

Esercizio sull'ottimizzazione e stima di costo

Si consideri il seguente schema relazionale contenente le ricette di una catena di ristoranti:

INGREDIENTE(Codice, Nome, Calorie);

COMPOSIZIONE(Ricetta, Ingrediente, Quantità)

RICETTA(CodiceRicetta, Nome, Regione, TempoPreparazione)

Nota: la quantità nella tabella COMPOSIZIONE è espressa in grammi

Vincoli di integrità: COMPOSIZIONE.Ricetta → RICETTA, COMPOSIZIONE.Ingrediente → INGREDIENTE

Data la seguente interrogazione SQL che consente di trovare gli ingredienti usati in ricette della Regione Veneto, riportando, il codice della ricetta e il nome e le calorie dell'ingrediente.

```
SELECT R.CodiceRicetta, I.Nome, I.Calorie
FROM RICETTA R JOIN COMPOSIZIONE C ON R.CodiceRicetta = C.Ricetta
      JOIN INGREDIENTE I ON C.Ingrediente = I.Codice
WHERE R.Regione = 'Veneto'
```

(4) Calcolare il costo dell'interrogazione in termini di numero di accessi a memoria secondaria sotto le seguenti ipotesi:

- la selezione su ricetta richiede una scansione sequenziale della tabella RICETTA
- l'ordine di esecuzione del join è (RICETTA ⋈ COMPOSIZIONE) ⋈ INGREDIENTE
- le operazioni di join vengono eseguite con la tecnica "Nested Loop Join" con una pagina di buffer disponibile per ogni tabella
- il risultato intermedio del primo join viene interamente memorizzato nel buffer
- $NP(\text{INGREDIENTE}) = 40$, $NP(\text{COMPOSIZIONE}) = 200$, $NP(\text{RICETTA}) = 12$
- $NR(\text{INGREDIENTE}) = 1200$, $NR(\text{COMPOSIZIONE}) = 13000$, $NR(\text{RICETTA}) = 260$
- $VAL(\text{Regione}, \text{RICETTA}) = 20$, $VAL(\text{Ricetta}, \text{COMPOSIZIONE}) = 260$

(2) Come cambia il costo se è disponibile un indice B+-tree sull'attributo Codice della tabella INGREDIENTE che ha profondità 2.

Si consideri il seguente schema relazionale contenente le ricette di una catena di ristoranti:

INGREDIENTE(Codice, Nome, Calorie);
COMPOSIZIONE(Ricetta, Ingrediente, Quantità)
RICETTA(CodiceRicetta, Nome, Regione, TempoPreparazione)

Applicando la formula per il costo della tecnica “Nested Loop Join” al caso specifico e secondo l’ordine di join indicato risulta:

$$\begin{aligned}\text{COSTO PRIMO JOIN} &= NP(\text{RICETTA}) + \\ &\quad NR(\text{RICETTA con selezione Regione='Veneto'}) * NP(\text{COMPOSIZIONE}) \\ \text{COSTO SECONDO JOIN} &= (\text{nessun costo di lettura della tabella esterna: è già nel buffer}) + \\ &\quad NR(\text{COMPOSIZIONE per le ricette con selezione Regione='Veneto'}) * \\ &\quad \quad \quad NP(\text{INGREDIENTE}) \\ \text{COSTO PRIMO JOIN} &= 12 + NR(\text{RICETTA}) / VAL(\text{Regione, RICETTA}) * 200 = \\ &\quad 12 + 260 / 20 * 200 = 12 + 13 * 200 = 2612 \\ \text{COSTO SECONDO JOIN} &= 0 + (\text{stima delle tuple di COMPOSIZIONE che rappresentano} \\ &\quad \text{ingredienti di ricette del Veneto}) * 40 = \\ &\quad (\text{numero medio di tuple di COMPOSIZIONE per ricetta} * \text{numero} \\ &\quad \text{di ricette del Veneto}) * 40 = \\ &\quad (NR(\text{COMPOSIZIONE}) / VAL(\text{Ricetta, COMPOSIZIONE}) * 13) * 40 = \\ &\quad (13000/260 * 13) * 40 = \\ &\quad = 650 * 40 = 26000\end{aligned}$$

COSTO TOTALE = 28612

(2) Come cambia il costo se è disponibile un indice B+-tree sull’attributo Codice della tabella INGREDIENTE che ha profondità 2.

COSTO PRIMO JOIN = come prima

$$\begin{aligned}\text{COSTO SECONDO JOIN} &= (\text{nessun costo di lettura della tabella esterna: è già nel buffer}) + \\ &\quad NR(\text{COMPOSIZIONE per le ricette con selezione Regione='Veneto'}) * \\ &\quad \quad \quad (\text{Profondità Indice} + \text{selettività di Codice in INGREDIENTE})\end{aligned}$$

$$\text{COSTO PRIMO JOIN} = 2612$$

$$\text{COSTO SECONDO JOIN} = (13000/260 * 13) * (2 + 1) = 650 * 3 = 1950$$

COSTO TOTALE = 4562