

Compito di Complementi di Basi di dati

14 settembre 2010

Esercizio 1:

Si supponga che per memorizzare alcuni dati anagrafici relativi ad un insieme di persone di interesse sia stata usata un'unica tabella *ANAGRAFE* con la seguente struttura:

ANAGRAFE(*CFPersona*, *Nome*, *Cognome*, *ViaResidenza*, *NCivicoResidenza*, *CAPResidenza*, *CittaResidenza*, *DataNascita*, *CittaNascita*)

Sia dato il seguente insieme di requisiti. Ogni persona sia identificata univocamente dal proprio codice fiscale. Ogni persona sia caratterizzata da un (unico) nome ed un (unico) cognome. Vi possano essere persone diverse con lo stesso nome e/o con lo stesso cognome. Ogni persona abbia un'unica residenza, un'unica città di nascita e un'unica data di nascita. Persone diverse possano avere la medesima residenza. Vi possano essere vie con lo stesso nome in città diverse, ma non nella stessa città. Ad ogni CAP corrisponda un'unica città, ma non valga il viceversa: a (parti diverse di) una stessa città possano essere associati CAP diversi.

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

Esercizio 2:

Si considerino i seguenti schedule:

$s_1: r_0(x), w_0(x), r_1(x), r_2(y), w_0(y), w_1(x);$

$s_2: r_0(x), w_1(x), r_1(x), w_0(x), w_2(x), r_2(x);$

$s_3: r_0(x), r_0(y), w_1(x), r_0(v), r_2(x), w_1(y), w_2(v), r_3(y), w_4(y), r_3(v).$

Stabilire se gli schedule dati sono o meno serializzabili rispetto alle viste, rispetto ai conflitti, rispetto al metodo del locking a due fasi, rispetto al metodo basato sui timestamp e rispetto al metodo del locking a due fasi stretto.

Esercizio 3:

Sia dato un file contenente $r = 4000000$ record, memorizzati su un disco con dimensione del blocco $B = 4096$ byte. I record abbiamo lunghezza fissa $R = 400$ byte e siano di tipo unspanned. La dimensione del campo chiave primaria sia $V1 = 15$ byte; la dimensione del puntatore ad un blocco sia $P = 5$ byte.

- (a) Si assuma che il file non sia ordinato rispetto ad alcun campo. Determinare il numero medio di accessi a blocco richiesto da una ricerca basata sulla chiave primaria.
- (b) Si assuma che il file non sia ordinato rispetto ad alcun campo. Determinare il numero medio di accessi a blocco richiesto da una ricerca basata su una chiave candidata (diversa dalla chiave primaria) di dimensione $V2 = 10$ byte.
- (c) Si assuma che il file sia ordinato rispetto alla chiave primaria. Determinare il numero medio di accessi a blocco richiesto da una ricerca basata sulla chiave primaria.
- (d) Si assuma che il file sia ordinato rispetto alla chiave primaria. 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. Determinare il numero medio di accessi a blocco richiesto da una ricerca basata su una chiave candidata (diversa dalla chiave primaria) di dimensione $V2 = 10$ byte.
- (f) Si assuma che il file non sia ordinato rispetto ad alcun campo. Determinare il numero medio di accessi a blocco richiesto da una ricerca con indice secondario costruito sulla chiave primaria.
- (g) Si assuma che il file non sia ordinato rispetto ad alcun campo. Determinare il numero medio di accessi a blocco richiesto da una ricerca con indice secondario costruito su una chiave candidata (diversa dalla chiave primaria) di dimensione $V2 = 10$ byte.
- (h) Si assuma che il file sia ordinato rispetto alla chiave primaria. Determinare il numero medio di accessi a blocco richiesto da una ricerca con indice secondario costruito su una chiave candidata (diversa dalla chiave primaria) di dimensione $V2 = 10$ byte.

Esercizio 4:

Descrivere brevemente le caratteristiche fondamentali delle procedure di ripristino da guasto denominate rispettivamente ripresa a caldo e ripresa a freddo, spiegando gli utilizzi dell'una e dell'altra. Successivamente, descrivere l'esecuzione della procedura di ripresa a caldo, mostrando la costruzione passo passo degli insiemi UNDO e REDO e l'applicazione delle azioni di ripristino, a fronte del seguente input:

$B(T_1), I(T_1, O_1, A_1), U(T_1, O_2, B_2, A_2), B(T_2), D(T_2, O_2, B_3), B(T_3), I(T_3, O_3, A_4), U(T_2, O_3, B_5, A_5),$
 $C(T_3), B(T_4), I(T_2, O_4, B_6), D(T_4, O_1, B_7), C(T_1), CK(T_2, T_4), C(T_2), B(T_5), U(T_4, O_4, B_8, A_8),$
 $D(T_5, O_3, B_9), B(T_6), A(T_5), U(T_6, O_2, B_{10}, A_{10}), C(T_4), I(T_6, O_4, A_{11}), guasto$

Esercizio 5 (facoltativo)

Si dimostri che i primi membri 'singleton' (vale a dire costituiti da un singolo attributo) sono insufficienti per le dipendenze funzionali (suggerimento: fornire un insieme di dipendenze funzionali che non sia equivalente ad alcun insieme di dipendenze funzionali del tipo $\{B_1 \rightarrow C_1, \dots, B_n \rightarrow C_n\}$, dove B_i e C_i , per $i = 1 \dots, n$, sono singoli attributi).