

Esercizi d'esame

Esercizio

- Si calcoli quanto tempo e' necessario per ordinare un file di 100.000 blocchi utilizzando 11 blocchi di memoria centrale

Soluzione

- Numero di passi = $1 + \lceil \log_{B-1} (\lceil N/B \rceil) \rceil$
 $= 1 + \lceil \log_{10} (\lceil 100.000/11 \rceil) \rceil = 1 + \lceil \log_{10} (9091) \rceil =$
 $= 1 + 4 = 5$

Numero di I/O = $5 * 2 * 100.000 = 1.000.000$

Tempo di accesso = $1.000.000 * 11 \text{ ms} = 11.000.000$
 $\text{ms} = 11.000 \text{ s} = 183,3 \text{ minuti}$

2

Esercizio

- Si abbiano le 4 relazioni E, F, G, H con le seguenti statistiche

E(a,b,c)	F(a,b,d)	G(a,c,d)	H(b,c,d)
T(E)=1000	T(F)=2000	T(G)=3000	T(H)=4000
V(E,a)=1000	V(F,a)=50	V(G,a)=50	V(H,b)=40
V(E,b)=50	V(F,b)=100	V(G,c)=300	V(H,c)=100
V(E,c)=20	V(F,d)=200	V(G,d)=500	V(H,d)=400

- Si calcoli la dimensione di $E \bowtie F \bowtie G \bowtie H$

Esercizio: soluzione

E(a,b,c)	F(a,b,d)	G(a,c,d)	H(b,c,d)
T(E)=1000	T(F)=2000	T(G)=3000	T(H)=4000
V(E,a)=1000	V(F,a)=50	V(G,a)=50	V(H,b)=40
V(E,b)=50	V(F,b)=100	V(G,c)=300	V(H,c)=100
V(E,c)=20	V(F,d)=200	V(G,d)=500	V(H,d)=400

- $T(E \bowtie F \bowtie G \bowtie H) =$
 $= \frac{1000 \times 2000 \times 3000 \times 4000}{1000 \times 50 \times 50 \times 100 \times 300 \times 100 \times 500 \times 400} = \frac{1}{62500}$

3

4

Esercizio

- Data la tabella R(a,b,c) si indichi qual'è il migliore algoritmo per eseguire la query

```
SELECT * FROM R WHERE a=cost1 or b=cost2
```

- nel caso in cui ci siano indici secondari su a e b,
 $B(R)=1000$, $T(R)=20.000$, $V(R,a)=100$, $V(R,b)=50$

Soluzione

- Costo della scansione=1000
- Costo dell'accesso tramite indici: $3+20.000/100$
 $+3+20.000/50=3+200+3+400=606$
- Conviene l'accesso tramite indici

5

Esercizio

- Date le relazioni

Impiegato(CodImp,CodDip,Nome,Cognome,Data_di_nascita,
Stipendio)

Dipartimento(CodDip,Nome,Indirizzo, N_affidenti)

- Si calcoli il costo di esecuzione della query

```
SELECT I.Nome, I.Cognome, D.Nome  
FROM Impiegato AS I, Dipartimento AS D
```

```
WHERE I.CodDip=D.CodDip
```

- supponendo che vengano utilizzati gli algoritmi di Block-Based Nested-Loops Join, Sort Join, Hash Join Ibrido e Join con Indice.
- Si supponga di avere $M=100$ buffer di memoria centrale disponibili, che $B(\text{Impiegato})=1500$, $T(\text{Impiegato})=15.000$, $B(\text{Dipartimento})=200$, $T(\text{Dipartimento})=2000$ e si supponga di avere un indice primario su Dipartimento.CodDip.

6

Esercizio: soluzione

- Block-Based Nested-Loops Join:
 - Costo= $B(\text{Dipartimento})+B(\text{Dipartimento})B(\text{Impiegato})/(M-1)$
 $= 200 + 200 * 1500/100=3200$
- Sort Join:
 - può essere fatto se $B(\text{Dipartimento})+B(\text{Impiegato}) \leq M^2$
 $B(\text{Dipartimento})+B(\text{Impiegato}) \leq 10.000$ $1700 \leq 10.000$ ok
 - Costo= $3(B(\text{Dipartimento})+B(\text{Impiegato}))=3(1500+200)=5100$
- Hash Join Ibrido:
 - può essere fatto se $B(\text{Dipartimento}) \leq M^2$
 $200 \leq 10.000$ ok
 - Costo= $(3-2M/B(\text{Dipartimento}))(B(\text{Impiegato})+B(\text{Dipartimento}))=$
 $(3-2 * 100/200)(1500+200)=(3-1)(1700)=3400$
- Index Join:
 - Costo= $B(\text{Impiegato})+T(\text{Impiegato}) * 1=1.500+15.000=16.500$

7

Esercizio (29/11/05)

- Siano date le relazioni

Cliente(CodCli,Nome,Cognome,DataDiNascita,Città)

Venditore(CodVend,Nome,Cognome,Città,Auto)

Supervisore(CodSup,Nome,Cognome,Auto)

- la query

```
SELECT C.*, V.*, S.*
```

```
FROM Cliente AS C, Venditore AS V, Supervisore AS S
```

```
WHERE C.Città=V.Città AND V.Auto=S.Auto
```

- e i parametri:

- buffer di memoria centrale disponibili: $M=100$
- dimensione del buffer: $B=500$ bytes
- numero di tuple: $T(\text{Cliente})=20.000$, $T(\text{Venditore})=1.000$,
 $T(\text{Supervisore})=500$
- dimensione delle tuple: $S(\text{Cliente})=200$ bytes, $S(\text{Venditore})=125$,
 $S(\text{Supervisore})=125$, $S(\text{Città})=25$, $S(\text{Auto})=30$
- numero di valori: $V(\text{Cliente,Città})=1.000$, $V(\text{Venditore,Città})=500$,
 $V(\text{Venditore,Auto})=100$, $V(\text{Supervisore,Auto})=80$

8

Esercizio

- Si stabilisca qual'è l'ordine migliore con cui eseguire i singoli join supponendo di utilizzare sempre l'Hash Join Ibrido. In particolare, occorrerà calcolare il costo delle due sequenze:
(Cliente)▷(Venditore)▷(Supervisore e
(Venditore)▷(Supervisore)▷(Cliente).
- Si supponga che i record siano impaccati nei buffer e che tutto lo spazio nei buffer sia occupato dai record. Quindi, il numero di blocchi di una relazione è dato da $B(R)=T(R)*S(R)/B$.

9

Esercizio: soluzione

Calcoliamo innanzitutto il numero di blocchi occupati da ciascuna relazione:

$$B(\text{Cliente})=T(\text{Cliente})*S(\text{Cliente})/B \\ =20.000*200/500=8.000$$

$$B(\text{Venditore})=1.000*125/500=250$$

$$B(\text{Supervisore})=500*125/500=125$$

10

Esercizio: soluzione

Prima sequenza: (Cliente)▷(Venditore)▷(Supervisore)
Costo((Cliente)▷(Venditore))=(3-2*100/250)*(8.000+250)=18.150
Sia X=(Cliente)▷(Venditore)
 $S(X)=S(\text{Cliente})+S(\text{Venditore})-S(\text{Città})=200+125-25=300$
 $T(X)=T(\text{Cliente})*T(\text{Venditore})/\max(V(\text{Cliente},\text{Città}),V(\text{Venditore},\text{Città}))=20.000*1.000/1.000=20.000$
 $B(X)=20.000*300/500=12.000$
Costo(X)▷(Supervisore)=(3-2*100/125)(125+12.000)=16.975
Costo((Cliente)▷(Venditore)▷(Supervisore))=18.150+16.975=35.125

11

Esercizio: soluzione

Seconda sequenza: (Venditore)▷(Supervisore)▷(Cliente)
Costo((Venditore)▷(Supervisore))=(3-2*100/125)*(250+125)=525
Sia Z=(Venditore)▷(Supervisore)
 $S(Z)=125+125-30=220$
 $T(Z)=1.000*500/100=5.000$
 $B(Z)=5000*220/500=2.200$
Costo(Z)▷(Cliente)=(3-2*100/2200)(2200+8000)=29.673
Costo((Venditore)▷(Supervisore)▷(Cliente))=525+29.673=30.198
La sequenza migliore per realizzare il join è la seconda sequenza.

12

Esercizio (9/1/06)

- Siano date le relazioni
Impiegato(CodImp, Nome, Cognome, DataDiNascita, Dipartimento, Stipendio)
Consulente(CodCons, Nome, Cognome, Dipartimento, Stipendio, Indirizzo)
- la query
SELECT I.*, C.*
FROM Impiegato AS I, Consulente AS C
WHERE I.Dipartimento=C.Dipartimento AND I.Nome="Francesco"
- e i parametri:
 - buffer di memoria centrale disponibili: $M=100$
 - dimensione del buffer: $B=500$ bytes
 - numero di tuple: $T(\text{Impiegato})=250.000$, $T(\text{Consulente})=800$,
 - dimensione delle tuple: $S(\text{Impiegato})=160$ bytes, $S(\text{Consulente})=250$, $S(\text{Dipartimento})=50$
 - numero di valori: $V(\text{Impiegato}, \text{Dipartimento})=2.000$, $V(\text{Consulente}, \text{Dipartimento})=500$, $V(\text{Impiegato}, \text{Nome})=400$

13

Esercizio

- Si stabilisca qual'è l'ordine migliore con cui eseguire le operazioni supponendo di utilizzare l'Hash Join Ibrido. In particolare, occorrerà calcolare il costo delle due sequenze:
 $(\sigma_{\text{Impiegato.Nome}='Francesco'}(\text{Impiegato}) \triangleright \triangleleft \text{Consulente})$ e $\sigma_{\text{Impiegato.Nome}='Francesco'}(\text{Impiegato}) \triangleright \triangleleft \text{Consulente}$.
- Si supponga che i record siano impaccati nei buffer e che tutto lo spazio nei buffer sia occupato dai record. Si supponga inoltre che ci sia un indice secondario su Impiegato.Nome. Si supponga che non vengano costruiti indici su Impiegato $\triangleright \triangleleft$ Consulente.

14

Esercizio: soluzione

Calcoliamo innanzitutto il numero di blocchi occupati da ciascuna relazione:

$$B(\text{Impiegato}) = T(\text{Impiegato}) * S(\text{Impiegato}) / B \\ = 250.000 * 160 / 500 = 80.000$$

$$B(\text{Consulente}) = 800 * 250 / 500 = 400$$

Prima sequenza: $(\sigma_{\text{Impiegato.Nome}='Francesco'}(\text{Impiegato}) \triangleright \triangleleft \text{Consulente})$

Sia $X = \sigma_{\text{Impiegato.Nome}='Francesco'}(\text{Impiegato})$

$$\text{Costo}(X) = 3 + T(\text{Impiegato}) / V(\text{Impiegato}, \text{Nome}) = 3 + 250.000 / 400 \\ = 3 + 625 = 628.$$

$$T(X) = 250.000 / 400 = 625$$

$$B(X) = 625 * 160 / 500 = 200$$

$$\text{Costo}(X \triangleright \triangleleft \text{Consulente}) = (3 - 2 * 100 / 200) * (200 + 400) = 1.200$$

$$\text{Costo}((\sigma_{\text{Impiegato.Nome}='Francesco'}(\text{Impiegato}) \triangleright \triangleleft \text{Consulente}) = 628 + 1.200 \\ = 1.828$$

15

Esercizio: soluzione

Seconda sequenza:

$$\sigma_{\text{Impiegato.Nome}='Francesco'}(\text{Impiegato}) \triangleright \triangleleft \text{Consulente}$$

Sia $Y = \text{Impiegato} \triangleright \triangleleft \text{Consulente}$

$$\text{Costo}(Y) = (3 - 2 * 100 / 400) * (80.000 + 400) = 201.000$$

$$S(Y) = S(\text{Impiegato}) + S(\text{Consulente}) - S(\text{Dipartimento}) = 160 + 250 - 50 = 360$$

$$T(Y) = T(\text{Impiegato}) * T(\text{Consulente}) / \max(V(\text{Impiegato}, \text{Dipartimento}), \\ V(\text{Consulente}, \text{Dipartimento})) = 250.000 * 800 / 2.000 = 100.000$$

$$B(Y) = 100.000 * 360 / 500 = 72.000$$

$$\text{Costo}(\sigma_{\text{Impiegato.Nome}='Francesco'}(Y)) = B(Y) = 72.000$$

$$\text{Costo}(\sigma_{\text{Impiegato.Nome}='Francesco'}(\text{Impiegato}) \triangleright \triangleleft \text{Consulente}) = 72.000 + 20 \\ 1.000 = 273.000$$

La sequenza migliore per realizzare l'operazione è la prima.

16

Esercizio (12/7/08)

- Siano date le relazioni
CD(CodiceCD, CodiceMus, Titolo, Anno, Durata)
Musicista(CodiceMus, CasaDiscografica, Nazionalità, Età)
- la query
SELECT C.*, M.*
FROM CD AS C, Musicista AS M
WHERE C.CodiceMus=M.CodiceMus AND C.Durata>30 AND
C.Durata<70
- e i parametri:
 - buffer di memoria centrale disponibili: M=20.000
 - dimensione del buffer: B=4.000 bytes
 - numero di tuple: T(CD)= 83102, T(Musicista)= 40000,
 - dimensione delle tuple: S(CD)=2000 bytes, S(Musicista)=3000,
 - S(CodMusista)=200
 - numero di valori: V(CD,CodiceMus)=40000,
 - Min(CD,Durata)=25 Max(CD,Durata)=80

17

Esercizio

- Si calcoli il costo minimo di ciascuna delle seguenti sequenze:
 $(\sigma_{Durata>30 \text{ AND } Durata<70} CD) \triangleright \triangleleft Musicista$
 $\sigma_{Durata>30 \text{ AND } Durata<70} (CD \triangleright \triangleleft Musicista)$
- Si supponga che i record siano impaccati nei buffer e che tutto lo spazio nei buffer sia occupato dai record. Si supponga che CD e Musicista siano ordinate sulla base del campo CodiceMus. Se ne tenga conto nel calcolo del costo del join.
- Si supponga di avere un indice secondario su CD.Durata. Per il calcolo del costo e della cardinalità della selezione su Durata si utilizzi la stima dei valori nel range. Si noti che CodiceMus è chiave primaria di Musicista.

18

Esercizio: soluzione

$B(CD) = T(CD) * S(CD) / B = 83102 * 2000 / 4000 = 41551$
 $B(Musicista) = 40000 * 3000 / 4000 = 30000$
Prima sequenza: $(\sigma_{Durata>30 \text{ AND } Durata<70} CD) \triangleright \triangleleft Musicista$
Sia $X = \sigma_{Durata>30 \text{ AND } Durata<70} CD$
Utilizzando l'indice secondario su Durata
 $f = (70 - 30 - 1) / (80 - 25 - 1) = 0.722$
 $Costo(X) = 3 + f * T(CD) = 3 + 0.722 * 83102 = 60003$
 $T(X) = f * T(CD) = 0.722 * 83102 = 60000$
 $B(X) = 60000 * 2000 / 4000 = 30000$

19

Esercizio: soluzione

Né X né Musicista stanno in memoria centrale, quindi si prova con il join nested-loop basato sui blocchi e con l'hash join ibrido. Inoltre X non è ordinato su CodiceMus
 $Costo_{JNL}(X \triangleright \triangleleft Musicista) = B(Musicista) + B(X) * B(Musicista) / M = 30000 + 30000 * 30000 / 20000 = 75000$
 $Costo_{HJI}(X \triangleright \triangleleft Musicista) = (3 - 2M/B(Musicista)) * (B(X) + B(Musicista)) = (3 - 2 * 20000 / 30000) * (30000 + 30000) = 100000$
Costo totale = $60003 + 75000 = 135.003$

20

Esercizio: soluzione

Seconda sequenza: $\sigma_{Durata>30 \text{ AND } Durata<70} (CD \bowtie Musicista)$

Sia $Z = CD \bowtie Musicista$

CD e Musicista sono ordinati quindi posso usare il sort-based join (sort join senza il passo preliminare di ordinamento)

$Costo_{SJ}(Z) = B(CD) + B(Musicista) = 41551 + 30000 = 71551$

$T(Z) = T(CD) * T(Musicista) / \max\{V(CD, CodiceMus), V(Musicista, CodiceMus)\} =$

$83102 * 40000 / \max\{40000, 40000\} = 83102$

21

Esercizio: soluzione

$S(Z) = S(CD) + S(Musicista) - S(CodiceMus) = 2000 + 3000 - 200 = 4800$

$B(Z) = 83102 * 4800 / 4000 = 99722$

Dato che non ci sono indici su Z:

$Costo(\sigma_{Durata>30 \text{ AND } Durata<70} (CD \bowtie Musicista)) = B(Z) = 99722$

Costo totale = $71551 + 99722 = 171273$

22

Esercizio (13/2/08)

- Siano date le relazioni

Paziente(CodicePaz, Nome, Cognome, Età, CodiceMed)

Medico(CodiceMed, Nome, Cognome, Indirizzo)

- la query

```
SELECT P.*, C.*
```

```
FROM Paziente AS P, Medico AS M
```

```
WHERE P.CodiceMed=M.CodiceMed AND P.Età>=10  
AND P.Età<=20
```

23

Esercizio

- e i parametri:

– buffer di memoria centrale disponibili: $M=1000$

– dimensione del buffer: $B=1.000$ bytes

– numero di tuple: $T(\text{Paziente})=80000$, $T(\text{Medico})=10000$,

– dimensione delle tuple: $S(\text{Paziente})=500$ bytes, $S(\text{Medico})=400$,

– $S(\text{CodiceMed})=50$

- Si calcoli il costo minimo di ciascuna delle seguenti sequenze:

• $\sigma_{Età>=10 \text{ AND } Età<=20} (\text{Paziente} \bowtie \text{Medico})$

• $(\sigma_{Età>=10 \text{ AND } Età<=20} \text{Paziente}) \bowtie \text{Medico}$

24

Esercizio

- 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 Medico.CodiceMed e un indice secondario su Paziente.Età. Per il calcolo del costo e della cardinalità della selezione su Età si utilizzi la stima dei valori nel range, sapendo che il range di Età è [1,105].

25

Esercizio: soluzione

$$B(\text{Paziente}) = T(\text{Paziente}) * S(\text{Paziente}) / B \\ = 80000 * 500 / 1000 = 40000$$

$$B(\text{Medico}) = 10000 * 400 / 1000 = 4000$$

$$\text{Prima sequenza: } (\sigma_{\text{Età} \geq 10 \text{ AND } \text{Età} \leq 20} \text{ Paziente}) \triangleright \triangleleft \\ \text{Medico}$$

$$\text{Sia } X = \sigma_{\text{Età} \geq 10 \text{ AND } \text{Età} \leq 20} \text{ Paziente}$$

C'e' un indice secondario su Paziente.Età

$$F = (20 - 10 + 1) / (105 - 1 + 1) = 11 / 105 = 0.105$$

$$\text{Costo}(X) = 3 + f * T(\text{Paziente}) = 3 + 0.105 * 80000 = 8403$$

$$T(X) = 0.105 * 80000 = 8400$$

$$B(X) = 0.105 * 40000 = 4200$$

26

Esercizio: soluzione

Né X né Medico 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 su CodiceMed.

$$\text{CostoJNL}(X \triangleright \triangleleft \text{Medico}) = \\ B(\text{Medico}) + B(X) * B(\text{Medico}) / M = 4000 + 4200 * 4000 / 1000 = 20200$$

$$\text{CostoHJI}(X \triangleright \triangleleft \text{Medico}) = (3 - 2M / B(\text{Medico})) (B(X) + B(\text{Medico})) = (3 - \\ 2 * 1000 / 4000) * (4000 + 4200) = 20500$$

$$\text{CostoIJ}(X \triangleright \triangleleft \text{Medico}) = B(X) + T(X) * \\ \lceil B(\text{Medico}) / V(\text{Medico}, \text{CodiceMed}) \rceil = 4200 + 8400 * 1 = 12600$$

$$\text{Costo totale} = 8403 + 12600 = 21.003$$

27

Esercizio: soluzione

Seconda sequenza: $\sigma_{\text{Età} \geq 10 \text{ AND } \text{Età} \leq 20} (\text{Paziente} \triangleright \triangleleft \text{Medico})$

Sia Z = Paziente $\triangleright \triangleleft$ Medico

Né Paziente né Medico 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 su CodiceMed.

$$\text{CostoJNL}(Z) = B(\text{Medico}) + B(\text{Paziente}) * B(\text{Medico}) / M = \\ 4000 + 4000 * 40000 / 1000 = 164.000$$

$$\text{CostoHJI}(Z) = (3 - 2M / B(\text{Medico})) (B(\text{Medico}) + B(\text{Paziente})) = \\ (3 - 2 * 1000 / 4000) * (4000 + 40000) = 110.000$$

$$\text{CostoIJ}(Z) = B(\text{Paziente}) + T(\text{Paziente}) \\ \lceil B(\text{Medico}) / V(\text{Medico}, \text{CodiceMed}) \rceil = 40000 + 80000 * 1 = 120.000$$

28

Esercizio: soluzione

$$T(Z) = T(\text{Medico}) * T(\text{Paziente}) / \max\{V(\text{Medico}, \text{CodiceMed}), V(\text{Paziente}, \text{CodiceMed})\} \\ = 80000 * 10000 / 10000 = 80000$$

$$S(Z) = S(\text{Medico}) + S(\text{Paziente}) - S(\text{CodiceMed}) = 400 + 500 - 200 = 700$$

$$B(Z) = 80000 * 700 / 1000 = 56000$$

Dato che non ci sono indici su Z:

$$\text{Costo: } (\sigma_{\text{Età} \geq 10 \text{ AND } \text{Età} \leq 20} Z) = B(Z) = 56000$$

$$\text{Costo totale} = 56000 + 110000 = 166.000$$

29

Esercizio

- Dato il seguente log
- | | |
|-------------------|---------------------|
| 1. B(T1) | 11. U(T4,O6,B4,A5) |
| 2. U(T1,O1,B1,A1) | 12. I(T4,O7,A6) |
| 3. B(T2) | 13. U(T4,O2,B5,A7) |
| 4. I(T1,O2,A2) | 14. C(T3) |
| 5. B(T3) | 15. I(T2,O8,A9) |
| 6. D(T3,O3,B2) | 16. A(T1) |
| 7. U(T2,O4,B3,A3) | 17. U(T4,O3,B7,A10) |
| 8. CK(T1,T2,T3) | 18. guasto |
| 9. I(T3,O5,A4) | |
| 10. B(T4) | |

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

30

Esercizio: soluzione

- | | |
|---------------------|--------------------------------------|
| 1. B(T1) | 8 UNDO={T1,T2,T3} REDO={} |
| 2. U(T1,O1,B1,A1) | 10 B(T4)->UNDO={T1,T2,T3,T4} REDO={} |
| 3. B(T2) | 14 C(T3)->UNDO={T1,T2,T4} REDO={T3} |
| 4. I(T1,O2,A2) | UNDO |
| 5. B(T3) | 17 O3=B7 |
| 6. D(T3,O3,B2) | 15 D(O8) |
| 7. U(T2,O4,B3,A3) | 13 O2=B5 |
| 8. CK(T1,T2,T3) | 12 D(O7) |
| 9. I(T3,O5,A4) | 11 O6=B4 |
| 10. B(T4) | 7 O4=B3 |
| 11. U(T4,O6,B4,A5) | 4 D(O2) |
| 12. I(T4,O7,A6) | 2 O1=B1 |
| 13. U(T4,O2,B5,A7) | REDO |
| 14. C(T3) | 6 D(O3) |
| 15. I(T2,O8,A9) | 9 I(O5,A4) |
| 16. A(T1) | |
| 17. U(T4,O3,B7,A10) | |

31

Esercizio

- Dire se il seguente schedule
 $S1 = r1(a), r2(a), r3(a), w1(b), w2(b), w3(b)$
 - e' view-equivalente al seguente schedule seriale
 $S2 = r2(a), w2(b), r1(a), w1(b), r3(a), w3(b)$
- Soluzione
- Si', i due schedule sono view equivalenti.
 - In S1 la relazione legge-da e' vuota e le scritture finali comprendono solo la scrittura di b da parte di T3. S2 ha la stessa relazione legge-da e la stessa scrittura finale su b.

32

Esercizio

- Dire se il seguente schedule
 $S1=r1(a), r3(d), w1(b), r2(b), w3(b), r4(b), w2(c), r5(c), w4(e), r5(e), w5(b)$
- E' view-equivalente allo schedule seriale:
 $S2=r1(a), w1(b), r2(b), w2(c), r3(d), w3(b), r4(b), w4(e), r5(c), r5(e), w5(b)$

Soluzione

- La relazione legge-da per S1 ha le seguenti coppie: $\{(r2(b),w1(b)),(r4(b),w3(b)),(r5(c),w2(c)),(r5(e),w4(e))\}$
- e le seguenti scritture finali: $\{w5(b), w2(c), w4(e)\}$
- S2 ha la stessa relazione legge-da e le stesse scritture finali

33

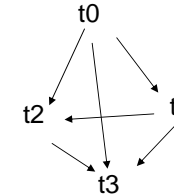
Esercizio

Dato il seguente schedule:

$w0(x) r1(x) r0(y) w1(y) w1(x) r2(x) w3(x) r2(y)$

si costruisca il grafo dei conflitti e si dica se e' conflict serializzabile.

Soluzione



Lo schedule e' conflict serializzabile

34

Esercizio

- Dire se il seguente schedule e' in 2PL
 $S1 = r1(x) r2(y) w1(z) r2(x) r3(x) w2(z) w2(y) r1(y)$
- Nel caso rispetti il 2PL, si indichi una sequenza di acquisizione e rilascio di lock che rispetti sia lo schedule che il two-phase locking. Nel caso non lo rispetti, si indichi almeno uno dei conflitti.

35

Soluzione

$r1(x) r2(y) w1(z) r2(x) r3(x) w2(z) w2(y) r1(y)$

Transazione 1	Transazione 2	Transazione 3
r_lock(x)		
	r_lock(y)	
w_lock(z)		
unlock(x)		
	r_lock(x)	
		r_lock(x)
unlock(z)		
	w_lock(z)	
	w_lock(y)	
	unlock(y)	
r_lock(y) -> NO		

Lo schedule non rispetta il two-phase locking. Infatti poichè sia possibile $w2(z)$, la transazione 1 deve rilasciare il lock esclusivo su z, ma non lo può fare perchè deve effettuare $r1(y)$ alla fine dello schedule e deve acquisire il lock in lettura su y dopo $w2(y)$

36

Esercizio (13/6/07)

- Dato il seguente schedule:
- r2(x) w3(y) w1(x) r5(p) w5(x) w2(z) r4(w) r3(z) r5(y) w4(p)
- 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.

37

Soluzione

r2(x) w3(y) w1(x) r5(p) w5(x) w2(z) r4(w) r3(z) r5(y) w4(p)
 e' in 2PL perche' ha la seguente sequenza di acquisizione e rilascio di lock

Transazione 1	Transazione 2	Transazione 3	Transazione 4	Transazione 5
	r_lock(x)			
		w_lock(y)		
	w_lock(z)			
	unlock(x)			
w_lock(x)				
unlock(x)				
				r_lock(p)
				w_lock(x)
	unlock(z)			
			r_lock(w)	

38

Soluzione

r2(x) w3(y) w1(x) r5(p) w5(x) w2(z) r4(w) r3(z) r5(y) w4(p)

Transazione 1	Transazione 2	Transazione 3	Transazione 4	Transazione 5
		r_lock(z)		
		unlock(y)		
				r_lock(y)
				unlock(p)
			w_lock(p)	
			unlock(p)	
		unlock(z)		
			unlock(w)	
				unlock(x)
				unlock(y)

39

Esercizio

- Dire se il seguente schedule e' in 2PL
 S1=r1(a), r3(d), w1(b), r2(b), w3(b), r4(b), w2(c), r5(c), w4(e), r5(e), w5(b)
- Nel caso lo sia, si indichi una sequenza di acquisizione e rilascio di lock compatibile con lo schedule e che rispetti il two-phase locking

40

Soluzione

r1(a), r3(d), w1(b), r2(b), w3(b), r4(b), w2(c), r5(c), w4(e), r5(e), w5(b)
 S1 e' in 2PL perche' ha la seguente sequenza di acquisizione e rilascio di lock

Transazione 1	Transazione 2	Transazione 3	Transazione 4	Transazione 5
r_lock(a)				
		r_lock(d)		
w_lock(b)				
unlock(b)				
unlock(a)				
	r_lock(b)			
	w_lock(c)			
	unlock(b)			
		w_lock(b)		

41

Soluzione

r1(a), r3(d), w1(b), r2(b), w3(b), r4(b), w2(c), r5(c), w4(e), r5(e), w5(b)

Transazione 1	Transazione 2	Transazione 3	Transazione 4	Transazione 5
		unlock(b)		
		unlock(d)		
			r_lock(b)	
	unlock(c)			
				r_lock(c)
			w_lock(e)	
			unlock(b)	
			unlock(e)	
				r_lock(e)
				w_lock(b)
				unlock(e)
				unlock(b)

42

Esercizio

- Dato il seguente database multidimensionale

Vendite mens.	Gen 03	Feb 03	Mar 03	Apr 03	Mag 03
Mocassino	165	178	193	205	244
Polacchino	154	201	212	245	255
Nike Jordan	54	88	45	24	65
Puma Buffon	56	64	55	52	64

- Si esegua una operazione di roll-up sulla tabella passando al livello successivo nella gerarchia della dimensione prodotto, sapendo che al livello successivo Mocassino e Polacchino fanno parte di Scarpa elegante e Nike Jordan e Puma Buffon fanno parte di Scarpa sportiva

43

Esercizio: soluzione

Vendite mens.	Gen 03	Feb 03	Mar 03	Apr 03	Mag 03
Scarpa elegante	319	379	405	450	499
Scarpa sportiva	110	152	100	76	129

44

Esercizio

- Dato il seguente database

Transaction ID	Purchased Items
1	{1, 5, 6}
2	{2, 3}
3	{4, 5}
4	{2, 5, 6}
5	{1, 5, 6}

- e la seguente regola associativa:
- $5, 6 \Rightarrow 1$
- si calcoli il supporto e la confidenza della regola

45

Esercizio: soluzione

Transaction ID	Purchased Items
1	{1, 5, 6}
2	{2, 3}
3	{4, 5}
4	{2, 5, 6}
5	{1, 5, 6}

- $5, 6 \Rightarrow 1$
- supporto=supporto({1,5,6})=2/5=0,4
- confidenza=2/3=0,66

46

Esercizio

- Si consideri il problema di effettuare il clustering di punti in R utilizzando l'algoritmo EM.
- Si supponga di essere nel passo di Expectation, che i cluster cercati siano due e che i valori dei parametri siano:
- $\mu_A=4, \sigma_A=2$ e $\mu_B=8, \sigma_B=3, p_A=0.4$
- Si calcoli la probabilita' di appartenenza al cluster A dell'istanza $x=6$

47

Esercizio: soluzione

$$\Pr(A|x) = \frac{\Pr(x|A)\Pr(A)}{\Pr(x)} = \frac{f(x; \mu_A, \sigma_A)p_A}{\Pr(x)}$$

con

$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$\Pr(A|x)P(x) = f(6; 4, 2) * 0.4 = 0,068$$

$$\Pr(B|x)P(x) = f(6; 8, 3) * 0.6 = 0,111$$

$$\Pr(A|x) = 0,068 / (0,068 + 0,111) = 0,382$$

48