

Capitolo 3

Esercizio 3.1 Considerare una relazione $R(A, \underline{B}, \underline{C}, D, E)$. Indicare quali delle seguenti proiezioni hanno certamente lo stesso numero di ennuple di R :

1. $\pi_{ABCD}(R)$
2. $\pi_{AC}(R)$
3. $\pi_{BC}(R)$
4. $\pi_C(R)$
5. $\pi_{CD}(R)$.

Soluzione:

- | | | |
|----|-----------------|------------|
| 1. | $\pi_{ABCD}(R)$ | SÌ |
| 2. | $\pi_{AC}(R)$ | NO |
| 3. | $\pi_{BC}(R)$ | SÌ |
| 4. | $\pi_C(R)$ | NO |
| 5. | $\pi_{CD}(R)$ | NO. |

Esercizio 3.2 Considerare le relazioni $R_1(\underline{A}, B, C)$ e $R_2(\underline{D}, E, F)$ aventi rispettivamente cardinalità N_1 e N_2 . Assumere che sia definito un vincolo di integrità referenziale fra l'attributo C di R_1 e la chiave D di R_2 . Indicare la cardinalità di ciascuno dei seguenti join (specificare l'intervallo nel quale essa può variare):

1. $R_1 \bowtie_{A=D} R_2$
2. $R_1 \bowtie_{C=D} R_2$
3. $R_1 \bowtie_{A=F} R_2$
4. $R_1 \bowtie_{B=E} R_2$.

Soluzione:

1. $R_1 \bowtie_{A=D} R_2$ **compresa fra 0 e il minimo fra N_1 e N_2**
2. $R_1 \bowtie_{C=D} R_2$ **esattamente N_1**
3. $R_1 \bowtie_{A=F} R_2$ **compresa fra 0 e N_2**
4. $R_1 \bowtie_{B=E} R_2$ **compresa fra 0 e $N_1 \cdot N_2$.**

Esercizio 3.3 Considerare le seguenti relazioni (tutte senza valori nulli):

- $R_1(\underline{A}, B, C)$, con vincolo di integrità referenziale fra C e R_2 e con cardinalità $N_1 = 100$
- $R_2(\underline{D}, E, F)$, con vincolo di integrità referenziale fra F e R_3 e con cardinalità $N_2 = 200$
- $R_3(\underline{G}, H, I)$, con cardinalità $N_3 = 50$.

Indicare la cardinalità del risultato di ciascuna delle seguenti espressioni (specificando l'intervallo nel quale essa può variare)

1. $\pi_{AB}(R_1)$
2. $\pi_E(R_2)$
3. $\pi_{BC}(R_1)$
4. $\pi_G(R_3)$
5. $R_1 \bowtie_{A=D} R_2$
6. $R_1 \bowtie_{C=D} R_2$
7. $R_3 \bowtie_{I=A} R_1$
8. $(R_3 \bowtie_{I=A} R_1) \bowtie_{C=D} R_2$
9. $(R_3 \bowtie_{I=A} R_1) \bowtie_{C=E} R_2$.

Soluzione:

1. $|\pi_{AB}(R_1)| = 100$
la cardinalità dell'operazione è esattamente pari a 100 poiché la proiezione coinvolge la chiave della relazione.
2. $1 \leq |\pi_E(R_2)| \leq 200$
la cardinalità dell'operazione è compresa tra 1 e 200. 1 come cardinalità minima in quanto non possono essere presenti valori nulli e non essendo coinvolta la chiave tutti i valori di E potrebbero essere uguali. 200 come cardinalità massima poiché tutti i valori di E potrebbero essere diversi ed al più saranno in numero tanti quanti la chiave.
3. $1 \leq |\pi_{BC}(R_1)| \leq 100$
la cardinalità dell'operazione è compresa tra 1 e 100. 1 come cardinalità minima in quanto non possono essere presenti valori nulli e non essendo coinvolta la chiave tutti i valori della proiezione su B e C potrebbero essere uguali. 100 come cardinalità massima poiché tutti i valori della proiezione su B e C potrebbero essere diversi ed al più saranno in numero tanti quanti sono gli elementi della chiave.
4. $|\pi_G(R_3)| = 50$
la cardinalità dell'operazione è esattamente 50 poiché G è chiave per R_3 .
5. $0 \leq |R_1 \bowtie_{A=D} R_2| \leq 100$
la cardinalità dell'operazione è compresa tra 0 e 100. 0 come cardinalità minima in quanto il join potrebbe essere vuoto. 100 come cardinalità massima poiché al più tutti i valori di A si combineranno al più con un valore di D poiché sia A che D sono delle chiavi per le due relazioni.

6. $|R_1 \bowtie_{C=D} R_2| = 100$
 la cardinalità dell'operazione è esattamente 100 poiché esiste un vincolo di integrità referenziale tra C e D e quindi ogni valore di C si combina con esattamente un valore di D.
7. $0 \leq |R_3 \bowtie_{I=A} R_1| \leq 50$
 la cardinalità dell'operazione è compresa tra 0 e 50 poiché al più ogni valore di I si combina con esattamente un valore di A.
8. $0 \leq |(R_3 \bowtie_{I=A} R_1) \bowtie_{C=D} R_2| \leq 50$
 riutilizzando il risultato ottenuto al punto precedente abbiamo che la cardinalità della primo join è compresa tra 0 e 50. La cardinalità del secondo join rimane invariata rispetto a quella ottenuta con il primo in quanto ogni valore di C si combina esattamente con un valore di D e sarà quindi compresa 0 e 50.
9. $0 \leq |(R_3 \bowtie_{I=A} R_1) \bowtie_{C=E} R_2| \leq 10000$
 riutilizzando il risultato ottenuto al punto precedente abbiamo che la cardinalità del primo join è compresa tra 0 e 50. La cardinalità del secondo join sarà compresa tra 0 e 10000 in quanto se i valori di C e D sono tutti diversi si avrà un join vuoto, mentre se sono tutti uguali si avrà il prodotto cartesiano delle tuple.

Esercizio 3.4 Date le relazioni R_1 (A,B,C), R_2 (E,F,G,H), R_3 (J,K), R_4 (L,M) aventi rispettivamente cardinalità N_1 , N_2 , N_3 e N_4 quali vincoli di chiave e di integrità referenziale vanno definiti (se possibile) affinché nei casi seguenti valgano le condizioni indicate?

1. $|R_1 \bowtie_{B=G} R_2| = N_1$
2. $|R_2 \bowtie_{G=B} R_1| = N_1$
3. $|\pi_J(R_3)| = N_3$
4. $|\pi_J(R_3)| < N_3$
5. $|\pi_L(R_4) \bowtie_{L=J} R_3| = N_4$
6. $|R_4 \bowtie_{M=K} R_3| = N_3$
7. $|R_1 \bowtie_{BC=GK} R_2| = N_2$
8. $|R_1 \bowtie_{BC=GH} R_2| = N_1$
9. $0 \leq |R_1 \bowtie_{A=F} R_2| \leq N_1 \cdot N_2$
10. $|R_1 \bowtie_{A=F} R_2| = N_1 \cdot N_2$.

Soluzione:

1. $|R_1 \bowtie_{B=G} R_2| = N_1$
 B chiave, G chiave e vincolo di integrità referenziale tra B e G.
2. $|R_2 \bowtie_{G=B} R_1| = N_1$
 B chiave, G chiave e vincolo di integrità referenziale tra B e G.
3. $|\pi_J R_3| = N_3$
 J chiave.
4. $|\pi_J R_3| < N_3$
 Non è possibile imporre vincoli che garantiscano lo strettamente minore.
5. $|\pi_L(R_4) \bowtie_{L=J} R_3| = N_4$
 L chiave, J chiave e vincolo di integrità referenziale tra L e J.
6. $|R_4 \bowtie_{M=K} R_3| = N_3$
 K chiave, M chiave e vincolo di integrità referenziale tra K e M.
7. $|R_1 \bowtie_{BC=GK} R_2| = N_2$
 BC chiave, GK chiave e vincolo di integrità referenziale tra GK e BC.
8. $|R_1 \bowtie_{BC=GH} R_2| = N_1$
 BC chiave, GH chiave e vincolo di integrità referenziale tra BC e GH.
9. $0 \leq |R_1 \bowtie_{A=F} R_2| \leq N_1 \cdot N_2$
 Nessun vincolo, perché la cardinalità è sempre nell'intervallo.
10. $|R_1 \bowtie_{A=F} R_2| = N_1 \cdot N_2$
 Non è possibile imporre vincoli in quanto A e F dovrebbero essere non chiave e con un unico valore.

Esercizio 3.5 Con riferimento ai punti 1 e 2 dell'esercizio precedente, considerando i vincoli di integrità imposti in ogni punto spiegare le differenze che si avrebbero nei risultati delle operazioni nel caso di join destro e join sinistro e come cambia di conseguenza la cardinalità del risultato.

Soluzione:

1. Nel caso di join destro il risultato rimane invariato in quanto tutte le tuple di R_1 già partecipano al risultato per il vincolo di integrità referenziale tra B e G. Se invece prendiamo in considerazione un join sinistro il risultato cambia in quanto in esso saranno presenti tutte le tuple di R_2 che prima non partecipavano al risultato. La cardinalità del risultato finale sarà dunque pari a N_2 .
2. Nel caso di join sinistro la cardinalità del risultato sarà pari a N_2 in quanto partecipano al risultato anche le tuple di N_2 che altrimenti non partecipavano al risultato. Nel caso di join destro ci ritroviamo nella situazione precedente quindi, considerando la presenza del vincolo di integrità referenziale tra due attributi B e G chiavi la cardinalità sarà pari a N_1 .

Esercizio 3.6

Considerare lo schema di base di dati contenente le relazioni:

Film(CodiceFilm, Titolo, Regista, Anno, CostoNoleggio)
Artisti(CodiceAttore, Cognome, Nome, Sesso, DataNascita, Nazionalità)
Interpretazioni(CodiceFilm, CodiceAttore, Personaggio)

1. Mostrare una base di dati su questo schema per la quale i join fra le varie relazioni siano tutti completi.
2. Supponendo che esistano due vincoli di integrità referenziale fra la relazione **Interpretazioni** e le altre due, discutere i possibili casi di join non completo.
3. Mostrare un prodotto cartesiano che coinvolga relazioni in questa base di dati.
4. Mostrare una base di dati per la quale uno (o più) dei join sia vuoto.

Soluzione:

FILM

CodiceFilm	Titolo	Regista	Anno	CostoNoleggio
145684	Armageddon	15434	1997	5000000
457343	La vita è bella	67532	1998	1000000
563822	Ronin	34573	1997	2000000

ARTISTI

CodiceAttore	Cognome	Nome	Sesso	DataNascita	Nazionalità
67532	Benigni	Roberto	M	14/03/1950	Italiana
12456	DeNiro	Robert	M	22/04/1951	Americana
45673	Braschi	Nicoletta	F	1/05/1954	Italiana
67777	Willis	Bruce	M	3/2/1959	Americana
12345	Tyler	Liv	F	18/02/1962	Americana

INTERPRETAZIONI

CodiceFilm	CodiceAttore	Personaggio
457343	67532	Guido
457343	45673	Dora
145684	67777	Harry
145684	12345	Grace
563822	12456	Sam

1. In questa istanza della base di dati, i due join (naturali) tra **INTERPRETAZIONI** e **ARTISTI** e tra **INTERPRETAZIONI** e **FILM** sono completi (in quanto non vi sono tuple che non partecipano al risultato)
2. Se inserissimo tutti i registi nella tabella **ARTISTI**, allora il join con **INTERPRETAZIONI** non sarebbe completo. Il vincolo di integrità referenziale in questo schema è posto su **CodiceAttore** e su **CodiceFilm**. Nella tabella **INTERPRETAZIONI** non ha senso una tupla in cui i valori **CodiceFilm** e **CodiceAttore** non abbiano corrispondenza nelle tabelle **FILM** e **ARTISTI**. È comunque possibile ammettere tuple della tabella **ARTISTI** senza corrispondenza in

INTERPRETAZIONI (ad esempio, nel caso dell'introduzione dei registi tra gli artisti). Questa situazione causa join incompleti.

3. Un esempio di prodotto cartesiano tra **ARTISTI** e **FILM** sulla base di dati è qui riportato:

CodiceFilm	Titolo	Reg.	Anno	CostoNo leggio	CodiceAttore	Cognome	Nome	Sesso	DataNascita	Nazionalità
145684	Armageddon	15434	1997	5000000	67532	Benigni	Roberto	M	14/03/50	Italiana
457343	La vita è bella	67532	1998	1000000	67532	Benigni	Roberto	M	14/03/50	Italiana
563822	Ronin	34573	1997	2000000	67532	Benigni	Roberto	M	14/03/50	Italiana
145684	Armageddon	15434	1997	5000000	12456	DeNiro	Robert	M	22/04/51	Americana
457343	La vita è bella	67532	1998	1000000	12456	DeNiro	Robert	M	22/04/51	Americana
563822	Ronin	34573	1997	2000000	12456	DeNiro	Robert	M	22/04/51	Americana
145684	Armageddon	15434	1997	5000000	45673	Braschi	Nicoletta	F	1/05/54	Italiana
457343	La vita è bella	67532	1998	1000000	45673	Braschi	Nicoletta	F	1/05/54	Italiana
563822	Ronin	34573	1997	2000000	45673	Braschi	Nicoletta	F	1/05/54	Italiana
145684	Armageddon	15434	1997	5000000	67777	Willis	Bruce	M	3/2/59	Americana
457343	La vita è bella	67532	1998	1000000	67777	Willis	Bruce	M	3/2/59	Americana
563822	Ronin	34573	1997	2000000	67777	Willis	Bruce	M	3/2/59	Americana
145684	Armageddon	15434	1997	5000000	12345	Tyler	Liv	F	18/02/62	Americana
457343	La vita è bella	67532	1998	1000000	12345	Tyler	Liv	F	18/02/62	Americana
563822	Ronin	34573	1997	2000000	12345	Tyler	Liv	F	18/02/62	Americana

4. Un esempio di Base di dati con join vuoti può essere il seguente: i valori nella tabella **INTERPRETAZIONI** non hanno corrispondenza nelle altre tabelle:

FILM

CodiceF	Titolo	Regista	Anno	CostoNoleggio
145684	Armageddon	15434	1997	5000000
457343	La vita e bella	67532	1998	1000000
563822	Ronin	34573	1997	2000000

ARTISTI

CodiceAttore	Cognome	Nome	Sesso	DataNascita	Nazionalità
67532	Benigni	Roberto	M	14/03/1950	Italiana
12456	DeNiro	Robert	M	22/04/1951	Americana
45673	Braschi	Nicoletta	F	1/05/1954	Italiana
67777	Willis	Bruce	M	3/2/1959	Americana
12345	Tyler	Liv	F	18/02/1962	Americana

INTERPRETAZIONI

CodiceFilm	CodiceAttore	Personaggio
478384	67500	Peter
467343	42223	Dora
185682	67754	Harry
945684	99845	John
963822	12000	Mark

Esercizio 3.7

Con riferimento allo schema nell'esercizio 3.1, formulare in algebra relazionale, in calcolo su domini, in calcolo su tuple e in Datalog le interrogazioni che trovano:

1. i titoli dei film nei quali Henry Fonda sia stato interprete;
2. i titoli dei film per i quali il regista sia stato anche interprete;
3. i titoli dei film in cui gli attori noti siano tutti dello stesso sesso.

Soluzione:

1.

Algebra Relazionale:

$$\Pi_{\text{Titolo}} (\text{FILM} \triangleright \triangleleft (\sigma_{(\text{Nome} = \text{"Henry"}) \wedge (\text{Cognome} = \text{"Fonda"})} (\text{ARTISTI}) \triangleright \triangleleft \text{INTERPRETAZIONI}))$$

Calcolo dei Domini:

{ Titolo: t | FILM (CodiceFilm: fn, Titolo: t, Regista: d, Anno: y, CostoNoleggio: pc) \wedge
 ARTISTI (CodiceAttore: an, Cognome: cogn, Nome: n, Sesso: s,
 DataNascita: b, Nazionalità: naz) \wedge
 INTERPRETAZIONI (CodiceFilm: fn, CodiceAttore: an, Personaggio: ch) \wedge
 (cogn = "Fonda") \wedge (n = "Henry") }

Calcolo delle Tuple:

{ F.titolo | F(FILM), A(ARTISTI), I(INTERPRETAZIONI) |
 F.CodiceFilm = I.CodiceFilm \wedge A.CodiceAttore = I.CodiceAttore \wedge
 A.Cognome = "Fonda" \wedge A.Nome = "Henry" }

Datalog:

FILMCONFONDA (Titolo: t) \leftarrow
 FILM (CodiceFilm: fn, Titolo: t, Regista: d, Anno: y, CostoNoleggio: pc),
 ARTISTI (CodiceAttore: an, Cognome: "Fonda", Nome: "Henry", Sesso: s, DataNascita: b,
 Nazionalità : naz),
 INTERPRETAZIONI (CodiceFilm : fn, CodiceAttore : an, Personaggio: ch)

2.

Algebra Relazionale:

$$\Pi_{\text{Titolo}} (\sigma_{(\text{Regista} = \text{CodiceAttore})} (\text{INTERPRETAZIONI} \triangleright \triangleleft \text{FILM}))$$

Calcolo dei Domini:

{ Titolo: t | FILM (CodiceFilm : fn, Titolo: t, Regista: d, Anno: y, CostoNoleggio: pc) \wedge
 INTERPRETAZIONI (CodiceFilm : fn, CodiceAttore: d, Personaggio: ch) }

Calcolo delle tuple:

{ F.Titolo | F(FILM), I(INTERPRETAZIONI) |
 F.CodiceFilm = I.CodiceFilm \wedge F.Regista = I.CodiceAttore }

Datalog:

FILMCONREGISTAARTISTA(Titolo: t) ←
 FILM (CodiceFilm:fn, Titolo:t,Regista:d, Anno: y,CostoNoleggio: pc)
 INTERPRETAZIONI (CodiceFilm : fn, CodiceAttore: d, Personaggio: ch)

3.

Algebra Relazionale:

$\Pi_{\text{Titolo}}(\text{FILM}) -$

$\Pi_{\text{Titolo}}(\text{FILM}) \triangleright \triangleleft \sigma_{\text{Sex} < \text{Sex1}}((\text{ARTISTI} \triangleright \triangleleft \text{INTERPRETAZIONI}) \triangleright \triangleleft$
 $\rho_{\text{Sex1} \leftarrow \text{Sex}}(\Pi_{\text{CodiceFilm, Sex}}(\text{ARTISTI} \triangleright \triangleleft \text{INTERPRETAZIONI}))$

Calcolo dei Domini:

{ Titolo: t | FILM (CodiceFilm: fn, Titolo:t, Regista:d, Anno: y, CostoNoleggio: pc) ∧
 ¬∃t1(∃d1(∃y1(∃pd1(FILM(CodiceFilm: fn,Titolo:t1,Regista:d1, Anno: y1, CostoNoleggio:pd1)
 ∧
 ARTISTI (CodiceAttore: an1,Cognome: sur1,Nome: n1, Sesso: s1,
 DataNascita: b1, Nazionalità :nat1) ∧
 ARTISTI (CodiceAttore: an2,Cognome: sur2,Nome: n2, Sesso: s2,
 DataNascita: b2, Nazionalità :nat2) ∧
 INTERPRETAZIONI (CodiceFilm: fn, CodiceAttore: an1, Personaggio: ch1)
 ∧
 INTERPRETAZIONI (CodiceFilm: fn, CodiceAttore: an2, Personaggio: ch2)
 ∧
 (s1 ≠ s2))))) }

Calcolo delle Tuple:

{ F.Titolo | F(FILM) |
 ¬(∃F1(FILM)(∃A1(ARTISTI) (∃A2(ARTISTI) (∃I 1(INTERPRETAZIONI)
 (∃I 2(INTERPRETAZIONI) ∧
 A1.CodiceAttore=I1.CodiceAttore∧F1.CodiceFilm=I1.CodiceFilm ∧
 A2.CodiceAttore=I2. CodiceAttore ∧I1.CodiceFilm =I2.CodiceFilm ∧
 A1.Sesso≠A2.Sesso))))) }

Datalog:

SESSODIVERSO (CodiceFilm: fn) ←
 FILM (CodiceFilm: fn, Titolo: t,Regista: d, Anno: y, CostoNoleggio: pc),
 ARTISTI (CodiceAttore: an1, Cognome:sur1 , Nome: n1, Sesso: s1, DataNascita:b1,
 Nazionalità :nat1),
 ARTISTI (CodiceAttore: an2, Cognome:sur2 , Nome: n2, Sesso: s2, DataNascita:b2,
 Nazionalità :nat2),
 INTERPRETAZIONI (FilmNumber: fn, CodiceAttore: an1, Personaggio: ch1),
 INTERPRETAZIONI (FilmNumber: fn, CodiceAttore: an2, Personaggio: ch2),
 (s1 ≠ s2)

STESSOSESSO(Titolo: t) ←

FILM (CodiceFilm: fn, Titolo: t,Regista: d, Anno: y, CostoNoleggio: pc),
 NOT SESSODIVERSO (CodiceFilm: fn)

Esercizio 3.8

Si consideri lo schema di base di dati che contiene le seguenti relazioni:

DEPUTATI (Codice, Cognome, Nome, Commissione, Provincia, Collegio)

COLLEGI (Provincia, Numero, Nome)

PROVINCE (Sigla, Nome, Regione)

REGIONI (Codice, Nome)

COMMISSIONI (Numero, Nome, Presidente)

Formulare in algebra relazionale, in calcolo dei domini e in calcolo delle tuple le seguenti interrogazioni:

1. Trovare nome e cognome dei presidenti di commissioni cui partecipa almeno un deputato eletto in una provincia della Sicilia;
2. Trovare nome e cognome dei deputati della commissione Bilancio;
3. Trovare nome, cognome e provincia di elezione dei deputati della commissione Bilancio;
4. Trovare nome, cognome, provincia e regione di elezione dei deputati della commissione Bilancio;
5. Trovare le regioni in cui vi sia un solo collegio, indicando nome e cognome del deputato ivi eletto;
6. Trovare i collegi di una stessa regione in cui siano stati eletti deputati con lo stesso nome proprio.

Soluzione:

1.

Algebra Relazionale:

$$\begin{aligned} & \Pi_{\text{Nom, Cogn}} \\ & ((\rho_{\text{Nom, Cogn} \leftarrow \text{Nome, Cognome}}(\text{DEPUTATI})) \triangleright \triangleleft_{\text{Presidente=Codice}} \\ & (\text{COMMISSIONI} \triangleright \triangleleft_{\text{Numero=Comm}} (\rho_{\text{Comm} \leftarrow \text{Commissione}}(\text{DEPUTATI} \triangleright \triangleleft_{\text{Provincia=Sigla}} \\ & (\text{PROVINCE} \triangleright \triangleleft_{\text{Regione=Codice}} (\\ & \sigma_{\text{Nome='Sicilia'}}(\text{REGIONI})))))) \end{aligned}$$

Calcolo dei Domini:

```
{ Cognome: surp, Nome: fnp |
DEPUTATI (Codice: np, Cognome: surp, Nome: fnp,
Commissione: cmp, Provincia: sigl, Collegio: cstp) ∧
COMMISSIONI( Numero: cm, Nome: nameC, Presidente: np) ∧
COLLEGI( Provincia: usl22 , Numero: usl23 , Nome: usl32) ∧
PROVINCE( Sigla: sigl ,Nome: usl1 ,Regione: co) ∧
REGIONI(Codice:co, Nome: RegName) ∧
( RegName = "Sicilia") }
```

Calcolo delle Tuple:

```
{ PRES.(Cognome, Nome) | PRES (DEPUTATI), D(Deputati), COM(COMMISSIONI),
COL (COLLEGI), P(PROVINCE), R(REGIONI) |
(PRES.Codice=COM.Presidente) ∧ (COM.Numero=D.Numero) ∧
(D.Provincia=P.Sigla) ∧ (P.Regione=R.Codice) ∧ (R.Nome = "Sicilia") }
```

2.

Algebra Relazionale:

$$\Pi_{\text{NomeC,Cognome}} ((\rho_{\text{NomeC} \leftarrow \text{Nome}}(\text{DEPUTATI})) \triangleright \triangleleft_{\text{Commissione=Numero}} (\sigma_{\text{Nome}=\text{"Bilancio"}}(\text{COMMISSIONE})))$$

Calcolo dei Domini:

{Nome: fn, Cognome: sur |
 DEPUTATI(Codice: n, Cognome: sur, Nome: fn,
 Commissione: cm, Provincia:co,Collegio: cst) ∧
 COMMISSIONE(Numero: cm, Nome: nomeC, Presidente: np) ∧
 (nomeC= "Bilancio") }

Calcolo delle tuple:

{D.(Nome,Cognome) | D(DEPUTATI), C(COMMISSIONI) |
 (D.Commissione=C.Numero) ∧ (C.Nome= "Bilancio") }

3.

Algebra Relazionale:

$$\Pi_{\text{NomeC,Cognome,nom1}} ((\rho_{\text{Nom1} \leftarrow \text{Nome}}(\text{PROVINCIA})) \triangleright \triangleleft_{\text{Sigla=Provincia}} ((\rho_{\text{Nome1} \leftarrow \text{Nome}}(\text{DEPUTATI})) \triangleright \triangleleft_{\text{Commissione=Numero}} (\sigma_{\text{Nome}=\text{"Bilancio"}}(\text{COMMISSIONE})))$$

Calcolo dei Domini:

{Nome: fn, Cognome: sur, Provincia: prv |
 DEPUTATI(Codice: n, Cognome: sur, Nome: fn,
 Commissione: cm, Provincia:co,Collegio: cst) ∧
 PROVINCE(Sigla: usl1 ,Nome: prv ,Regione: co) ∧
 COMMISSIONE(Numero: cm, Nome: nameC, Presidente: usl2) ∧
 (nameC= "Bilancio") }

Calcolo delle tuple:

{D.(Nome,Cognome), P.(Nome), | D(DEPUTATI),
 C(COMMISSIONI), P(PROVINCE) |
 (D.Commissione=C.Numero) ∧ (C.Nome= "Bilancio") ∧
 (D.Provincia= P.Sigla) }

4.

Algebra Relazionale:

$$\Pi_{\text{NomeC,Cognome,nom1,rega}} ((\rho_{\text{Rege} \leftarrow \text{Nome}}(\text{REGIONE})) \triangleright \triangleleft_{\text{Codice=Regione}} (\rho_{\text{Nom1} \leftarrow \text{Nome}}(\text{PROVINCIA})) \triangleright \triangleleft_{\text{Sigla=Provincia}} ((\rho_{\text{Nome1} \leftarrow \text{Nome}}(\text{DEPUTATI})) \triangleright \triangleleft_{\text{Commissione=Numero}} (\sigma_{\text{Nome}=\text{"Bilancio"}}(\text{COMMISSIONE})))$$

Calcolo dei Domini:

{ Nome: fn, Cognome: sur, Provincia: prv, Regione: rege |
 DEPUTATI(Codice: n, Cognome: sur, Nome: fn,
 Commissione: cm, Provincia:co,Collegio: cst) ∧
 PROVINCE(Sigla: usl1 ,Nome: prv ,Regione: co)∧
 REGIONI(Codice:co, Nome: rege)∧
 COMMISSIONE(Numero: cm, Nome: nameC, Presidente: usl2) ∧
 (nameC= “Bilancio”) }

Calcolo delle tuple:

{ D.(Nome,Cognome), P.(Nome), R.(Nome) | D(DEPUTATI),
 C(COMMISSIONI), R(REGIONI), P(PROVINCE) |
 (D.Commissione=C.Numero) ∧(C.Nome= “Bilancio”) ∧
 (D.Provincia= P.Sigla)∧(P.Regione= R.Codice) }

5. Algebra Relazionale:

$\Pi_{\text{RegioneC, Nome, Cognome}}$
 $(\text{DEPUTATI} \triangleright \triangleleft_{\text{Provincia=ProvinciaC} \wedge \text{Collegio=NumeroC}}$
 $(\text{REGIONI} \triangleright \triangleleft_{\text{Codice=RegioneC}}$
 $((\rho_{\text{SiglaC, NomeC, RegioneC, NumeroC, Nome1C} \leftarrow \text{Sigla, Nome, Regione, Numero, Nome1}$
 $(\text{PROVINCE} \triangleright \triangleleft_{\text{Sigla=Provincia}} (\rho_{\text{Nome1} \leftarrow \text{Nome}} \text{COLLEGI}))))$
 $-$
 $\Pi_{\text{SiglaC, NomeC, RegioneC, NumeroC, Nome1C}}$
 $((\text{PROVINCE} \triangleright \triangleleft_{\text{Sigla=Provincia}} (\rho_{\text{Nome1} \leftarrow \text{Nome}} \text{COLLEGI}))$
 $\triangleright \triangleleft_{\text{Regione=RegioneC}}$
 $(\rho_{\text{SiglaC, NomeC, RegioneC, NumeroC, Nome1C} \leftarrow \text{Sigla, Nome, Regione, Numero, Nome1}$
 $(\text{PROVINCE} \triangleright \triangleleft_{\text{Sigla=Provincia}} (\rho_{\text{Nome1} \leftarrow \text{Nome}} \text{COLLEGI}))))))$

Calcolo dei Domini:

{ Nome: fn, Cognome: sur, Regione: rege |
 DEPUTATI(Codice: n, Cognome: sur, Nome: fn,
 Commissione: cm, Provincia:prov,Collegio: coll) ∧
 PROVINCE(Sigla: prov,Nome: prv ,Regione: co)∧
 REGIONI(Codice:co, Nome: rege)∧
 PROVINCE(Sigla: n1 ,Nome: nom13 ,Regione: co)∧
 COLLEGI(Provincia: n1 ,Numero: coll,Nome: usl31)∧
 ¬(∃ n1(∃ co PROVINCE(Sigla: n1 ,Nome: nom1 ,Regione: co)∧
 COLLEGI(Provincia: n1 ,Numero: usl11 ,Nome: usl21)∧
 PROVINCE(Sigla: n21 ,Nome: nom2 ,Regione: co)∧
 COLLEGI(Provincia: n2 ,Numero: usl12 ,Nome: usl22) ∧usl11 <> usl12}

Calcolo delle tuple:

$$\{ D.(Nome, Cognome), R.(Nome) \mid D(DEPUTATI), \\ C(COLLEGI), R(REGIONI), C1(COLLEGI), \\ P1(PROVINCE), C2(COLLEGI), P2(PROVINCE) \mid \\ (D.Collegio = C.Numero) \\ \neg (\exists P1(PROVINCE) ((P2.Sigla = C2.Provincia) \wedge (P1.Sigla = C1.Provincia) \wedge \\ (P1.Regione = P2.Regione) \wedge (C1.nome \neq C2.nome))) \}$$

6. Algebra Relazionale:

$$\begin{aligned} & (\Pi_{NomeColl1} \\ & (\sigma_{NomeD1=NomeD2} \\ & (\rho_{NomeD2, NomeColl2, Regione2 \leftarrow \cdot, NomeD, Nome, Regione} \\ & (\Pi_{Regione, NomeD, Provincia, CollegioNome} (\rho_{NomeD, ProvinciaD \leftarrow Nome, Provincia} DEPUTATI) \\ & \triangleright \triangleleft_{(Provincia=ProvinciaD \wedge Collegio=Numero)} \\ & (COLLEGI \triangleright \triangleleft_{Provincia=Sigla} (\rho_{NomeP \leftarrow Nome} PROVINCE)))))) \\ & \triangleright \triangleleft_{Regione2=Regione1} \\ & (\rho_{NomeD1, NomeColl1, Regione1 \leftarrow \cdot, NomeD, Nome, Regione} \\ & (\Pi_{Regione, NomeD, Provincia, CollegioNome} (\rho_{NomeD, ProvinciaD \leftarrow Nome, Provincia} DEPUTATI) \\ & \triangleright \triangleleft_{(Provincia=ProvinciaD \wedge Collegio=Numero)} \\ & (COLLEGI \triangleright \triangleleft_{Provincia=Sigla} (\rho_{NomeP \leftarrow Nome} PROVINCE)))))) \end{aligned}$$

Calcolo dei Domini:

$$\{ Nome: nomecoll \mid \\ DEPUTATI(Codice: n1, Cognome: sur1, Nome: fn, \\ Commissione: cm1, Provincia: n1, Collegio: coll1) \wedge \\ PROVINCE(Sigla: n1, Nome: prv1, Regione: rege) \wedge \\ COLLEGI(Provincia: n1, Numero: coll1, Nome: nomecoll1) \wedge \\ DEPUTATI(Codice: n1, Cognome: sur2, Nome: fn, \\ Commissione: cm2, Provincia: n2, Collegio: coll2) \wedge \\ PROVINCE(Sigla: n2, Nome: prv2, Regione: rege) \wedge \\ COLLEGI(Provincia: n2, Numero: coll2, Nome: Nomecoll2) \}$$

Calcolo delle tuple:

$$\{ C1.(Nome) \mid D1(DEPUTATI), P1(PROVINCE), C1(COLLEGI), \\ D2(DEPUTATI), P2(PROVINCE), C2(COLLEGI) \mid \\ (C1.Provincia = P1.Sigla) \wedge (C2.Provincia = P2.Sigla) \wedge \\ (D1.Provincia = C1.Provincia) \wedge (D1.Collegio = C1.Numero) \wedge \\ (D2.Provincia = C2.Provincia) \wedge (D2.Collegio = C2.Numero) \wedge \\ (P2.Regione = P1.Regione) \wedge (D1.Nome = D2.Nome) \}$$

Esercizio 3.9

Mostrare come le interrogazioni nell'esercizio 3.3 possano trarre vantaggio, nella specifica, dalla definizione di viste.

Soluzione:

I punti 2, 3 e 4 dell'esercizio 3.3 sono molto simili. Per rendere più veloci le interrogazioni è utile definire una vista con le parti in comune, inserendo in una tabella COMBILANCIO tutte le informazioni necessarie:

```

COMBILANCIO = (
  (ρRege←Nome( REGIONE))▷◁Codice=Regione
  (ρNom1←Nome( PROVINCIA))▷◁Sigla=Provincia
  ((ρNome1←Nome(DEPUTATI))▷◁Commissione=Numero
  (σNome="Bilancio"( COMMISSIONE)))
  
```

A questo punto le interrogazioni dell'esercizio 3.3 si risolvono più semplicemente:

- 3.2 $\Pi_{NomeC,Cognome} (COMBILANCIO)$
- 3.3 $\Pi_{NomeC,Cognome,nom1} (COMBILANCIO)$
- 3.4 $\Pi_{NomeC,Cognome,nom1,rege} (COMBILANCIO)$

Esercizio 3.10

Si consideri lo schema di base di dati sulle relazioni:

MATERIE (Codice, Facoltà, Denominazione, Professore)

STUDENTI (Matricola, Cognome, Nome, Facoltà)

PROFESSORI (Matricola, Cognome, Nome)

ESAMI (Studente, Materia, Voto, Data)

PIANIDISTUDIO (Studente, Materia, Anno)

Formulare, in algebra relazionale, in calcolo su domini, in calcolo su tuple e in Datalog le interrogazioni che producono:

1. gli studenti che hanno riportato in almeno un esame una votazione pari a 30, mostrando, per ciascuno di essi, nome e cognome e data della prima di tali occasioni;
2. per ogni insegnamento della facoltà di ingegneria, gli studenti che hanno superato l'esame nell'ultima seduta svolta;
3. gli studenti che hanno superato tutti gli esami previsti dal rispettivo piano di studio;
4. per ogni insegnamento della facoltà di lettere, lo studente (o gli studenti) che hanno superato l'esame con il voto più alto;
5. gli studenti che hanno in piano di studio solo gli insegnamenti della propria facoltà;
6. nome e cognome degli studenti che hanno sostenuto almeno un esame con un professore che ha il loro stesso nome proprio.

Soluzione:

1)

Algebra Relazionale:

$$\begin{aligned} & \Pi_{\text{Nome, Cognome, Data}} \\ & ((\Pi_{\text{Studente, Data}} (\sigma_{\text{Voto} = '30'}(\text{ESAMI})) - \\ & \Pi_{\text{Studente, Data}} ((\sigma_{\text{Voto} = '30'}(\text{ESAMI}) \triangleright \triangleleft_{(\text{Studente}=\text{Studente1}) \wedge (\text{Data} > \text{Data1})} \\ & \rho_{\text{Studente1, Corso1, Voto1, Data1} \leftarrow \text{Studente, Corso, Voto, Data}} \sigma_{\text{Voto} = '30'}(\text{ESAMI}))) \\ & \triangleright \triangleleft_{\text{Studente}=\text{Matricola}}(\text{STUDENTE})) \end{aligned}$$

Calcolo dei Domini:

$$\begin{aligned} & \{ \text{Nome: fn, Cognome: sur, Data: d} \mid \\ & \text{STUDENTE}(\text{Matricola: n, Nome: fn, Cognome: sur, Facoltà: f}) \wedge \\ & \text{ESAMI}(\text{studente: n, Corso: c, Voto: g, Data: d}) \wedge (g = '30') \wedge \\ & \neg(\exists c1 (\exists d1 (\text{ESAMI}(\text{Studente: n, Corso: c1, Voto: g, Data: d1}) \wedge (d1 < d)))) \} \end{aligned}$$

Calcolo delle Tuple:

$$\begin{aligned} & \{ S.(\text{Nome, Cognome}), E.(\text{Data}) \mid S(\text{STUDENTE}), E(\text{ESAMI}) \mid \\ & (S.\text{matricola} = E.\text{Studente}) \wedge (E.\text{voto} = '30') \wedge \\ & \neg(\exists E1(\text{ESAMI}) ((E1.\text{Studente} = S.\text{matricola}) \wedge (E1.\text{voto} = '30') \wedge (E1.\text{Data} < E.\text{Data}))) \} \end{aligned}$$

Datalog:

```
OTHERA (Studente: n, Data: d) ←
    ESAMI(Studente: n, Corso: c, Voto: '30', Data: d ),
    ESAMI(Studente: n, Corso: c1, Voto: '30', Data: d1 ),
    (d > d1)

FIRSTA (Nome: fn, Cognome: sur, Data: d) ←
    STUDENTE(Matricola: n, Nome: fn, Cognome: sur, Facoltà: f),
    ESAMI(Studente: n, Corso: c, Voto: '30', Data: d ),
    not OTHERA(Studente: n, Data: d)
```

2)

Algebra Relazionale:

$$\sigma_{(Voto \geq 18) \wedge (Facoltà = 'Ingegneria')}$$

$$(\Pi_{Cognome, Nome, Materia, Voto, Data}$$

$$(STUDENTI \bowtie_{Studente = Matricola} (\Pi_{Studente, Materia, Voto, Data} (\sigma_{(D \geq Data)}$$

$$(ESAMI \bowtie_{Materia = M} (\rho_{S, M, V, D \leftarrow Studente, Materia, Voto, Data} (ESAMI))))))$$

Calcolo Domini:

```
{ Nome: fn, Cognome: sur, Materia: c, Voto: g, Data: d |
  STUDENTE(Matricola: sn, Nome: fn, Cognome: sur, Facoltà: f) ∧
  ESAMI(studente: sn, Materia: c, Voto: g, Data: d ) ∧
  ESAMI(studente: sn, Materia: c, Voto: v1, Data: d1) ∧
  (f='Ingegneria') ∧ (d ≥ d1 ) }
```

Calcolo delle Tuple:

```
{ S.(Nome, Cognome, Materia, Voto, Data) |
  S (STUDENTI), E(ESAMI), E1(ESAMI) |
  (E.Materia=E1.Materia) ∧ (E.Data > E1.Data) ∧ (E.Studente=S.Matricola) ∧
  (Facoltà="Ingegneria") ∧ (E.Voto ≥ 18) }
```

Datalog:

```
BRAVISTUDENTI(Nome: fn, Cognome: sur, Materia: c, Voto: g , Data: d) ←
    STUDENTI(Matricola: sn, Nome: fn, Cognome: sur, Facoltà: sf),
    ESAMI(Studente: sn, Materia: c, Voto: g, Data: d ),
    ESAMI(Studente: sn, Materia: c, Voto: g1, Data: d1 ),
    (d ≥ d1), (sf = 'Ingegneria'), (g ≥ 18)
```

3)

Algebra Relazionale:

$$\Pi_{\text{Studente}}(\text{PIANIDISTUDIO}) - (\Pi_{\text{Studente}}(\Pi_{\text{Studente,Materia}}(\text{PIANIDISTUDIO}) - \Pi_{\text{Studente,Materia}}(\text{ESAMI})))$$

Calcolo dei Domini:

$$\{ \text{Studente: n} \mid \text{PIANIDISTUDIO}(\text{Studente:n, Materia:c, Anno:y}) \wedge (\forall c1(\forall y1(\exists g1, \exists d1(\neg(\text{PIANIDISTUDIO}(\text{Studente: n, Materia: c1, Anno: y1}) \vee (\text{ESAMI}(\text{Studente: n, Materia: c1, Voto: g1, Data: d1})))))) \}$$

Calcolo delle Tuple:

$$\{ S.\text{Studente} \mid P(\text{PIANIDISTUDIO}) \mid (\forall P(\text{PIANIDISTUDIO}) (\exists E(\text{ESAMI}) (\neg(S.\text{Studente}=P.\text{Studente}) \vee ((P.\text{Studente}=E.\text{Studente}) \wedge (S1.\text{Materia}=E.\text{Materia})))) \}$$

Datalog:

$$\text{ESAMISTUDENTE}(\text{Studente: n, Materia: c}) \leftarrow \text{ESAMI}(\text{Studente: n, Materia: c, Voto: g, Data: d})$$

$$\text{ESAMIINCOMPLETISTUDENTE}(\text{Studente: n}) \leftarrow \text{PIANIDISTUDIO}(\text{Studente:n, Materia:c, Anno:y}), \text{NOT ESAMISTUDENTE}(\text{Studente: n, Materia:c})$$

$$\text{TUTTIESAMI}(\text{Studente: n}) \leftarrow \text{PIANIDISTUDIO}(\text{Studente:n, Materia:c, Anno:y}) \text{ NOT ESAMIINCOMPLETISTUDENTE}(\text{Studente: n})$$

4)

Algebra Relazionale :

$$\Pi_{\text{Studente,Materia}}(\text{ESAMI} \bowtie_{\text{Materia=Codice}} \sigma_{\text{Facoltà="Lettere"}}(\text{MATERIE})) - \Pi_{\text{Studente,Materia}} \sigma_{(\text{Voto} < \text{Voto1}) \wedge (\text{Facoltà} = \text{"Lettere"})}(\text{MATERIE} \bowtie_{\text{Codicer=Materia}} \text{ESAMI} \bowtie_{\text{Materia=Materia1}} \rho_{\text{Studente1, Materia1, Voto1, Data1} \leftarrow \text{Studente, Materia, Voto, Data}}(\text{ESAMI}))$$

Calcolo dei Domini:

$$\{ \text{Studente: sn, Materia: c} \mid \text{ESAMI}(\text{Studente: sn, Materia: c, Voto: g, Data: d}) \wedge \\ \text{MATERIE}(\text{Codice: c, Facoltà: f, Denominazione: ct, Professore: t}) \wedge (f \neq \text{"Lettere"}) \wedge \\ \neg(\exists \text{sn1}(\exists \text{d1}(\exists \text{g1}(\text{ESAMI}(\text{Studente: sn1, Materia: c, Date: d1, Voto: g1}) \wedge (\text{g1} > \text{g})))) \}$$

Calcolo delle Tuple:

$$\{ \text{E.Studente, E.Materia} \mid \text{E}(\text{ESAMI}), \text{M}(\text{MATERIE}) \mid \\ (\text{E.Materia} = \text{M.Codice}) \wedge (\text{M.Facoltà} = \text{"Lettere"}) \wedge \\ ((\exists \text{E1}(\text{ESAMI}) \neg (\text{E1.Materia} = \text{E.Materia}) \wedge (\text{E1.Voto} > \text{E.Voto})) \}$$

Datalog:

STUDENTINONMIGLIORI(Studente: sn, Materia: c) ←
 ESAMI(Studente: sn, Materia: c, Voto: g, Data: d),
 MATERIE(Number: c, Facoltà: "Lettere", Denominazione: ct, Professore: t),
 ESAMI(Studente: sn1, Materia: c, Voto: g1, Data: d1),
 (g < g1)

STUDENTIMIGLIORI(Studente: sn, Materia: c) ←
 ESAMI(Studente: sn, Materia: c, Voto: g, Data: d),
 MATERIE(Codice: c, Facoltà: "Lettere", Denominazione: ct, Professore: t),
 NOT STUDENTINONMIGLIORI(Studente: sn, Materia: c)

5)

Algebra Relazionale:

$$\Pi_{\text{Studente}}(\text{PIANIDISTUDIO}) - \\ \Pi_{\text{Studente}}(\sigma_{\text{SFacoltà} \neq \text{Facoltà}}(\rho_{\text{SMatricola, SFacoltà} \leftarrow \text{Matricolar, Facoltà}}(\text{STUDENTI}) \\ \triangleright \triangleleft_{\text{SMatricola} = \text{Studente}} \text{PIANIDISTUDIO} \\ \triangleright \triangleleft_{\text{Materia} = \text{Codice}} \text{MATERIE}))$$

Calcolo dei Domini:

$$\{ \text{Studente: n} \mid \text{PIANIDISTUDIO}(\text{Studente: n, Materia: c, Anno: y}) \wedge \\ \neg(\exists \text{fn}(\exists \text{sur}, \exists \text{sf}(\exists \text{f}, (\exists \text{ct}, \exists \text{t}(\text{STUDENTI}(\text{Matricola: n, Nome: fn, Cognome: sur, Facoltà: sf}) \wedge \\ \text{MATERIE}(\text{Codice: c, Facoltà: f, Denominazione: ct, Professore: t}) \wedge (\text{sf} \neq \text{f})))))) \}$$

Calcolo delle Tuple:

$$\{ \text{SP.Studente} \mid \text{SP}(\text{PIANIDISTUDIO}) \mid \neg(\exists \text{S}(\text{STUDENTI}))(\exists \text{C}(\text{MATERIE})) \\ ((\text{SP.Materia} = \text{C.Codice}) \wedge (\text{S.Matricola} = \text{SP.Studente}) \wedge (\text{C.Facoltà} \neq \text{S.Facoltà})) \}$$

Datalog:

DIFFERENTEFACOLTA(Studente: n) ←
 PIANIDISTUDIO(Studente: n, Materia: c, Anno: y),

STUDENTS(Matricola: n, Nome: fn, Cognome: sur, Facoltà: sf) ,
 MATERIE (Codice: c, Facoltà: f, Denominazione: ct, Professore: t),
 (sf≠ f)

STESSAFACOLTA(Studiante: n) ←
 PIANIDISTUDIO (Studiante:n, Materia :c , Anno: y),
 NOT DIFFERENTEFACOLTA(Studiante: n)

6)

Algebra Relazionale:

$$\Pi_{\text{Nome,Cognome}} (\sigma_{\text{Nome}=\text{NomeP}} (\\ (\text{ESAMI} \bowtie_{\text{Studiante}=\text{Matricola}} \text{STUDENTI}) \bowtie_{\text{Materia}=\text{Codice}} \\ (\rho_{\text{NomeP,CognomeP} \leftarrow \text{Nome,Cognome}} (\text{PROFESSORI} \bowtie_{\text{Matricola}=\text{Professore}} \text{MATERIE}))))$$

Calcolo dei Domini:

{ Nome: fn, Cognome: sur |
 STUDENTI(Matricola: n, Nome: fn, Cognome: sur, Facoltà: sf) ∧
 ESAMI(Studiante: n, Materia: c, Voto: g, Data: d) ∧
 MATERIE(Codice: c, Facoltà: f, Denominazione: ct, Professore: t) ∧
 PROFESSORI(Matricola: t, Nome:fn, Cognome: tfn) }

Calcolo delle Tuple:

{ S.(Nome,Cognome) |
 S(STUDENTI), E (ESAMI), M(MATERIE), P(PROFESSORI) |
 (S.Matricola=E.Studiante) ∧ (E.Materia=M.Codice) ∧ (M.Professore=P.Matricola) ∧
 (P.Nome=S.Nome) }

Datalog:

STESSONOME (Nome: fn, Cognome: sur) ←
 STUDENTI(Matricola: n, Nome: fn, Cognome: sur, Facoltà: sf) ,
 ESAMI(Studiante: n, Materia: c, Voto: g, Data: d) ,
 MATERIE(Codice: c, Facoltà: f, Denominazione: ct, Professore: t),
 PROFESSORI(Matricola: t, Cognome:surf, Nome: fn)

Esercizio 3.11

Con riferimento al seguente schema di base di dati:

CITTÀ (Nome, Regione, Abitanti)

ATTRAVERSAMENTI (Città, Fiume)

FIUMI (Fiume, Lunghezza)

formulare, in algebra relazionale, in calcolo sui domini, in calcolo su tuple e in Datalog le seguenti interrogazioni:

1. Visualizza nome, regione e abitanti per le città che hanno più di 50000 abitanti e sono attraversate dal Po oppure dall'Adige;
2. Trovare le città che sono attraversate da (almeno) due fiumi, visualizzando il nome della città e quello del più lungo di tali fiumi.

Soluzione:

1) Algebra Relazionale:

$$\Pi_{\text{Nome, Regione, Abitanti}} (\sigma_{(\text{Fiume}=\text{"Po"}) \vee (\text{Fiume}=\text{"Adige"})} (\text{ATTRAVERSAMENTO})) \triangleright \triangleleft_{\text{Città}=\text{Nome}} \sigma_{\text{Abitanti} > 50000} (\text{CITTÀ})$$

Calcolo dei Domini:

$$\{ \text{Nome: n, Regione: reg, Abitanti: p} \mid \text{CITTÀ (Name: n, Regione: reg, Abitanti: p)} \wedge \text{ATTRAVERSAMENTO (Città: n, Fiume: r)} \wedge (p > 50000) \wedge ((r = \text{"Po"}) \vee (r = \text{"Adige"})) \}$$

Calcolo delle Tuple:

$$\{ C.(\text{Nome, Regione, Abitanti}) \mid C(\text{CITTÀ}), A(\text{ATTRAVERSAMENTO}) \mid (C.\text{Nome} = A.\text{Città}) \wedge (C.\text{Abitanti} > 50000) \wedge ((A.\text{Fiume} = \text{"Po"}) \vee (A.\text{Fiume} = \text{"Adige"})) \}$$

Datalog:

$$\text{POEADIGE (Città: c)} \leftarrow \text{ATTRAVERSAMENTO (Città: c, Fiume: r),} \\ (r = \text{"Po"})$$

2) Algebra Relazionale:

$$\Pi_{\text{Città, Fiume}} (\sigma_{\text{Fiume} \neq \text{Fiume1}} (\text{ATTRAVERSAMENTO})) \triangleright \triangleleft_{\text{Città}=\text{Città1}} \rho_{\text{Città1, Fiume1} \leftarrow \text{Città, Fiume}} (\text{ATTRAVERSAMENTO})) -$$

$$\Pi_{\text{Città, Fiume}} (\sigma_{(\text{Fiume} \neq \text{Fiume1}) \wedge (\text{Lughezza} < \text{Lughezza1})} ((\text{FIUMI}) \triangleright \triangleleft \text{ATTRAVERSAMENTO}))$$

$$\triangleright \triangleleft_{\text{Città}=\text{Città1}}$$

$\rho_{\text{Città1, Fiume1, Lunghezza1} \leftarrow \text{Città, Fiume, Lunghezza} (\text{FIUMI} \triangleright \triangleleft \text{ATTRaversamento}))}$

Calcolo dei Domini:

$\{ \text{Città: } n, \text{ Fiume: } r \mid \text{ATTRaversamento}(\text{Città: } n, \text{ Fiume: } r) \wedge$
 $\text{ATTRaversamento}(\text{Città: } n, \text{ Fiume: } r1) \wedge (r \neq r1) \wedge$
 $\text{FIUMI}(\text{Fiume: } r, \text{ Lunghezza: } l) \wedge$
 $(\forall r2 (\forall l2 \neg ((\text{ATTRaversamento}(\text{Città: } n, \text{ Fiume: } r2) \wedge$
 $\text{FIUMI}(\text{Fiume: } r2, \text{ Lunghezza: } l2)) \vee (l2 < l))) \}$

Calcolo delle Tuple:

$\{ C.(\text{Città} , \text{ Fiume}) \mid$
 $C(\text{ATTRaversamento}), C1(\text{ATTRaversamento}), F(\text{FIUMI}) \mid$
 $(C. \text{Città} = C1. \text{Città}) \wedge (C. \text{Fiume} \neq C1. \text{Fiume}) \wedge (C. \text{Fiume} = F. \text{Fiume}) \wedge$
 $(\forall C2(\text{ATTRaversamento}) (\forall F2(\text{FIUMI}) (C2. \text{Città} \neq C. \text{Città})$
 $\wedge (F2. \text{Fiume} = F. \text{Fiume}) \wedge (F2. \text{Lunghezza} < F. \text{Lunghezza})))) \}$

Datalog:

$\text{FIUMECORTO} (\text{Città} : c, \text{ Fiume} : r) \leftarrow$
 $\text{ATTRaversamento} (\text{Città} : c, \text{ Fiume} : r),$
 $\text{FIUMI} (\text{Fiume} : r, \text{ Lunghezza} : l),$
 $\text{ATTRaversamento} (\text{Città} : c, \text{ Fiume} : r1),$
 $\text{FIUMI} (\text{Fiume} : r1, \text{ Lunghezza} : l1),$
 $(l < l1)$

$\text{FIUMELUNGO} (\text{Città} : c, \text{ Fiume} : r) \leftarrow$
 $\text{ATTRaversamento} (\text{Città} : c, \text{ Fiume} : r),$
 $\text{ATTRaversamento} (\text{Città} : c, \text{ Fiume} : r1),$
 $(r \neq r1),$
 $\text{NOT FIUMECORTO} (\text{Città} : c, \text{ Fiume} : r)$

Esercizio 3.12

Con riferimento al seguente schema di base di dati:

AFFLUENZA (Affluente, Fiume)

FIUMI (Fiume, Lunghezza)

formulare l'interrogazione in Datalog che trova tutti gli affluenti, diretti e indiretti dell'Adige.

Soluzione:

TUTTIAFFLUENTI (Affluenti: t) ←
AFFLUENZA (Affluenti : t, Fiume: “Adige”)

TUTTIAFFLUENTI (Affluenti : t) ←
AFFLUENZA (Affluenti : t, Fiume:r),
TUTTIAFFLUENTI (Affluenti : r)

Esercizio 3.13

Si consideri lo schema relazionale composto dalle seguenti relazioni:

PROFESSORI (Codice, Cognome, Nome)

CORSI (Codice, Denominazione, Professore)

STUDENTI (Matricola, Cognome, Nome)

ESAMI (Studente, Corso, Data, Voto)

Formulare, con riferimento a tale schema, le espressioni dell'algebra, del calcolo relazionale su tuple e del Datalog, che producano:

1. Gli esami superati dallo studente Pico della Mirandola (supposto unico), con indicazione, per ciascuno, della denominazione del corso, del voto e del cognome del professore;
2. i professori che tengono due corsi (e non più di due), con indicazione di cognome e nome del professore e denominazione dei due corsi.

Soluzione:

1) Algebra Relazionale:

$$\Pi_{\text{Denominazione, Voto, CognomeP, NomeP}} \left(\sigma_{\text{Cognome} = \text{"della Mirandola"} \wedge \text{Nome} = \text{"Pico"}} \right. \\
\left. (\text{STUDENTI} \triangleright \triangleleft_{\text{Matricola} = \text{Studente}} \text{ESAMI} \triangleright \triangleleft_{\text{Corso} = \text{CCodice}} \rho_{\text{CCodice} \leftarrow \text{Codice}} (\text{CORSI}) \right. \\
\left. \triangleright \triangleleft_{\text{Professore} = \text{TCodice}} \rho_{\text{TCodice, T.CognomeP, NomeP} \leftarrow \text{Codice, Cognome, Nome}} (\text{PROFESSORI})) \right)$$

Calcolo delle Tuple:

$$\{ \text{C.Denominazione, E.Voto, P.}(\text{Cognome, Nome}) \mid \text{C}(\text{CORSI}), \\
\text{E}(\text{ESAMI}), \text{P}(\text{PROFESSORI}), \text{S}(\text{STUDENTI}) \mid \\
(\text{S.Matricola} = \text{E.Studente}) \wedge (\text{E.Corso} = \text{C.Codice}) \wedge (\text{C.Professore} = \text{P.Codice}) \wedge \\
(\text{S.Cognome} = \text{"Della Mirandola"}) \wedge (\text{S.Nome} = \text{"Pico"}) \}$$

Datalog:

ESAMIPICO(Denominazione: cn, Voto: g, Cognome: tsur, Nome: tname) ←
 STUDENTI(Matricola: n, Cognome: "Della Mirandola", Nome: "Pico"),
 ESAMI (Studente: n, Corso: c, Voto: g, Data: d),
 CORSI (Codice: c, Denominazione:cn, Professore: t),
 PROFESSORI (Codice: t, Cognome: tsur, Nome: tname)

2) Algebra Relazionale:

$$\Pi_{\text{Cognome, Nome, Denominazione, Denominazione1}} \left(\Pi_{\text{Professore, Denominazione, Denominazione1}} (\text{CORSI} \triangleright \triangleleft_{(\text{Professore} = \text{Professore1}) \wedge (\text{Codice} \neq \text{Codice1})} \right. \\
\rho_{\text{Codice1, Denominazione1, Professore1} \leftarrow \text{Codice, Denominazione, Professore}} (\text{CORSI}) \left. \right) - \\
\Pi_{\text{Professore, Denominazione, Denominazione1}} \left(\sigma_{\text{Codice} \neq \text{Codice2}} (\text{CORSI} \triangleright \triangleleft_{(\text{Professore} = \text{Professore1}) \wedge (\text{Codice} \neq \text{Codice1})} \right)$$

$$\begin{aligned} & \rho_{\text{Codice1, Denominazione1, Professore1} \leftarrow \text{Codice, Denominazione, Professore}} (\text{CORSI}) \\ & \triangleright \triangleleft (\text{Professore1} = \text{Professore2}) \wedge (\text{Codice1} \neq \text{Codice2}) \wedge (\text{Codice1} \neq \text{Codice2}) \\ & \rho_{\text{Codice2, Denominazione2, Professore2} \leftarrow \text{Codice, Denominazione, Professore}} (\text{CORSI})) \\ & \triangleright \triangleleft_{\text{Professore} = \text{PCodice}} \rho_{\text{PCodice} \leftarrow \text{Codice}} (\text{PROFESSORI}) \end{aligned}$$

Calcolo delle tuple:

$$\begin{aligned} & \{ P.(\text{Nome}, \text{Cognome}), C.(\text{Denominazione}, C1. \text{Denominazione}) \mid \\ & P(\text{PROFESSORI}), C(\text{CORSI}), C1(\text{CORSI}) \mid \\ & (P. \text{Codice}) = (C. \text{Professore}) \wedge (P. \text{Codice}) = (C1. \text{Professore}) \wedge \\ & (C. \text{Codice} \neq C1. \text{Codice}) \wedge \\ & \neg (\exists C2(\text{CORSI}) ((C2. \text{Professore} = P. \text{Codice}) \wedge \\ & (C2. \text{Codice} \neq C. \text{Codice}) \wedge (C2. \text{Codice} \neq C1. \text{Codice}) \wedge (C. \text{Codice} \neq C1. \text{Codice})) \} \end{aligned}$$

Datalog:

$$\begin{aligned} & \text{PIUDIDUE}(\text{Professore} : t) \leftarrow \\ & \quad \text{CORSI}(\text{Codice} : n, \text{Denominazione} : cn, \text{Professore} : t), \\ & \quad \text{CORSI}(\text{Codice} : n1, \text{Denominazione} : cn1, \text{Professore} : t), \\ & \quad \text{CORSI}(\text{Codice} : n1, \text{Denominazione} : cn1, \text{Professore} : t), \\ & \quad (n \neq n1), (n \neq n2), (n1 \neq n2) \\ & \text{ESATTAMENTEDUE}(\text{Nome} : fn, \text{Cognome} : sn, \text{Corso1} : cn1, \text{Corso2} : cn2) \leftarrow \\ & \quad \text{PROFESSORI}(\text{Codice} : t, \text{Nome} : fn, \text{Cognome} : sn), \\ & \quad \text{CORSI}(\text{Codice} : n1, \text{Denominazione} : cn1, \text{Professore} : t), \\ & \quad \text{CORSI}(\text{Codice} : n2, \text{Denominazione} : cn2, \text{Professore} : t), \\ & \quad (n1 \neq n2), \\ & \quad \text{NOT PIUDIDUE}(\text{Professore} : t) \end{aligned}$$

Esercizio 3.14

Considerare uno schema relazionale contenente le relazioni

$R_1(ABC)$, $R_2(DG)$, $R_3(EF)$

Formulare in calcolo relazionale su tuple e su domini l'interrogazione realizzata in algebra relazionale dalla seguente espressione:

$$(R_3 \bowtie_{G=E} R_2) \cup (\rho_{DG \leftarrow AC} (\Pi_{ACEF} (R_1 \bowtie_{B=F} R_3)))$$

Soluzione:

Questa espressione non è esprimibile in calcolo sulle tuple a causa dell'unione tra due diverse tabelle. In calcolo sui domini l'espressione diventa:

$$\{ D: d, G: g, E: e, F: f \mid R_3(E:e, F:f) \wedge (R_2(D: d, G: g) \wedge (g=e)) \vee (R_1(A: d, B: b, C: g) \wedge (b=f)) \}$$

Esercizio 3.15

Con riferimento allo schema dell'esercizio 3.9, formulare in algebra relazionale le interrogazioni realizzate in calcolo su domini dalle seguenti espressioni:

$$\begin{aligned}
 & \{ H: g, B: b \mid R_1(A: a, B: b, C: c) \wedge R_2(D: c, G: g) \} \\
 & \{ A: a, B: b \mid R_2(D: a, G: b) \wedge R_3(E: a, F: b) \} \\
 & \{ A: a, B: b \mid R_1(A: a, B: b, C: c) \wedge \\
 & \quad \exists a' (R_1(A: a', B: b, C: c) \wedge a \neq a') \} \\
 & \{ A: a, B: b \mid R_1(A: a, B: b, C: c) \wedge \\
 & \quad \forall a' (\neg R_1(A: a', B: b, C: c) \vee a = a') \} \\
 & \{ A: a, B: b \mid R_1(A: a, B: b, C: c) \wedge \\
 & \quad \neg \exists a' (R_1(A: a', B: b, C: c) \wedge a \neq a') \}
 \end{aligned}$$

Soluzione:

$$\begin{aligned}
 & \rho_{H \leftarrow G} (\Pi_{BG} (R_1 \bowtie_{C=D} R_2)) \\
 & \rho_{AB \leftarrow DG} (\Pi_{DG} (R_2 \bowtie_{(D=E) \wedge (G=F)} R_3)) \\
 & \Pi_{AB} (\sigma_{A \neq A1} (R_1 \bowtie_{(B=B1) \wedge (C=C1)} \rho_{A1, B1, C1 \leftarrow ABC}(R_1))) \\
 & \Pi_{AB} (R_1) - \Pi_{AB} (\sigma_{A \neq A1} (R_1 \bowtie_{(B=B1) \wedge (C=C1)} \rho_{A1, B1, C1 \leftarrow ABC}(R_1))) \\
 & \Pi_{AB} (R_1) - \Pi_{AB} (\sigma_{A \neq A1} (R_1 \bowtie_{(B=B1) \wedge (C=C1)} \rho_{A1, B1, C1 \leftarrow ABC}(R_1)))
 \end{aligned}$$

Esercizio 3.16

Trasformare la seguente espressione dall'algebra, che fa riferimento allo schema

$$R_1(AB), R_2(CDE), R_3(FGH)$$

con l'obiettivo di ridurre le dimensioni dei risultati intermedi:

$$\Pi_{ADH}(\sigma_{(B=C) \wedge (E=F) \wedge (A>20) \wedge (G=10)}((R_1 \bowtie R_3) \bowtie R_2))$$

Soluzione:

$$\Pi_{ADH}(\sigma_{A>20}(R_1) \bowtie \Join_{B=C} \Pi_{CDH}(R_2 \bowtie \Join_{E=F} \Pi_{FH}(\sigma_{G=10}(R_3))))$$

Esercizio 3.17 Considerare la seguente base di dati relazionale:

- FARMACI (Codice, NomeFarmaco, PrincipioAttivo, Produttore, Prezzo)
- PRODUTTORI (CodProduttore, Nome, Nazione)
- SOSTANZE (ID, NomeSostanza, Categoria)

con vincoli di integrità referenziale fra **Produttore** e la relazione PRODUTTORI, fra **PrincipioAttivo** e la relazione SOSTANZE.

Formulare in algebra relazionale le seguenti interrogazioni:

1. l'interrogazione che fornisce, per i farmaci il cui principio attivo è nella categoria "sulfamidico," il nome del farmaco e quello del suo produttore;
2. l'interrogazione che fornisce, per i farmaci con produttore italiano, il nome del farmaco e quello della sostanza del suo principio attivo.

Soluzione:

1. $A = \text{FARMACI} \bowtie_{\text{PrincipioAttivo}=\text{ID}} (\sigma_{\text{Categoria}='sulfamidico'} \text{SOSTANZE});$

$\pi_{\text{NomeFarmaco}, \text{Nome}} (\text{PRODUTTORI} \bowtie_{\text{CodProduttore}=\text{Produttore}} A).$

2. $B = \text{FARMACI} \bowtie_{\text{Produttore}=\text{CodProduttore}} (\sigma_{\text{Nazione}='Italia'} \text{PRODUTTORI});$

$\pi_{\text{NomeFarmaco}, \text{NomeSostanza}} (\text{SOSTANZE} \bowtie_{\text{ID}=\text{PrincipioAttivo}} B).$

Esercizio 3.18 Mostrare, con riferimento alla base di dati dell'esercizio 2.10, il risultato finale e quelli intermedi della seguente espressione dell'algebra relazionale:

$\pi_{Cognome}(IMPIEG \bowtie_{Matr=Imp} (PARTECIP \bowtie_{Prog=Cod} \sigma_{Costo>65}(PROG)))$

Soluzione:

$\sigma_{Costo>65}(PROG)$

ID	Titolo	Costo
A	Luna	70
C	Giove	90

$PARTECIP \bowtie_{Prog=Cod} \sigma_{Costo>65}(PROG)$

Imp	Prog	ID	Titolo	Costo
101	A	A	Luna	70
103	A	A	Luna	70

$IMPIEG \bowtie_{Matr=Imp} (PARTECIP \bowtie_{Prog=Cod} \sigma_{Costo>65}(PROG))$

Matr	Cognome	Nome	Età	Imp	Prog	ID	Titolo	Costo
101	Rossi	Mario	35	101	A	A	Luna	70
103	Gialli	Mario	34	103	A	A	Luna	70

$\pi_{Cognome}(IMPIEG \bowtie_{Matr=Imp} (PARTECIP \bowtie_{Prog=Cod} \sigma_{Costo>65}(PROG)))$

Cognome
Rossi
Gialli

Esercizio 3.19 Considerare la seguente base di dati relazionale:

- CLIENTI (Codice, Nome, Indirizzo, Città)
- NOLEGGI (Cliente, Auto, DataPrelievo, DataRestituzione) con vincolo di integrità referenziale fra l'attributo Auto e la relazione AUTOVETTURE
- AUTOVETTURE (Targa, Modello, Colore, AnnoImmatricolazione, Costo-Giornaliero) con vincolo di integrità referenziale fra l'attributo Cliente e la relazione CLIENTI

formulare in algebra relazionale:

1. l'interrogazione che restituisce i dati dei clienti che hanno noleggiato almeno un'autovettura nell'anno 2006
2. l'interrogazione che restituisce i clienti che hanno noleggiato più di un'autovettura
3. l'interrogazione che restituisce i clienti che hanno noleggiato autovetture di un solo modello.

Soluzione:

1. $A = \sigma_{(DataPrelievo \geq '01/01/2006') \wedge (DataPrelievo \leq '31/12/2006')} NOLEGGI;$

$CLIENTI \bowtie_{Codice=Cliente} A$

2. $\pi_{Cliente} (\sigma_{Auto \neq Auto'} (NOLEGGI \bowtie_{Cliente=Cliente'} (\rho_{X' \leftarrow X} NOLEGGI)))$

3. $V = NOLEGGI \bowtie_{Auto=Targa} AUTOVETTURE$

$V_1 = V \bowtie_{Cliente=Cliente'} (\rho_{X' \leftarrow X} V)$

$\pi_{Cliente} NOLEGGI - \pi_{Cliente} (\sigma_{Modello \neq Modello'} (V_1))$

Esercizio 3.20 Considerando la seguente base di dati:

- FORNITORI (CF, Nome, Indirizzo, Città)
- PRODOTTI (CP, Nome, Marca, Modello)
- CATALOGO (CF, CP, Costo) con vincoli di integrità referenziale fra l'attributo CF e la relazione FORNITORI e fra l'attributo CP e la relazione PRODOTTI

formulare in algebra relazionale le seguenti interrogazioni:

1. trovare Nome, Marca e Modello dei prodotti acquistabili con meno di 2.000;
2. trovare i nomi dei fornitori che distribuiscono prodotti IBM (IBM è la marca di un prodotto);
3. trovare i codici di tutti i prodotti che sono forniti da almeno due fornitori;
4. trovare i nomi dei fornitori che distribuiscono tutti i prodotti presenti nel catalogo;
5. trovare i nomi dei fornitori che forniscono tutti i prodotti IBM presenti nel catalogo.

Soluzione:

$$1. \pi_{Nome, Marca, Modello} (\sigma_{Costo < 2.000} PRODOTTI \bowtie CATALOGO)$$

$$2. R = (FORNITORI \bowtie CATALOGO) \bowtie (\pi_{CP, Marca} PRODOTTI)$$

$$\pi_{Nome} (\sigma_{Marca = 'IBM'}(R))$$

$$3. COPIACATALOGO = \rho_{CF' \leftarrow CF} (\pi_{CP, CF} CATALOGO)$$

$$\pi_{CP} (\sigma_{CF > CF'} (CATALOGO \bowtie COPIACATALOGO))$$

4. Formuliamo innanzitutto l'interrogazione R che trova i fornitori a cui manca almeno un prodotto del catalogo:

$$R = \pi_{CF} ((\pi_{CF} (FORNITORI) \bowtie \pi_{CP} (CATALOGO)) - \pi_{CF, CP} (CATALOGO))$$

$$\pi_{Nome} ((\pi_{CF} (FORNITORI) - R) \bowtie FORNITORI)$$

5. La soluzione di questa interrogazione è uguale a quella del punto precedente usando al posto della relazione CATALOGO la relazione CATALOGO IBM definita come segue:

$$CATALOGO_{IBM} = \pi_{CP, CF} (\sigma_{Marca = 'IBM'} (CATALOGO \bowtie PRODOTTI))$$

Esercizio 3.21 Data la seguente istanza di basi di dati, mostrare i risultati intermedi e quelli finali delle interrogazioni definite nell'esercizio 3.19.

FORNITORI			
Nome	CodiceFornitore	Indirizzo	Città
Ladroni	001	Via Ostense	Roma
Risparmietti	002	Viale Marconi	Roma
Teloporto	010	Via Roma	Milano

CATALOGO		
CodiceFornitore	CodiceProdotto	Costo
001	0002	3.200
001	0003	2.200
002	0001	1.900
002	0002	2.500
002	0003	1.800
010	0001	2.200
010	0003	2.000

Prodotti			
CodiceProdotto	Nome	Marca	Modello
0001	Notebook	IBM	390
0002	Desktop	IBM	510
0003	Desktop	ACER	730

Soluzione:

1. Trovare Nome, Marca e Modello dei prodotti acquistabili con meno di 2.000.

Passo 1: $\text{PRODOTTI} \bowtie \text{CATALOGO}$

CP	Nome	Marca	Modello	CF	Costo
0001	Notebook	IBM	390	002	1.900
0001	Notebook	IBM	390	010	2.200
0002	Desktop	IBM	510	002	2.500
0002	Desktop	IBM	510	001	3.200
0003	Desktop	ACER	730	001	2.200
0003	Desktop	ACER	730	010	2.000
0003	Desktop	ACER	730	002	1.800

Passo 2: $\sigma_{\text{Costo} < 2.000} (\text{PRODOTTI} \bowtie \text{CATALOGO})$

CP	Nome	Marca	Modello	CF	Costo
0001	Notebook	IBM	390	002	1.900
0003	Desktop	ACER	730	002	1.800

Passo 3: $\pi_{\text{Nome}, \text{Marca}, \text{Modello}} (\sigma_{\text{Costo} < 2.000} (\text{PRODOTTI} \bowtie \text{CATALOGO}))$

Nome	Marca	Modello
Notebook	IBM	390
Desktop	ACER	730

2. Trovare i nomi dei fornitori che distribuiscono prodotti IBM (IBM è la marca di un prodotto).

Passo1: FORNITORI \bowtie CATALOGO

Nome	Indirizzo	Città	CF	CP	Costo
Ladroni	Via Ostense	Roma	001	0003	2.200
Ladroni	Via Ostense	Roma	001	0002	3.200
Risparmietti	Viale Marconi	Roma	002	0001	1.900
Risparmietti	Viale Marconi	Roma	002	0002	2.500
Risparmietti	Viale Marconi	Roma	002	0003	1.800
Teloporto	Via Roma	Milano	010	0001	2.200
Teloporto	Via Roma	Milano	010	0003	2.000

Passo 2: $R = (\text{FORNITORI} \bowtie \text{CATALOGO}) \bowtie (\pi_{CP, \text{Marca}} \text{PRODOTTI})$

Nome	Indirizzo	Città	CF	Costo	CP	Marca
Ladroni	Via Ostense	Roma	001	2.200	0003	ACER
Risparmietti	Viale Marconi	Roma	002	1.900	0001	IBM
Risparmietti	Viale Marconi	Roma	002	2.500	0002	IBM
Teloporto	Via Roma	Milano	010	2.200	0001	IBM
Ladroni	Via Ostense	Roma	001	3.200	0002	IBM
Teloporto	Via Roma	Milano	010	2.000	0003	ACER
Risparmietti	Viale Marconi	Roma	002	1.800	0003	ACER

Passo 3: $\sigma_{\text{Marca}='IBM'}(R)$

Nome	Indirizzo	Città	CF	Costo	CP	Marca
Risparmietti	Viale Marconi	Roma	002	1.900	0001	IBM
Teloporto	Via Roma	Milano	010	2.200	0001	IBM
Risparmietti	Viale Marconi	Roma	002	2.500	0002	IBM
Ladroni	Via Ostense	Roma	001	3.200	0002	IBM

Passo 4: $\pi_{\text{Nome}}(\sigma_{\text{Marca}='IBM'}(R))$

Nome
Ladroni
Risparmietti
Teloporto

3. Trovare i nomi dei fornitori che distribuiscono prodotti IBM (IBM è la marca di un prodotto).

Passo 1: $\text{COPIACATALOGO} = \rho_{CF' \leftarrow CF}(\pi_{CP, CF} \text{CATALOGO})$

CF'	CP
001	0002
001	0003
002	0001
002	0002
002	0003
010	0001
010	0003

Passo 2: CATALOGO \bowtie COPIACATALOGO

CF'	CP	CF
002	0001	002
010	0001	002
002	0001	010
010	0001	010
002	0002	002
001	0002	002
002	0002	001
001	0002	001
001	0003	001
010	0003	001
002	0003	001
001	0003	010
010	0003	010
002	0003	010
001	0003	002
010	0003	002
002	0003	002

Passo 3: $\sigma_{CF > CF'}(\text{CATALOGO} \bowtie \text{COPIACATALOGO})$

CF'	CP	CF
001	0002	002
002	0001	010
001	0003	010
002	0003	010
001	0003	002

Passo 4: $\pi_{CP}(\sigma_{CF > CF'}(\text{CATALOGO} \bowtie \text{COPIACATALOGO}))$

CodiceProdotto
0001
0002
0003

4. Trovare i codici di tutti i prodotti che sono forniti da almeno due fornitori.

Passo 1: $\pi_{CF}(\text{FORNITORI}) \bowtie \pi_{CP}(\text{CATALOGO})$
 Prodotto cartesiano tra CF di FORNITORI e CP di CATALOGO.

CF	CP
001	0001
001	0002
001	0003
002	0001
002	0002
002	0003
010	0001
010	0002
010	0003

Passo 2: $(\pi_{CF}(\text{FORNITORI}) \bowtie \pi_{CP}(\text{CATALOGO})) - \pi_{CF,CP}(\text{CATALOGO})$
 Attraverso questa interrogazione si ottiene una relazione che contiene i CF di Fornitori associati ai CP dei prodotti che non hanno nel catalogo, ossia i CF dei fornitori che non hanno almeno uno dei prodotti nel catalogo.

CF	CP	-	CF	CP	=	CF	CP
001	0001		001	0003		001	0001
001	0002		002	0001		001	0002
001	0003		002	0002		010	0002
002	0001		010	0001			
002	0002		001	0002			
002	0003		010	0003			
010	0001		002	0003			
010	0002						
010	0003						

Passo 3:
 $R = \pi_{CF}(\pi_{CF}(\text{FORNITORI}) \bowtie \pi_{CP}(\text{CATALOGO})) - \pi_{CF,CP}(\text{CATALOGO})$

CF
001
010

Passo 4 $(\pi_{CF}(\text{FORNITORI}) - R)$

CF
001

Passo 5: $\pi_{Nome}((\pi_{CF}(\text{FORNITORI}) - R) \bowtie \text{FORNITORI})$

Nome
Risparmietti

5. Trovare i nomi dei fornitori che forniscono tutti i prodotti IBM presenti nel

catalogo.

Passo 1: $CATALOGO_{IBM} = \pi_{CP,CF}(\sigma_{Marca='IBM'}(CATALOGO \bowtie PRODOTTI))$

CP	CF
0001	002
0001	010
0002	002
0002	001

Passo 2:

Nome
Risparmietti