

GERARCHIE DI GENERALIZZAZIONE

71

Gerarchie di generalizzazione

- Una gerarchia di generalizzazione è un legame logico tra un'entità padre E ed alcune entità figlie $E_1 E_2 \dots E_n$ dove:
 - E è la generalizzazione di $E_1 E_2 \dots E_n$
 - $E_1 E_2 \dots E_n$ sono specializzazioni di Etale per cui:
 - ogni istanza di E_k è anche istanza di E
 - una istanza di E può essere una istanza di E_k
- Le entità figlio ereditano le proprietà (attributi, relazioni, identificatori) dell'entità padre.

72

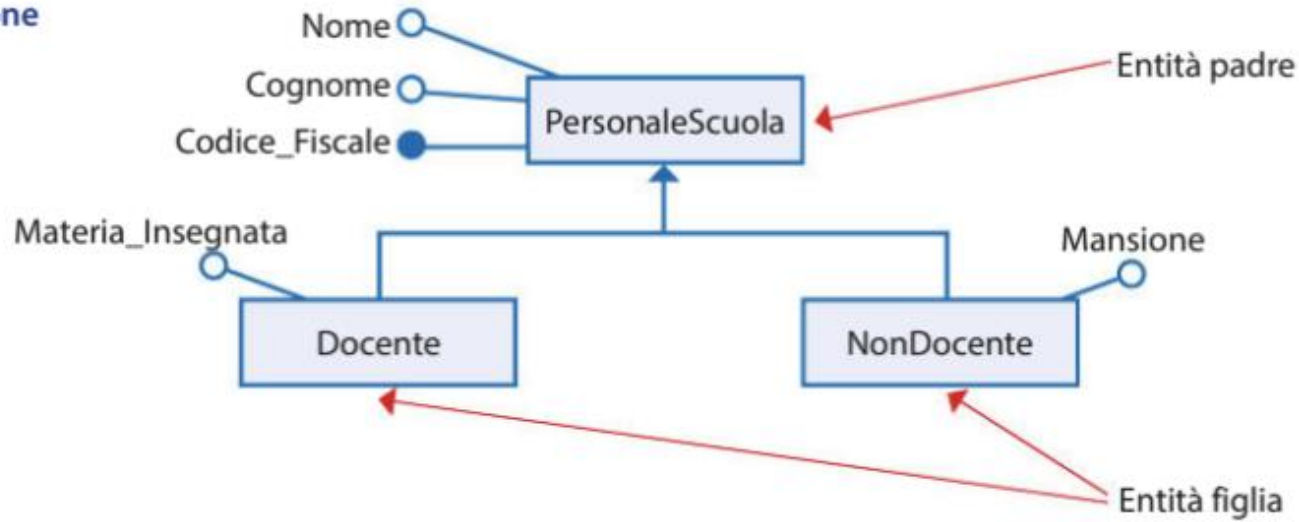
Quando alcune entità possono avere delle istanze in comune, oppure quando nasce l'esigenza di voler particolareggiare o generalizzare qualche entità, è **utile procedere con un'organizzazione per gerarchie**.

Una gerarchia ha una struttura ad albero: la radice rappresenta l'entità che contiene le caratteristiche comuni (**entità padre**), mentre i vari nodi (**entità figlie**) rappresentano le entità che contengono caratteristiche più specifiche.

Per creare una gerarchia si attua un **processo di specializzazione**, ossia si "specializza" l'entità padre creando entità figlie sempre più specifiche (aggiungendo agli attributi del padre altri attributi che le specializzano). Percorrendo la gerarchia dal basso verso l'alto si attua automaticamente un **processo di generalizzazione** in quanto si risale l'albero passando dalle entità più specializzate a quelle più generiche (si risale verso gli attributi comuni).

Ad esempio, l'entità *PersonaleScuola* può essere pensata come una generalizzazione delle entità *Docente* e *NonDocente*. In questo modo è possibile rappresentare il fatto che istanze di *Docente* e *NonDocente* possono anche essere classificate come istanze di *PersonaleScuola*. Questo approccio consente di prescindere dalle proprietà che le rendono semanticamente diverse (ad esempio *Materia_Insegnata* per i docenti e *Mansione* per i non docenti) ed evidenziare, invece, le proprietà comuni, ad esempio *Codice_Fiscale*, *Cognome*, *Nome*.

Specializzazione



Generalizzazione

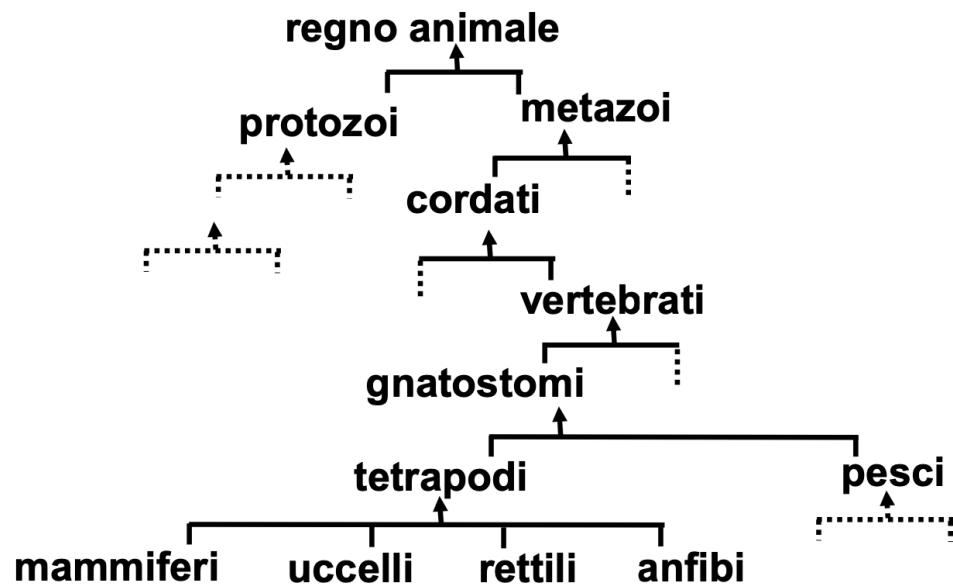
Rifletti

Nel modello ER un'associazione di generalizzazione è anche conosciuta con il termine **associazione ISA** (dall'inglese *is a*, cioè "è un").

Naturalmente il processo di specializzazione può proseguire, nel senso che una entità figlia può essere a sua volta una entità padre per altre entità modellando, così, gerarchie a più livelli. Ad esempio, l'entità figlia *NonDocente* potrebbe essere specializzata ulteriormente in *PersonaleSegreteria* e *CollaboratoreScolastico*.

Ogni istanza di un'entità figlia è anche istanza dell'entità padre. Per questo motivo, ogni entità figlia **eredita** tutte le proprietà (attributi) dell'entità padre. Ad esempio, nel nostro caso l'entità *NonDocente*, oltre all'attributo *Mansione*, possiede anche gli attributi *Cognome*, *Nome* e *Codice_Fiscale*.

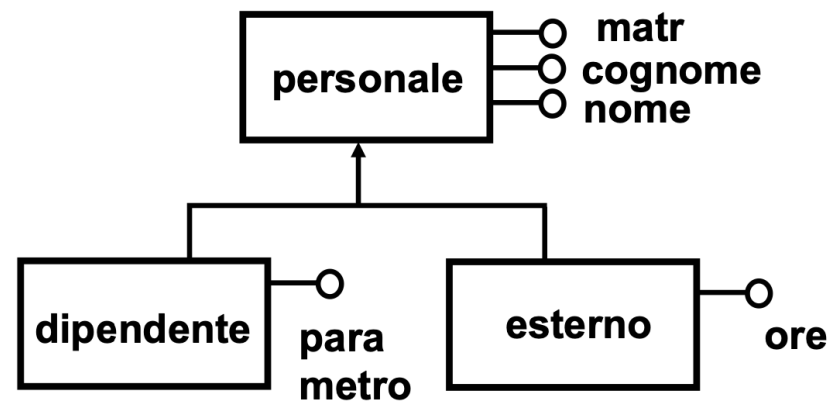
Una delle gerarchie più note



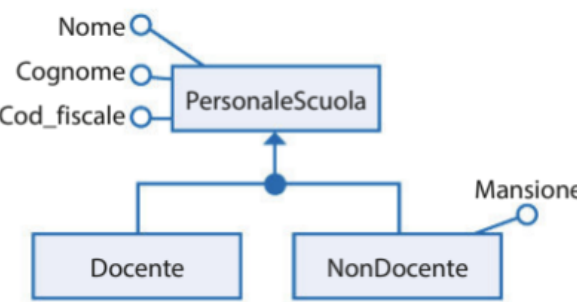
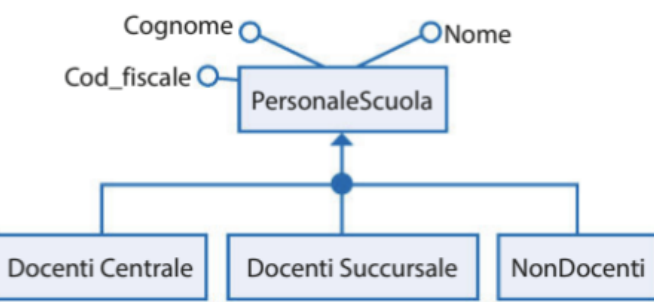
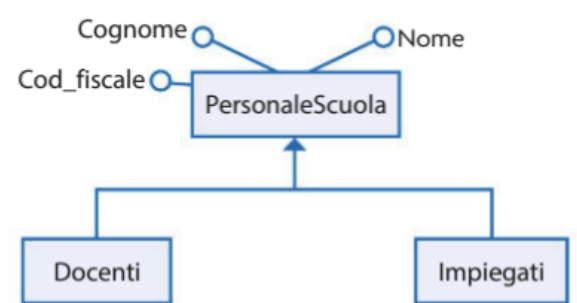
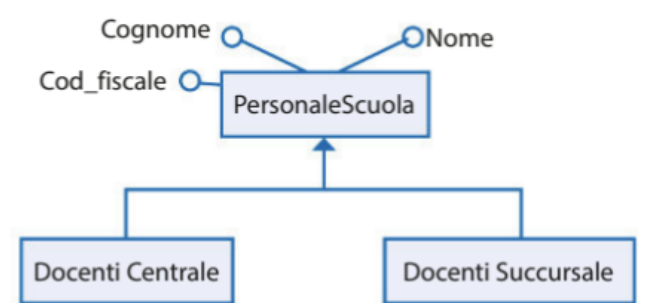
73

Esempio di gerarchia

un'azienda si avvale dell'opera di professionisti esterni, quindi il suo personale si suddivide in esterni e dipendenti:



74

		DISGIUNZIONE	
		Esclusiva	Sovrapposta
COPERTURA	Totale	 <p>Totale (i Docenti e i NonDocenti costituiscono tutto il personale della scuola). Esclusiva (una persona della scuola è un Docente o un NonDocente).</p>	 <p>Totale (l'unione delle tre figlie concorrono a formare tutte le istanze di PersonaleScuola). Sovrapposta (alcuni docenti possono insegnare sia nella sede Centrale sia in Succursale).</p>
	Parziale	 <p>Parziale (i docenti e gli impiegati non concorrono a formare tutto il personale della scuola). Esclusiva (il docente non può ricoprire il ruolo di impiegato e viceversa).</p>	 <p>Parziale (i soli docenti non concorrono a formare tutto il personale della scuola). Sovrapposta (alcuni docenti possono insegnare sia nella sede Centrale sia in Succursale).</p>

Le generalizzazioni possono essere classificate in base a due proprietà, **copertura** e **disgiunzione**, che sono ortogonali tra loro; quindi è possibile avere quattro configurazioni di generalizzazione:

- la generalizzazione ha copertura **totale** se ogni istanza della classe padre è una istanza di almeno una delle figlie, altrimenti è **parziale**;
- la generalizzazione ha disgiunzione **esclusiva** se ogni istanza della classe padre è una istanza di al più una delle classi figlie, altrimenti si dice **sovrapposta**.

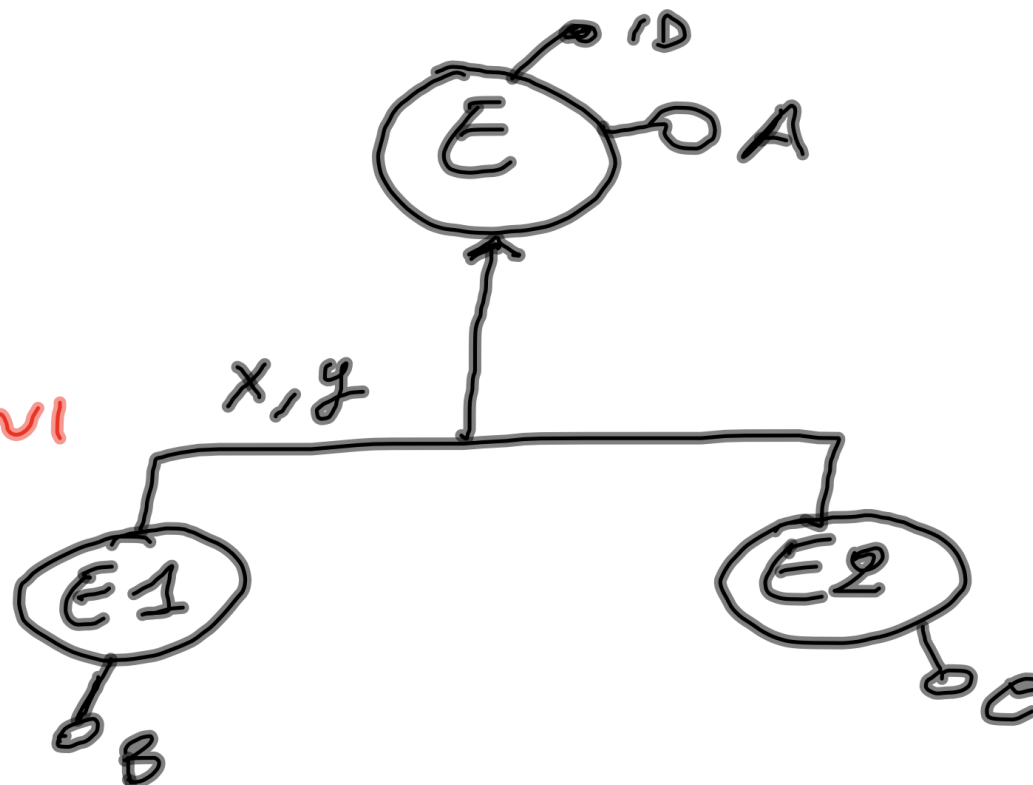
GERARCHIE: DEFINIZIONI

$x: \begin{cases} T \\ P \end{cases}$ TOTALE
PARZIALE

$y: \begin{cases} E \\ S \end{cases}$ ESCLUSIVA
SOVRAPPOSIZIONI

$\cdot \rightarrow T, E$ T, S

$\cdot \rightarrow P, E$ P, S



x

Proprieta' delle gerarchie

- ***t* sta per totale:** ogni istanza dell'entità padre deve far parte di una delle entità figlie
 - nell'esempio il personale si divide (completamente) in esterni e dipendenti
- ***p* sta per parziale:** le istanze dell'entità padre possono far parte di una delle entità figlie
 - nell'esempio i cacciatori e pescatori sono un sottoinsieme dei cittadini

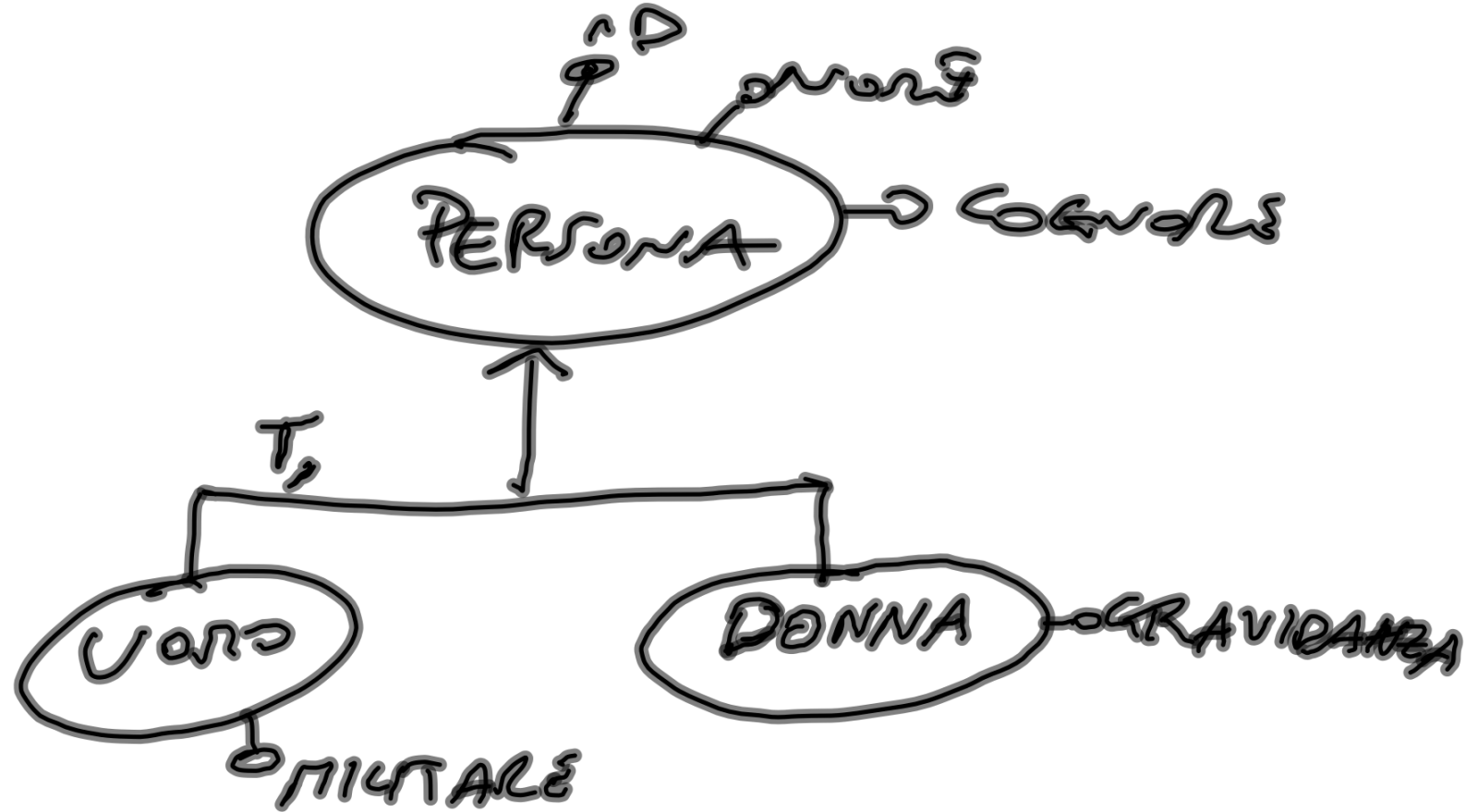
y

Proprieta' delle gerarchie

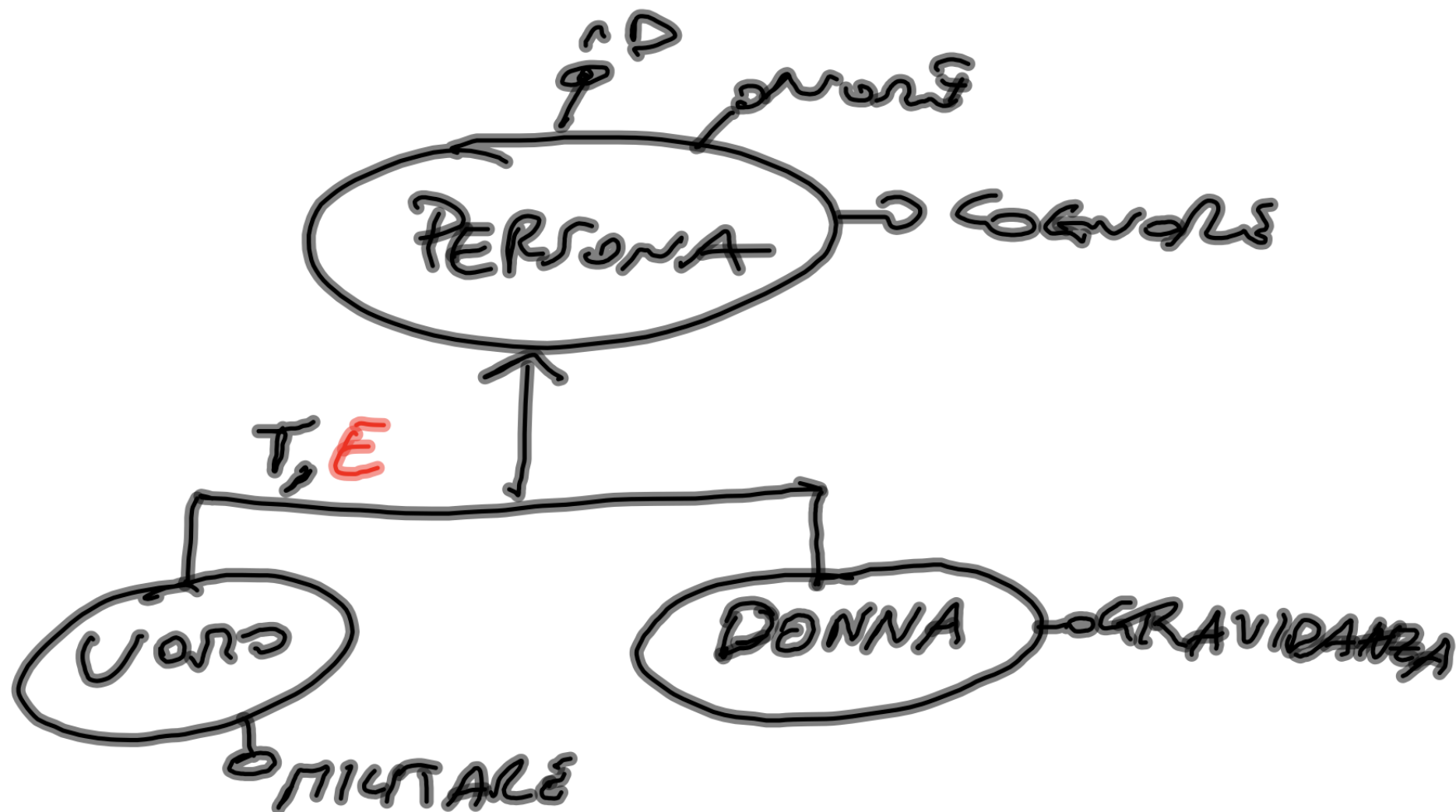
- ***e* sta per esclusiva:** ogni istanza dell'entità padre non può far parte di più di una delle entità figlie
 - nell'esempio si esclude che una istanza di personale possa appartenere ad entrambe le sottoclassi
- ***O* sta per overlapping:** ogni istanza dell'entità padre può far parte di più entità figlie
 - nell'esempio un cittadino può essere al tempo stesso cacciatore e pescatore

*si trova anche con la parola «sovrapposta» nei libri italiani

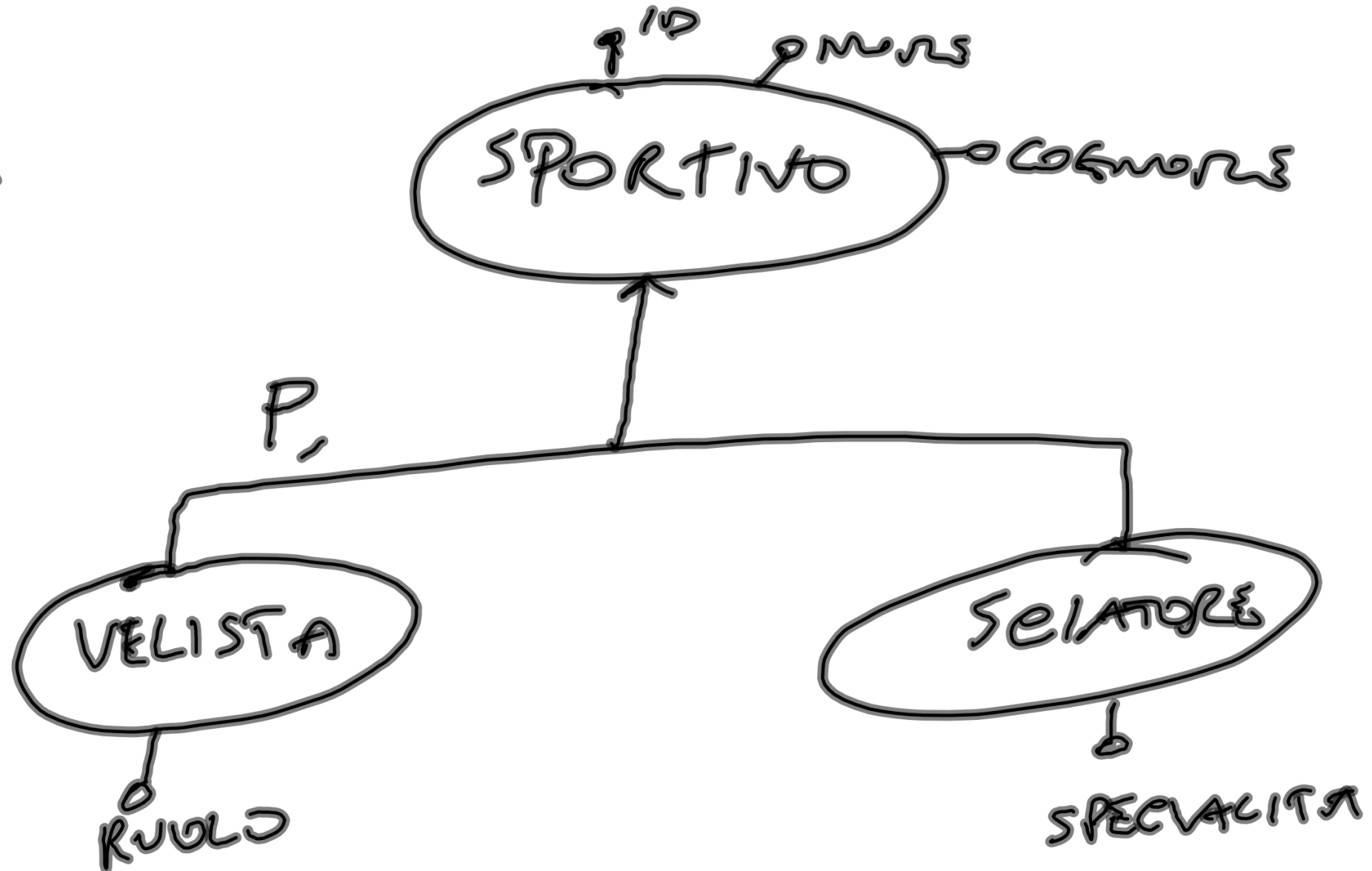
TOTALE : ogni istanza dell'entità padre deve far parte di una delle entità figlie



ESCLUSIVA: ogni istanza dell'entità padre deve
far parte di una sola delle entità figlie



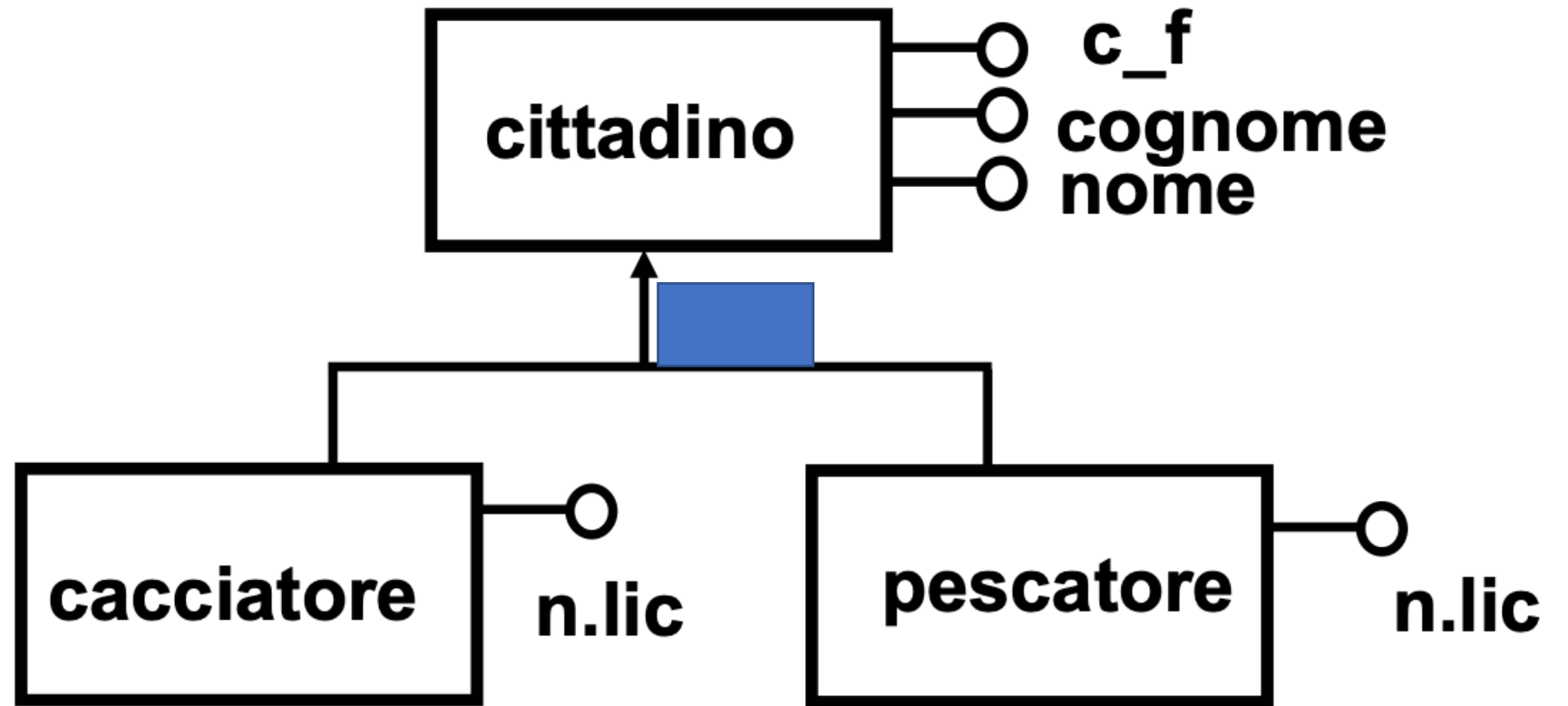
PARZIALE: le istanze dell'entità padre possono
far parte di una delle entità figlie



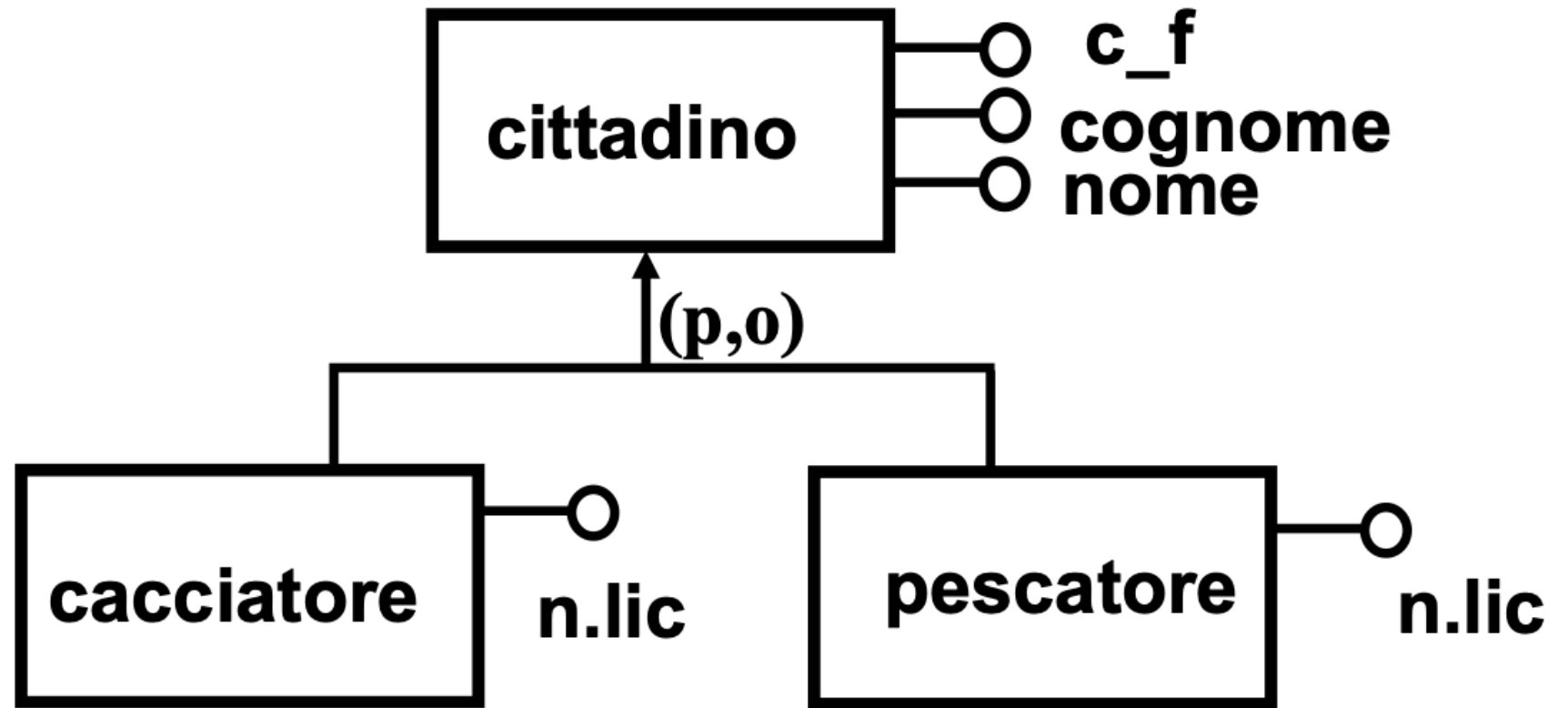
SOVRAPPOSIZIONI: ogni istanza dell'entità padre
può far parte di una o più entità figlie



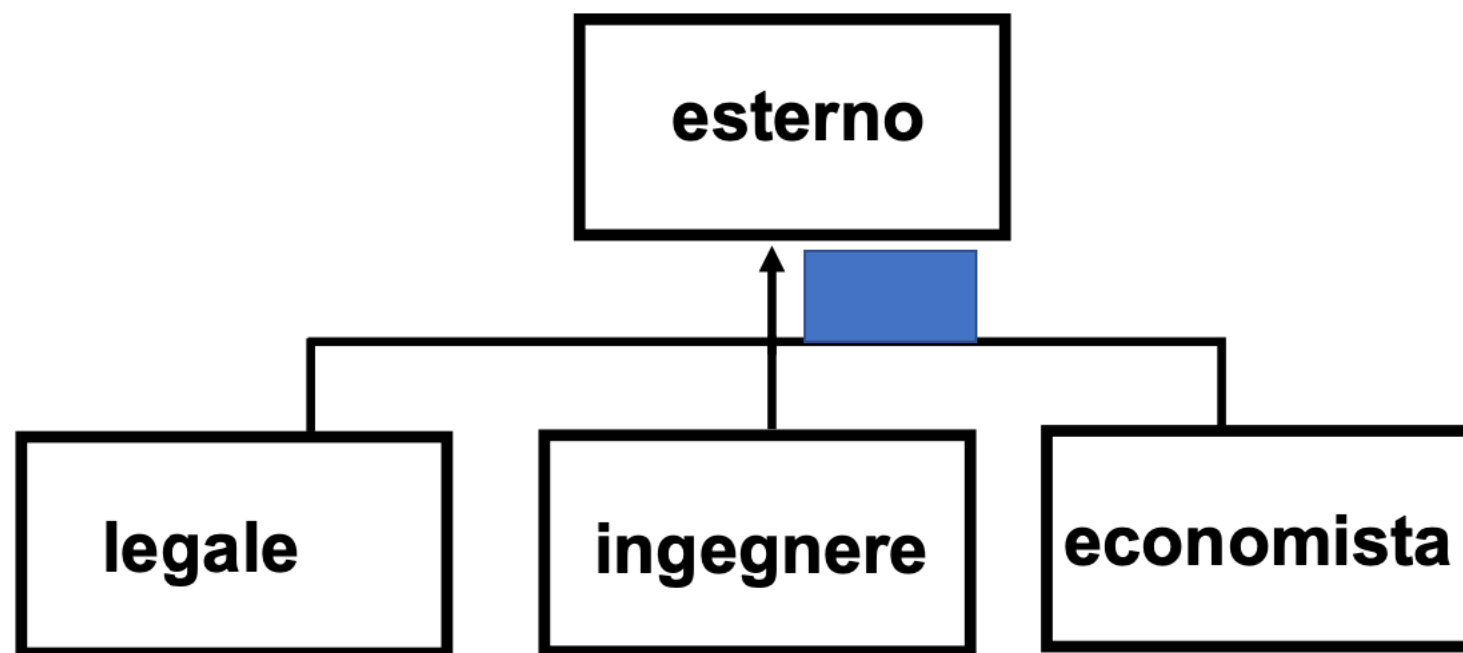
Prova tu



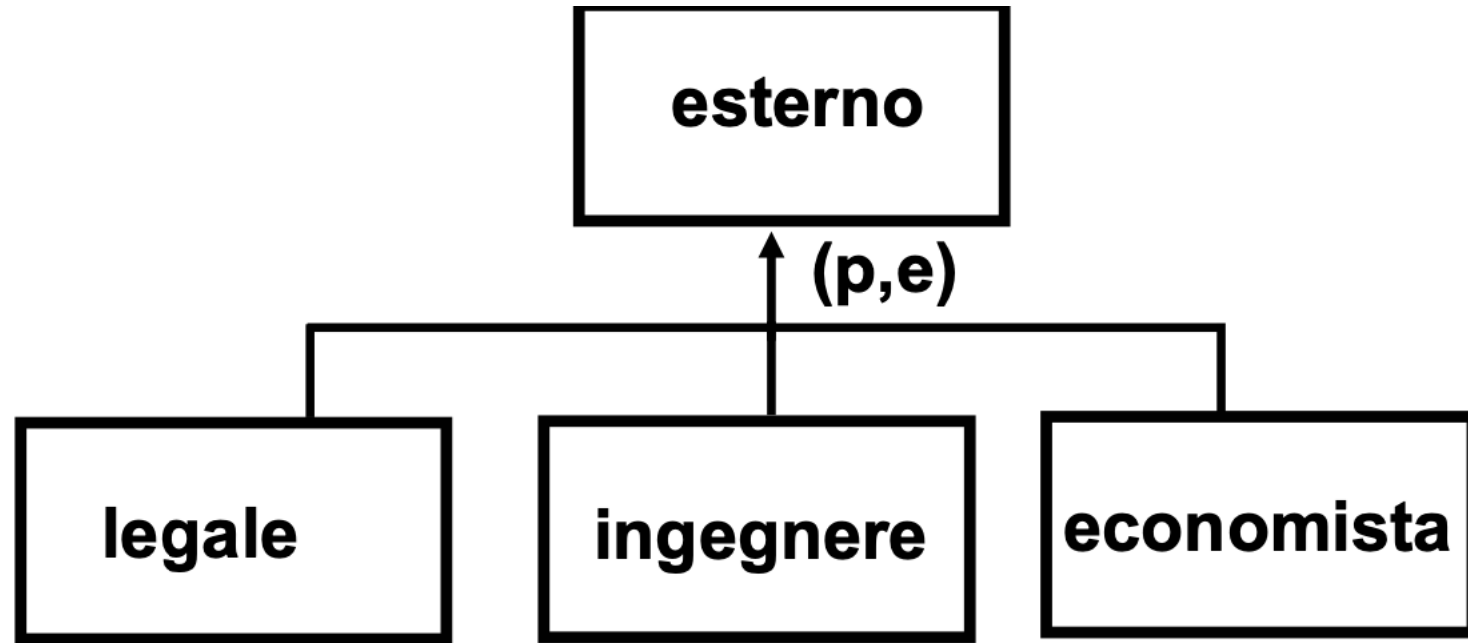
Prova tu



Prova tu

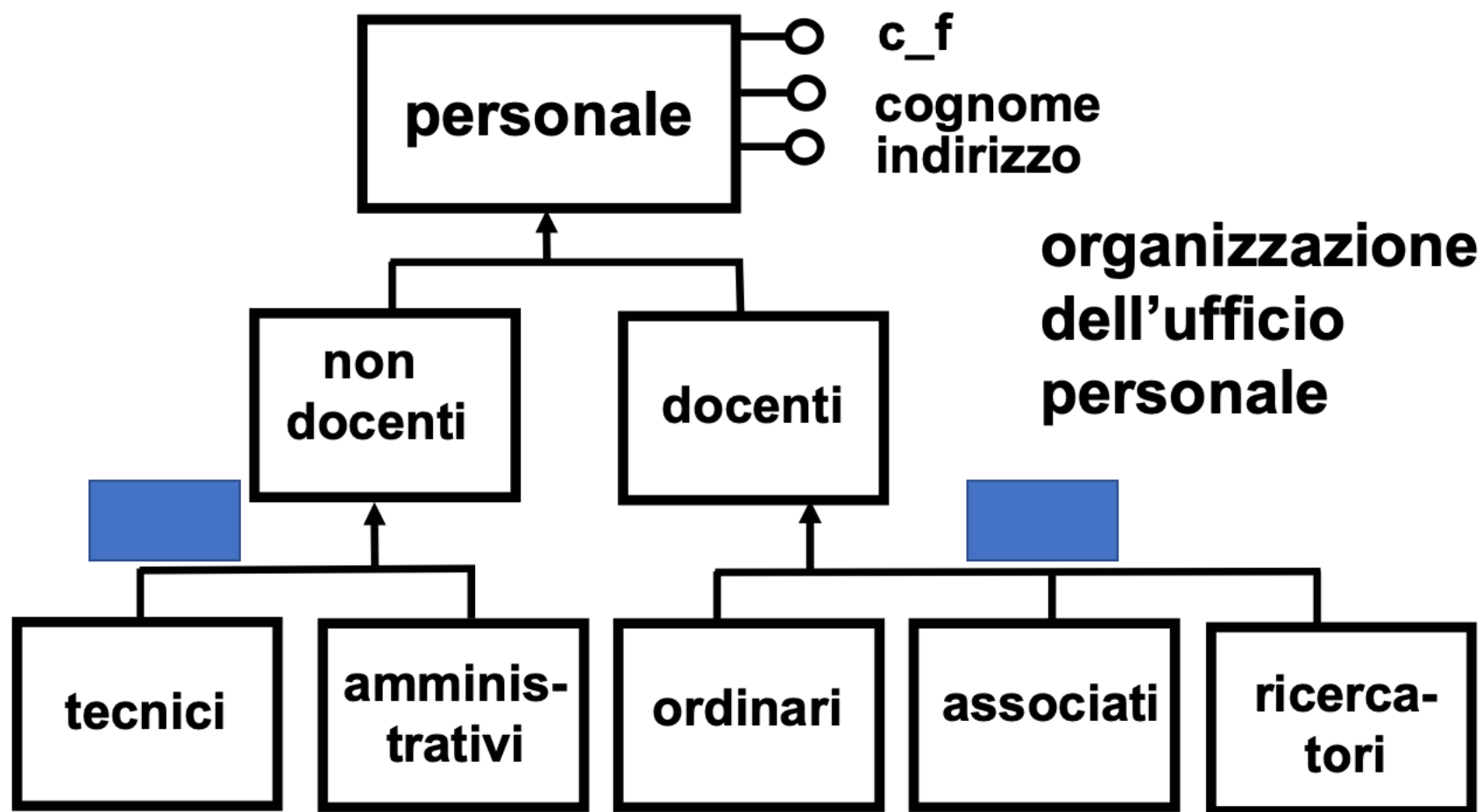


Prova tu

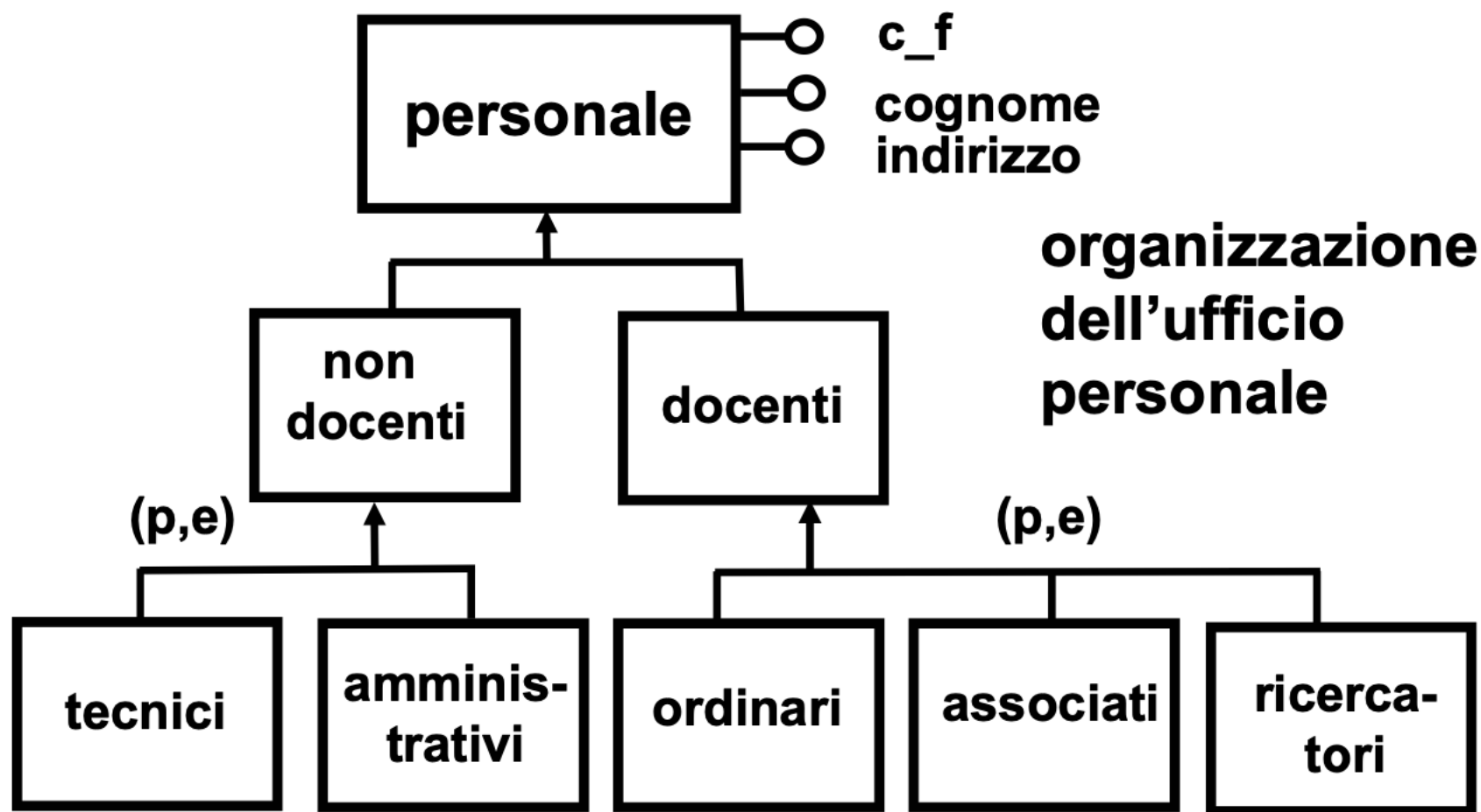


p: possono esistere esterni generici che non sono né legali, né ingegneri, né economisti ma non interessa stabilire una sottoclasse ad hoc

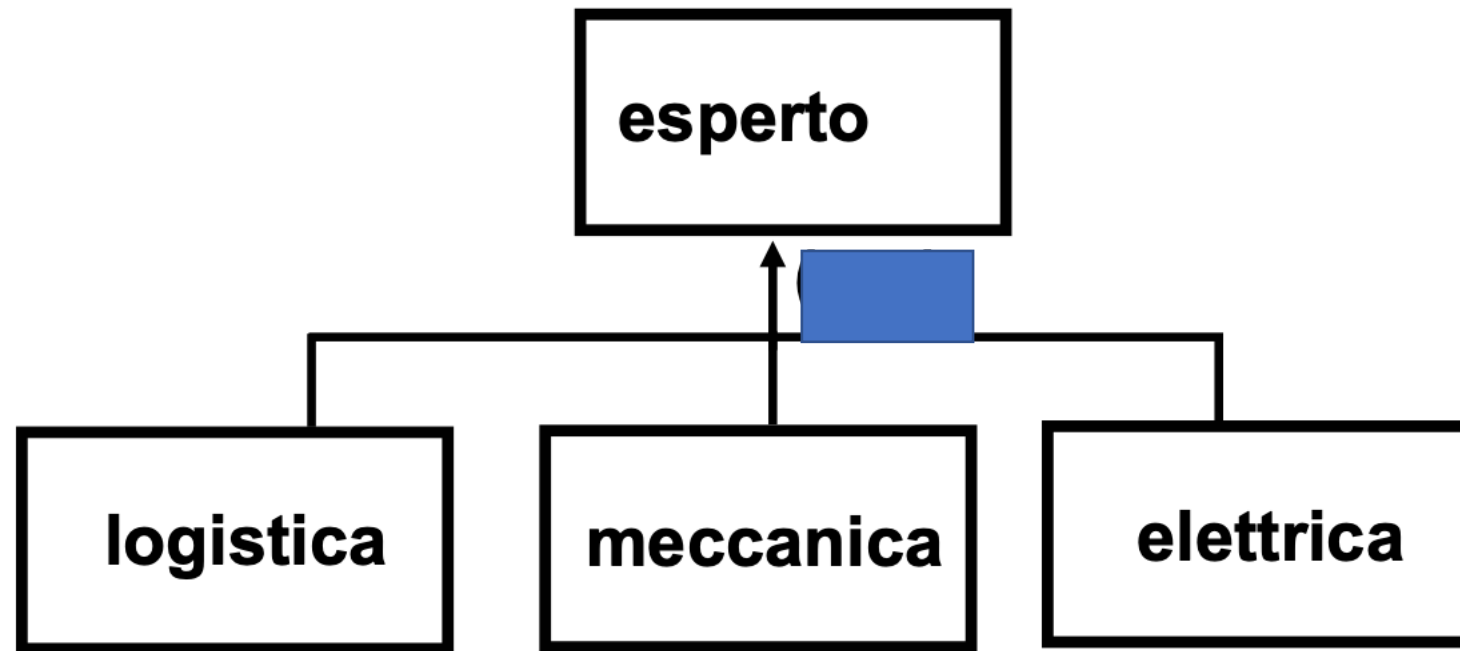
Prova tu



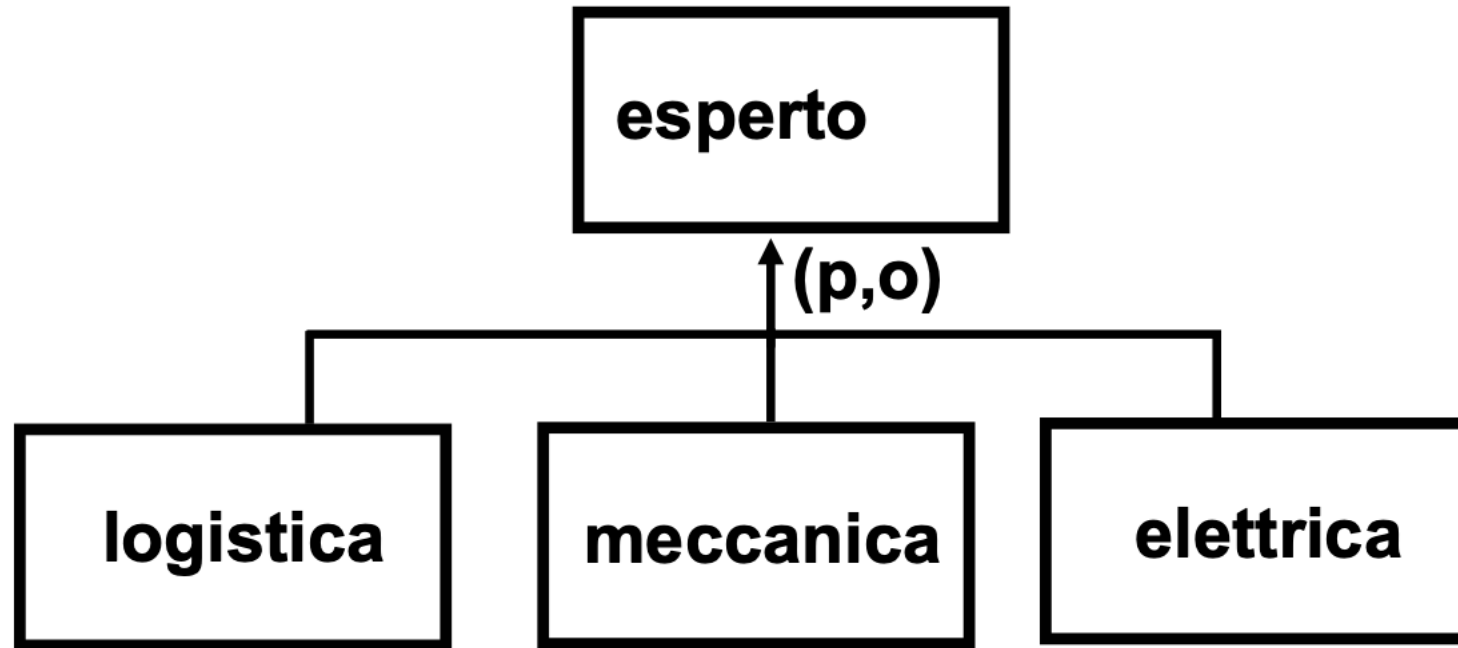
Prova tu



Prova tu



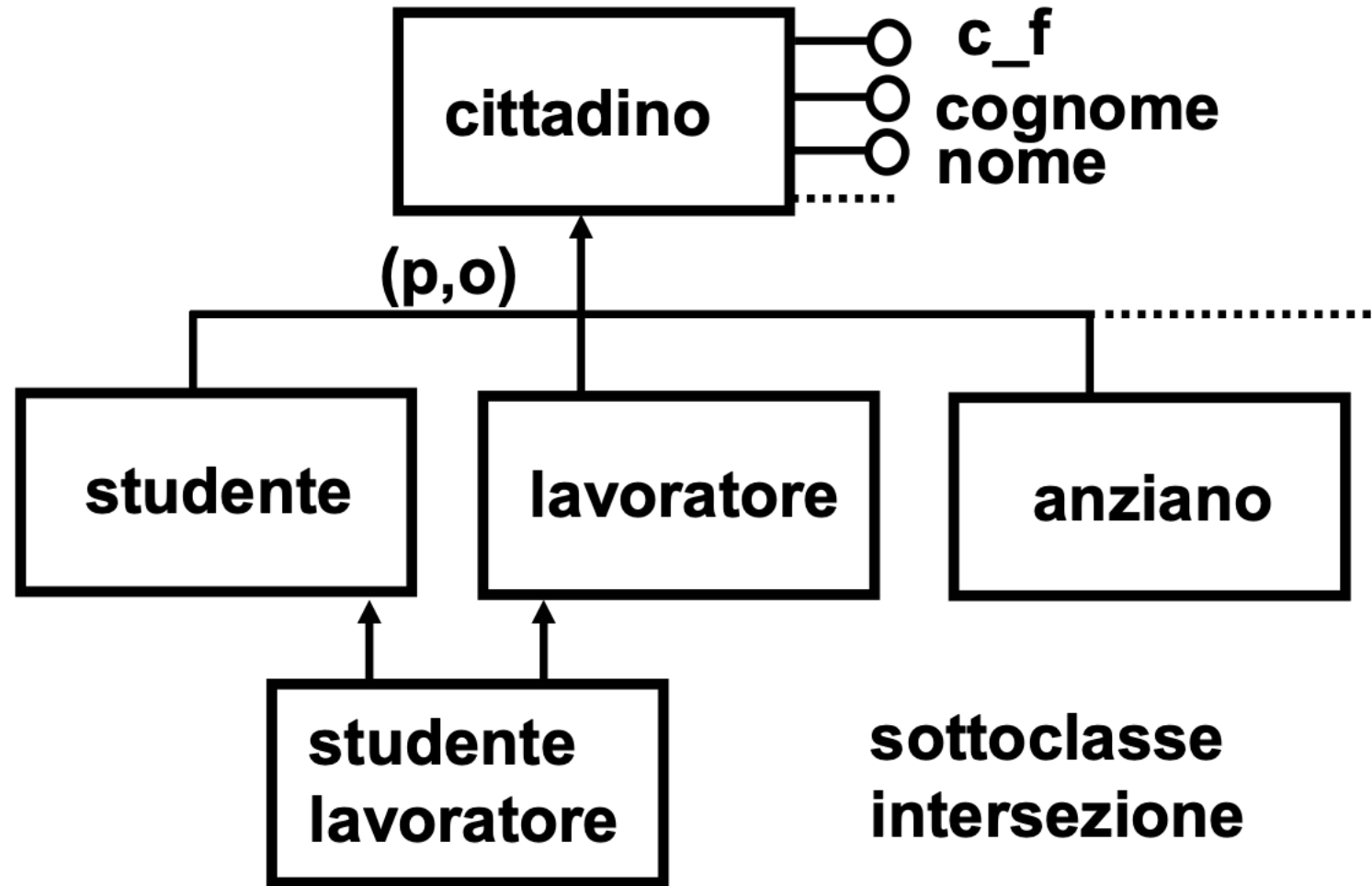
Prova tu



p : possono esistere esperti sia meccanici, sia elettrici, sia della logistica

O: le tre qualifiche non si escludono

Prova tu



Esercizio

Una concessionaria di automobili svolge operazioni di compravendita di auto nuove e usate. Sarà necessario raccogliere tutte le informazioni attinenti alle automobili. Le informazioni da registrare sono diverse in funzione del tipo di macchina; ad esempio, per le auto usate è necessario conoscere dati in merito alle riparazioni, mentre per le auto nuove è utile conoscere gli optional in dotazione.

Le auto sono caratterizzate da prezzi che variano in funzione dello stato dell'auto (per quelle usate) e della dotazione (per quelle nuove). Memorizzeremo, inoltre, anche le informazioni anagrafiche dei proprietari, supponendo che ogni automobile possa averne uno solo.

Per la riparazione delle automobili realizzeremo una lista di interventi, individuando per ognuno il livello di gravità. La lista sarà accompagnata anche da un preventivo di spesa.

Dovendo trattare auto sia nuove che usate, per ottenere una memorizzazione efficiente delle informazioni relative alle automobili, scegliamo di utilizzare un'associazione per generalizzazione, attraverso la quale memorizzare nell'entità padre le informazioni comuni a tutte le automobili e nelle entità figlie solo quelle specifiche alle auto nuove e a quelle usate.

Nel database vanno pertanto memorizzate le informazioni che consentono di:

- registrare le immatricolazioni di nuove automobili;
- registrare le informazioni che riguardano le riparazioni;
- elencare le automobili usate caratterizzate da un prezzo inferiore a un valore fornito in input;
- elencare le riparazioni da effettuare per una determinata auto usata;
- elencare le auto nuove o usate di ogni marca presente in concessionaria;
- elencare gli optional presenti su ogni automobile.

