

Compito di Complementi di Basi di Dati

5 aprile 2006

Esercizio 1:

Si dimostri che due insiemi di dipendenze funzionali F e G sono equivalenti se (e solo se) $F \subseteq G^+$ e che $G \subseteq F^+$.

Successivamente, si considerino la relazione $R(A, B, C, D, E)$ e i due insiemi di dipendenze funzionali $F = \{B \rightarrow CDE, CD \rightarrow B, E \rightarrow A\}$ e $G = \{B \rightarrow ACE, E \rightarrow AD, CD \rightarrow B\}$.

- Stabilire se F e G sono o meno equivalenti.
- Determinare le chiavi candidate e gli attributi primi e non primi di (R, F) e (R, G) .
- Stabilire se (R, F) è o meno in 2NF.
- Stabilire se (R, F) è o meno in 3NF.
- Stabilire se (R, F) è o meno in BCNF.
- Nel caso in cui (R, F) non sia in 3NF, fornire una scomposizione lossless join in 3NF di (R, F) che conservi le dipendenze.
- Stabilire se la scomposizione di cui al punto precedente è o meno in BCNF?

Ogni risposta va adeguatamente motivata (non è sufficiente una risposta di tipo sì/no).

Esercizio 2:

Si illustrino brevemente le diverse possibili modalità di interazione con le basi di dati da parte dei programmi applicativi, mettendo in evidenza punti di contatto e di differenziazione. Successivamente, si descrivano le fasi tipiche dell'interazione con una base di dati da parte di un programma in linguaggio C attraverso l'uso di SQL/CLI.

Esercizio 3:

Si stabilisca se i seguenti schedule appartengono o meno a VSR, CSR, 2PL, 2PL stretto e TS. Nel caso in cui uno schedule appartenga sia a VSR che a CSR, verificare se tutti gli schedule seriali equivalenti rispetto alle viste allo schedule dato sono ad esso equivalenti anche rispetto ai conflitti.

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

Esercizio 4:

Si consideri un file contenente 12000000 record di dimensione prefissata pari a 120 byte, memorizzati in blocchi di dimensione pari a 1024 byte in modo unspanned. Si assuma che il file sia ordinato rispetto ad un campo chiave $V1$ di dimensione pari a 15 byte.

1. Si determini il numero di accessi a blocco richiesti da una ricerca basata su un indice secondario costruito su un campo chiave non ordinante V_2 di dimensione pari a 9 byte, con dimensione del puntatore a record pari a 7 byte.
2. Successivamente, si determini la dimensione e il numero di accessi a blocco richiesti da una ricerca basata su un indice multilivello statico ottenuto a partire dall'indice secondario definito al punto precedente.
3. Infine, si determini la dimensione e il numero di accessi a blocco richiesti da una ricerca basata su un B^+ -albero, con campo di ricerca il campo chiave V_2 , puntatore ai dati (record) 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%.