

Basi di Dati, 16/11/2004, primo compito, compito A

1) Si vogliono gestire i dati necessari alla programmazione didattica di un'Università. L'Università offre un insieme di Corsi di Laurea. Di un Corso di Laurea interessano il nome, il codice che lo identifica, la o le Facoltà che lo gestiscono, e le materie insegnate. Di una Facoltà interessa il nome, il codice, ed il docente che la presiede. La stessa materia può essere insegnata in più Corsi di Laurea. Di ogni materia interessa il nome, il codice che la identifica, ed il numero di crediti. Per ciascuna materia, in ciascun anno accademico, può essere attivato un corso, più di uno, oppure nessuno. Ogni corso è attivato per una specifica materia. Inoltre, per ogni corso interessa l'Anno Accademico di attivazione, il docente responsabile, e l'insieme degli altri Docenti che insegnano in quel corso. Di un docente interessano un codice, che lo identifica, nome, cognome, ed un insieme di recapiti telefonici. Un docente può essere interno od esterno. Per un docente esterno interessa l'ente da cui dipende. Per un docente interno interessa la Facoltà a cui afferisce. Solo un docente che afferisce ad una Facoltà può presiederla.

a) Si disegni lo schema concettuale che descrive la situazione, indicando in modo testuale i vincoli non catturati graficamente.

b) Si traduca lo schema concettuale in uno schema relazionale grafico e testuale, usando la notazione $R(\underline{A}, B^*, \dots)$

2) Si consideri la base di dati:

Concerti (IdConcerto, NomeC, Città, Giorno, Mese, Anno, Spettatori)

GruppiConcerti (IdGruppo*, IdConcerto*)

Gruppi (IdGruppo, NomeG, AnnoFondazione)

GruppiMusicisti(IdGruppo*, IdMusicista*)

Musicisti(IdMusicista, NomeM, Nazionalità, AnnoNascita, Compenso)

a) Si disegni lo schema relazionale in formato grafico (rettangoli e frecce).

b) Si scrivano le interrogazioni SQL che restituiscono le seguenti informazioni:

1. Per ciascun concerto tenuto a Viareggio, il nome del concerto ed il numero di gruppi che vi hanno partecipato
2. Il nome dei concerti ai quali hanno partecipato gruppi fondati dopo il 2000
3. I nomi dei gruppi che, dopo il 2000, hanno suonato solamente in concerti con almeno 5000 spettatori.
4. I nomi dei gruppi che hanno suonato solamente dopo il 2000 e in concerti con almeno 5000 spettatori
5. Per ogni gruppo che ha tenuto almeno tre concerti, il nome del gruppo ed il totale dei compensi per i partecipanti di Nazionalità italiana (per semplicità assumiamo che il gruppo abbia qualche partecipante di Nazionalità italiana)
6. Le coppie di Nomi dei gruppi che hanno suonato almeno una volta nello stesso concerto tra il 2000 ed il 2004
7. (**opzionale**) I nomi dei musicisti tali che per ogni concerto con più di 1000 spettatori c'era un gruppo in cui avesse suonato il musicista

c) si disegni l'albero di sintassi astratta di un'espressione algebrica ("albero logico") per l'interrogazione 2.

Basi di Dati, 16/11/2004, primo compito, compito B

- 1) Si vogliono gestire i dati necessari alla gestione dei corsi di una Facoltà e degli elenchi delle lezioni tenute. La Facoltà offre un insieme di Corsi di Laurea. Di un Corso di Laurea interessano il nome, il codice che lo identifica, e le materie insegnate. La stessa materia può essere insegnata in più Corsi di Laurea. Di ogni materia interessa il nome, il codice che la identifica, ed il numero di crediti. Per ciascuna materia, in ciascun anno accademico, può essere attivato un corso, più di uno, oppure nessuno. Ogni corso è attivato per una specifica materia. Inoltre, per ogni corso interessa l'Anno Accademico di attivazione, il docente responsabile, e l'insieme degli altri Docenti che insegnano in quel corso. Di un docente interessano un codice, che lo identifica, nome, cognome, ed un insieme di recapiti telefonici. Un docente può essere interno od esterno. Per un docente interno interessa anche l'anzianità di servizio. Solo un docente interno può essere responsabile di un corso. Per ogni corso il sistema mantiene un elenco delle lezioni relative. Per ogni lezione interessa conoscere data, ora, argomento, corso, ed i docenti presenti. Uno dei docenti presenti è presente come "docente", ed è uno di quelli che insegnano nel corso, incluso il responsabile. Gli altri (se ve ne sono altri) sono presenti come "assistenti", e non sono necessariamente inclusi tra coloro che insegnano nel corso.

a) Si disegni lo schema concettuale che descrive la situazione, indicando in modo testuale i vincoli non catturati graficamente.

b) Si traduca lo schema concettuale in uno schema relazionale grafico e testuale, usando la notazione $R(\underline{A}, B^*, \dots)$

- 2) Si consideri la base di dati:

Concerti (IdConcerto, NomeC, Città, Giorno, Mese, Anno, Spettatori)

GruppiConcerti (IdGruppo*, IdConcerto*)

Gruppi (IdGruppo, NomeG, AnnoFondazione)

GruppiMusicisti(IdGruppo*, IdMusicista*)

Musicisti(IdMusicista, NomeM, Nazionalità, AnnoNascita, Compenso)

a) Si disegni lo schema relazionale in formato grafico (rettangoli e frecce).

b) Si scrivano le interrogazioni SQL che restituiscono le seguenti informazioni:

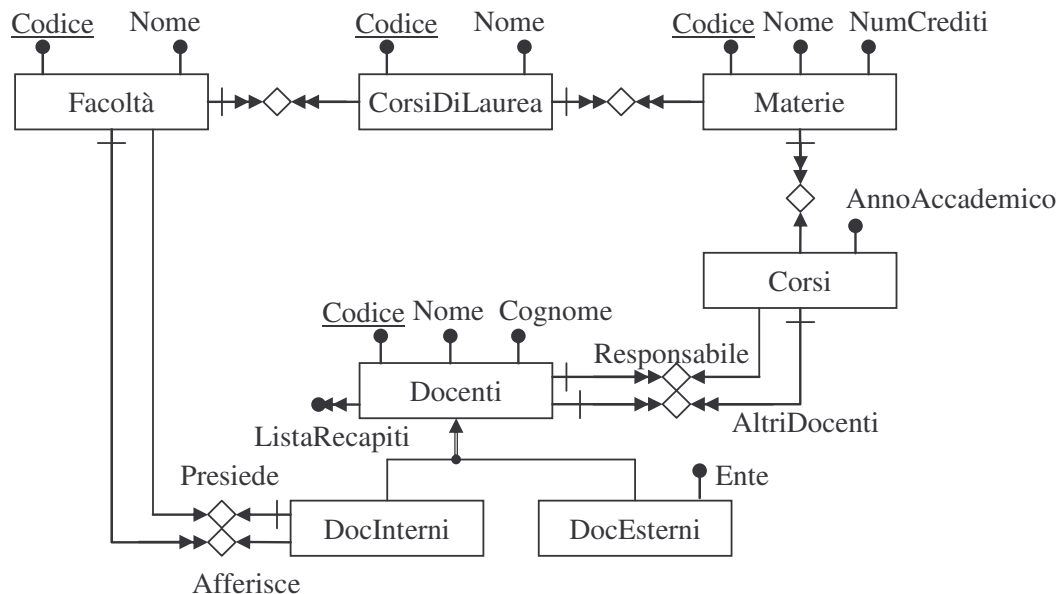
1. Le coppie di Nomi dei musicisti che hanno suonato almeno una volta nello stesso gruppo, fondato in un anno compreso tra il 2000 ed il 2004
2. Per ciascun gruppo fondato nel 2003, il nome del gruppo ed il numero di musicisti che lo formano
3. Il nome dei musicisti che hanno suonato in gruppi fondati dopo il 1998
4. I nomi dei gruppi nei quali tutti i musicisti di nazionalità italiana hanno un compenso superiore a 10000.
5. I nomi dei gruppi nei quali tutti i musicisti sono di nazionalità italiana ed hanno un compenso superiore a 10000.
6. Per ogni gruppo costituito da almeno tre musicisti, il nome del gruppo ed il totale degli spettatori nei concerti tenuti nel 2002 (per semplicità assumiamo che il gruppo abbia tenuto qualche concerto nel 2002)
7. (**opzionale**) I nomi dei concerti tali che per ogni musicista con Compenso maggiore di 10000 era presente un gruppo in cui avesse suonato il musicista

c) Si disegni l'albero di sintassi astratta di un'espressione algebrica ("albero logico") per l'interrogazione 3.

Basi di Dati, 16/11/2004, primo compitino, compito A, Soluzioni, V1.1

- 2) Si vogliono gestire i dati necessari alla programmazione didattica di un'Università. L'Università offre un insieme di Corsi di Laurea. Di un Corso di Laurea interessano il nome, il codice che lo identifica, la o le Facoltà che lo gestiscono, e le materie insegnate. Di una Facoltà interessa il nome, il codice, ed il docente che la presiede. La stessa materia può essere insegnata in più Corsi di Laurea. Di ogni materia interessa il nome, il codice che la identifica, ed il numero di crediti. Per ciascuna materia, in ciascun anno accademico, può essere attivato un corso, più di uno, oppure nessuno. Ogni corso è attivato per una specifica materia. Inoltre, per ogni corso interessa l'Anno Accademico di attivazione, il docente responsabile, e l'insieme degli altri Docenti che insegnano in quel corso. Di un docente interessano un codice, che lo identifica, nome, cognome, ed un insieme di recapiti telefonici. Un docente può essere interno od esterno. Per un docente esterno interessa l'ente da cui dipende. Per un docente interno interessa la Facoltà a cui afferisce. Solo un docente che afferisce ad una Facoltà può presiederla.

a) Si disegni lo schema concettuale che descrive la situazione, indicando in modo testuale i vincoli non catturati graficamente

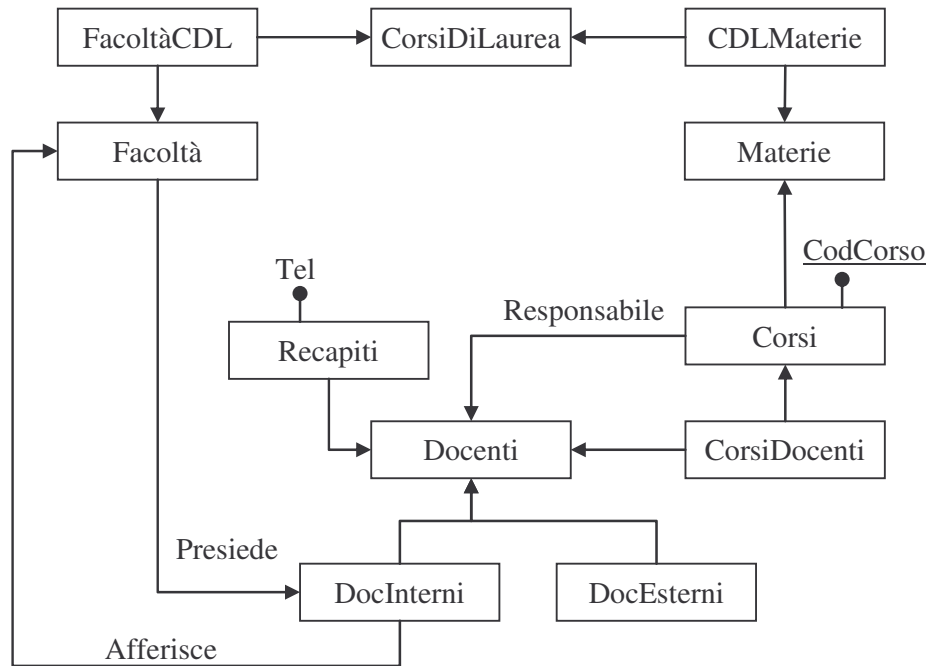


Vincoli non espressi graficamente:

- 1) Solo un docente che afferisce ad una Facoltà può presiederla
- 2) Il docente responsabile di un corso non è compreso nell'associazione AltriDocenti

b) Si traduca lo schema concettuale in uno schema relazionale grafico e testuale, usando la notazione $R(\underline{A}, B^*, \dots)$

Schema relazionale in formato grafico:



Schema relazionale in formato testuale:

Facoltà (CodFac, NomeFac, Preside*)
 CorsiDiLaurea (CodCdL, NomeCdL)
 FacoltàCDL (CodFac*, CodCdL*)
 Materie (CodMat, NomeMat, Crediti)
 CDLMaterie (CodCdL*, CodMat*)
 Corsi (CodCorso, CodMat*, Responsabile*, AnnoAcc)
 Docenti (CodDoc, NomeDoc, Cognome)
 CorsiDocenti (CodCorso*, CodDoc*)
 Recapiti (CodDoc*, Tel)
 DocInterni (CodDoc*, Afferenza*)
 DocEsterni (CodDoc*, Ente)

2) Si consideri la base di dati:

Concerti (IdConcerto, NomeC, Città, Giorno, Mese, Anno, Spettatori)

GruppiConcerti (IdGruppo*, IdConcerto*)

Gruppi (IdGruppo, NomeG, AnnoFondazione)

GruppiMusicisti(IdGruppo*, IdMusicista*)

Musicisti(IdMusicista, NomeM, Nazionalità, AnnoNascita, Compenso)

d) Si disegni lo schema relazionale in formato grafico (rettangoli e frecce).

e) Si scrivano le interrogazioni SQL che restituiscono le seguenti informazioni:

1. Per ciascun concerto tenuto a Viareggio, il nome del concerto ed il numero di gruppi che vi han partecipato

```
SELECT C.NomeC, Count(*)
FROM Concerti C, GruppiConcerti GC
WHERE C.IdConcerto = GC.IdConcerto And C.Città = "Viareggio"
GROUP BY C.IdConcerto, C.NomeC
```

2. Il nome dei concerti ai quali hanno partecipato gruppi fondati dopo il 2000

```
SELECT C.NomeC
FROM Concerti C, GruppiConcerti GC, Gruppi G
WHERE C.IdConcerto = GC.IdConcerto and GC.IdGruppo = G.IdGruppo
and G.AnnoFondazione > 2000
```

3. I nomi dei gruppi che, dopo il 2000, hanno suonato solamente in concerti con almeno 5000 spettatori.

```
SELECT G.NomeG
FROM Gruppi G
WHERE not exists (SELECT *
                  FROM GruppiConcerti GC, Concerti C
                  WHERE G.IdGruppo = GC.IdGruppo and
                        GC.IdConcerto = C.IdConcerto
                        and C.Anno > 2000 and C.Spettatori < 5000)
```

4. I nomi dei gruppi che hanno suonato solamente dopo il 2000 e in concerti con almeno 5000 spettatori

```
SELECT G.NomeG
FROM Gruppi G
WHERE not exists (SELECT *
                  FROM GruppiConcerti GC, Concerti C
                  WHERE G.IdGruppo = GC.IdGruppo and
                        GC.IdConcerto = C.IdConcerto
                        and not (C.Anno > 2000 and C.Spettatori >= 5000))
```

5. Per ogni gruppo che ha tenuto almeno tre concerti, il nome del gruppo ed il totale dei compensi per i partecipanti di Nazionalità italiana(per semplicità assumiamo che il gruppo abbia qualche partecipante di Nazionalità italiana)

```
SELECT G.NomeG, Sum(M.Compenso)
FROM Gruppi G, GruppiMusicisti GM, Musicisti M
WHERE G.IdGruppo = GM.IdGruppo and GM.IdMusicista = M.IdMusicista
and M.Nazionalità = 'Italiano'
and 3 <= (SELECT count(*) FROM GruppiConcerti GC
          WHERE G.IdGruppo = GC.IdGruppo)
GROUP BY G.IdGruppo, G.NomeG
oppure
```

```

SELECT G.NomeG, Sum(M.Compenso)
FROM Gruppi G, GruppiMusicisti GM, Musicisti M
WHERE G.IdGruppo = GM.IdGruppo and GM.IdMusicista = M.IdMusicista
      and M.Nazionalità = 'Italiana'
GROUP BY G.IdGruppo, G.NomeG
HAVING 3 <= (SELECT count(*) FROM GruppiConcerti GC
              WHERE G.IdGruppo = GC.IdGruppo)

```

6. *Le coppie di Nomi dei gruppi che hanno suonato almeno una volta nello stesso concerto tra il 2000 ed il 2004*

```

SELECT DISTINCT G1.NomeG, G2.NomeG
FROM Gruppi G1, GruppiConcerti GC1, Concerti C,
      GruppiConcerti GC2, Gruppi G2
WHERE G1.IdGruppo = GC1.IdGruppo and GC1.IdConcerto = C.IdConcerto
      and C.IdConcerto = GC2.IdConcerto and GC2.IdGruppo = G2.IdGruppo
      and C.Anno >= 2000 and C.Anno <= 2004 and G1.NomeG < G2.NomeG

```

7. *(opzionale) I nomi dei musicisti tali che per ogni concerto con più di 1000 spettatori c'era un gruppo in cui avesse suonato il musicista*

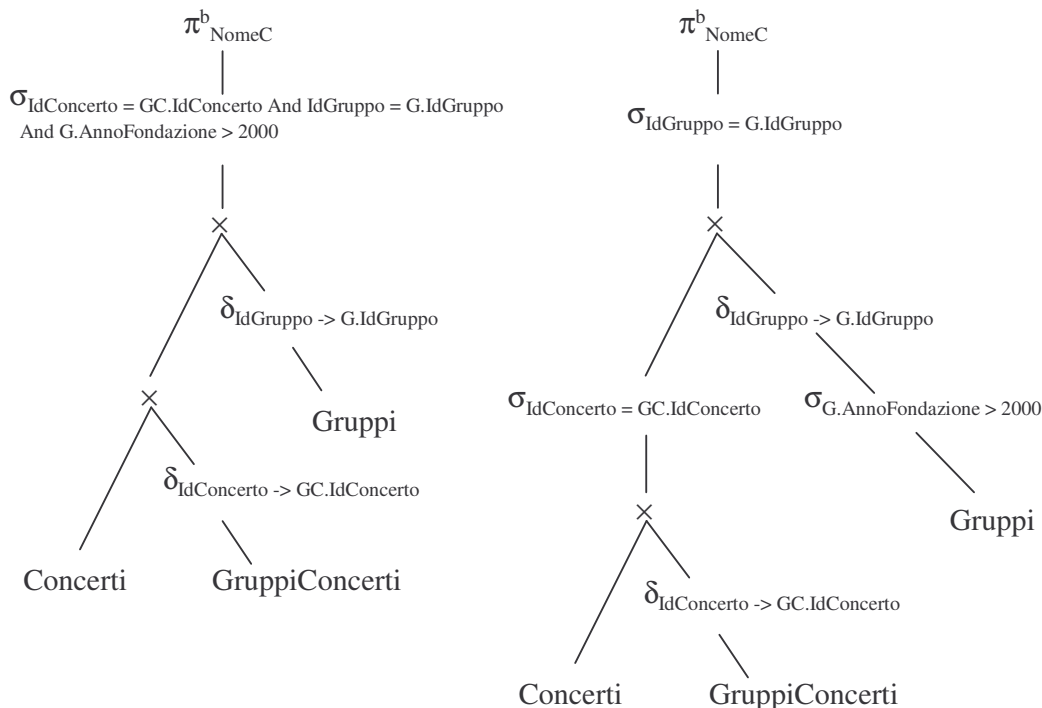
```

SELECT M.NomeM
FROM Musicisti M
WHERE not exists (SELECT *
                  FROM Concerti C
                  WHERE C.Spettatori > 1000 and not exists (SELECT *
                                                            FROM GruppiConcerti GC, GruppiMusicisti GM
                                                            WHERE C.IdConcerto = GC.IdConcerto
                                                                  and GC.IdGruppo = GM.IdGruppo
                                                                  and GM.IdMusicista = M.IdMusicista))

```

- f) *si disegni l'albero di sintassi astratta di un'espressione algebrica ("albero logico") per l'interrogazione 2.*

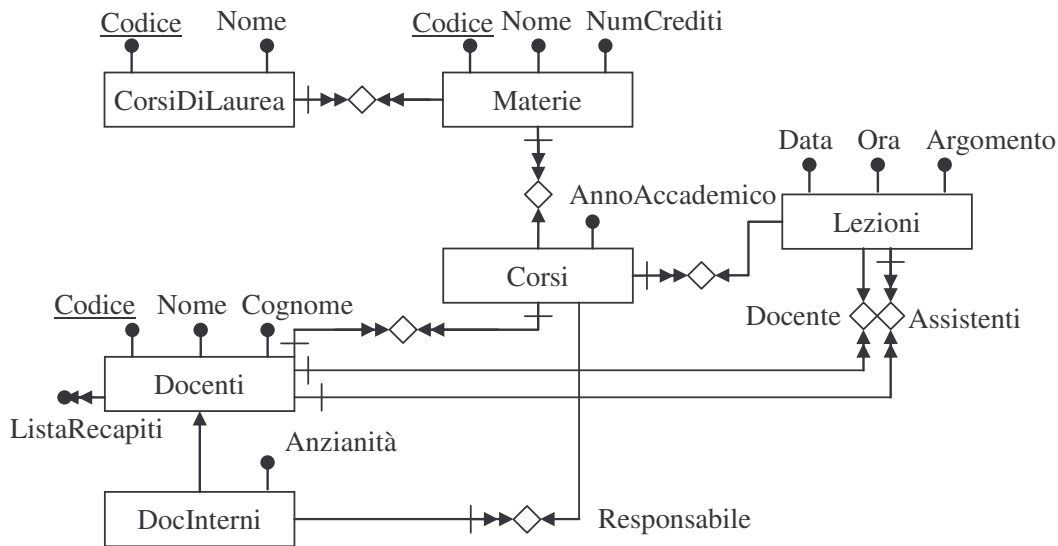
I due alberi seguenti vanno entrambi bene:



Basi di Dati, 16/11/2004, primo compitino, compito B

- 2) Si vogliono gestire i dati necessari alla gestione dei corsi di una Facoltà e degli elenchi delle lezioni tenute. La Facoltà offre un insieme di Corsi di Laurea. Di un Corso di Laurea interessano il nome, il codice che lo identifica, e le materie insegnate. La stessa materia può essere insegnata in più Corsi di Laurea. Di ogni materia interessa il nome, il codice che la identifica, ed il numero di crediti. Per ciascuna materia, in ciascun anno accademico, può essere attivato un corso, più di uno, oppure nessuno. Ogni corso è attivato per una specifica materia. Inoltre, per ogni corso interessa l'Anno Accademico di attivazione, il docente responsabile, e l'insieme degli altri Docenti che insegnano in quel corso. Di un docente interessano un codice, che lo identifica, nome, cognome, ed un insieme di recapiti telefonici. Un docente può essere interno od esterno. Per un docente interno interessa anche l'anzianità di servizio. Solo un docente interno può essere responsabile di un corso. Per ogni corso il sistema mantiene un elenco delle lezioni relative. Per ogni lezione interessa conoscere data, ora, argomento, corso, ed i docenti presenti. Uno dei docenti presenti è presente come "docente", ed è uno di quelli che insegnano nel corso, incluso il responsabile. Gli altri (se ve ne sono altri) sono presenti come "assistenti", e non sono necessariamente inclusi tra coloro che insegnano nel corso.

a) Si disegni lo schema concettuale che descrive la situazione, indicando in modo testuale i vincoli non catturati graficamente.



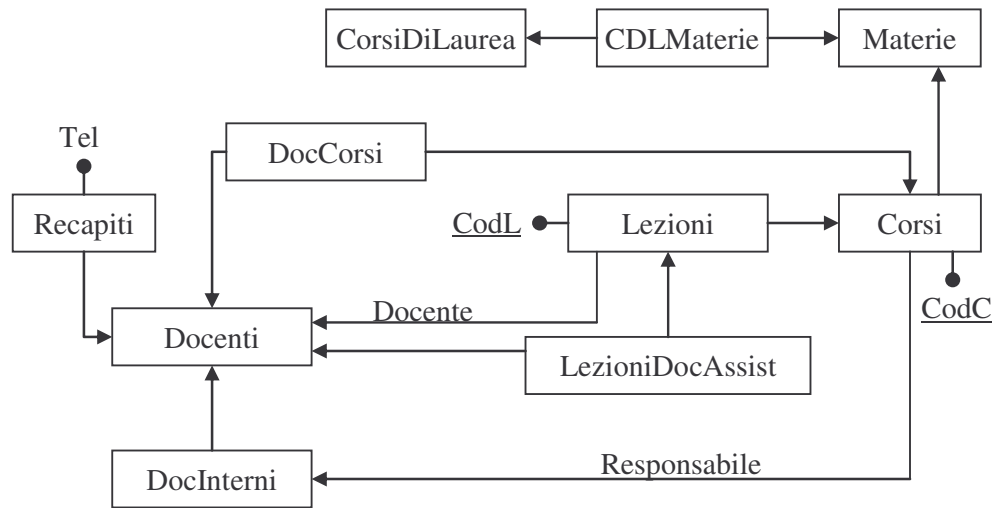
Vincoli non espressi graficamente:

- Il docente responsabile di un corso non è compreso nell'istanza dell'associazione tra Corsi e Docenti relativa allo stesso corso (ovvero: $\text{forall } c \text{ in Corsi: } c.\text{Responsabile} \text{ not in } c.\text{Docenti}$)
- Il docente presente come "docente" è uno di quelli che insegnano nel corso, incluso il responsabile ($\text{forall } l \text{ in Lezioni: } l.\text{Docente} \text{ in } (l.\text{Corsi}.\text{Docenti} \text{ union } l.\text{Corsi}.\text{Responsabile})$)
- Il docente presente a lezione come "docente" non è presente alla stessa lezione come "assistente" ($\text{forall } l \text{ in Lezioni: } l.\text{Docente} \text{ not in } l.\text{Assistenti}$)

Attenzione, la frase: *Gli altri (se ve ne sono altri) sono presenti come "assistenti", e non sono necessariamente inclusi tra coloro che insegnano nel corso* non è un vincolo, perché esprime una possibilità, e non un obbligo

b) Si traduca lo schema concettuale in uno schema relazionale grafico e testuale, usando la notazione $R(\underline{A}, B^*, \dots)$

Schema relazionale in formato grafico:



Schema relazionale in formato testuale:

CorsiDiLaurea ($\underline{\text{CodCdL}}$, NomeCdL)
 Materie ($\underline{\text{CodMat}}$, NomeMat, Crediti)
 CDLMaterie ($\underline{\text{CodCdL}^*}$, $\underline{\text{CodMat}^*}$)
 Corsi ($\underline{\text{CodC}}$, CodMat^* , Responsabile^* , AnnoAcc)
 Docenti ($\underline{\text{CodDoc}}$, NomeDoc, Cognome)
 Recapiti ($\underline{\text{CodDoc}^*}$, Tel)
 DocInterni ($\underline{\text{CodDoc}^*}$, Anzianità)
 CorsiDocenti ($\underline{\text{CodC}^*}$, $\underline{\text{CodDoc}^*}$)
 Lezioni ($\underline{\text{CodL}}$, CodC^* , Docente^*)
 LezioniDocAss ($\underline{\text{CodL}^*}$, $\underline{\text{CodDoc}^*}$)

2) Si consideri la base di dati:

Concerti (IdConcerto, NomeC, Città, Giorno, Mese, Anno, Spettatori)

GruppiConcerti (IdGruppo*, IdConcerto*)

Gruppi (IdGruppo, NomeG, AnnoFondazione)

GruppiMusicisti (IdGruppo*, IdMusicista*)

Musicisti (IdMusicista, NomeM, Nazionalità, AnnoNascita, Compenso)

d) Si disegni lo schema relazionale in formato grafico (rettangoli e frecce).

e) Si scrivano le interrogazioni SQL che restituiscono le seguenti informazioni:

1. Le coppie di Nomi dei musicisti che hanno suonato almeno una volta nello stesso gruppo, fondato in un anno compreso tra il 2000 ed il 2004

```
SELECT DISTINCT M1.NomeM, M2.NomeM
FROM      Musicisti M1, GruppiMusicisti GM1, Gruppi G,
          GruppiMusicisti GM2, Musicisti M2
WHERE     M1.IdMusicista = GM1.IdMusicista and GM1.IdGruppo = G.IdGruppo
          and G.IdGruppo = GM2.IdGruppo and GM2.IdMusicista = M2.IdMusicista
          and G.AnnoFondazione >= 2000 and G.AnnoFondazione <= 2004
          and M1.NomeM < M2.NomeM
```

2. Per ciascun gruppo fondato nel 2003, il nome del gruppo ed il numero di musicisti che lo formano

```
SELECT G.NomeG, Count(*)
FROM      Gruppi G, GruppiMusicisti GM
WHERE     G.IdGruppo = GM.IdGruppo And G. AnnoFondazione = 2003
GROUP BY  G.IdGruppo, G.NomeG
```

3. Il nome dei musicisti che hanno suonato in gruppi fondati dopo il 1998

```
SELECT M.NomeM
FROM      Musicisti M, GruppiMusicisti GM, Gruppi G
WHERE     M.IdMusicista = GM.IdMusicista and GM.IdGruppo = G.IdGruppo
          And G.AnnoFondazione > 1998
```

4. I nomi dei gruppi nei quali tutti i musicisti di nazionalità italiana hanno un compenso superiore a 10000.

```
SELECT G.NomeG
FROM      Gruppi G
WHERE     not exists (SELECT *
                     FROM      GruppiMusicisti GM, Musicisti M
                     WHERE     G.IdGruppo = GM.IdGruppo
                             GM.IdMusicista = M.IdMusicista
                             and M.Nazionalità = 'Italiano' and M.Compenso <= 10000)
```

5. I nomi dei gruppi nei quali tutti i musicisti sono di nazionalità italiana ed hanno un compenso superiore a 10000.

```
SELECT G.NomeG
FROM      Gruppi G
WHERE     not exists (SELECT *
                     FROM      GruppiMusicisti GM, Musicisti M
                     WHERE     G.IdGruppo = GM.IdGruppo
                             GM.IdMusicista = M.IdMusicista
                             and not (M.Nazionalità = 'Italiano' and M.Compenso > 10000))
```

6. Per ogni gruppo costituito da almeno tre musicisti, il nome del gruppo ed il totale degli spettatori nei concerti tenuti nel 2002 (per semplicità assumiamo che il gruppo abbia tenuto qualche concerto nel 2002)

```

SELECT G.NomeG, Sum(C.Spettatori)
FROM  Concerti C, GruppiConcerti GC, Gruppi G
WHERE C.IdConcerto = GC.IdConcerto and GC.IdGruppo = G.IdGruppo
      And C.Anno = 2002 and 3 <= (SELECT count(*) FROM GruppiMusicisti GM
                                WHERE G.IdGruppo = GM.IdGruppo)

GROUP BY  G.IdGruppo, G.NomeG
oppure
SELECT G.NomeG, Sum(C.Spettatori)
FROM  Concerti C, GruppiConcerti GC, Gruppi G
WHERE C.IdConcerto = GC.IdConcerto and GC.IdGruppo = G.IdGruppo
      And C.Anno = 2002
GROUP BY  G.IdGruppo, G.NomeG
HAVING  3 <= (SELECT count(*) FROM GruppiMusicisti GM
              WHERE G.IdGruppo = GM.IdGruppo)

```

7. (**opzionale**) *I nomi dei concerti tali che per ogni musicista con Compenso maggiore di 10000 era presente un gruppo in cui avesse suonato il musicista*

```

SELECT C.NomeC
FROM  Concerti C
WHERE not exists (SELECT *
                  FROM  Musicisti M
                  WHERE M.Compenso > 10000 and not exists (SELECT *
                                                            FROM GruppiConcerti GC, GruppiMusicisti GM
                                                            WHERE C.IdConcerto = GC.IdConcerto
                                                           and GC.IdGruppo = GM.IdGruppo
                                                           and GM.IdMusicista = M.IdMusicista))

```

- f) *Si disegni l'albero di sintassi astratta di un'espressione algebrica ("albero logico") per l'interrogazione 3.*

Analogo alla soluzione di 2-(f) nel compito A