

# Compito di Complementi di Basi di Dati

10 febbraio 2010

## Esercizio 1:

Si supponga che per memorizzare i dati relativi alla gestione di un insieme di stazioni per il rifornimento di carburante della regione Friuli Venezia Giulia sia stata usata un'unica tabella *STAZIONE* con la seguente struttura:

*STAZIONE*(*CodiceStazione*, *Azienda*, *ComuneStazione*, *ComuneAzienda*, *Pompa*,  
*Carburante*, *Dipendente*)

Sia dato il seguente insieme di requisiti. Ogni stazione di rifornimento sia identificata univocamente da un codice e sia caratterizzata dal comune in cui si trova e dall'azienda che la possiede. Ogni azienda sia identificata univocamente da un codice e sia caratterizzata dal comune ove ha sede. In ogni stazione vi siano una o più pompe per l'erogazione del carburante, ciascuna identificata da un numero (pompa numero 1, pompa numero 2, ..). Ogni pompa eroghi un unico (tipo di) carburante. Ogni azienda disponga di un certo insieme di dipendenti. Ogni dipendente lavori per un'unica azienda e sia assegnato ad una ed una sola delle stazioni di proprietà dell'azienda.

- Determinare le dipendenze funzionali della relazione *STAZIONE*.
- Determinare le chiavi candidate e gli attributi primi e non primi di *STAZIONE*.
- Stabilire se *STAZIONE* è o meno in 3NF.
- Stabilire se *STAZIONE* è o meno in 2NF.
- Stabilire se *STAZIONE* è o meno in BCNF.
- Nel caso in cui *STAZIONE* non sia in 3NF, fornire una scomposizione lossless join in 3NF di *STAZIONE* che conservi le dipendenze.
- (facoltativo) Stabilire se la scomposizione di cui al punto precedente è o meno in BCNF?

## Esercizio 2:

Si descrivano le modalità di gestione della concorrenza in SQL:1999. In particolare, si descrivano i diversi possibili livelli di isolamento. Successivamente, si stabilisca se i seguenti schedule appartengono o meno a 2PL stretto, 2PL, TS, CSR e VSR:

- $s_1 : r_3(y), w_1(y), r_1(x), w_5(x), w_1(t), w_4(x), r_2(z), r_3(z), w_5(z), r_5(t), w_2(z), r_4(t);$
- $s_2 : r_2(x), r_1(x), w_2(x), r_4(z), r_3(x), w_1(x), r_3(y), r_3(x), w_1(y), w_1(z), r_5(y), w_5(x), r_5(z);$
- $s_3 : r_2(y), w_2(y), r_3(z), r_1(x), w_1(x), r_3(y), w_3(y), r_2(z), w_3(z), r_2(x), r_1(y), w_2(x), w_1(y);$

## Esercizio 3:

Siano date le tabelle:

*Azienda*(*nomeAzienda*, *amministratoreDelegato*, *provinciaSede*)

*Dipendente*(*cf*, *nome*, *cognome*, *recapito*, *azienda*)

Si costruisca una vista materializzata *Personale*, che tenga traccia del numero di dipendenti di ogni azienda, e due trigger che aggiornino il contenuto di tale vista a fronte di inserimenti e cancellazioni relativi alla tabella *Dipendente*.

#### **Esercizio 4:**

Si consideri un file contenente 10000000 record di dimensione prefissata pari a 100 byte, memorizzati in blocchi di dimensione pari a 1024 byte in modo unspanned. La dimensione del campo chiave primaria  $V$  sia 9 byte; la dimensione del puntatore a blocco  $P$  sia 6 byte.

- (a) Si assuma che il file sia ordinato rispetto alla chiave primaria  $V$ . Determinare il numero medio di accessi a blocco richiesto da una ricerca basata sulla chiave primaria.
- (b) Si assuma che il file sia ordinato rispetto alla chiave primaria  $V$ . Determinare il numero medio di accessi a blocco richiesto da una ricerca basata su un campo  $V'$  non chiave di dimensione 9 byte.
- (c) Si assuma che il file sia ordinato rispetto alla chiave primaria  $V$ . Determinare il numero medio di accessi a blocco richiesto da una ricerca basata su un campo chiave  $V'$ , diverso dalla chiave primaria, di dimensione 9 byte.
- (d) Si assuma che il file sia ordinato rispetto alla chiave primaria  $V$ . Determinare il numero medio di accessi a blocco richiesto da una ricerca con indice primario.
- (e) Si assuma che il file sia ordinato rispetto alla chiave primaria  $V$ . Determinare il numero medio di accessi a blocco richiesto da una ricerca con indice secondario costruito su un campo chiave  $V'$ , diverso dalla chiave primaria, di dimensione 9 byte.
- (f) Si determinino dimensione e struttura di un indice multilivello statico ottenuto a partire dall'indice secondario di cui al punto (e).
- (g) Si determini la dimensione di un  $B$ -albero, con campo di ricerca il campo chiave primaria  $V$ , puntatore ai dati di dimensione pari a 7 byte e puntatore ausiliario di dimensione pari a 6 byte, assumendo che ciascun nodo del  $B$ -albero sia pieno al 70%.