

# Compito di Complementi di Basi di Dati

19 settembre 2005

Anno Accademico 2004/2005

## Esercizio 1:

Sia data una relazione  $R(A, B, C, D)$ , con associato l'insieme di dipendenze funzionali  $F = \{AC \rightarrow D, B \rightarrow CD, C \rightarrow B, BC \rightarrow A\}$ .

- Determinare una copertura minimale di  $F$ .
- Stabilire se l'insieme  $F$  è o meno equivalente all'insieme  $G = \{AB \rightarrow D, B \rightarrow AC, C \rightarrow BD\}$ .
- Determinare le chiavi candidate, gli attributi primi e gli attributi non primi della relazione  $R$ .
- Stabilire se  $R$  è o meno in BCNF.
- Stabilire se  $R$  è o meno in 3NF.
- Stabilire se  $R$  è o meno in 2NF.
- Nel caso in cui  $R$  non sia in 3NF, fornire una scomposizione lossless join di  $R$  in 3NF che conservi le dipendenze.

## Esercizio 2:

Si stabilisca se i seguenti schedule appartengono o meno a VSR, CSR, TS, 2PL e 2PL stretto (nel caso di schedule appartenenti a VSR, si fornisca almeno uno schedule seriale equivalente; lo stesso nel caso di schedule appartenenti a CSR):

- $s_1 : w_1(x), r_2(x), w_1(z), r_2(z), w_2(x), r_3(x), r_0(z), w_0(z)$ ;
- $s_2 : w_1(y), r_2(x), w_3(x), r_4(x), r_2(y), w_3(y), r_2(v), w_4(v), r_5(v), w_5(y)$ .

## Esercizio 3:

Si calcolino i profili dei risultati intermedi e del risultato finale della seguente interrogazione<sup>1</sup>:

$\pi_{Nome, Punt_i}(\pi_{Nome, Codice}(Candidati) \bowtie (\sigma_{Tema='Lingua\_Inglese'}(\pi_{Tema, Codice, Punt_i}(ProveConcorso))))$ ,  
relativa alle tabelle:

$Candidati(Nome, Codice, Provincia, Data\_di\_Nascita)$ ;

$ProveConcorso(Tema, Codice, Data, Punt_i)$ ,

---

<sup>1</sup>Si ricordi che la formula  $col(n, m, k)$  che calcola il numero di valori distinti presenti in  $k$  oggetti estratti a partire da  $n$  oggetti diversi di  $m$  valori distinti, distribuiti omogeneamente, ammette la seguente approssimazione:

- $col(n, m, k) = k$  se  $k \leq m/2$ ;
- $col(n, m, k) = (k + m)/3$  se  $m/2 \leq k \leq 2m$ ;
- $col(n, m, k) = m$  se  $k \geq 2m$ .

aventi i seguenti profili:

$CARD(Candidati) = 750$ ,  $SIZE(Candidati) = 35$ ,

$CARD(ProveConcorso) = 45000$ ,  $SIZE(ProveConcorso) = 32$ ,

$SIZE(Nome, Candidati) = 18$ ,  $SIZE(Codice, Candidati) = 5$ ,

$SIZE(Provincia, Candidati) = 2$ ,  $SIZE(Data_di_Nascita, Candidati) = 10$ ,

$SIZE(Tema, ProveConcorso) = 15$ ,  $SIZE(Codice, ProveConcorso) = 5$ ,

$SIZE(Data, ProveConcorso) = 10$ ,  $SIZE(Punti, ProveConcorso) = 2$ ,

$VAL(Nome, Candidati) = 700$ ,  $VAL(Codice, Candidati) = 750$ ,

$VAL(Provincia, Candidati) = 40$ ,  $VAL(Data_di_Nascita, Candidati) = 720$ ,

$VAL(Tema, ProveConcorso) = 30$ ,  $VAL(Codice, ProveConcorso) = 750$ ,

$VAL(Data, ProveConcorso) = 250$ ,  $VAL(Punti, ProveConcorso) = 13$ .

#### Esercizio 4:

Si consideri un file contenente 10000000 record di dimensione prefissata pari a 50 byte, memorizzati in blocchi di dimensione pari a 1024 byte in modo unspanned. Si determinino dimensione e struttura di un indice multilivello statico ottenuto a partire da un indice primario costruito su un campo chiave V di 9 byte, con dimensione del puntatore a blocco pari a 6 byte. Successivamente, si determini la dimensione di un *B*-albero, con campo di ricerca il campo chiave V di 9 byte, 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%.

**Anno Accademico 2003/2004**

#### Esercizio 1:

Sia dato il seguente schema relazionale:

*Parcheggio*(Numero, posizione, Posti, AnnoApertura, TariffaOraria);

*AddettoSicurezza*(CF, Parcheggio, Categoria, Stipendio);

*AddettoAmministrazione*(CF, Categoria, Stipendio).

Ogni parcheggio sia identificato univocamente dal suo numero. Si assuma che ogni addetto alla sicurezza lavori in un unico parcheggio e che ogni parcheggio possa avere più addetti alla sicurezza. Al contrario, si assuma che gli amministrativi si occupino globalmente dell'insieme dei parcheggi. La posizione di un parcheggio indichi la sua collocazione all'interno di una griglia di suddivisione del territorio e sia espressa da una stringa di due simboli costituita da una lettera seguita da un numero (posizione B5, E7, ..). Si assuma che più parcheggi possano essere collocati nella stessa posizione. La categoria indichi il livello professionale dell'addetto (C, D, ..). Si assuma che addetti della stessa categoria possano avere stipendi diversi.

Definire preliminarmente le chiavi primarie ed esterne delle relazioni date. Successivamente, formulare opportune interrogazioni in SQL che permettano di determinare (senza usare l'operatore CONTAINS e usando solo se necessario le funzioni aggregate):

- i parcheggi aperti dopo il 1990, non collocati in posizione F5, con una capienza compresa tra 200 e 1500 auto (estremi inclusi);
- le posizioni con un solo parcheggio disponibile;
- le coppie di addetti che appartengono alla stessa categoria, ma hanno uno stipendio diverso;

- (d) i parcheggi che hanno esattamente lo stesso numero di addetti alla sicurezza;
- (e) per ogni posizione, il numero di parcheggi presenti, il numero totale di posti disponibili e l'ammontare complessivo degli stipendi pagati per gli addetti alla sicurezza.

### **Esercizio 2:**

Si descrivano sinteticamente le regole di traduzione delle relazioni del modello Entità/Relazioni nel modello relazionale.

### **Esercizio 3:**

Si vuole progettare una base di dati di supporto alla gestione degli interventi di assistenza clienti effettuati da una data software house.

La software house possiede un certo numero di clienti, appartenenti a diverse categorie (aziende, enti pubblici, liberi professionisti, ..). Di ogni cliente la software house possiede indirizzo, recapito telefonico ed estremi della persona di riferimento.

Ogni cliente può formulare una richiesta di assistenza attraverso gli appositi moduli disponibili on-line. Ogni richiesta è identificata univocamente da un codice numerico generato automaticamente dal sistema. In ogni richiesta, il cliente deve specificare il sistema per il quale richiede l'intervento di assistenza e la tipologia del problema (scegliendola in un insieme predefinito di tipologie).

Ad ogni richiesta di assistenza da parte di un cliente, corrispondono uno o più interventi di assistenza da parte della software house. Ogni intervento relativo ad una data richiesta è identificato da un numero progressivo (intervento 1, relativo alla richiesta 523, intervento 2, relativo alla richiesta 523, ..). Gli interventi possono essere svolti sia in remoto (il cliente viene contattato telefonicamente e il problema viene risolto attraverso tale interazione) o presso la sede del cliente. Di ogni intervento vengono registrati il giorno in cui ha luogo (si assume che ogni intervento venga eseguito in giornata) e la durata

Ogni intervento viene effettuato da un tecnico della software house. Se una richiesta di assistenza comporta l'esecuzione di più interventi, questi possono essere svolti da tecnici diversi. Ogni tecnico è identificato univocamente dal suo codice fiscale. Per ogni tecnico vengono specificate le tipologie di problemi che è in grado di risolvere (tali tipologie appartengono ad un insieme predefinito di possibili tipologie). L'esecuzione di un intervento relativo ad una data richiesta deve ovviamente essere assegnata ad un tecnico che ha le competenze necessarie (corrispondenza tra tipologia del problema specificata nella richiesta e tipologie di problemi per i quali il tecnico ha le competenze necessarie).

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.

### **Esercizio 4:**

Si stabilisca se i seguenti schedule appartengono o meno a VSR, CSR, TS, 2PL e 2PL stretto (nel caso di schedule appartenenti a VSR, si fornisca almeno uno schedule seriale equivalente; lo stesso nel caso di schedule appartenenti a CSR):

1.  $s_1 : w_1(x), r_2(x), w_1(z), r_2(z), w_2(x), r_3(x), r_0(z), w_0(z);$
2.  $s_2 : w_1(y), r_2(x), w_3(x), r_4(x), r_2(y), w_3(y), r_2(v), w_4(v), r_5(v), w_5(y).$