

Compito di Basi di dati - 9 CFU

8 luglio 2014

Esercizio 1:

Sia dato il seguente schema di una base di dati relazionale contenente alcune informazioni sui corsi di laurea di una data università:

studente(matricola, nome, cognome, corsoDiLaurea, annoImmatricolazione);

insegnamento(codiceInsegnamento, nomeInsegnamento, corsoDiLaurea, docente);

esame(insegnamento, studente, voto, lode).

Si assuma che ogni studente sia identificato dalla sua matricola e sia caratterizzato da nome, cognome, corso di laurea a cui è iscritto e anno di immatricolazione. Si assuma, inoltre, che ogni insegnamento sia identificato da un codice (as esempio, ASD203) e sia caratterizzato da un nome (ad esempio, Algoritmi e strutture dati), dal corso di laurea al quale appartiene (ad esempio, Tecnologie web e multimediali) e dal docente che lo tiene. Si assuma che possano esservi più insegnamenti con lo stesso nome (ad esempio, più insegnamenti di Reti di Calcolatori), non necessariamente tenuti dallo stesso docente. Si assuma anche che un docente possa tenere più insegnamenti (l'insegnamento BD203 di Basi di Dati, l'insegnamento BD204 di Basi di Dati e l'insegnamento ASD132 di Algoritmi e Strutture Dati). Si assuma, infine, che di ogni esame superato positivamente vengano registrati l'insegnamento, lo studente, il voto ottenuto (compreso tra 18 e 30) e l'eventuale lode (attributo booleano).

Definire preliminarmente le chiavi primarie, le eventuali altre chiavi candidate e, se ve ne sono, le chiavi esterne delle relazioni date. Successivamente, formulare opportune interrogazioni in algebra relazionale che permettano di determinare (senza usare l'operatore di divisione e usando solo se e quando necessario le funzioni aggregate):

- gli studenti che, in tutti gli esami registrati, hanno ottenuti solo voti compresi tra 24 e 27 (estremi inclusi);
- gli studenti che hanno ottenuto un voto maggiore di 29 in più di 2 esami;
- gli studenti che hanno superato un sottoinsieme proprio degli esami superati dallo studente identificato dal numero di matricola 987654.

Esercizio 2:

Con riferimento all'Esercizio 1, formulare opportune interrogazioni in SQL che permettano di determinare quanto richiesto (senza usare l'operatore CONTAINS e usando solo se e quando necessario le funzioni aggregate).

Esercizio 3:

Si voglia modellare il seguente insieme di informazioni riguardanti un sistema unificato per la gestione delle prenotazioni di un tavolo per la cena in un insieme di ristoranti di proprietà di una data società.

- Ogni ristorante sia identificato univocamente dal proprio nome e sia caratterizzato da un indirizzo, da uno e più recapiti telefonici, da un indirizzo di email e da un numero di fax (opzionale).
- Ogni prenotazione sia caratterizzata da un codice di prenotazione univoco e sia caratterizzata dalla data in cui viene effettuata la prenotazione, dalla data della cena, dal tavolo prenotato, dal cliente che ha effettuato

la prenotazione, dal numero di persone che parteciperà alla cena e dal menù ordinato. Si assuma che non sia possibile prenotare più di un tavolo con una singola prenotazione (ad esempio, se un cliente vuole riservare 3 tavoli, dovrà effettuare 3 prenotazioni distinte).

- Ogni tavolo di un dato ristorante sia identificato univocamente da un numero (tavolo numero 12 del ristorante Al Gambero Rosso) e da un numero di posti.
- Ogni menù sia identificato univocamente da un codice e sia caratterizzato da un nome, da una breve descrizione e da un certo insieme di portate. Ogni portata sia individuata univocamente da un nome (ad esempio, risotto agli asparagi, frutta di stagione, ..) e sia caratterizzata dalla collocazione all'interno del menù (antipasto, primo piatto, ..) e da un insieme di ingredienti. Si assuma che una stessa portata possa comparire in più menù.
- Ogni cliente sia identificato univocamente da un codice fiscale e sia caratterizzato da un nome, da un cognome e da un recapito telefonico.

Si definisca uno schema Entità-Relazioni che descriva il contenuto informativo del sistema, illustrando con chiarezza le eventuali assunzioni fatte. Lo schema dovrà essere completato con attributi ragionevoli per ciascuna entità (identificando le possibili chiavi) e relazione. Vanno specificati accuratamente i vincoli di cardinalità e partecipazione di ciascuna relazione. Si definiscano anche eventuali regole di derivazione e/o vincoli di integrità necessari per codificare alcuni dei requisiti attesi del sistema.

Esercizio 4:

Si illustrino brevemente i limiti delle tecniche VSR e CSR. Successivamente, si stabilisca se i seguenti schedule appartengono o meno a 2PL stretto, 2PL e TS:

1. $s_1 : r_3(y), w_1(y), r_1(x), w_5(x), r_2(z), w_1(t), r_3(z), w_4(x), r_5(t), w_5(z), r_4(t), w_2(z);$
2. $s_2 : r_2(x), r_1(x), w_2(x), r_4(z), r_3(x), w_1(x), r_3(y), w_1(y), r_3(x), w_1(z), r_5(y), w_5(x), r_5(z);$
3. (facoltativo) $s_3 : r_2(y), w_2(y), r_3(z), r_1(x), w_1(x), r_3(y), w_3(y), r_2(z), w_3(z), r_2(x), w_2(x), r_1(y), w_1(y).$

Esercizio 5:

Si consideri un file contenente 10000000 record di dimensione prefissata pari a 400 byte, memorizzati in blocchi di dimensione pari a 4096 byte in modo unspanned. La dimensione del campo chiave primaria V sia 14 byte; la dimensione del puntatore a blocco P sia 6 byte. Si chiede di confrontare fra loro le seguenti soluzioni, in termini di numero medio di accessi a blocco e di dimensione dell'indice.

- (a) Ricerca basata su un indice multilivello statico ottenuto a partire da un indice secondario costruito sul campo chiave primaria V .
- (b) Ricerca basata su 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%.
- (c) Ricerca basata su 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%.