

# Progettazione del Software

Emiliano Casalicchio

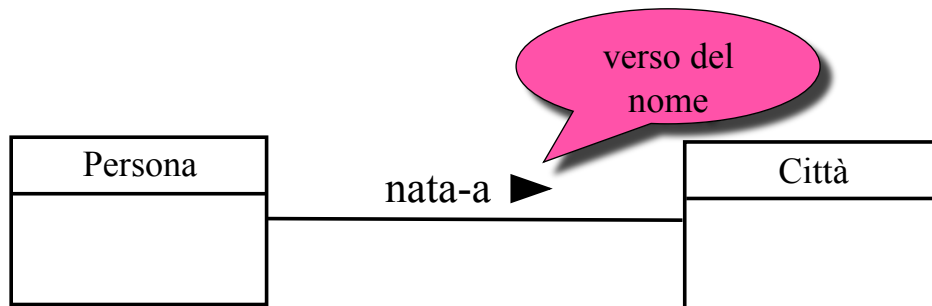
*Dipartimento di Informatica e Sistemistica  
SAPIENZA Università di Roma – Sede di Rieti*

<http://www.ce.uniroma2.it/courses/PSW>

- Abbiamo introdotto il concetto di classe, oggetto e relazione
- Abbiamo introdotto i criteri per individuare classi, attributi e metodi
- Vediamo ora un alcuni di aspetti pratici relativa al modello delle classi

## Nomi di associazioni

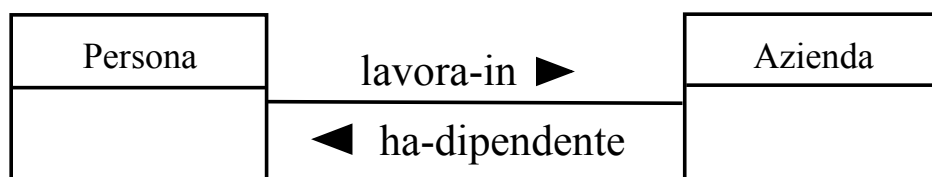
- Alcune volte è interessante specificare un **verso** per il nome della associazione



- Attenzione: la notazione riportata sopra **non significa che l'associazione è navigabile (attraversabile) solo in un verso**
- In altre parole, il verso **non è una caratteristica del significato della associazione**, ma dice semplicemente che il nome scelto per la associazione evoca un verso (nell'esempio, il verso è dalla Persona alla Città)

## Nomi di associazioni

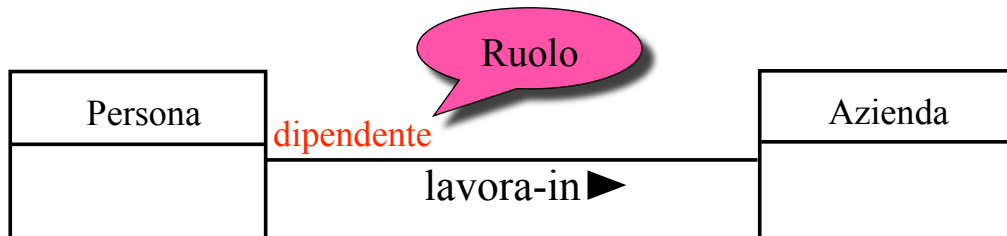
- È anche possibile assegnare due nomi con i relativi versi alla stessa associazione



- Osserviamo ancora che le frecce che simboleggiano il verso non aggiungono nulla al significato della associazione (che formalmente si può considerare sempre una relazione matematica), ma aiutano ad interpretare il senso dei nomi scelti per l'associazione
- Le frecce che simboleggiano il verso si indicano anche nel link che sono istanze delle associazioni

## Ruoli nelle associazioni

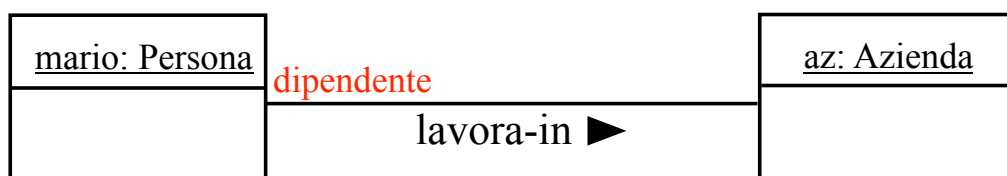
- È possibile aggiungere alla associazione una informazione che specifica il ruolo che una classe gioca nella associazione



- Il ruolo si indica con un nome posizionato lungo la linea che rappresenta l'associazione, vicino alla classe alla quale si riferisce
- Nell'esempio, **dipendente** è il ruolo che la persona gioca nell'associazione "lavora-in" con Azienda

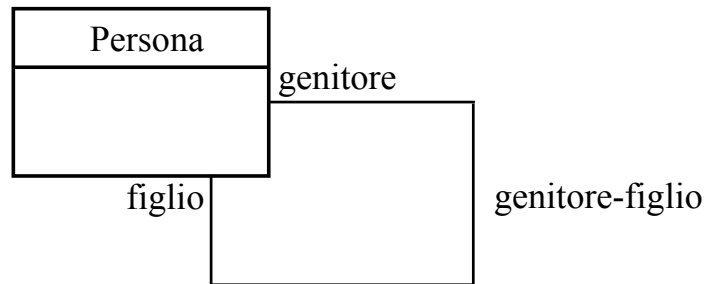
## Ruoli nei link

- Se nella associazione  $A$  è indicato il ruolo giocato dalla classe  $C$ , tale ruolo sarà indicato (vicino alla corrispondente istanza di  $C$ ) in ogni link che è istanza di  $A$
- Esempio:



## Ancora sui ruoli

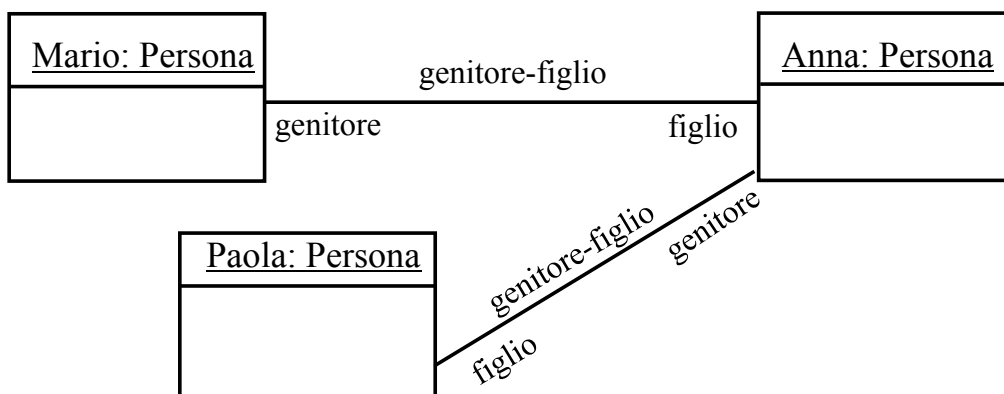
- Analogamente al verso, il ruolo è generalmente opzionale, e non aggiunge nulla al significato dell'associazione
- L'unico caso in cui il ruolo è **obbligatorio** è quello in cui l'associazione insiste più volte sulla stessa classe, e rappresenta una relazione non simmetrica
- Esempio:



- Se non fossero indicati i ruoli nell'associazione “genitore-figlio”, non sapremmo interpretare correttamente i link che sono istanze dell'associazione

## Osservazioni sui ruoli

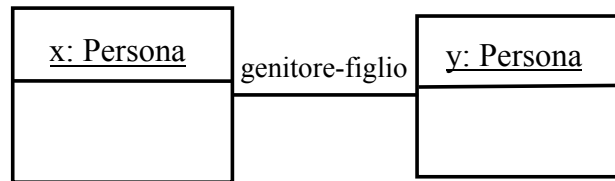
Esempio della necessità dei ruoli nei link:



Se non fossero indicati i ruoli nell'associazione “genitore-figlio”, non sapremmo interpretare correttamente i link che sono istanze dell'associazione

## Importanza dei ruoli

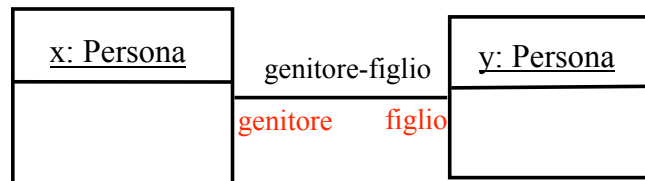
**Senza ruoli non è chiaro chi è il genitore e chi il figlio.** In altre parole, dalla coppia  $\langle x,y \rangle$  in genitore-figlio non si evincono i ruoli di  $x$  e  $y$



**Con i ruoli, non c'è più ambiguità.** Formalmente, il link è ora una coppia etichettata:

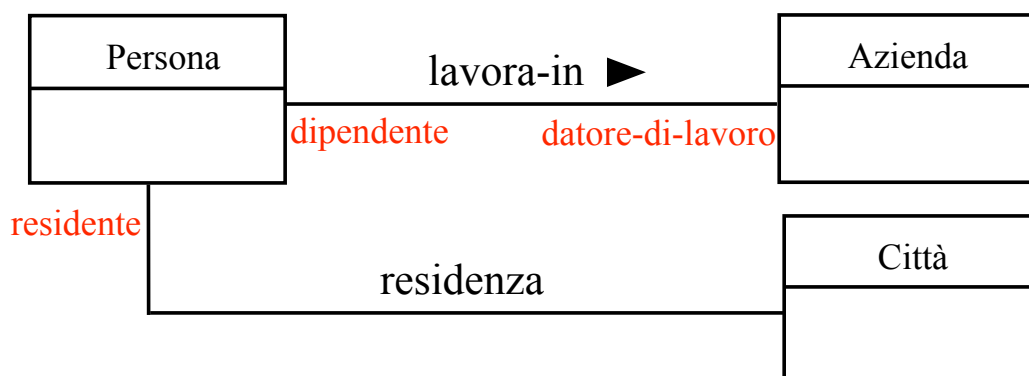
$\langle \text{genitore}:x, \text{figlio}:y \rangle$

dalla quale si evincono i ruoli di  $x$  e  $y$



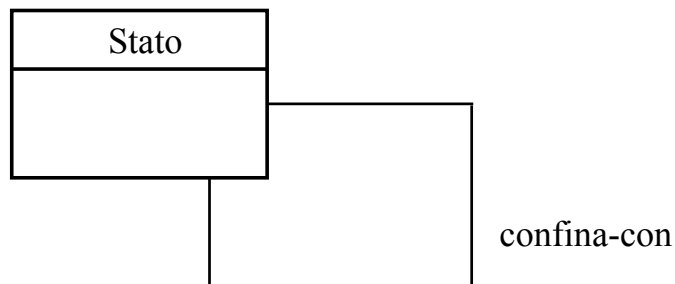
## Importanza dei ruoli

Anche nei casi in cui non è strettamente necessario, il ruolo può essere utile per aumentare la leggibilità del diagramma



## Casi di ruoli non significativi

- Ci sono casi in cui l'associazione insiste più volte sulla stessa classe, ma il ruolo non è significativo, in particolare quando l'associazione rappresenta una **relazione simmetrica**
- Esempio:



## Esercizio

Considerare le seguenti relazioni fra persone:

1. Essere amico
2. Essere coniuge
3. Essere collega di lavoro
4. Essere superiore (nell'ambito del lavoro)

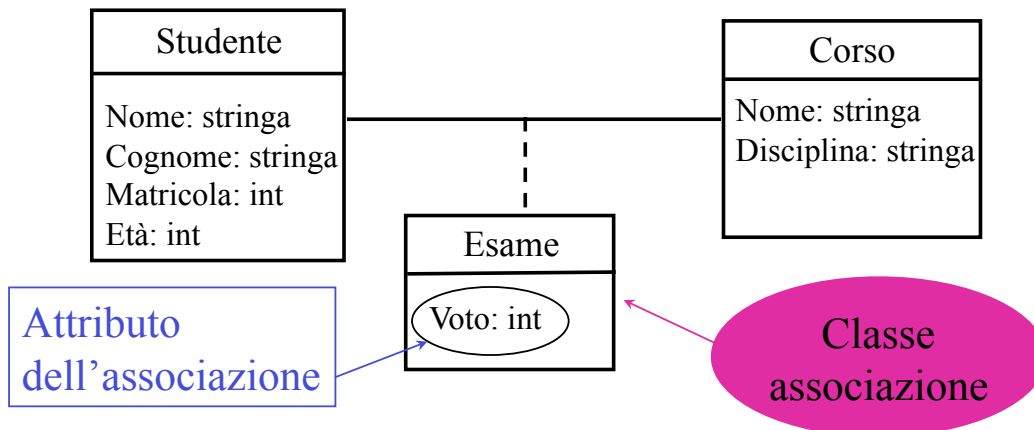
Per quali fra esse prevedereste dei ruoli?

In tal caso, quali nomi scegliereste?

## Attributi di associazione

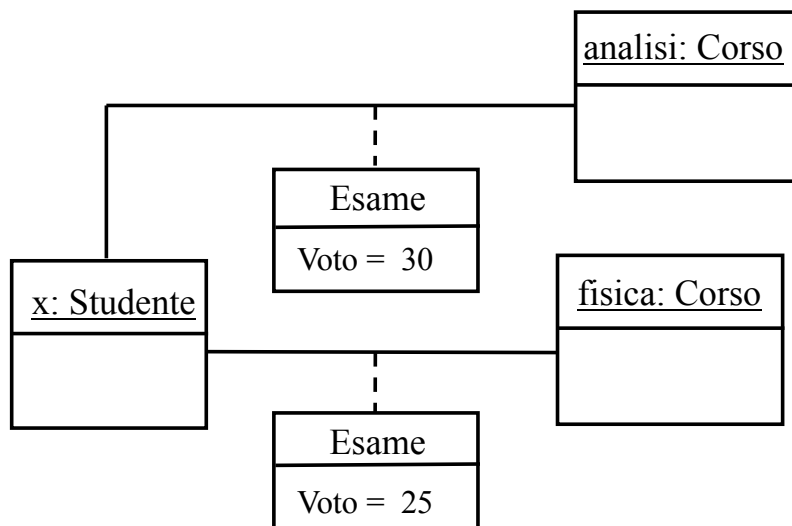
Analogamente alle classi, anche **le associazioni possono avere attributi**. Formalmente, **un attributo di una associazione è una funzione** che associa ad ogni link che è istanza dell'associazione un valore di un determinato tipo

**Esempio:** Voto non è una proprietà nè di Studente, nè di Corso, ma della associazione Esame tra Studente e Corso



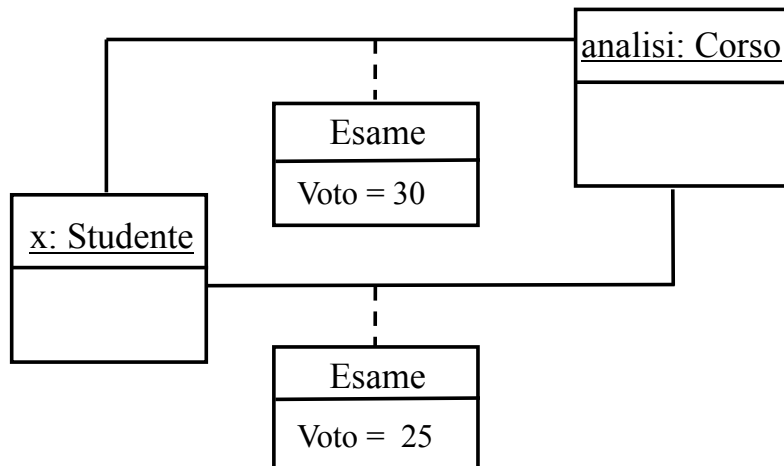
## Attributi nei link

Ovviamente, se una associazione ha un attributo, ogni link che è istanza dell'associazione avrà un valore per quell'attributo



## Attributi nei link

Con la presenza degli attributi, il significato dell'associazione (relazione matematica) non cambia. Quindi questo è un errore:



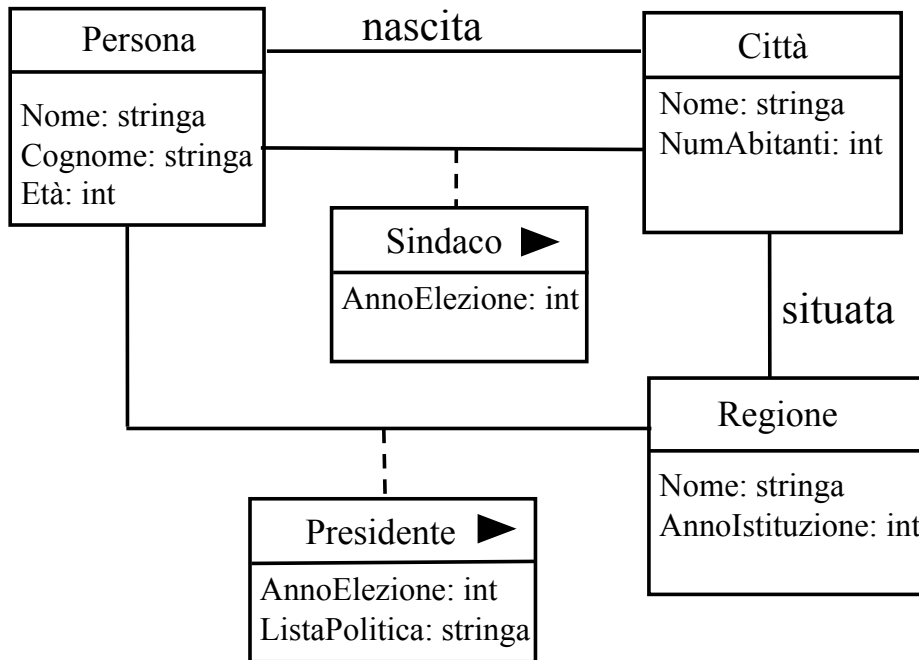
## Esercizio 3

Tracciare il diagramma delle classi corrispondenti alle seguenti specifiche:

*Si vogliono modellare le persone (con nome, cognome, età), le città di nascita (con nome, numero di abitanti, e sindaco, compresa l'indicazione dell'anno di elezione), e le regioni in cui si trovano le città (con nome, anno di istituzione, e presidente, con l'anno di elezione e lista politica in cui è stato eletto).*



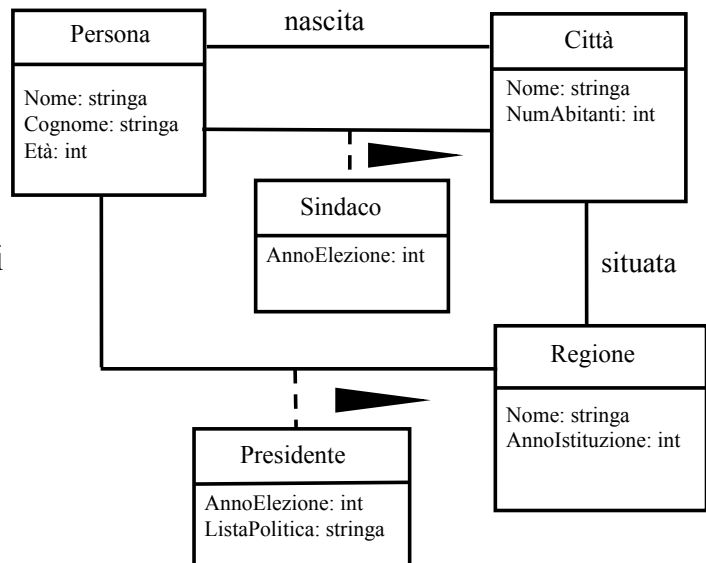
## Soluzione dell'esercizio 3



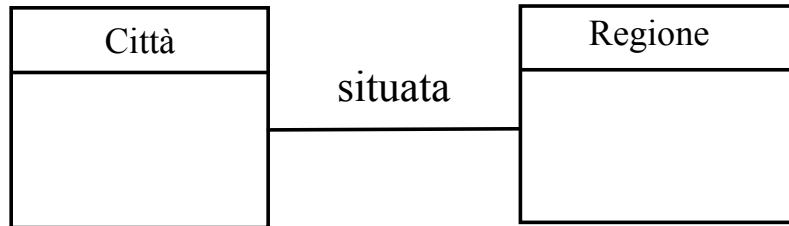
## Esercizio

- Con riferimento al diagramma delle classi precedente, è possibile:

- Rappresentare che Antonio è nato a Milano ed è sindaco di Roma, eletto nel 1987?
- Inoltre, che Francesco è nato a Roma ed è sindaco di Roma, eletto nel 1992?
- Inoltre, che Walter è nato a Roma ed è sindaco di Roma, eletto nel 1999?
- Inoltre, che Francesco è stato eletto nuovamente sindaco di Roma nel 2003?



## Significato delle associazioni

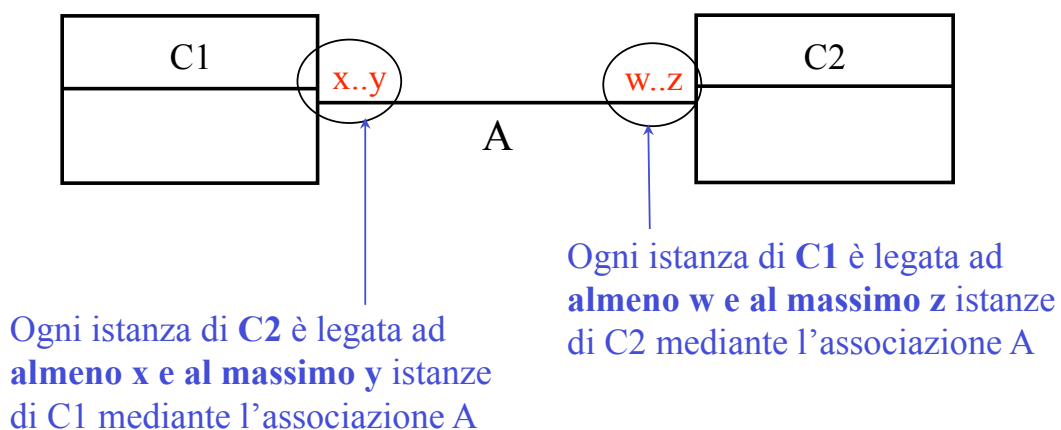


Quali delle seguenti cose ci dice questo diagramma delle classi?

- Ogni città è situata in una regione
- Ogni regione ha delle città in essa situate
- Alcune città sono situate in una regione
- Alcune regioni hanno città situate in esse
- Nessuna delle precedenti (specificare)

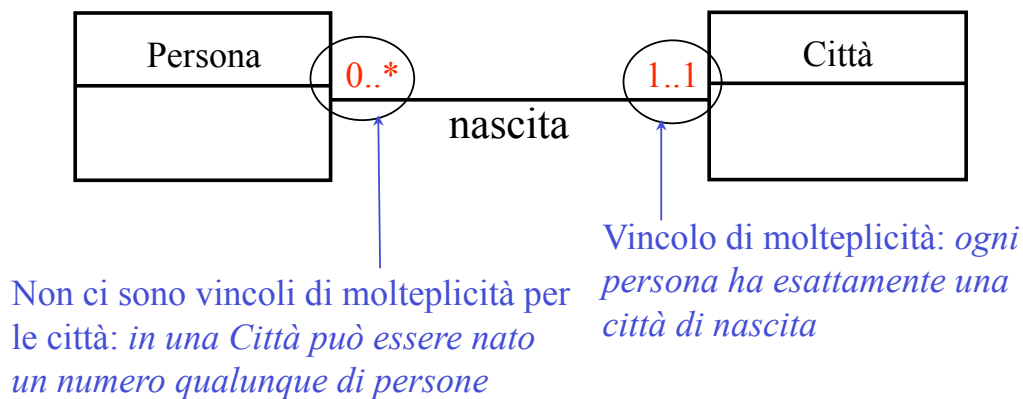
## Molteplicità delle associazioni

- Per specificare con maggiore precisione il significato delle **associazioni binarie** (non ennarie – vedi dopo) si possono definire i **vincoli di molteplicità** (o semplicemente molteplicità) delle associazioni



## Molteplicità delle associazioni

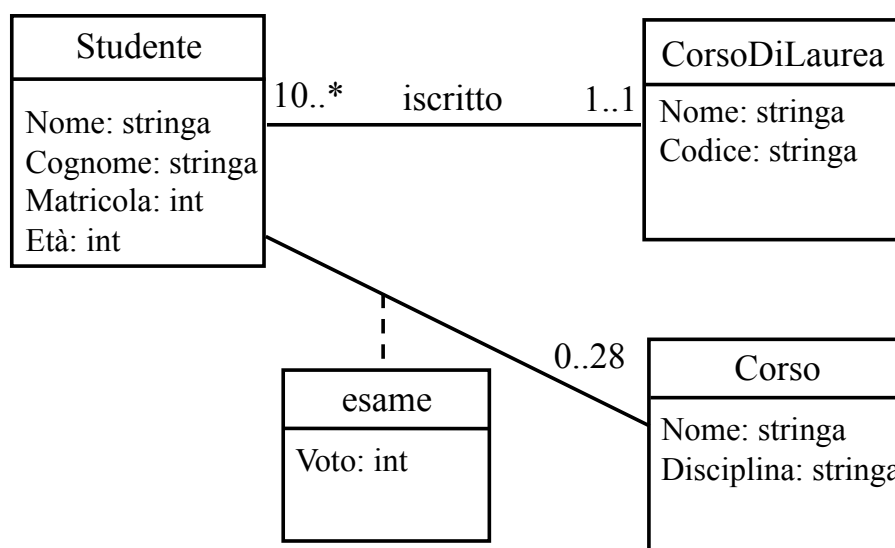
- Esempio: ogni istanza di Persona deve essere legata ad esattamente una istanza (cioè ad almeno una e al massimo una) istanza di Città da link della associazione “nascita”



## Molteplicità delle associazioni: notazione

- Le molteplicità si definiscono solo per le associazioni binarie
- Possibili molteplicità:
  - 0 .. \* (nessun vincolo, si può evitare di indicare)
  - 0 .. 1 (nessun limite per il minimo, e al massimo una)
  - 1 .. \* (al minimo una, e nessun limite per il massimo)
  - 1 .. 1 (esattamente una)
  - 0 .. y (nessun limite per il minimo, e al massimo y, con y intero > 0)
  - x .. \* (al minimo x, con x intero ≥ 0, e nessun limite per il massimo)
  - x .. y (al minimo x e al massimo y, con x ,y interi, x ≥ 0 e y ≥ x)

## Esempi di molteplicità

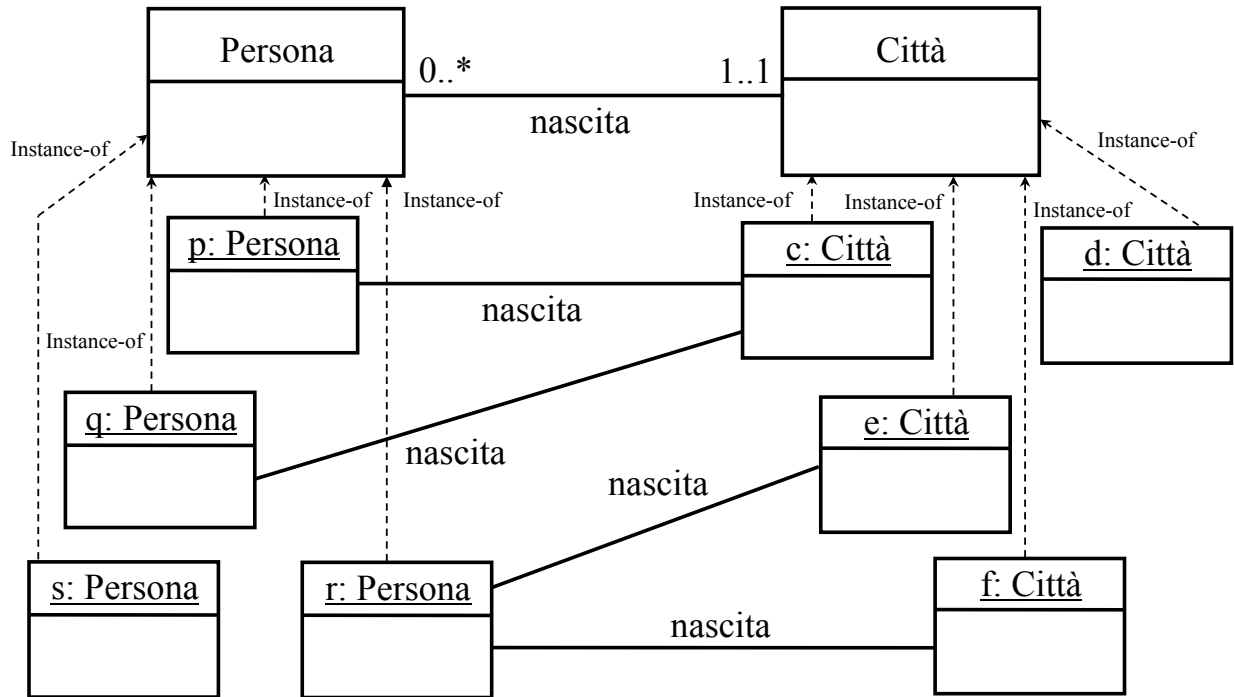


- Ogni studente   iscritto ad un corso di laurea
- Ogni corso di laurea ha almeno 10 iscritti
- Ogni studente pu  sostenere al massimo 28 esami

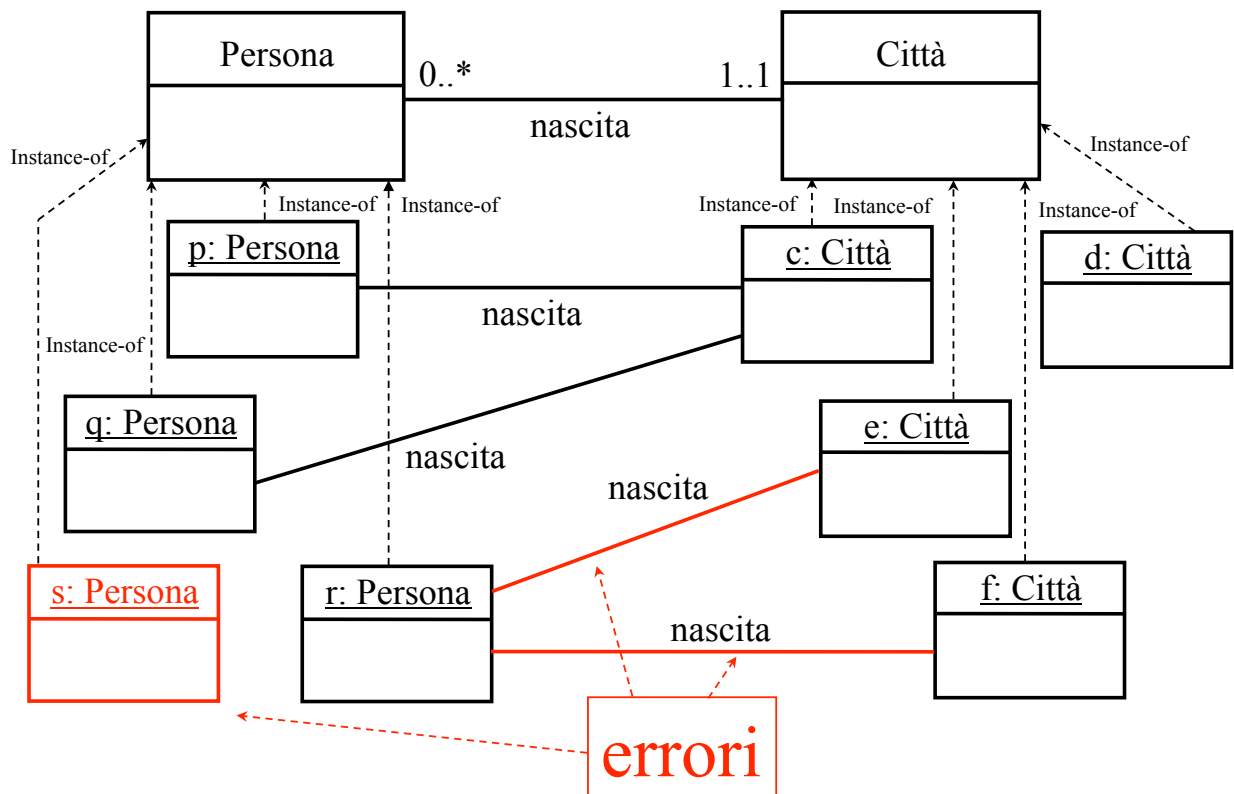
## Esercizio

- Con riferimento al diagramma delle classi precedente,   possibile:
  1. Che uno studente non sia iscritto ad alcun corso di laurea?
  2. Che uno studente sia iscritto a due corsi di laurea differenti?
  3. Che ci siano esattamente 11 studenti, tutti iscritti allo stesso corso di laurea?
  4. Che ci siano esattamente 3 studenti?
  5. Che ci siano esattamente 10 studenti?

## Esercizio 4: individuare gli errori

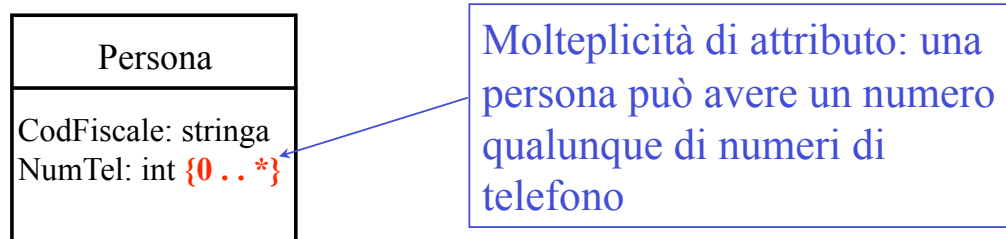


## Soluzione dell'esercizio 4



## Molteplicità di attributi

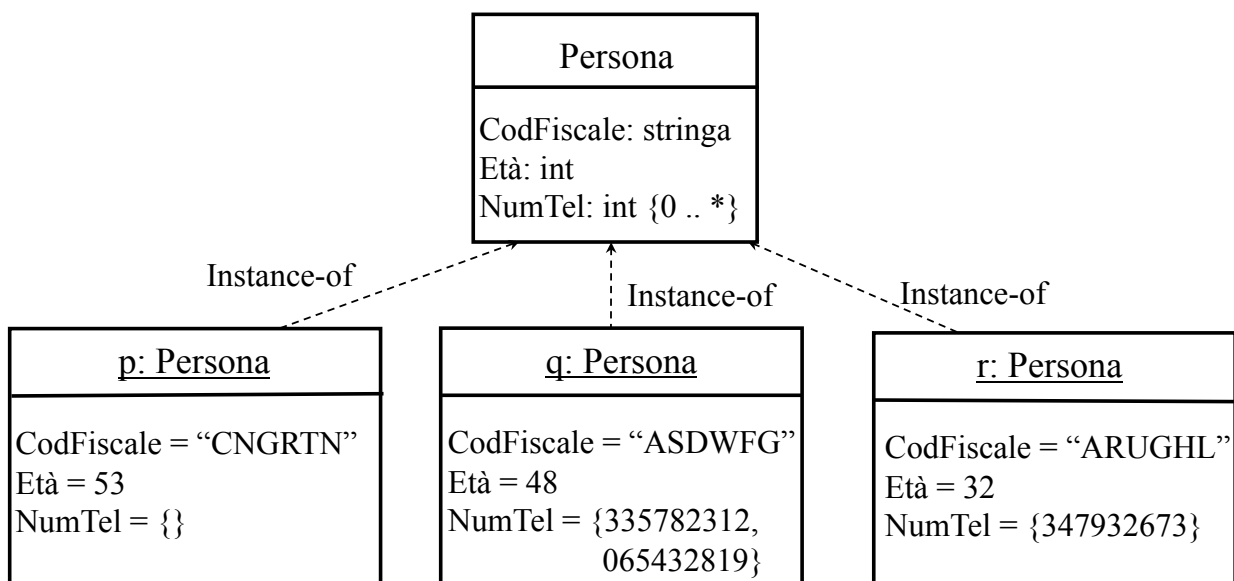
- Si possono specificare anche le molteplicità degli attributi. Se le **molteplicità** di un attributo  $B$  di tipo  $T$  di una classe  $C$  **non** vengono **indicate**, vuol dire che  $B$  associa ad ogni istanza di  $C$  esattamente un valore di  $T$  (come detto prima), che è equivalente a dire che la molteplicità è **1..1**
- Al contrario, se un attributo  $B$  di tipo  $T$  di una classe  $C$  ha **molteplicità  $x..y$** , allora  $B$  associa ad ogni istanza di  $C$  al minimo  $x$  e al massimo  $y$  valori di tipo  $T$



Un attributo di tipo  $T$  della classe  $C$  con molteplicità diversa da  $\{1..1\}$  si dice **multivalore**, e formalmente non è una funzione totale, ma una relazione tra la classe  $C$  ed il tipo  $T$

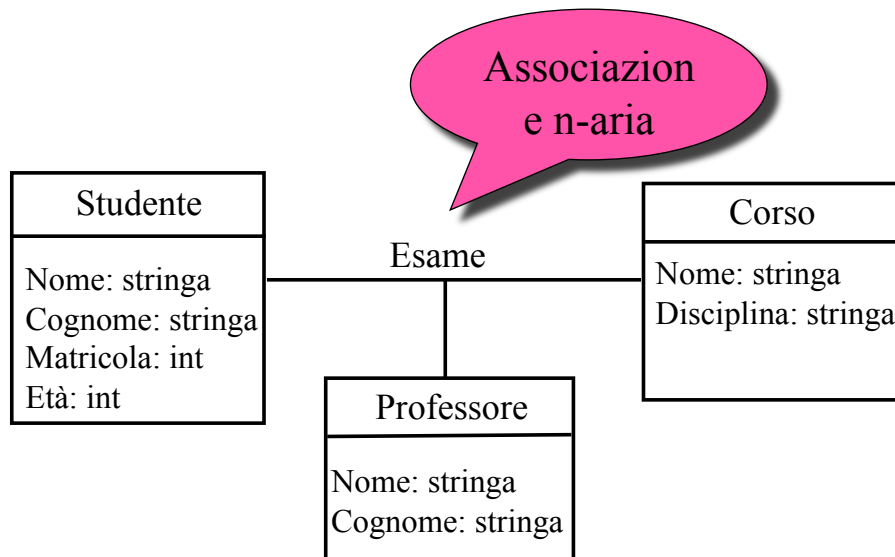
## Attributi multivalore nelle istanze

Nelle istanze, il valore di un attributo multivalore si indica mediante un insieme



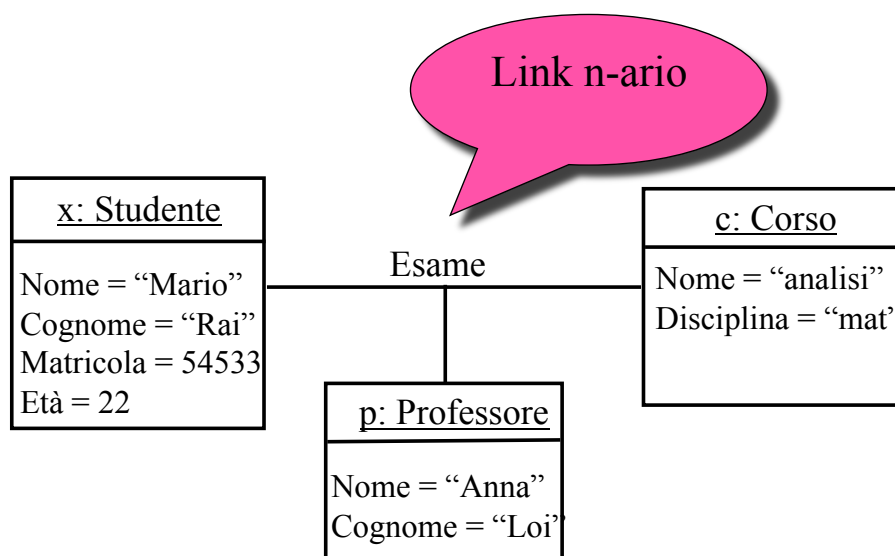
## Associazioni n-arie

Una associazione può essere definita su tre o più classi. In tale caso l'associazione si dice **n-aria**, e modella una **relazione matematica tra n insiemi**



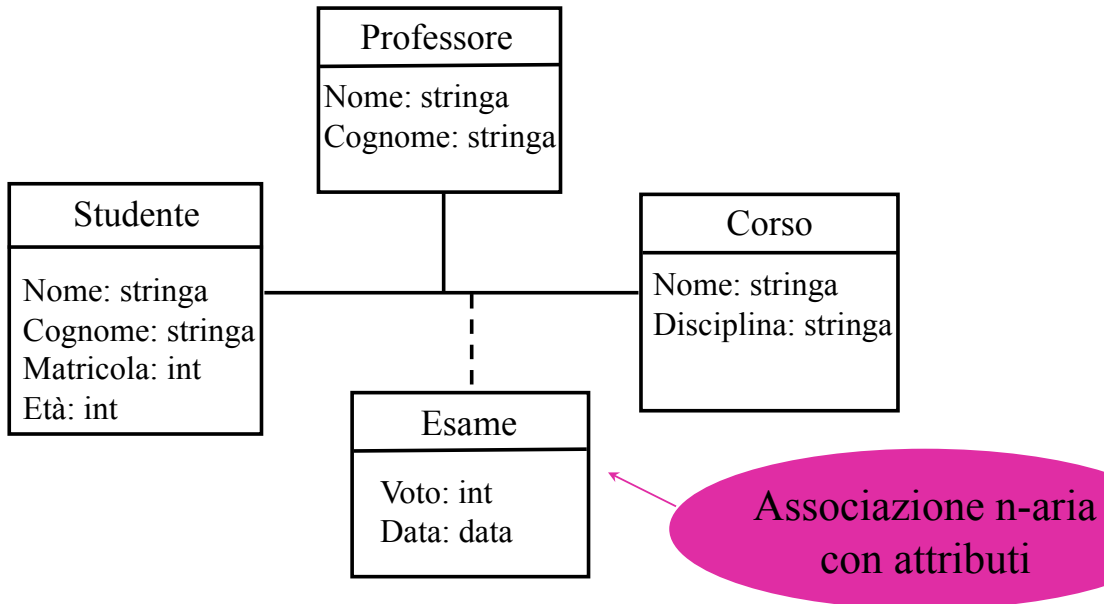
## Istanze di associazioni n-arie: link n-ari

Ogni istanza di una associazione n-aria è un **link n-ario**, cioè che coinvolge **n oggetti** (è una *ennupla*)



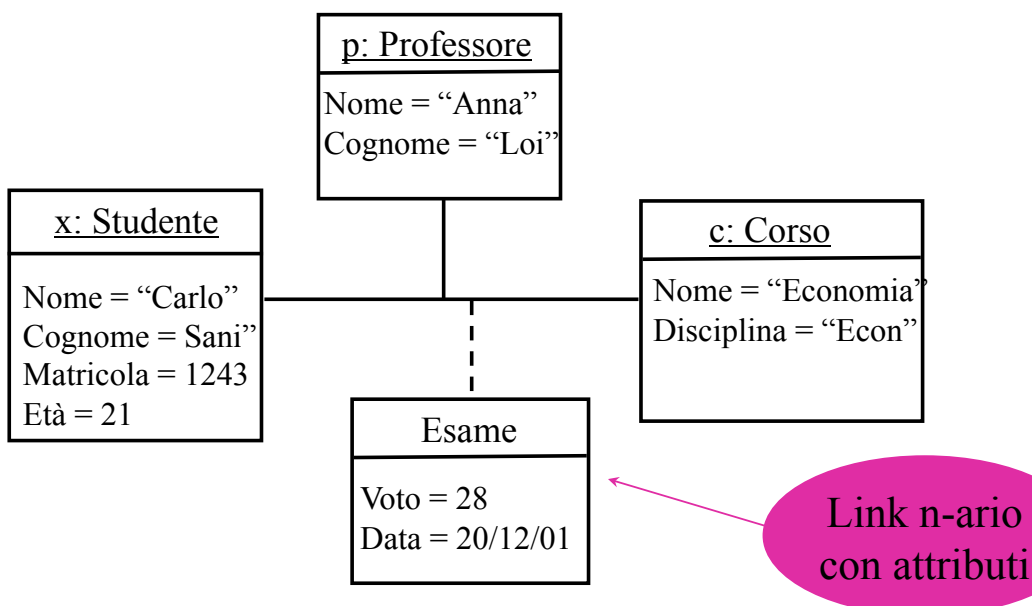
## Associazioni n-arie con attributi

Ovviamente, anche le associazioni n-arie possono avere attributi



## Link n-ari con attributi

I link che sono istanze di associazioni n-arie con attributi, hanno un valore per ogni attributo





## Associazioni n-arie e molteplicità

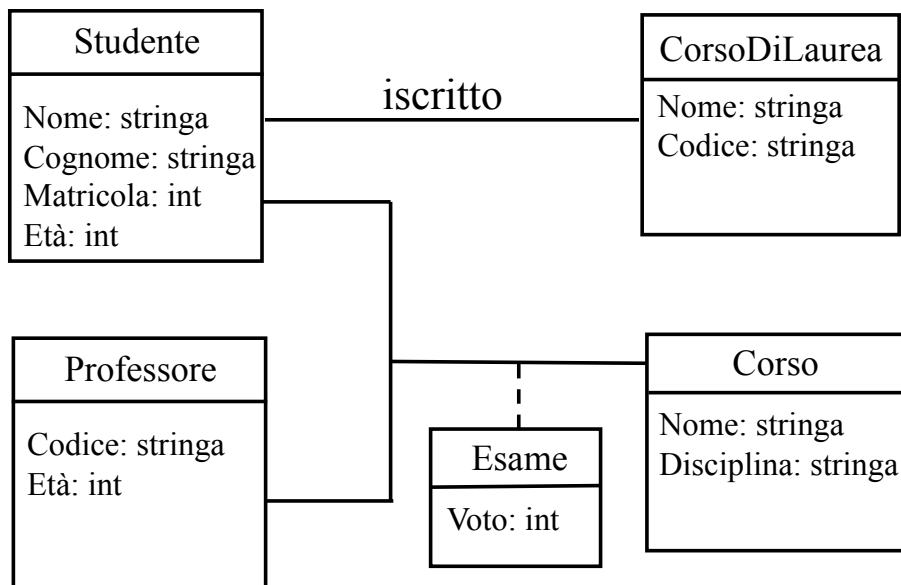
- Ci sono diversi vincoli di molteplicità che sarebbe di interesse esprimere su associazioni n-arie (*vedi Corso di Basi di Dati*)...
- ...tuttavia noi in questo corso non li studieremo in modo specifico, ne considereremo la notazione per esprimerli in UML.
- Qualora avessimo bisogno di specificare un vincolo di molteplicità lo faremo in linguaggio testuale con un commento.

## Esempio

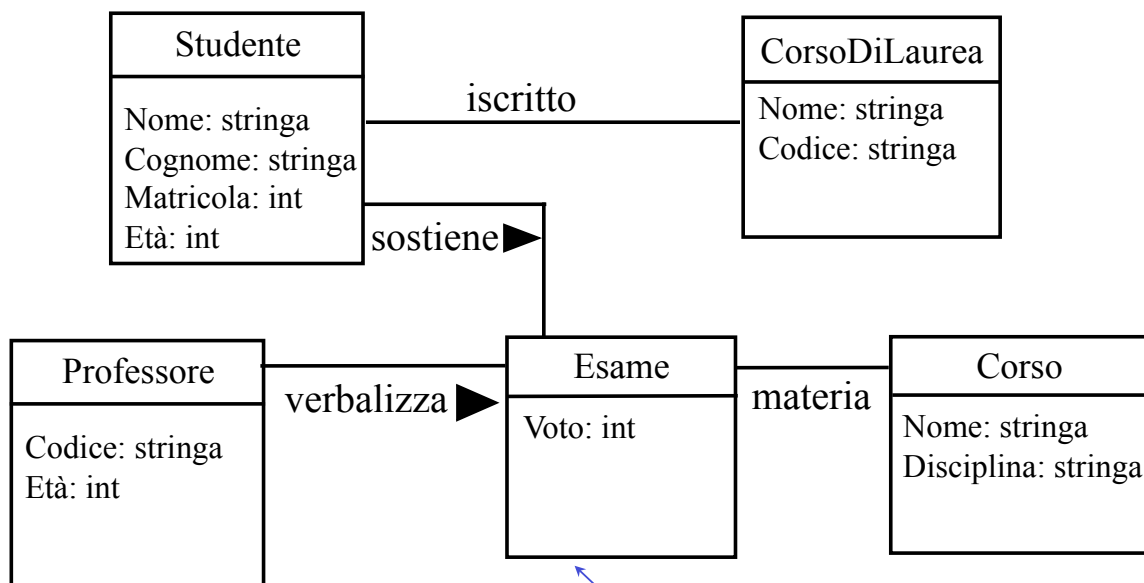
Tracciare il diagramma delle classi corrispondenti alle seguenti specifiche:

*Si vogliono modellare gli studenti (con nome, cognome, numero di matricola, età), il corso di laurea in cui sono iscritti, ed i corsi di cui hanno sostenuto l'esame, con il professore che ha verbalizzato l'esame, ed il voto conseguito. Di ogni corso di laurea interessa il codice e il nome. Di ogni corso interessa il nome e la disciplina a cui appartiene (ad esempio: matematica, fisica, informatica, ecc.). Di ogni professore interessa codice ed età.*

## Diagramma delle classi per l'esempio

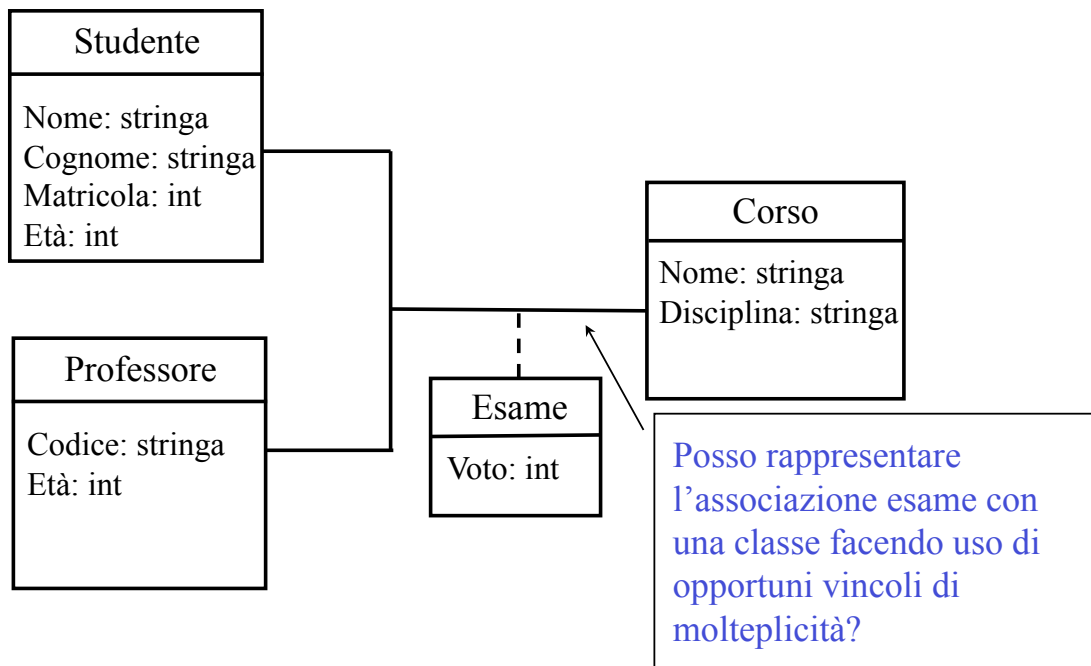


## Soluzione scorretta per l'esempio

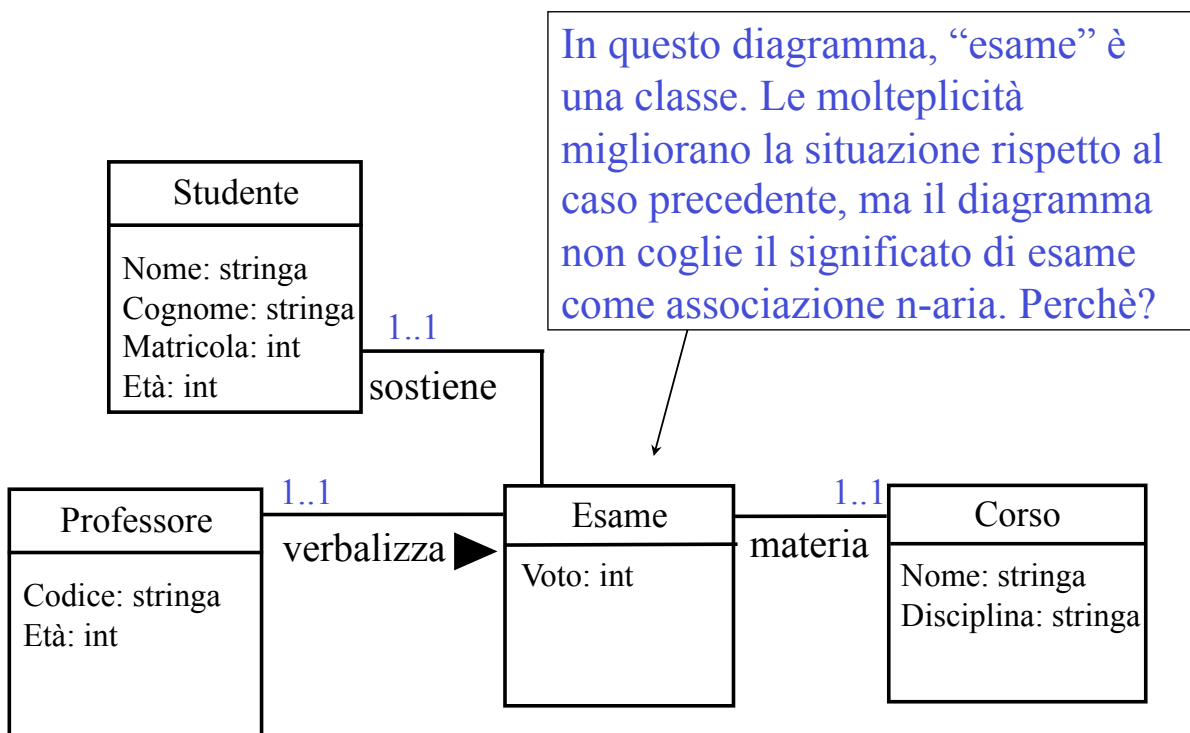


In questo diagramma, "esame" è una classe. Qual è l'errore?

## Torniamo alla associazione n-aria

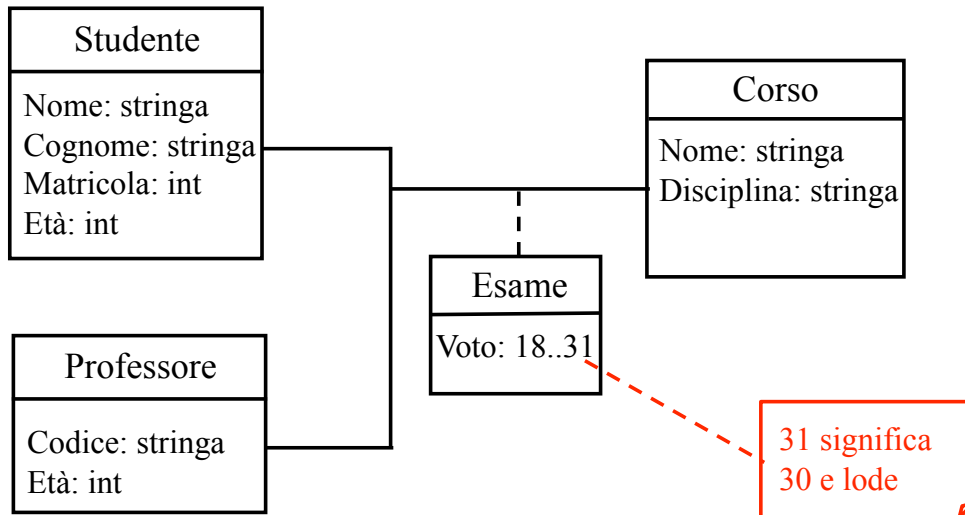


## Soluzione scorretta



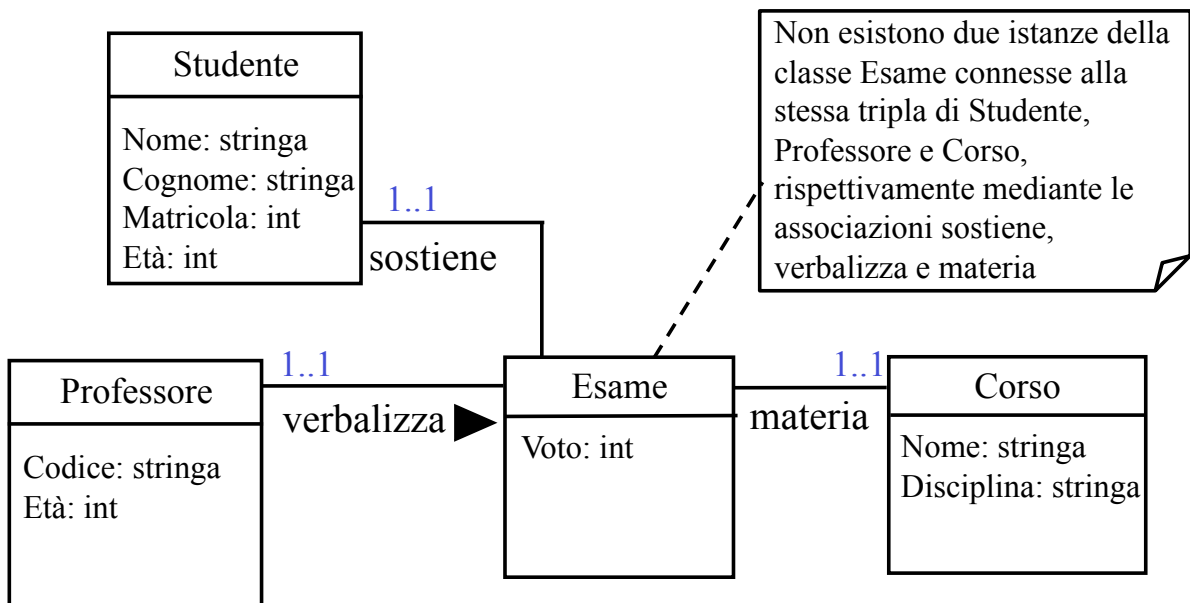
## Commenti in UML

In UML, quando si vuole specificare una caratteristica che non è possibile rappresentare esplicitamente nel diagramma con i meccanismi visti finora, si può usare la nozione di **commento**



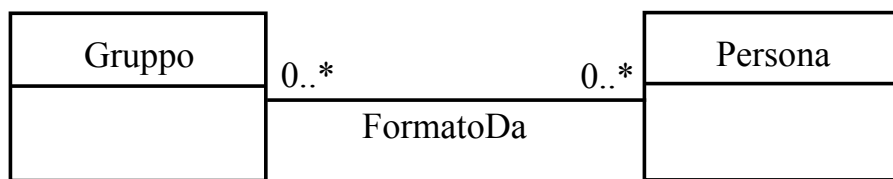
## Esempio di uso dei commenti

In questo modo, il diagramma modella il concetto di Esame in modo equivalente ad una associazione n-aria tra Studente, Professore e Corso



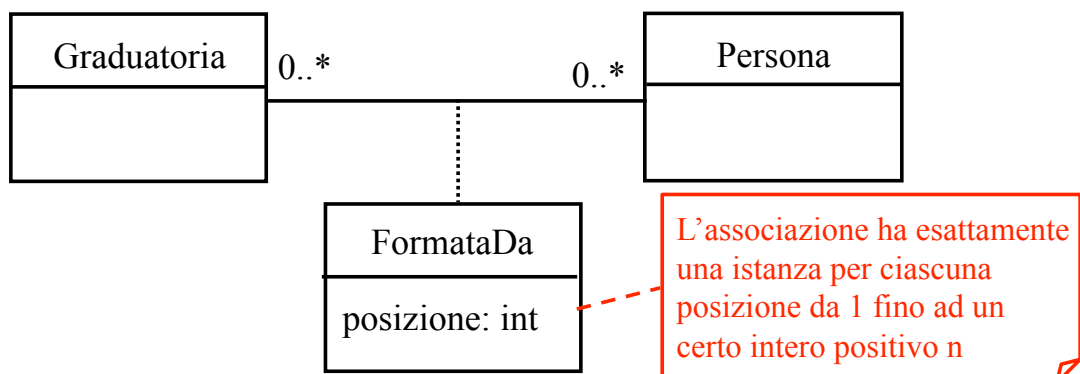
## Associazioni ordinate (0)

- Supponiamo di voler descrivere gruppi di persone ...
- Un gruppo è formato da persone. Ogni persona può apparire in un gruppo al più una volta (ovviamente ciascuna persona può fare parte di 0, 1, molti gruppi)
- In UML possiamo rappresentare questo scenario come segue:



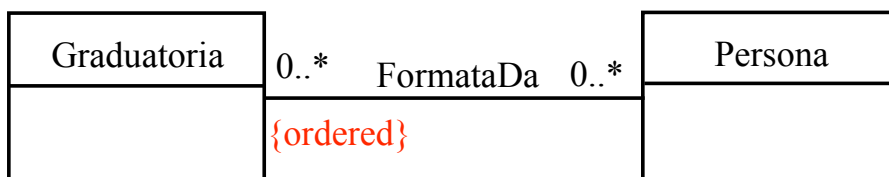
## Associazioni ordinate (1)

- Consideriamo ora invece una graduatoria di persone...
- Una graduatoria ha un certo numero di posizioni ciascuno occupato da una sola persona (*non consideriamo pari merito in questo esempio*) e una persona può apparire un una graduatoria al più una volta.
- Una possibile rappresentazione in UML è:



## Associazioni ordinate (2)

- La situazione descritta nell'esempio Graduatoria è molto comune, si pensi alla scaletta di un concerto, ad una presentazione formata da una sequenza di slide, ecc.
- L'attributo **posizione** serve **solo** a mantenere questo ordine (e per svolgere il suo lavoro deve sempre rispettare il vincolo del commento)
- In UML si può semplificare la descrizione utilizzando l'asserzione **{ordered}**



- **{ordered}** posto vicino a Graduatoria dice che data una istanza *g* di Graduatoria le istanze della associazione FormataDa che coinvolgono *g* sono ordinate (senza menzionare quale attributo utilizziamo per mantenere l'ordine).

## Associazioni ordinate (3)

- La soluzione con **{ordered}** è da preferire alla soluzione con un attributo esplicito “posizione” perchè
  - è più semplice (non fa uso di vincoli esterni -espressi nei commenti) ed è quindi **più leggibile**
  - **astrae** da come verrà mantenuta l'informazione sull'**ordine** evitando di introdurre uno specifico attributo (“posizione”) necessario a questo scopo

## Esercizio

Rappresentare in un diagramma delle classi UML playlist costituite da un nome (una stringa) e da un elenco di brani eventualmente ripetuti. Ciascun brano è caratterizzato dal nome (una stringa), la durata (un reale) e il nome del file (una stringa)

## Soluzione

