

7 COMPITO DI SISTEMI INFORMATIVI/BASI DI DATI II

17 giugno 2008 – Compito B (Tot. 16) Tempo: 2h

Esercizio 1 (punti 3)

Si consideri il seguente log:

- | | |
|----------------------|----------------------|
| 1. U(T1,O1,B1,A1) | 16. U(T5,O5,B16,A16) |
| 2. B(T3) | 17. D(T2,O3,B17) |
| 3. D(T3,O1,B3) | 18. B(T6) |
| 4. U(T3,O2,B4,A4) | 19. U(T5,O6,B19,A19) |
| 5. C(T1) | 20. I(T2,O7,A20) |
| 6. I(T3,O3,A6) | 21. U(T6,O2,B21,A21) |
| 7. U(T4,O3,B7,A7) | 22. U(T4,O2,B22,A22) |
| 8. B(T2) | 23. I(T6,O8,B23,A23) |
| 9. I(T2,O4,A9) | 24. D(T4,O5,B24) |
| 10. CK(T2,T3) | 25. C(T4) |
| 11. B(T4) | 26. I(T6,O9,A26) |
| 12. U(T2,O4,B12,A12) | 27. D(T2,O9,B27) |
| 13. I(T3,O5,A13) | 28. C(T2) |
| 14. B(T5) | |
| 15. C(T3) | |

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:

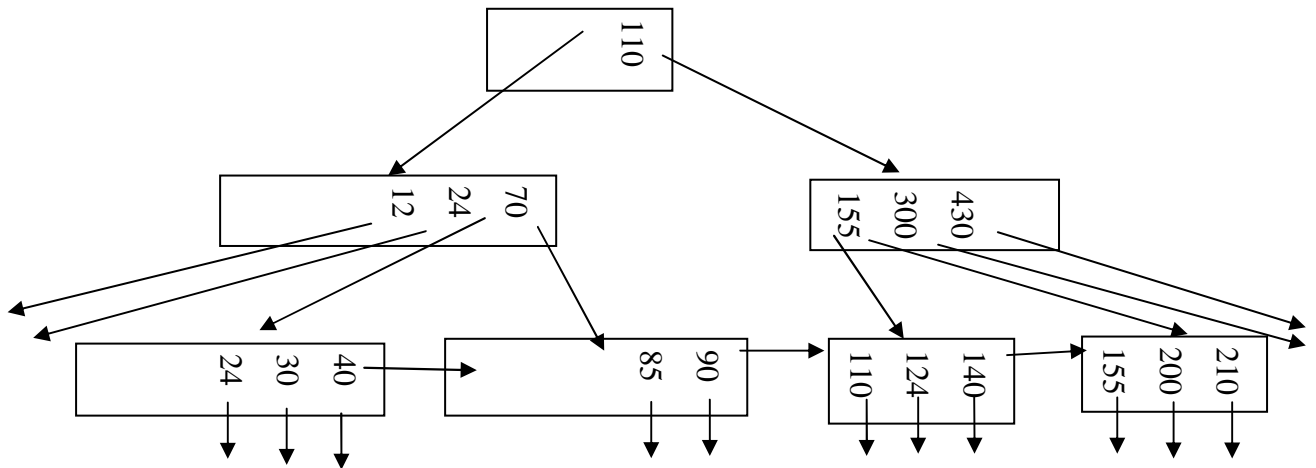
w1(x) r4(y) r2(x) w2(y) w3(x) r3(z) w2(w) r5(w) r5(y) w5(z)

si indichi se è view equivalente al seguente schedale seriale

w1(x) r4(y) r2(x) w2(y) w2(w) w3(x) r3(z) r5(w) r5(y) w5(z)

Esercizio 3 (punti 3)

Sia dato il seguente B+ tree di ordine 3.



Si mostri come si modifica l'albero nel caso di inserimento della chiave 25

Esercizio 4 (punti 6)

Siano date le relazioni

Prodotto(CodiceProd, Nome, Categoria, PuntoVendita)

Transazione(CodiceTra, Cliente, Data, PuntoVendita)

la query

SELECT P.*, C.*

FROM Prodotto AS P, Transazione AS T

WHERE P.PuntoVendita=T.PuntoVendita AND P.Categoria='Farinacei'

e i parametri:

buffer di memoria centrale disponibili: $M=1.000.000$

dimensione del buffer: $B=2.000$ bytes

numero di tuple: $T(\text{Prodotto})=4.000.000$, $T(\text{Transazione})=50.000.000$,

dimensione delle tuple: $S(\text{Prodotto})=1.000$ bytes, $S(\text{Transazione})=400$,

$S(\text{PuntoVendita})=80$

numero di valori $V(\text{Prodotto}, \text{PuntoVendita})=1.000$, $V(\text{Transazione}, \text{PuntoVendita})=230$

$V(\text{Prodotto}, \text{Categoria})=50$

Si calcoli il costo minimo di ciascuna delle seguenti sequenze:

$(\sigma_{\text{Categoria}='Farinacei'} \text{ Prodotto}) \triangleright \triangleleft \text{ Transazione}$

$\sigma_{\text{Categoria}='Farinacei'} (\text{Prodotto} \triangleright \triangleleft \text{ Transazione})$

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 secondario su Prodotto.Categoria e un indice primario su Transazione.PuntoVendita.

SOLUZIONE

Esercizio 1

1. U(T1,O1,B1,A1)
2. B(T3)
3. D(T3,O1,B3)
4. U(T3,O2,B4,A4)
5. C(T1)
6. I(T3,O3,A6)
7. U(T4,O3,B7,A7)
8. B(T2)
9. I(T2,O4,A9)
10. CK(T2,T3)
11. B(T4)
12. U(T2,O4,B12,A12)
13. I(T3,O5,A13)
14. B(T5)
15. C(T3)
16. U(T5,O5,B16,A16)
17. D(T2,O3,B17)
18. B(T6)
19. U(T5,O6,B19,A19)
20. I(T2,O7,A20)
21. U(T6,O2,B21,A21)
22. U(T4,O2,B22,A22)
23. U(T6,O8,B23,A23)
24. D(T4,O5,B24)
25. C(T4)
26. I(T6,O9,A26)
27. D(T2,O9,B27)
28. C(T2)

- 10 UNDO={T2,T3 } REDO={ }
- 11 UNDO={T2,T3,T4 } REDO={ }
- 14 UNDO={T2,T3,T4,T5 } REDO={ }
- 15 UNDO={T2, T4,T5 } REDO={T3 }
- 18 UNDO={T2, T4,T5,T6 } REDO={T3 }
- 25 UNDO={T2,T5,T6 } REDO={T3,T4 }
- 28 UNDO={T5,T6 } REDO={T2,T3,T4 }
- UNDO
- 26 D(O9)
- 23 O8=B23
- 21 O2=B21
- 19 O6=B19
- 16 O5=B16
- REDO
- 3 D(O1)
- 4 O2=A4
- 6 I(O3,A6)
- 7 O3=A7
- 9 I(O4,A9)
- 12 O4=A12
- 13 I(O5,A13)
- 17 D(O3)
- 20 I(O7,A20)
- 22 O2=A22
- 24 D(O5)
- 27 D(O9)

Esercizio 2

w1(x) r4(y) r2(x) w2(y) w3(x) r3(z) w2(w) r5(w) r5(y) w5(z)

"legge-da"={ (r2(x),w1(x)),(r5(w),w2(w)),(r5(y),w2(y)) }

scritture finali={ (t3,x),(t2,y),(t5,z),(t2,w) }

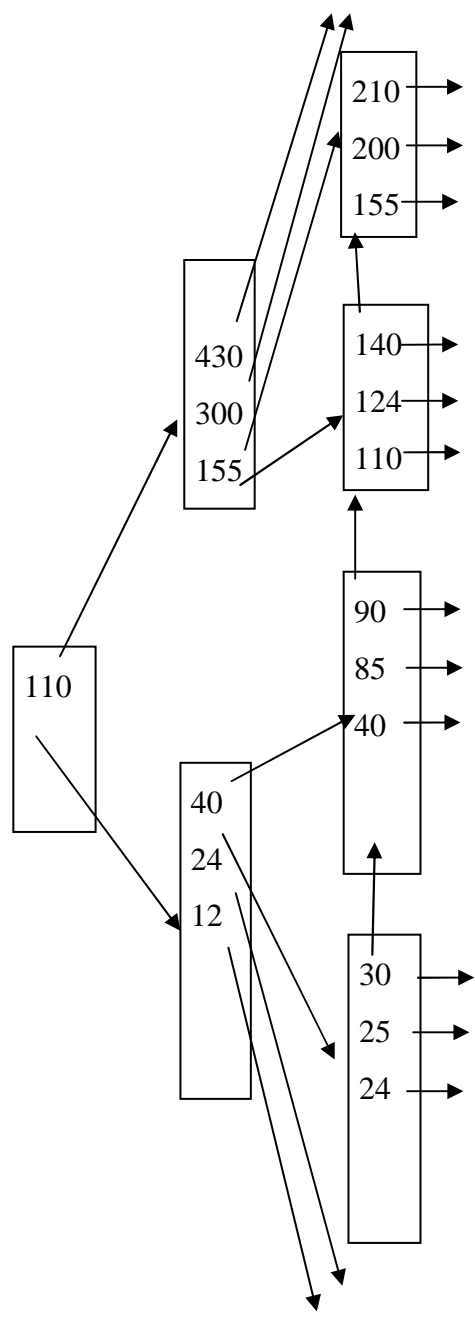
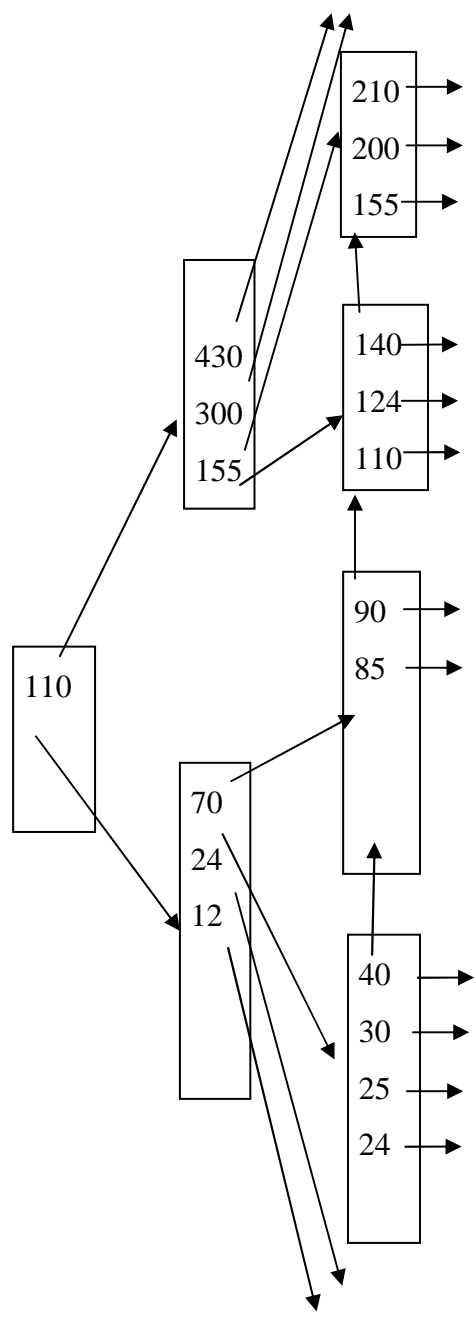
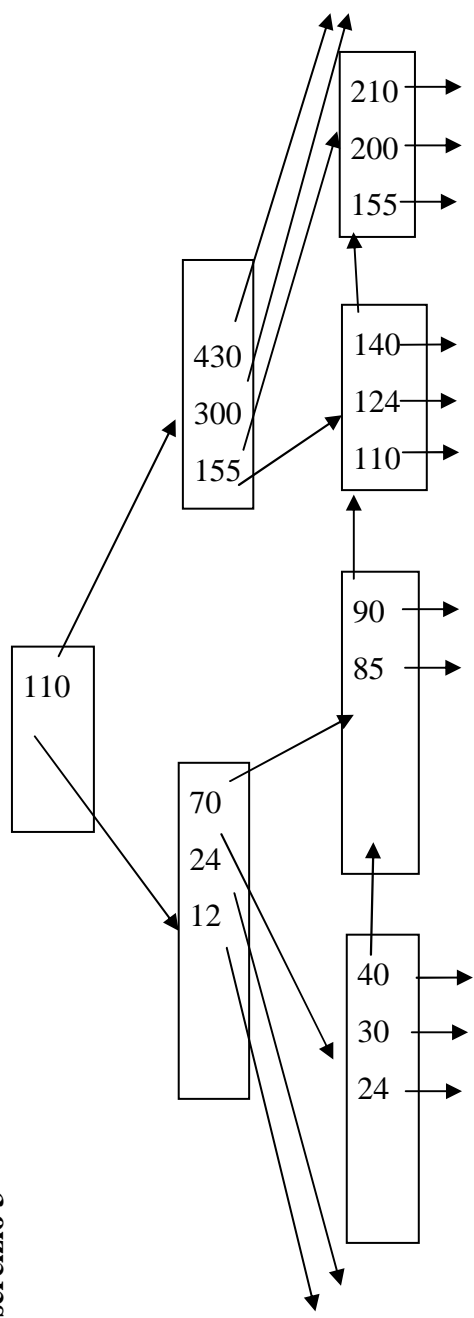
w1(x) r4(y) r2(x) w2(y) w2(w) w3(x) r3(z) r5(w) r5(y) w5(z)

"legge-da"={ (r2(x),w1(x)),(r5(w),w2(w)),(r5(y),w2(y)) }

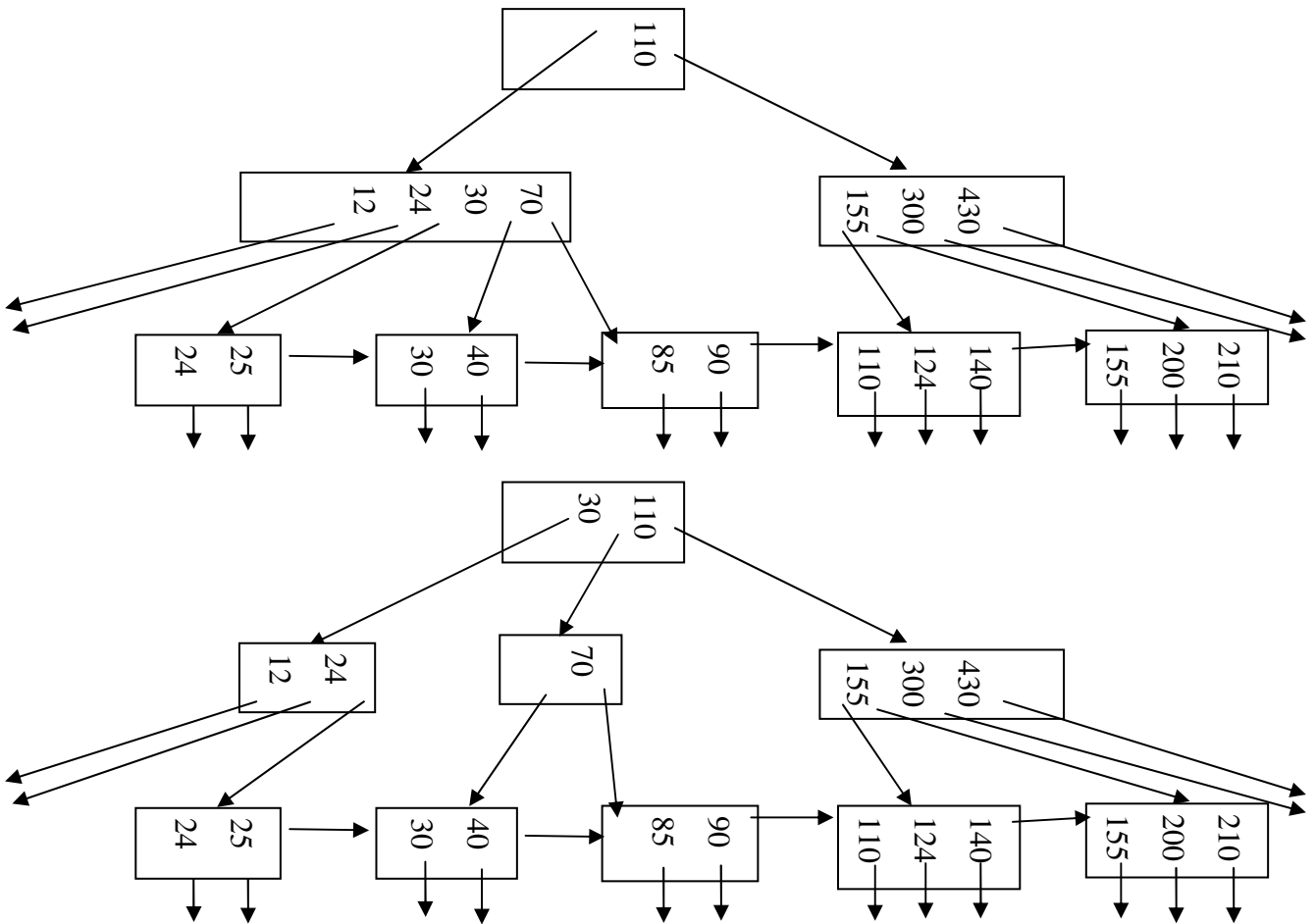
scritture finali={ (t3,x),(t2,y),(t5,z),(t2,w) }

I due schedule sono view-equivalenti

Esercizio 3



Oppure



Esercizio 4

buffer di memoria centrale disponibili: $M=1.000.000$

dimensione del buffer: $B=2.000$ bytes

numero di tuple: $T(\text{Prodotto})= 4.000.000$, $T(\text{Transazione})= 50.000.000$,

dimensione delle tuple: $S(\text{Prodotto})=1.000$ bytes, $S(\text{Transazione})=400$,

$S(\text{PuntoVendita})=80$

numero di valori $V(\text{Prodotto}, \text{PuntoVendita})=1.000$, $V(\text{Transazione}, \text{PuntoVendita})=230$

$V(\text{Prodotto}, \text{Categoria})=50$

Si calcoli il costo minimo di ciascuna delle seguenti sequenze:

$(\sigma_{\text{Categoria}='Farinacei'} \text{Prodotto}) \triangleright \triangleleft \text{Transazione}$

$\sigma_{\text{Categoria}='Farinacei'} (\text{Prodotto} \triangleright \triangleleft \text{Transazione})$

Calcoliamo innanzitutto il numero di blocchi occupati da ciascuna relazione:

$B(\text{Prodotto})=T(\text{Prodotto}) \cdot S(\text{Prodotto})/B = 4000000 \cdot 1000/2000=2.000.000$

$B(\text{Transazione}) = 50000000 \cdot 400/2000=10.000.000$

Prima sequenza: $(\sigma_{\text{Categoria}='Farinacei'} \text{Prodotto}) \triangleright \triangleleft \text{Transazione}$

Sia $X = \sigma_{\text{Categoria}='Farinacei'} \text{Prodotto}$

C'è un indice secondario su Categoria

$$\text{Costo}(X) = 3 + T(\text{Prodotto})/V(\text{Prodotto}, \text{Categoria}) = 3 + 4000000/50 = 80003$$

$$T(X) = 4000000/50 = 80.000$$

$$B(X) = 80000 * 1000 / 2000 = 40.000$$

X sta in memoria centrale, quindi si utilizza il join a un passo

$$\text{Costo}(X \triangleright \triangleleft \text{Transazione}) = B(\text{Transazione}) + B(X) = 10000000 + 40000 = 10.040.000$$

$$\text{Costo totale} = 80.003 + 10.040.000 = 10.120.003$$

Seconda sequenza: $\sigma_{\text{Categoria}='Farinacei'}$ (Prodotto $\triangleright \triangleleft$ Transazione)

Sia $Z = \text{Prodotto} \triangleright \triangleleft \text{Transazione}$

Né Prodotto né Transazione 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 visto che c'è un indice primario su Transazione.PuntoVendita.

$$\begin{aligned} \text{CostoJNL}(Z) &= B(\text{Prodotto}) + B(\text{Transazione}) * B(\text{Prodotto}) / M \\ &= 2000000 + 10000000 * 2000000 / 1000000 = 22.000.000 \end{aligned}$$

Hash join ibrido: si può fare perché $M > \sqrt{B(\text{Prodotto})} \Rightarrow 1000000 > \sqrt{2000000} \Rightarrow 1000000 > 1414$

$$\text{CostoHJI}(Z) = (3 - 2M/B(\text{Prodotto})) * (B(\text{Prodotto}) + B(\text{Transazione})) = (3 - 2 * 1000000 / 2000000) * (2000000 + 10000000) = 24.000.000$$

$$\begin{aligned} \text{CostoIJ}(Z) &= B(\text{Prodotto}) + T(\text{Prodotto}) \lceil B(\text{Transazione}) / V(\text{Transazione}, \text{PuntoVendita}) \rceil = \\ &= 2000000 + 4000000 * \lceil 10000000 / 230 \rceil = 2000000 + 4000000 * \lceil 43478,26087 \rceil = \\ &= 2000000 + 4000000 * 43479 = 173.918.000.000 \end{aligned}$$

$$T(Z) = T(\text{Prodotto}) * T(\text{Transazione}) / \max\{V(\text{Prodotto}, \text{PuntoVendita}), V(\text{Transazione}, \text{PuntoVendita})\} = 4000000 * 50000000 / 1000 = 200.000.000.000$$

$$S(Z) = S(\text{Prodotto}) + S(\text{Transazione}) - S(\text{PuntoVendita}) = 1000 + 400 - 80 = 1.320$$

$$B(Z) = 2000000000000 * 1320 / 2000 = 132.000.000.000$$

Dato che non ci sono indici su Z:

$$\text{Costo: } (\sigma_{\text{Categoria}='Farinacei'} Z) = B(Z) = 132.000.000.000$$

$$\text{Costo totale} = 22000000 + 132000000000 = 132.022.000.000$$