

Compito di Complementi di Basi di Dati

19 dicembre 2007

Esercizio 1:

Si supponga che per memorizzare i dati relativi all'offerta di ristoranti di una data città sia stata usata un'unica tabella *RISTORANTE* con la seguente struttura:

RISTORANTE(*NomeRistorante*, *Indirizzo*, *NumeroForchette*, *Proprietario*, *MembroPersonale*, *RuoloMembroPersonale*, *OreSettimanali*)

Sia dato il seguente insieme di requisiti. Ogni ristorante sia identificato univocamente dal nome e possieda un unico proprietario e un certo insieme di membri del personale. Di ogni ristorante venga, inoltre, memorizzato l'indirizzo e il livello, espresso dal numero di forchette ad esso assegnato (1,2,3, ..). Ogni proprietario sia identificato univocamente dal suo codice fiscale e possa possedere più ristoranti, tutti dello stesso livello. Ogni membro del personale sia identificato univocamente dal suo codice fiscale. Un membro del personale possa lavorare per più ristoranti, con la possibilità di ricoprire ruoli diversi in ristoranti diversi (ad esempio, capo cuoco in un ristorante, aiuto cuoco in un altro). Di ogni membro del personale si voglia sapere, per ogni ristorante presso cui lavora, il numero di ore settimanali di lavoro presso tale ristorante (si assuma che tale valore rimanga invariato nel tempo).

- Determinare le dipendenze funzionali della relazione *RISTORANTE*, indicando, per ciascuna di esse, il requisito codificato.
- Determinare le chiavi candidate e gli attributi primi e non primi di *RISTORANTE*.
- Stabilire se *RISTORANTE* è o meno in BCNF.
- Stabilire se *RISTORANTE* è o meno in 3NF.
- Stabilire se *RISTORANTE* è o meno in 2NF.
- Nel caso in cui *RISTORANTE* non sia in 3NF, fornire una scomposizione lossless join in 3NF di *RISTORANTE* che conservi le dipendenze.

Esercizio 2:

Stabilire se i seguenti schedule appartengono o meno a VSR, CSR, TS, 2PL e 2PL stretto. Nel caso appartengano a VSR e/o a CSR, determinare tutti gli schedule seriali ad essi equivalenti.

- $w_1(x), r_1(x), r_2(y), w_3(x), r_3(y), r_2(x), w_3(y), w_1(y)$
- $r_1(x), r_1(t), w_4(y), r_4(x), r_4(z), r_3(x), w_4(x), w_2(t), r_3(z), w_2(z), w_3(y), w_1(y)$
- $r_4(x), r_1(x), w_4(x), r_1(y), w_3(y), r_4(z), w_4(z), w_3(z), w_1(t), w_2(z), w_2(t)$

Esercizio 3:

Si discutano i problemi legati all'interazione tra trigger e transazioni. Successivamente, date le tabelle:

Città(*nomeCittà*, *provincia*, *sindaco*)

Persona(*cf*, *nome*, *cognome*, *via*, *numero*, *cittàResidenza*)

si costruisca una vista materializzata *Popolazione*, che tenga traccia del numero di residenti di ogni città, e due trigger che aggiornino il contenuto di tale vista a fronte di inserimenti e cancellazioni relativi alla tabella *Persona*.

Esercizio 4:

Si consideri un file ordinato con $r=50000$ record, memorizzato su un disco con dimensioni del blocco $B=1024$ byte. I record abbiano lunghezza fissa $R=150$ byte e siano di tipo unspanned. Il campo chiave del file abbia dimensione $V=12$ byte ed il puntatore a blocco abbia dimensione $P=8$ byte.

- (a) Determinare il numero di accessi a blocco richiesti da una ricerca basata sul valore del campo chiave in presenza delle seguenti organizzazioni del file: (i) non ordinato, (ii) ordinato rispetto alla chiave, (iii) con indice primario e (iv) con indice secondario costruito sul campo chiave.
- (b) Convertire il file con indice primario in uno con indice multilivello. Quanti livelli sono necessari? Calcolare il numero di blocchi associato ad ogni livello.
- (c) Convertire il file con indice secondario in uno con indice multilivello. Quanti livelli sono necessari? Calcolare il numero di blocchi associato ad ogni livello.