

[illegible]

B. Linguaggi Formali (6 punti)

4. Estrarre in Algebra Relazionale ottimizzata, Calcolo Relazionale e Datalog per ogni orchestra (codice e nome) il prezzo del concerto più caro (4 p.)

a) Algebra

CONCCARO := $(\Pi_{\text{CodO}, \text{PrezzoBiglietto}} \text{CONCERTO}) -$

$((\Pi_{\text{CodO}, \text{PrezzoBiglietto}} \text{CONCERTO}) \triangleright_{\text{CodO}=\text{CodO} \wedge \text{PrezzoBiglietto} < \text{PrezzoBiglietto}} (\Pi_{\text{CodO}, \text{PrezzoBiglietto}} \text{CONCERTO}))$

$\Pi_{\text{CodOrchestra}, \text{NomeO}, \text{PrezzoBiglietto}} (\Pi_{\text{CodOrchestra}, \text{NomeO}} \text{ORCHESTRA}) \triangleright_{\text{CodOrchestra}=\text{CodO}} \text{CONCCARO}$

b) Calcolo

```
{ t | ∃ tO ∈ ORCHESTRA, tC ∈ CONCERTO |  
  (t[CodOrchestra] = tO[CodOrchestra] ∧  
   t[NomeO] = tO[NomeO] ∧  
   t[PrezzoBiglietto] = tC[PrezzoBiglietto] ∧  
   tO[CodOrchestra] = tC[CodO] ∧  
   ¬ (∃ tC1 ∈ CONCERTO |  
     (tC1[CodO] = tC[CodO] ∧  
      tC1[PrezzoBiglietto] > tC[PrezzoBiglietto])  
   )  
}
```

c) Datalog

ConcNONCaro(CodO, Prezzo1) :- CONCERTO(____, CodO, __, Prezzo1),
CONCERTO(____, CodO, __, Prezzo2),
Prezzo1 < Prezzo2

ConcCaro(CodO, NomeO, Prezzo) :- ORCHESTRA(CodO, NomeO, __, __, __),
CONCERTO(____, CodO, __, Prezzo, __),
¬ ConcNONCaro(CodO, Prezzo)

? – ConcCaro (X,Y,Z)

5. Esprimere, in uno a scelta dei tre linguaggi formali (algebra ottimizzata), l'interrogazione che trova il codice e il nome delle orchestre il cui direttore non ha mai diretto più di una orchestra (2 p.)

Algebra

$(\Pi_{\text{CodOrchestra}, \text{NomeOrchestra}} \text{ORCHESTRA}) -$

$(\Pi_{\text{CodOrchestra}, \text{NomeOrchestra}} ((\Pi_{\text{CodOrchestra}, \text{NomeOrchestra}, \text{NomeDirettore}} \text{ORCHESTRA}) \triangleright_{\text{CodOrchestra} \neq \text{CodOrchestra} \wedge \text{NomeDirettore} = \text{NomeDirettore}} (\Pi_{\text{CodOrchestra}, \text{NomeDirettore}} \text{ORCHESTRA})))$

C. DOMANDA (1 punti)

Si esprima, tramite un'opportuna asserzione SQL, il vincolo per cui un'orchestra non può avere due concerti diversi nella stessa data.

```
CREATE ASSERTION NonStessaData  
CHECK ( NOT EXISTS ( SELECT *  
                     FROM CONCERTO  
                     GROUP BY CodOrchestra, Data  
                     HAVING COUNT(*) >= 2) )
```

Parte 2 : PROGETTAZIONE CONCETTUALE (su un foglio a parte rispetto alla Parte 1)

D. PROGETTO CONCETTUALE (10 punti) E LOGICO (4 punti)

- Ricordare che il progetto è valutato per completezza, correttezza, leggibilità, minimalità e autoesplicatività, e che anche il progetto logico è un **grafo** i cui nodi e archi devono essere disposti coerentemente col progetto concettuale.
- Ricordare anche di specificare gli **identificatori** di tutte le entità e le **cardinalità** di tutte le associazioni, e di disporre i due grafi su due facciate affiancate, in modo da poterli osservare simultaneamente.