

# Tecnologie delle Basi di Dati M

Appello del 24/7/2012

## Esercizio 1 (3 punti)

Sia dato il seguente elenco di record presenti nel log:

1. BEGIN(T1)
2. UPDATE(T1, PA, VA1, VA2)
3. BEGIN(T2)
4. UPDATE(T2, PB, VB1, VB2)
5. COMMIT(T2)
6. CHECKPOINT
7. END(T2)
8. UPDATE(T1, PC, VC1, VC2)
9. BEGIN(T3)
10. UPDATE(T3, PD, VD1, VD2)
11. COMMIT(T1)

si illustrino le azioni compiute al restart del sistema da un sistema che adotti una politica *Undo-Redo*, mostrando in particolare i record aggiunti al log

## Esercizio 2 (4 punti)

Date le relazioni con schema:

Impiegati(matricola, nome, cognome, residenza, stipendio)

Progetti(codice, nome, sede)

Assegnamenti(matricola, codice, mesi, funzione)

si ottimizzi l'esecuzione della seguente interrogazione SQL (si considerino solamente i piani che verrebbero generati da un ottimizzatore che utilizzi unicamente piani *left-deep*):

```
SELECT I.nome, I.cognome, A.codice, A.funzione
FROM Impiegati I, Assegnamenti A
WHERE A.matricola=I.matricola
      AND A.mesi>36
      AND I.residenza in ('Bologna', 'Milano')
```

tenendo conto che dai cataloghi della base di dati risulta:

- Numero di tuple Impiegati = 10K, numero di pagine Impiegati = 1K
- Numero di tuple Progetti = 100, numero di pagine Progetti = 10
- Numero di tuple Assegnamenti = 20K, numero di pagine Assegnamenti = 5K
- Indice unclustered su I.matricola: numero foglie = 100
- Indice clustered su I.residenza: numero foglie = 40, numero chiavi = 100
- Indice unclustered (TID ordinate) su A.mesi: numero foglie = 10, valore minimo = 1, valore massimo = 48
- Indice unclustered (TID ordinate) su A.matricola: numero foglie = 200

Si disegni infine l'albero corrispondente al piano di accesso di costo minimo e stimi il numero di risultati dell'interrogazione.

Suggerimento: per la formula di Cardenas si utilizzino i seguenti valori, validi per  $P = 5000$ :

R	$\Phi(R, 5000)$
500	475.86
1000	906.43
1500	1296.02
2000	1648.53
2500	1967.50
3000	2256.11
3500	2517.25
4000	2753.53
4500	2967.33
5000	3160.79

R	$\Phi(R, 5000)$
5500	3335.83
6000	3494.21
6500	3637.52
7000	3767.19
7500	3884.52
8000	3990.68
8500	4086.74
9000	4173.65
9500	4252.30
10000	4323.46

R	$\Phi(R, 5000)$
10500	4387.85
11000	4446.11
11500	4498.82
12000	4546.52
12500	4589.68
13000	4628.73
13500	4664.06
14000	4696.03
14500	4724.96
15000	4751.14

## Esercizio 3 (5 punti)

Si confrontino il linear hashing e l'extendible hashing, evidenziandone i rispettivi pregi e difetti.

## Esercizio 4 (3 punti)

Descrivere la realizzazione della scrittura di una pagina, il commit e l'abort in un sistema che adotti una politica *Undo-noRedo*.

### *Soluzione Esercizio 1*

#### **Fase di analisi:**

Il record di checkpoint necessariamente include T1 e T2 tra le transazioni attive e PA e PB tra le pagine sporche. La fase di analisi aggiunge T3 alle transazioni attive (togliendovi T2) e PC e PD alle pagine sporche (togliendovi PB).

#### **Fase di redo:**

Vengono rieseguite le operazioni 2., 8., e 10. (a meno che le pagine non abbiano un recLSN maggiore del LSN corrispondente). Viene aggiunto un record END(T1).

#### **Fase di undo:**

Viene aggiunto un record di compensazione COMP(T3, PD, VD2). Viene aggiunto un record END(T3).

In totale, record che vengono aggiunti sono i seguenti:

12. UPDATE(T1, PA, VA1, VA2)
13. UPDATE(T1, PC, VC1, VC2)
14. UPDATE(T3, PD, VD1, VD2)
15. END(T1)
16. COMP(T3, PD, VD2)
17. END(T3)

### *Soluzione Esercizio 2*

#### **Selettività dei predicati:**

Predicato su A.mesi =  $12/48 = 0.25$

Predicato su I.residenza =  $1/100 = 0.01$  per ogni valore di residenza

Predicato di join su matricola =  $1/10K$  (chiave esterna)

#### **Accesso a I:**

Costo scan sequenziale = **1000**

Costo indice su residenza:  $2 \times (NL \times 0.01 + NP \times 0.01) = 2 \times (40 \times 0.01 + 1000 \times 0.01) = 2 + 20 = 22$

Costo indice su matricola:  $1 + 1 = 2$

Numero tuple residue =  $NT \times 0.02 = 200$

#### **Accesso ad A:**

Costo scan sequenziale = **5000**

Costo indice su matricola:  $NL/10K + \Phi(NT/10K, NP) = 200/10K + \Phi(20K/10K, 5K) = 1 + 2 = 3$

Costo indice su mesi:  $NL \times 0.25 + \Phi(NT \times 0.25, NP) = 10/4 + \Phi(20K/4, 5K) = 3 + 3161 = 3164$

Numero tuple residue =  $NT \times 0.25 = 5000$

#### **Costi di join:**

I-A: costo = costo indice su residenza +  $200 \times$  costo indice A.matricola =  $22 + 200 \times 3 = 622$

A-I: costo = costo indice su mesi +  $5000 \times$  costo indice I.matricola =  $3164 + 5000 \times 2 = 13164$

Il numero di risultati dell'interrogazione è  $20K \times 0.25 \times 0.02 = 100$