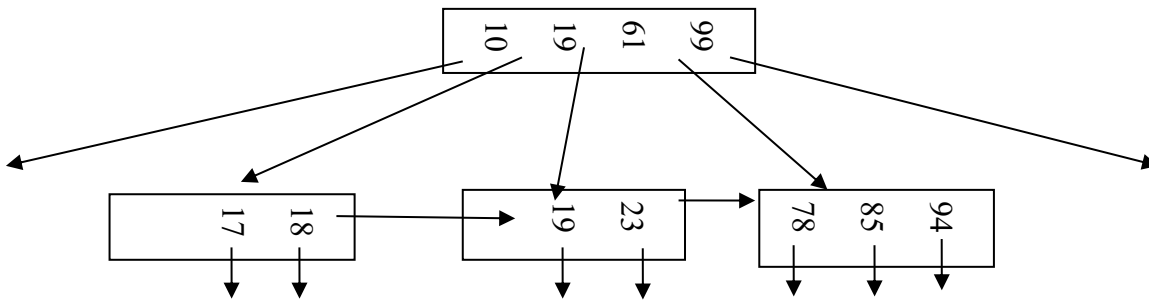
T1	T2	T3	T4	T5
	w lock(x)			
r lock(y)				
unlock(y)				
		w lock(y)		
		unlock(y)		
			r lock(y)	
			unlock(y)	
				w lock(y)
				r lock(z)

Lo schedule non è in 2 phase-locking perché T3 deve rilasciare il lock su y affinché T4 possa proseguire ma poi deve scrivere nuovamente su y

**Esercizio 3**

Cancellazione della chiave 61





#### Esercizio 4

buffer di memoria centrale disponibili:  $M=200000$

dimensione del buffer:  $B=20000$  bytes

numero di tuple:  $T(\text{Cliente})= 5000000$ ,  $T(\text{Assicuratore})= 500000$ ,

dimensione delle tuple:  $S(\text{Cliente})=10000$ ,  $S(\text{Assicuratore})=10000$ ,  $S(\text{CodAss})=50$

numero di valori  $V(\text{Cliente}, \text{CodAss})=400000$

$\text{Min}(\text{Cliente}, \text{Portafoglio})=1000$   $\text{Max}(\text{Cliente}, \text{Portafoglio})=100000$

Calcoliamo innanzitutto il numero di blocchi occupati da ciascuna relazione:

$B(\text{Cliente})=T(\text{Cliente})\cdot S(\text{Cliente})/B = 5000000\cdot 10000/20000=2500000$

$B(\text{Assicuratore})=T(\text{Assicuratore})\cdot S(\text{Assicuratore})/B = 500000\cdot 10000/20000=250000$

$(\sigma_{\text{Portafoglio} \leq 40000} \text{Cliente}) \triangleright \triangleleft \text{Assicuratore}$

$\sigma_{\text{Portafoglio} \leq 40000} (\text{Cliente} \triangleright \triangleleft \text{Assicuratore})$

Prima sequenza:  $(\sigma_{\text{Portafoglio} \leq 40000} \text{Cliente}) \triangleright \triangleleft \text{Assicuratore}$

Sia  $X = (\sigma_{\text{Portafoglio} \leq 40000} \text{Cliente})$

C'è un indice secondario su Cliente.Portafoglio, quindi proviamo con accesso tramite indice e con la scansione

$f=(40000-1000+1)/(100000-1000+1)= 0.394$

$\text{CostoI}(X) = 3+ f\cdot T(\text{Cliente})=3+0.394\cdot 5000000=1970003$

$\text{CostoS}(X) = B(\text{Cliente})= 2500000$

Costa meno l'accesso con indice quindi si usa quello.

$B(X) = 0.394\cdot 2500000=985000$

Né X né Assicuratore stanno in memoria centrale, quindi si prova con il join nested-loop basato sui blocchi e con l'hash join ibrido. Il join con indice usando l'indice primario su Cliente.CodAss non si può usare perché costruendo X è stato distrutto.

$\text{CostoJNL}(X \triangleright \triangleleft \text{Assicuratore})=B(\text{Assicuratore})+B(X)\cdot B(\text{Assicuratore})/M$   
 $=250000+985000\cdot 250000/200000=1481250$

$\text{CostoHJI}(X \triangleright \triangleleft \text{Assicuratore})=(3-2M/B(\text{Assicuratore}))(B(X)+B(\text{Assicuratore})) = (3-$   
 $2\cdot 200000/250000)\cdot (985000+250000) =1729000$

$$\text{Costo totale} = 1970003 + 1481250 = 3.451.253$$

Seconda sequenza:  $\sigma_{\text{Portafoglio} \leq 40000} (\text{Cliente} \triangleright \triangleleft \text{Assicuratore})$

Sia  $Z = \text{Cliente} \triangleright \triangleleft \text{Assicuratore}$

Né Cliente né Assicuratore stanno in memoria centrale, quindi si prova con il join nested-loop basato sui blocchi, con l'hash join ibrido e con il join con indice usando l'indice primario su Cliente.CodAss.

$$\begin{aligned} \text{CostoJNL}(Z) &= B(\text{Assicuratore}) + B(\text{Cliente}) * B(\text{Assicuratore}) / M = 250000 + 2500000 * 250000 / 200000 \\ &= 3375000 \end{aligned}$$

$$\begin{aligned} \text{CostoHJI}(Z) &= (3 - 2M/B(\text{Assicuratore})) (B(\text{Cliente}) + B(\text{Assicuratore})) = (3 - 2 * 200000 / 250000) * \\ &(2500000 + 250000) = 3850000 \end{aligned}$$

$$\begin{aligned} \text{CostoIJ}(Z) &= B(\text{Assicuratore}) + T(\text{Assicuratore}) \lceil B(\text{Cliente}) / V(\text{Cliente}, \text{CodAss}) \rceil \\ &= 250000 + 500000 * \lceil 2500000 / 400000 \rceil = 250000 + 500000 * 7 = 3750000 \end{aligned}$$

$$\begin{aligned} T(Z) &= T(\text{Assicuratore}) * T(\text{Cliente}) / \max \{ V(\text{Cliente}, \text{CodAss}), V(\text{Assicuratore}, \text{CodAss}) \} = \\ &500000 * 5000000 / 500000 = 5000000 \end{aligned}$$

$$S(Z) = S(\text{Cliente}) + S(\text{Assicuratore}) - S(\text{Filiale}) = 10000 + 10000 - 50 = 19950$$

$$B(Z) = 5000000 * 19950 / 20000 = 4987500$$

Dato che non ci sono indici su Z:

$$\text{Costo } \sigma_{\text{Portafoglio} \leq 40000} (Z) = B(Z) = 4987500$$

$$\text{Costo totale} = 3375000 + 4987500 = 8.362.500$$

La sequenza di costo minore è la prima