

## Prima prova di verifica del 8/11/2006

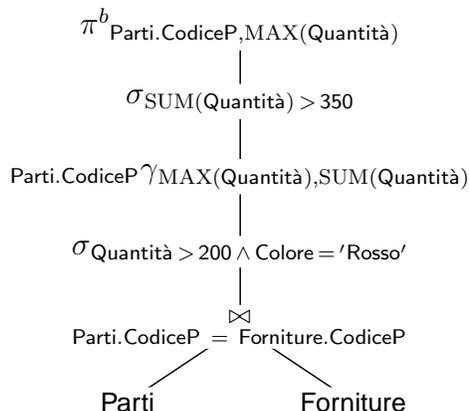
1. Si vuole progettare la base di dati per le risorse umane di un insieme di università. Di una persona interessano il codice fiscale, che la individua, il nome, il sesso, e l'anno di nascita. Una persona è in relazione con un'università, della quale interessano il nome, che la individua, la città ed i recapiti telefonici, che possono essere più di uno. Una persona può essere un dipendente o uno studente. Un dipendente può essere un docente o un tecnico-amministrativo. Dei dipendenti interessano l'anno di assunzione e lo stipendio. Di un docente interessa la qualifica e di un tecnico-amministrativo interessa il livello di inquadramento. Degli studenti interessano la matricola, l'anno di immatricolazione e il corso di laurea che frequenta. Degli studenti laureandi (ovvero, che hanno iniziato il lavoro di tesi) interessano il titolo della tesi e i docenti relatori della tesi, che possono essere più di uno. Un dipendente non può essere studente.

- (a) Si definisca lo schema concettuale della base di dati.
- (b) Si traduca lo schema concettuale in uno schema relazionale grafico e testuale, usando la notazione  $R(\underline{A}, B^*, \dots)$

2. Si consideri il seguente schema relazionale:

Fornitori(CodiceF, NomeF, CittàF);  
 Parti(CodiceP, NomeP, Colore, Peso)  
 Forniture(CodiceF\*, CodiceP\*, Quantità, CittàDestinazione)

Si scriva in SQL l'interrogazione con il seguente albero logico:



Si scrivano in SQL le seguenti interrogazioni (nelle interrogazioni, “fornitori di Milano” significa fornitori il cui attributo CittàF vale 'Milano'). .

- (a) Trovare il nome e il codice dei fornitori di Milano e il codice delle parti fornite. Si dia anche la rappresentazione in forma algebrica della soluzione.
- (b) Trovare il nome della parte e quello del fornitore per ogni parte destinata alla stessa città del fornitore che la fornisce. Si dia anche la rappresentazione in forma algebrica della soluzione.
- (c) Trovare il codice, il nome e il totale delle quantità delle parti di colore Rosso ricevute da qualche fornitore. Si dia anche la rappresentazione in forma algebrica della soluzione.
- (d) Per ogni città CF in cui è locato qualche fornitore e per ogni città di destinazione CD di qualche fornitura, trovare le triple (CF, CD, numero di forniture da CF a CD), se esistono almeno tre forniture da CF a CD.
- (e) Per ogni parte che ha almeno un fornitore a Pisa, trovare il codice e il numero totale di fornitori (compresi quelli che non sono a Pisa).
- (f) Trovare il codice e il nome di tutte le parti di colore Rosso che sono fornite **solo** da fornitori di Pisa (ovvero che sono fornite esclusivamente da fornitori di Pisa).
- (g) Trovare il nome dei fornitori di Pisa che forniscono tutte le parti.
- (h) Trovare il codice e il nome dei fornitori che forniscono almeno le parti fornite dal fornitore con codice F2.

## Soluzione prima prova di verifica del 8/11/2006

1. Il progetto concettuale è in Figura 1 e quello relazionale in Figura 2.

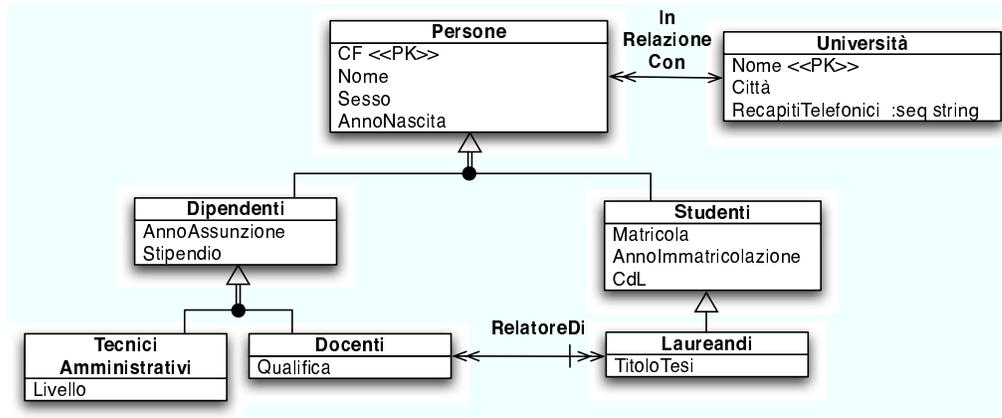


Figura 1: Schema concettuale

2. Interrogazione per l'albero logico:

```
SELECT Parti.CodiceP, MAX(Quantità)
FROM Parti, Forniture
WHERE Parti.CodiceP = Forniture.CodiceP
AND Quantità > 200 AND Colore = 'Rosso'
GROUP BY Parti.CodiceP
HAVING SUM(Quantità) > 350 ;
```

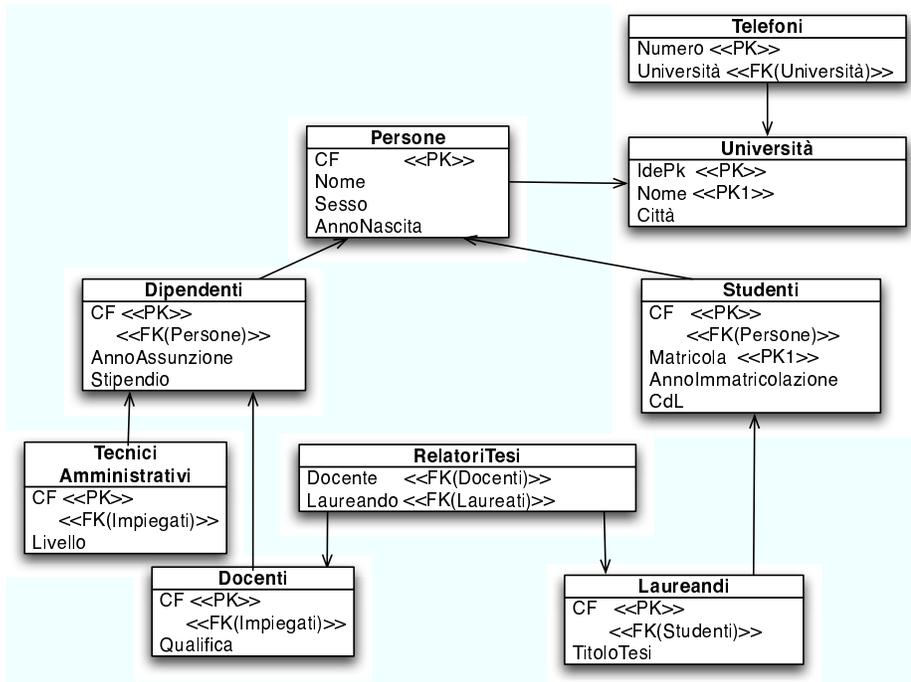


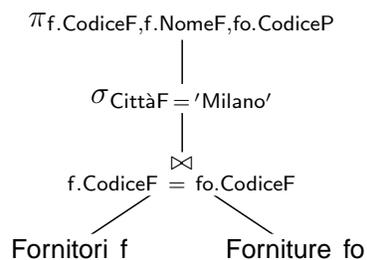
Figura 2: Schema relazionale

3. Altre interrogazioni:

- (a) Trovare il nome e il codice dei fornitori di Milano e il codice delle parti fornite. Si dia anche la rappresentazione in forma algebrica della soluzione.

```

SELECT DISTINCT f.CodiceF, f.NomeF, fo.CodiceP
FROM Fornitori f, Forniture fo
WHERE f.CodiceF = fo.CodiceF AND CittàF = 'Milano';
  
```

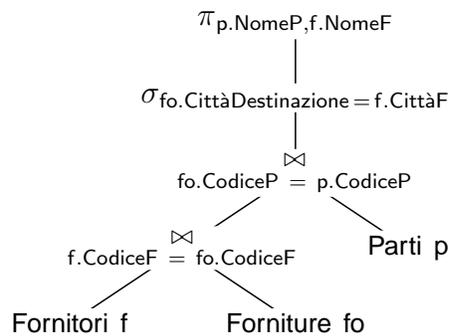


- (b) Trovare il nome della parte e quello del fornitore per ogni parte destinata alla stessa città del fornitore che la fornisce. Si dia anche la rappresentazione in forma algebrica della soluzione.

```

SELECT DISTINCT p.NomeP, f.NomeF
FROM Fornitori f, Forniture fo, Parti p
WHERE f.CodiceF = fo.CodiceF
AND fo.CodiceP = p.CodiceP
AND fo.CittàDestinazione = f.CittàF ;

```

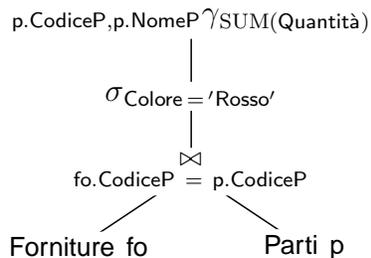


- (c) Trovare il codice, il nome e il totale delle quantità delle parti di colore Rosso ricevute da qualche fornitore. Si dia anche la rappresentazione in forma algebrica della soluzione.

```

SELECT p.CodiceP, p.NomeP, SUM(Quantità)
FROM Forniture fo, Parti p
WHERE fo.CodiceP = p.CodiceP AND p.Colore = 'Rosso'
GROUP BY p.CodiceP, p.NomeP;

```



- (d) Per ogni città CF in cui è locato qualche fornitore e per ogni città di destinazione CD di qualche fornitura, trovare le triple (CF, CD, numero di forniture da CF a CD), se esistono almeno tre forniture da CF a CD.

```
SELECT    f.CittàF, fo.CittàDestinazione, COUNT(*)
FROM      Fornitori f, Forniture fo
WHERE     f.CodiceF = fo.CodiceF
GROUP BY  f.CittàF, fo.CittàDestinazione
HAVING    COUNT(*) > 2;
```

- (e) Per ogni parte che ha almeno un fornitore a Pisa, trovare il codice e il numero totale di fornitori (compresi quelli che non sono a Pisa).

```
SELECT    fo.CodiceP, COUNT(*)
FROM      Forniture fo
WHERE     (FOR SOME fo2 IN Forniture, f In Fornitori
          WHERE fo2.CodiceP = fo.CodiceP AND f.CodiceF = fo2.CodiceF
          : f.CittàF = 'Pisa')
GROUP BY  fo.CodiceP;
```

```
SELECT    fo.CodiceP, COUNT(*)
FROM      Forniture fo
WHERE     EXISTS (
          SELECT *
          FROM  Forniture fo2, Fornitori f
          WHERE fo2.CodiceP = fo.CodiceP AND f.CodiceF = fo2.CodiceF
          AND f.CittàF = 'Pisa')
GROUP BY  fo.CodiceP;
```

- (f) Trovare il codice e il nome di tutte le parti di colore Rosso che sono fornite **solo** da fornitori di Pisa (ovvero che sono fornite esclusivamente da fornitori di Pisa).

```
SELECT    p.CodiceP, p.NomeP
FROM      Parti p, Forniture fo
WHERE     fo.CodiceP = p.CodiceP AND p.Colore = 'Rosso' AND
          (FOR ALL fo IN Forniture, f IN Fornitori f
          WHERE fo.CodiceP = p.CodiceP AND fo.CodiceF = f.CodiceF
          : f.CittàF = 'Pisa');
```

```
SELECT    p.CodiceP, p.NomeP
FROM      Parti p, Forniture fo
WHERE     fo.CodiceP = p.CodiceP AND p.Colore = 'Rosso' AND
          NOT EXISTS (
          SELECT *
          FROM  Forniture fo, Fornitori f
          WHERE fo.CodiceP = p.CodiceP AND fo.CodiceF = f.CodiceF
          AND NOT(f.CittàF = 'Pisa') );
```

(g) Trovare il nome dei fornitori di Pisa che forniscono tutte le parti.

```
SELECT f.NomeF
FROM   Fornitori f
WHERE  f.CittàF = 'Pisa' AND
      FOR ALL p IN Parti
      : FOR SOME fo IN Forniture WHERE fo.CodiceF = f.CodiceF AND fo.CodiceP = p.CodiceP;
```

```
SELECT f.NomeF
FROM   Fornitori f
WHERE  f.CittàF = 'Pisa' AND
      NOT EXISTS (
        SELECT *
        FROM   Parti p
        WHERE  NOT EXISTS (
          SELECT *
          FROM   Forniture fo
          WHERE  fo.CodiceF = f.CodiceF AND fo.CodiceP = p.CodiceP ));
```

(h) Trovare il codice e il nome dei fornitori che forniscono almeno le parti fornite dal fornitore con codice F2.

```
SELECT f.Codice, f.NomeF
FROM   Fornitori f
WHERE  NOT(f.CodiceF = 'F2' ) AND
      FOR ALL fo IN Forniture WHERE fo.CodiceF = 'F2'
      : FOR SOME ffo IN Forniture WHERE ffo.CodiceF = f.CodiceF AND ffo.CodiceP = fo.CodiceP;
```

```
SELECT f.Codice, f.NomeF
FROM   Fornitori f
WHERE  NOT(f.CodiceF = 'F2' ) AND
      NOT EXISTS (
        SELECT *
        FROM   Forniture fo
        WHERE  fo.CodiceF = 'F2'
        AND   NOT EXISTS (
          SELECT *
          FROM   Forniture ffo
          WHERE  ffo.CodiceF = f.CodiceF AND ffo.CodiceP = fo.CodiceP ));
```