

Progettazione del Software

Emiliano Casalicchio

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

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

La fase di analisi

- I Class e Object diagram: definizione e utilizzo

Intermezzo: relazione matematica

- Se S_1 e S_2 sono due insiemi, una relazione R tra S_1 e S_2 è un **sottoinsieme del prodotto cartesiano** tra S_1 e S_2

$$R \subseteq S_1 \times S_2$$

- Il **prodotto cartesiano**

$$S_1 \times S_2$$

tra S_1 e S_2 è l'insieme di tutte le coppie $\langle x, y \rangle$ tali che $x \in S_1$ e $y \in S_2$

- Ovviamente, la nozione si estende banalmente a relazioni tra n insiemi: $R \subseteq S_1 \times S_2 \times \cdots \times S_n$

Intermezzo: relazione matematica

- Consideriamo gli insiemi $S_1 = \{1, 2, 3\}$ $S_2 = \{a, b\}$

- **Prodotto cartesiano:**

$$S_1 \times S_2 = \{ \langle 1, a \rangle, \langle 1, b \rangle, \langle 2, a \rangle, \langle 2, b \rangle, \langle 3, a \rangle, \langle 3, b \rangle \}$$

- **Esempio** di relazione tra S_1 e S_2 :

$$R \subseteq S_1 \times S_2 = \{ \langle 1, a \rangle, \langle 1, b \rangle, \langle 2, a \rangle \}$$

- Si noti come una relazione R selezioni un sottoinsieme degli elementi del prodotto cartesiano, quelli che sono significativi rispetto al significato della relazione R

Intermezzo: relazione matematica

- Consideriamo gli insiemi
 $\text{studente} = \{\text{Paolo, Anna, Lucia}\}$
 $\text{esame} = \{\text{Analisi, Geometria}\}$
- Esempio di relazione “sostenuto” tra “studente” ed “esame”:
 $\text{sostenuto} = \{ \langle \text{Paolo, Analisi} \rangle, \langle \text{Anna, Analisi} \rangle, \langle \text{Anna, Geometria} \rangle \}$
- Si noti come, tra tutte le coppie del prodotto cartesiano tra “studente” ed “esame”, la relazione “sostenuto” seleziona un insieme di coppie, e cioè solo le coppie $\langle x, y \rangle$ tali che lo studente x ha sostenuto l’esame y

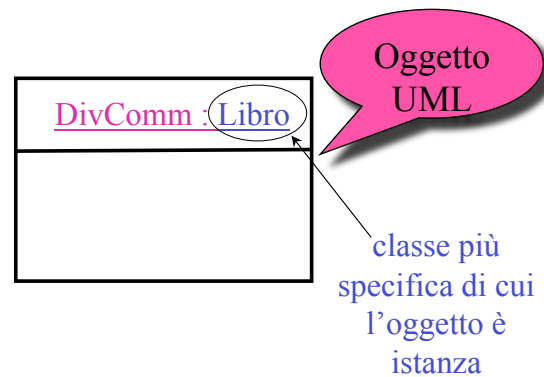
Diagramma delle classi e degli oggetti per l’analisi

- Nella fase di analisi ci si concentra sulle **classi** più che sugli oggetti
- Gli oggetti servono essenzialmente per descrivere elementi singoli particolarmente significativi (oltre che per scopi didattici)

Oggetti in UML

- Un **oggetto** in UML modella un elemento del dominio di analisi che
 - ha vita propria
 - è identificato univocamente mediante l'**identificatore** di oggetto
 - è istanza di una classe (la cosiddetta **classe più specifica** – vedremo che, in determinate circostanze, un oggetto è istanza di più classi, ma in ogni caso, tra le classi di cui un oggetto è istanza, esiste sempre la classe più specifica)

- DivComm è l'identificatore di oggetto
- Libro è la classe (più specifica) di cui l'oggetto è istanza
- Si noti la sottolineatura

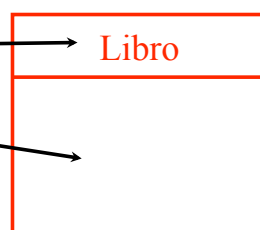


Classi in UML

- Una **classe** modella un insieme di oggetti omogenei (le **istanze** della classe) ai quali sono associate proprietà statiche e dinamiche (operazioni). Ogni **classe** e' descritta da:
 - un **nome**
 - un **insieme di proprietà "locali"** (astrazioni delle proprietà comuni degli oggetti che sono istanze delle classi)
- In realtà la visione di una classe come un insieme è un pò semplicistica; vedremo infatti che associata agli oggetti di una classe c'è anche una collezione di **operazioni**

nome della classe

le proprietà locali della classe sono descritte qui



Rapporto tra classi e istanze

- Tra un oggetto che è istanza di una classe C e la classe C si traccia un arco **Instance-of** (l'arco in realtà non è strettamente necessario, perchè la classe di cui l'oggetto è istanza è già indicata nell'oggetto)
- Ricordiamo che gli oggetti formano il livello **estensionale**, mentre le classi a livello **intensionale**

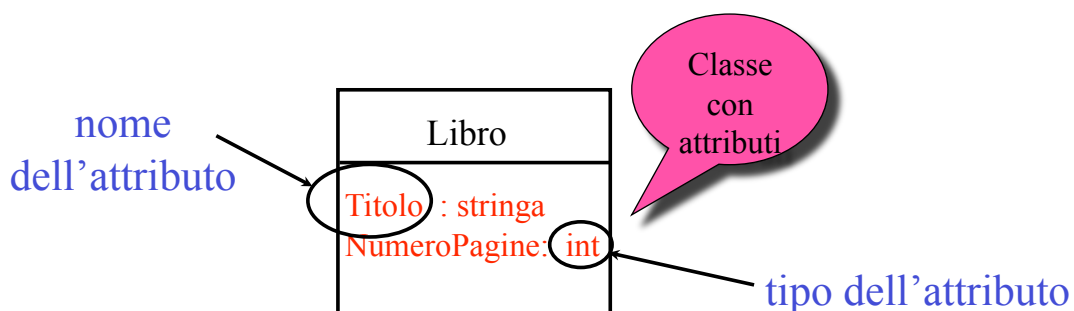


E.Casalicchio -- Progettazione del SW a.a. 2009/108/2009

9

Proprietà di classi: attributi in UML

- Un **attributo** modella una proprietà **locale** della classe ed è caratterizzato da un nome e dal tipo dei valori associati
- Ogni attributo di una classe stabilisce una proprietà locale **valida per tutte le istanze** della classe. Il fatto che la proprietà sia locale significa che è un proprietà **indipendente da altri oggetti**
- Formalmente, un attributo A della classe C si può considerare una **funzione** che associa un valore di tipo T ad ogni oggetto che è istanza di C

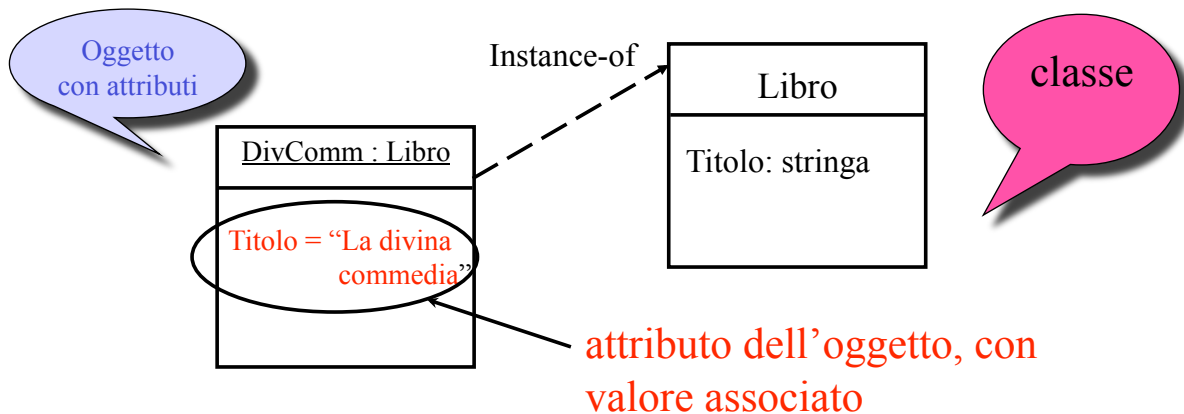


E.Casalicchio -- Progettazione del SW a.a. 2009/108/2009

10

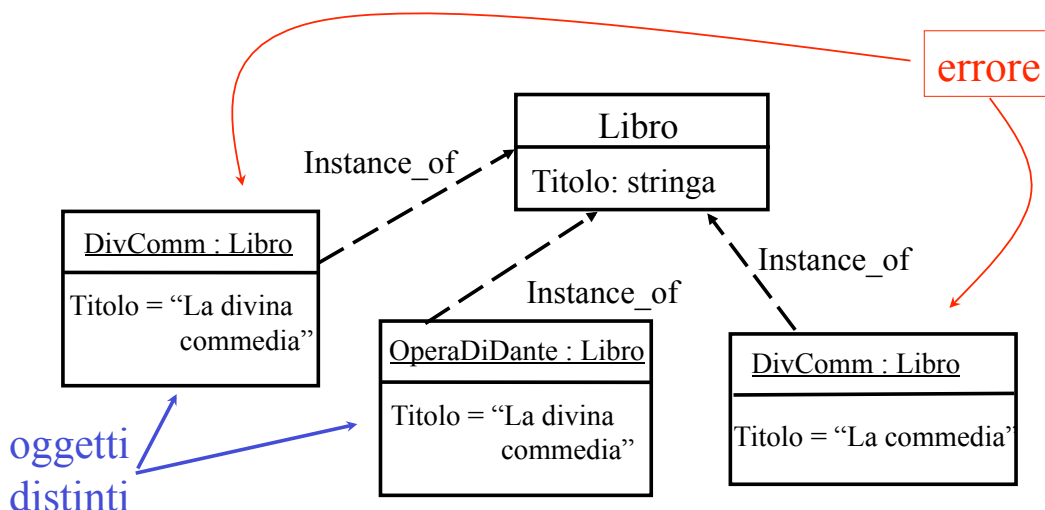
Attributi di oggetti

- Gli attributi di una classe determinano gli attributi delle sue istanze
- **Regola importante:** se una classe C ha un attributo A di tipo T , **ogni** oggetto che è istanza di C ha l'attributo A , con un valore associato di tipo T
- **Regola importante:** un oggetto X non può avere un valore per un attributo non definito nella classe di cui X è istanza

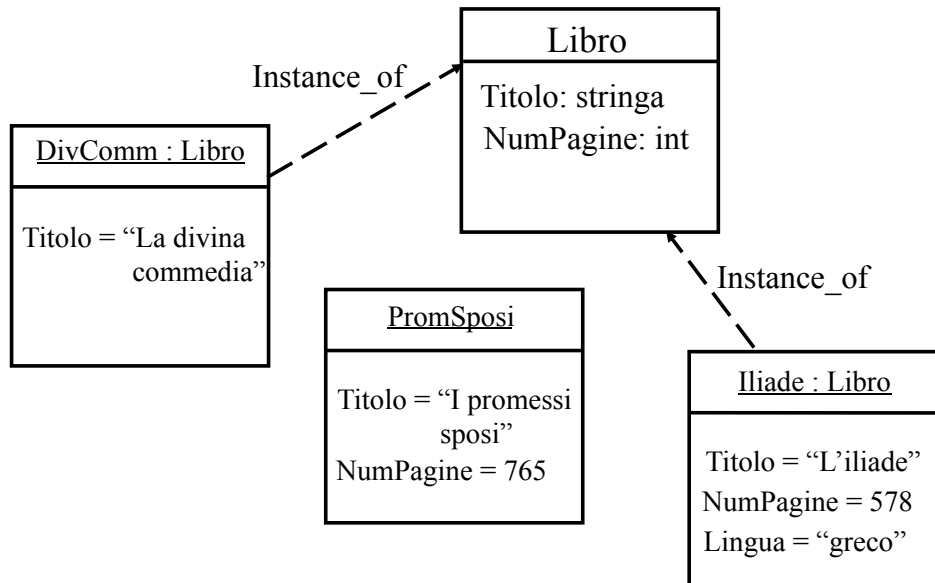


Importanza dell'identificatore di oggetto

- Due oggetti con identificatori **diversi** sono comunque **distinti**, *anche se hanno i valori di tutti gli attributi uguali*
- Due oggetti **diversi** devono avere **identificatori diversi**, anche se possono avere gli stessi valori per tutti gli attributi



Esercizio 1



Il diagramma è corretto? Se no, quali sono gli errori?

Errori dell'esercizio 1

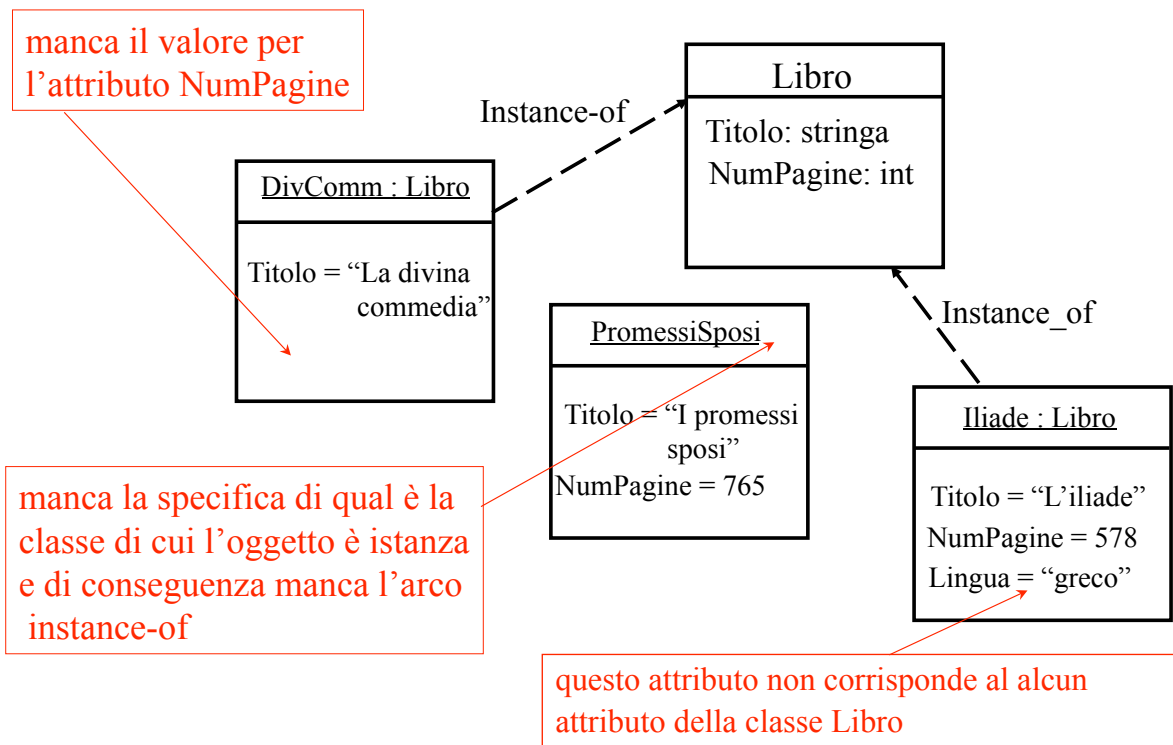
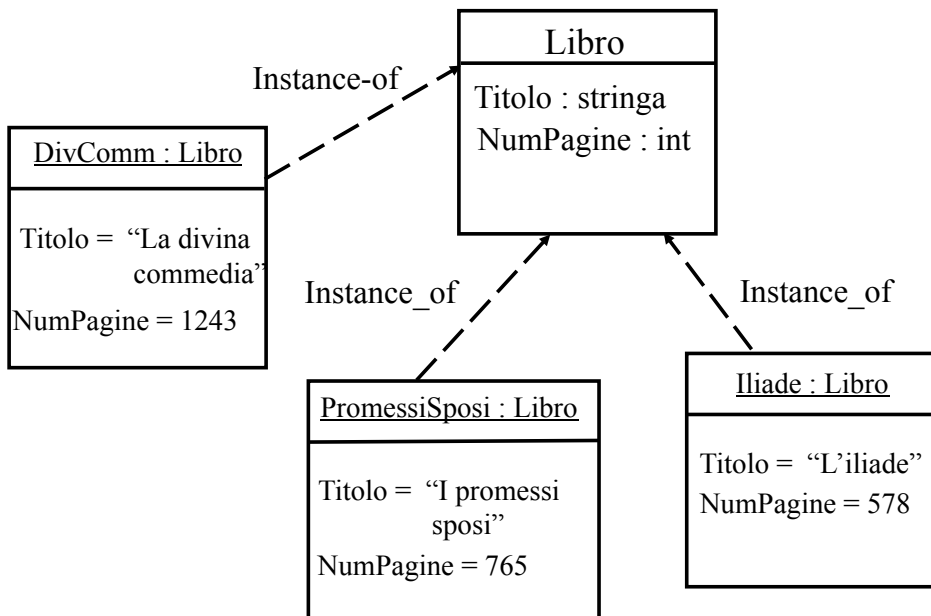


Diagramma corretto per l'esercizio 1



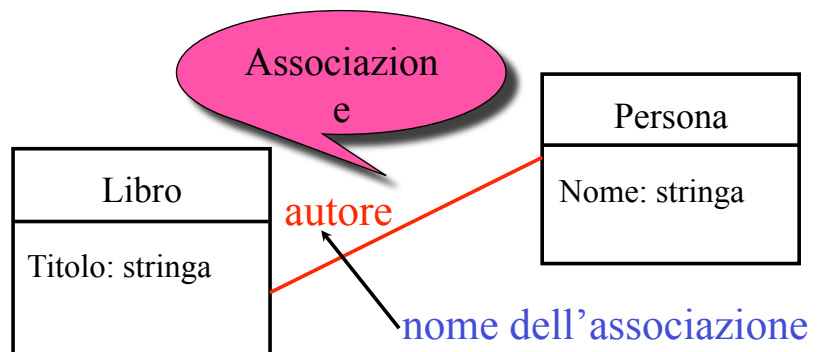
Osservazioni sulle classi

- Per tutto ciò che è stato detto finora, due classi diverse non possono avere istanze comuni. In altre parole, **classi diverse modellano insiemi disgiunti** (*torneremo su questo punto quando introdurremo la **generalizzazione***)
- Si noti la **distinzione tra oggetti** (istanze di classi) e **valori** (di un certo tipo):
 - un oggetto è istanza di una **classe** ed ha vita propria
 - un valore è un elemento di un **tipo**, ed ha senso solo se associato ad un oggetto tramite un attributo
- Il livello intensionale **determina** come è strutturato il livello estensionale

Proprietà di classi: associazioni in UML

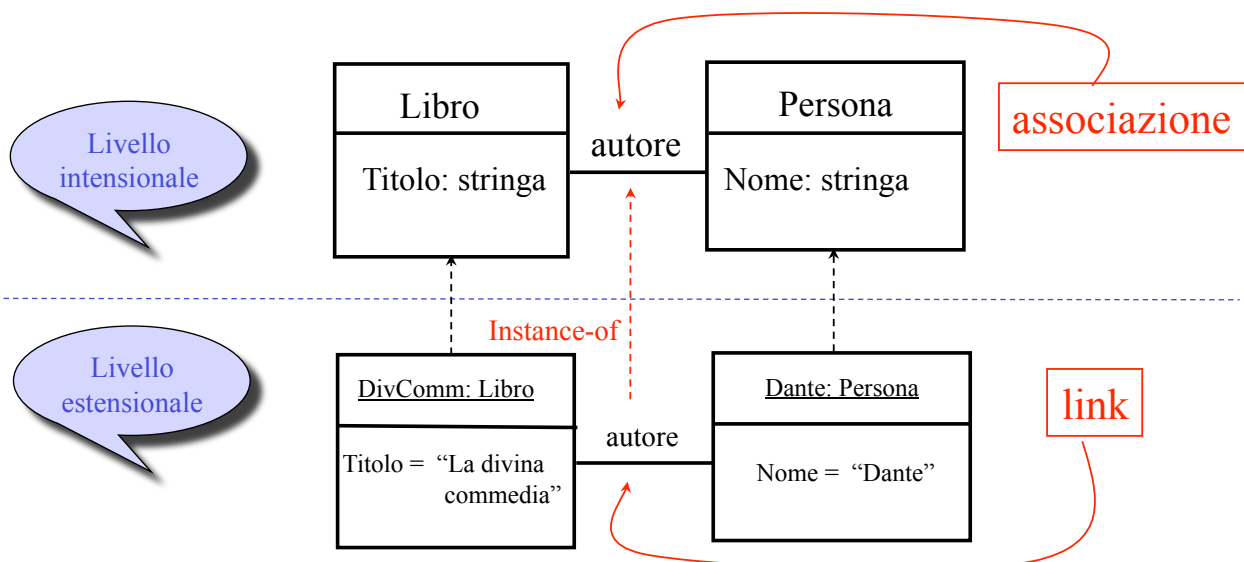
- Per il momento, ci limitiamo a discutere associazioni tra **due** classi (ma le associazioni possono coinvolgere N classi)
- Una **associazione** (o relazione) tra una classe C_1 ed una classe C_2 modella una relazione matematica tra l'insieme delle istanze di C_1 e l'insieme delle istanze di C_2
- Gli attributi modellano proprietà locali di una classe, le associazioni modellano **proprietà che coinvolgono altre classi**. Una associazione tra due classi modella una proprietà di **entrambe le classi**

Nota: “autore” è una proprietà sia di libro sia di persona



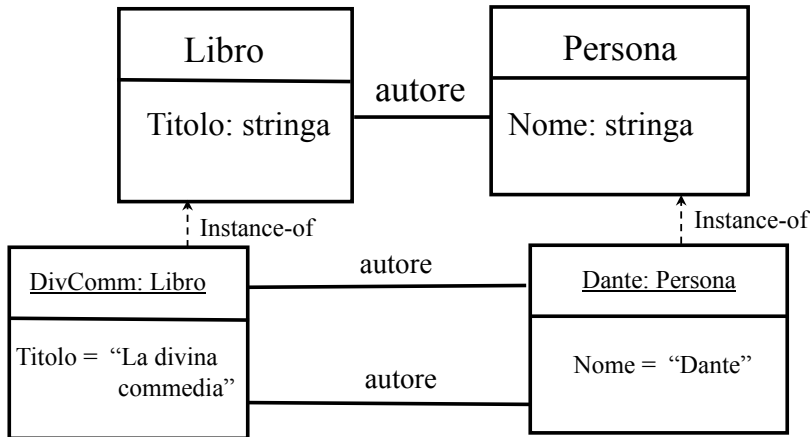
Istanze di associazioni: link

- Le istanze di associazioni si chiamano **link**: se A è una associazione tra le classi C_1 e C_2 , una istanza di A è un link tra due oggetti (in altre parole, una **coppia**), uno della classe C_1 e l'altro della classe C_2



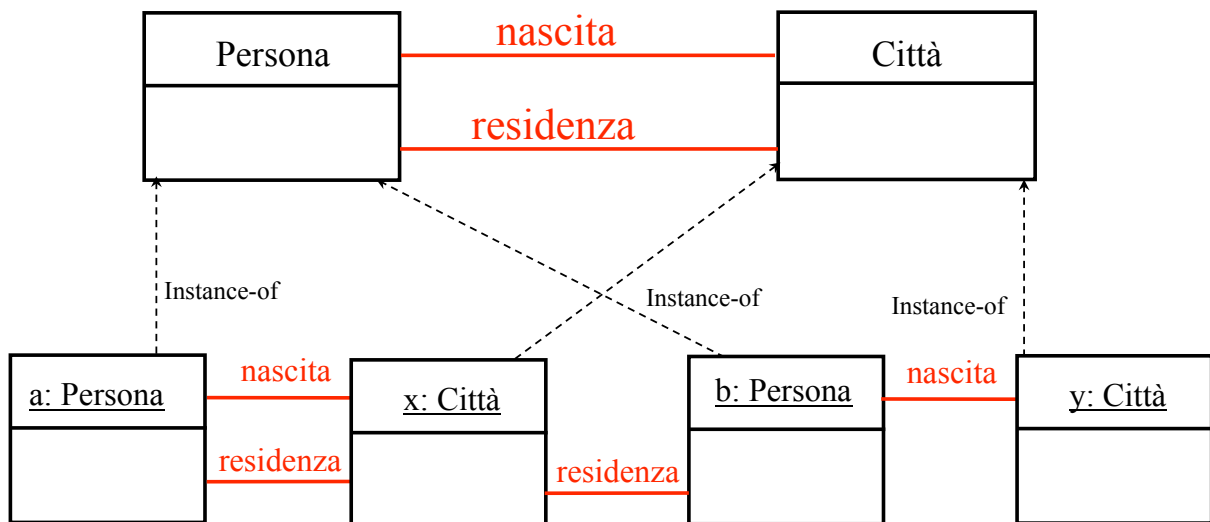
Istanze di associazioni: link

- Come gli oggetti sono istanze delle classi, così i link sono **istanze delle associazioni** (gli archi instance-of non sono necessari)
- Al contrario degli oggetti, però, i link non hanno identificatori espliciti: **un link è implicitamente identificato dalla coppia (o in generale dalla ennupla) di oggetti che esso rappresenta**
- Ciò implica, ad esempio, che il seguente diagramma è **errato**:



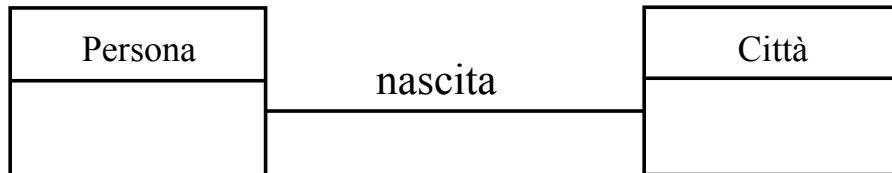
Più associazioni tra due classi

- Ovviamente, tra le stesse due classi possono essere definite più associazioni



Ancora sulle associazioni in UML

- Attenzione: una relazione R tra C_1 e C_2 non dice nulla sul numero di link di R che coinvolgono due istanze delle classi C_1 e C_2 . Ad esempio, dato questo diagramma:



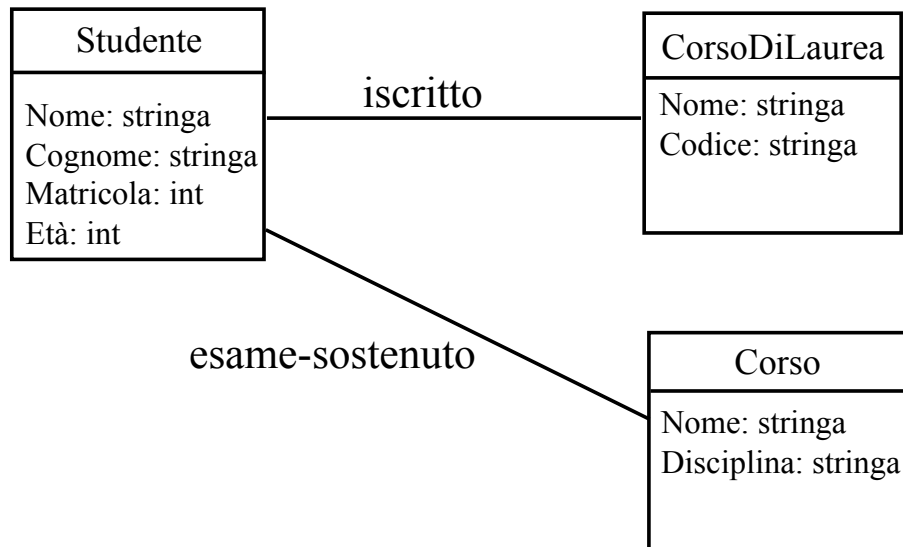
- Una istanza di Persona può essere legata a **zero, una, o più** istanze di Città da link di tipo “nascita”
- *Vedremo successivamente* come si possono specificare condizioni sul numero di link che coinvolgono un oggetto in UML (ad esempio per imporre che ogni istanza deve avere esattamente un link di tipo “nascita” con una istanza di Città)

Esempio

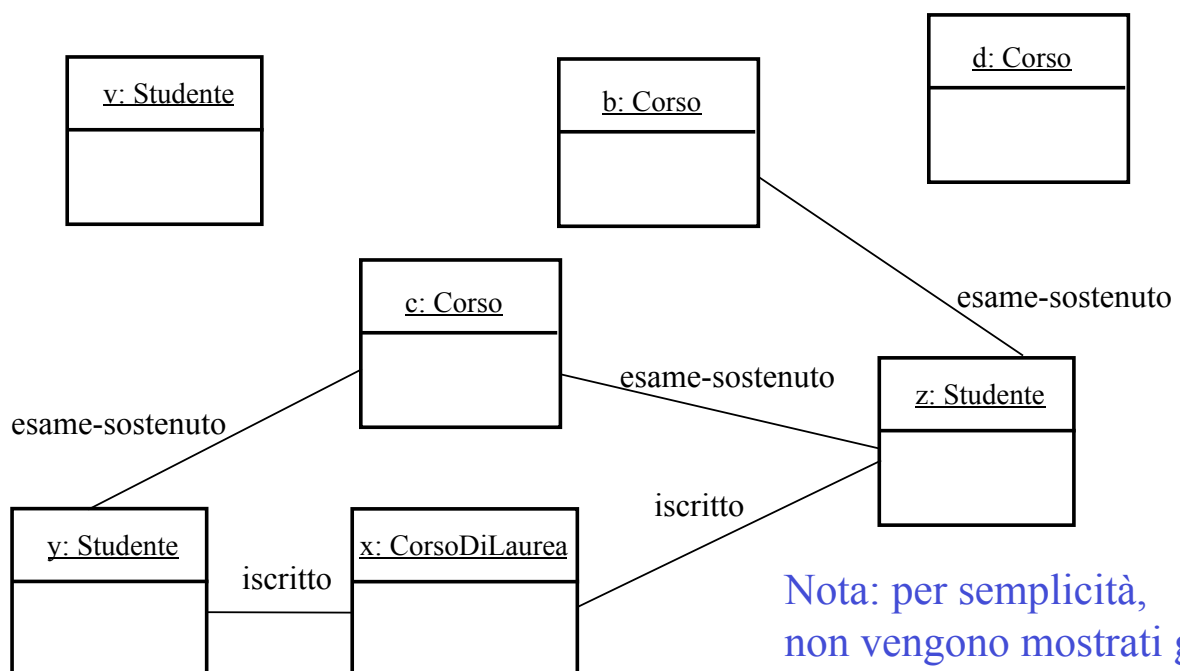
Tracciare il diagramma delle classi corrispondenti alle seguenti specifiche:

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. 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.).

Diagramma delle classi per l'esempio



Esempi di oggetti



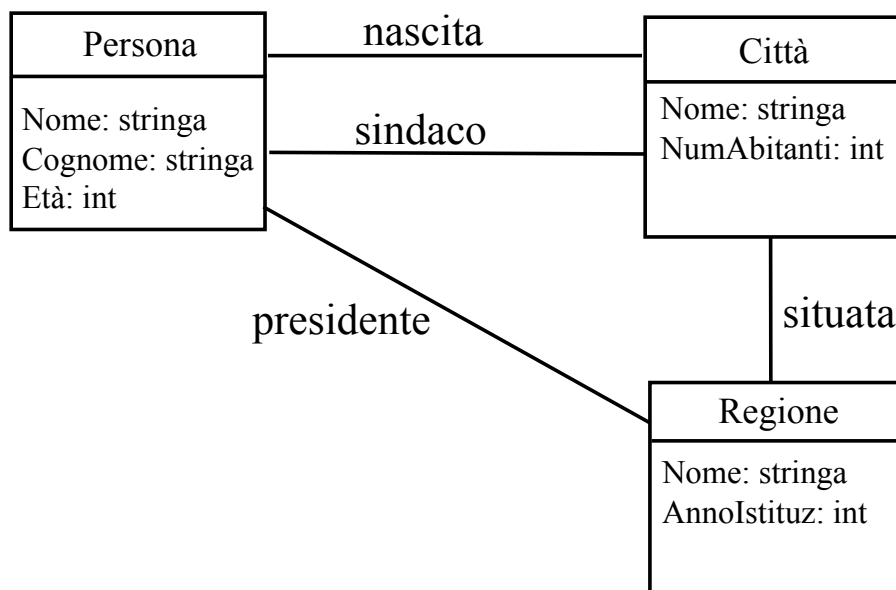
Nota: per semplicità, non vengono mostrati gli attributi ed i relativi valori

Esercizio 2

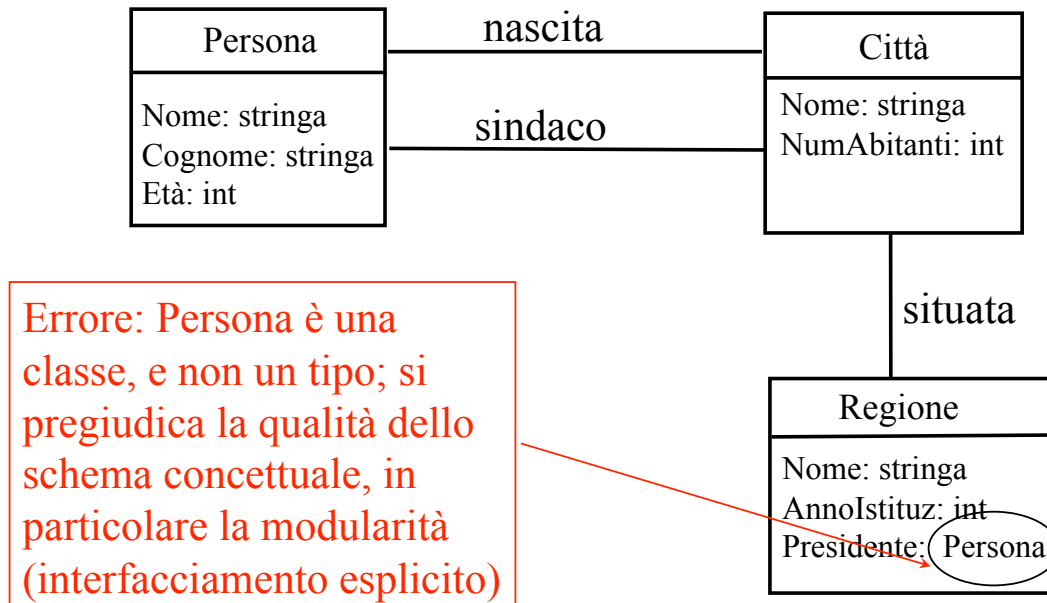
Tracciare il diagramma delle classi corrispondenti alle seguenti specifiche:

Si vogliono modellare le persone (con nome, cognome, età), le città di nascita (con nome, numero di abitanti, e sindaco), e le regioni in cui si trovano le città (con nome, anno di istituzione, e presidente).

Soluzione dell'esercizio 2



Possibile errore nell'esercizio 2



Possibile errore nell'esercizio 2

