Sistemi informativi II
Prof. Stefano Paraboschi
28-1-2004

A. Si ha il seguente schema relazionale, che descrive le prenotazioni dei laboratori effettuate dagli studenti di una università distribuita su più sedi

STUDENTI(Matr,Nome,Sede)
PRENOTAZIONI(Lab,Matr,Giorno,Turno)
LABORATORI(NomeLab,Sede,Capienza)

1. Progettarne la frammentazione, supponendo che ogni sede sia dotata di un proprio DBMS e che gli studenti possono talvolta prenotare anche aule in sedi diverse dalla propria. Discutere le soluzioni che possono evitare ai gestori dei laboratori di accedere a DBMS remoti per ottenere l'elenco nominativo degli studenti prenotati.
2. Formulare ai 3 livelli di frammentazione il comando SQL che sposta lo studente con matricola 47654 dalla sede di Bergamo a quella di Dalmine.

B. Date tre transazioni che eseguono le operazioni descritte:

t_1: $w_1(x)$
t_2: $r_2(y)$, $r_2(x)$, $w_2(x)$
t_3: $w_3(y)$, $w_3(x)$

1. Costruire uno schedule che sia CSR ma non 2PL.
2. Costruire uno schedule che sia CSR ma non TS.

C. Dato il seguente schema a oggetti:

```
create class Auto
  attributes Targa: string,
             Modello: string,
             Proprietario: *Cliente;

create class Posteggio
  attributes Mese: integer,
             Anno: integer,
             Posizione: string,
             Mezzo: *Auto;
```

```
create class Cliente
  attributes Nome: string,
             Indirizzo: string,
             Pagamenti: set(
               struct(Mese: integer,
                      Anno: integer,
                      Importo: integer);
```

1. Usando i costrutti offerti dal linguaggio SQL:1999 costruire una rappresentazione equivalente dello schema (eventualmente solo in forma grafica),
2. Estrarre (a scelta in OQL o SQL:1999) i clienti che non hanno pagato il posteggio almeno una volta nel 2001.

D. Descrivere l'organizzazione fisica ad accesso calcolato (*hash*), presentandone vantaggi e svantaggi.

E. Illustrare le particolarità del costrutto di *relationship* nel linguaggio ODL.