Esercitazioni per il corso di Logica Matematica

Luca Motto Ros

14 marzo 2005

Nota importante. Queste pagine contengono appunti personali dell'esercitatore e sono messe a disposizione nel caso possano risultare utili a qualcuno. In nessun caso vanno considerate come programma ufficiale del corso e in nessun caso eventuali errori, inesattezze o difformità rispetto alle lezioni qui contenuti possono essere considerati quale fonte autorevole in sede d'esame.

Logica del prim'ordine 1

La sintassi

Un linguaggio per la logica del prim'ordine è un insieme di simboli del tipo:

$$\begin{aligned}\mathcal{L} = \{&\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \exists, \forall, (,), \text{``,''}, x, y, z, \dots, a, b, c, \dots \\ &f, g, h, \dots, P, Q, R, \dots\}.\end{aligned}$$

In un linguaggio del prim'ordine vi sono diversi tipi di simboli:

Costanti logiche (o simboli fissi) Sono $\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \exists, \forall$. Questi simboli sono comuni ad ogni linguaggio del prim'ordine ed hanno un significato fisso (cioè non dipendente dal particolare linguaggio considerato). Come nel caso della logica proposizionale, sceglieremo alcune costanti logiche come *primitive* e definiremo le altre sulla base di queste. Dunque, in quel che segue, restringeremo l'insieme delle costanti logiche di un linguaggio a quelle primitive che sono: $\neg, \wedge, \exists$. Tutti gli altri simboli fissi verranno considerati *abbreviazioni* di formule contenenti solo le costanti logiche primitive. I simboli comuni ai linguaggi proposizionali continueranno a essere chiamati connettivi, mentre $\exists$ e $\forall$ si diranno *quantificatori* (rispettivamente, *esistenziale* e *universale*).

Simboli tipografici Sono $(,)$, "," e non hanno alcun significato particolare (si potrebbero anche eliminare), ma aiutano nella lettura delle formule.

Variabili L'insieme delle variabili $\mathcal{V}$, contiene tutte le variabili che servono. Esse vengono generalmente indicate con lettere minuscole dal fondo dell'alfabeto, come $x, y, z, \ldots$, e indicano *generici* elementi di un certo insieme in questione[1]. *Non* vi sono variabili che indicano sottoinsiemi: questo identifica $\mathcal{L}$ come un linguaggio del *prim'ordine*. Se vi fossero tali variabili, $\mathcal{L}$ si direbbe linguaggio del *second'ordine*.

Costanti L'insieme $\mathcal{C}_{\mathcal{L}}$ dei simboli per le costanti di un linguaggio $\mathcal{L}$ consiste di simboli (generalmente lettere minuscole dall'inizio dell'alfabeto, come $a, b, c, \ldots$) che rappresentano specifici elementi di un certo insieme in questione.

Funzioni Le lettere $f, g, h, \ldots$ sono generalmente usate per indicare simboli funzionali di $\mathcal{L}$ (il loro insieme verrà denotato $\mathcal{F}_{\mathcal{L}}$), ciascuno con una propria arietà fissata.

Relazioni I simboli $R, P, Q, \ldots$ formano l'insieme $\mathcal{R}_{\mathcal{L}}$ dei simboli relazionali di $\mathcal{L}$. A ciascuno di tali simboli è associata una arietà.

Dunque ciò che distingue un linguaggio del prim'ordine $\mathcal{L}$ da un altro linguaggio $\mathcal{L}'$ sono i relativi insiemi di simboli per costanti, funzioni e relazioni che abbiamo indicato con $\mathcal{C}_{\mathcal{L}}$, $\mathcal{F}_{\mathcal{L}}$ e $\mathcal{R}_{\mathcal{L}}$. Dunque, d'ora in poi, indicheremo un linguaggio per la logica del prim'ordine semplicemente esplicitando l'insieme

$$\mathcal{L} = \mathcal{C}_{\mathcal{L}} \cup \mathcal{F}_{\mathcal{L}} \cup \mathcal{R}_{\mathcal{L}}.$$

Accanto a questi simboli, generalmente si aggiunge l'identità come *simbolo fisso*. In realtà esso corrisponderebbe ad un simbolo relazionale binario da porre in $\mathcal{R}_{\mathcal{L}}$, ma poiché si usa di continuo in ogni struttura matematica, e sempre con lo stesso significato, è conveniente aggiungerlo alla lista dei simboli fissi. Quando si considera l'identità in questo modo, si parla più propriamente di logica del prim'ordine *con uguaglianza*.

Le *formule* dei linguaggi del prim'ordine verranno generalmente indicate con lettere greche minuscole φ, ψ e θ.

Prima di definire le formule, diamo una definizione induttiva dei *termini*.

Definizione 1. *Un $\mathcal{L}$-*termine *t è una stringa finita di simboli di $\mathcal{L}$ ottenuta a partire da $\mathcal{V}$ e $\mathcal{C}_{\mathcal{L}}$ mediante l'applicazione delle funzioni di $\mathcal{F}_{\mathcal{L}}$. In altre parole:*

1. *ogni variabile o (simbolo di) costante di $\mathcal{L}$ è un $\mathcal{L}$-termine;*

[1] Come vedremo in seguito, l'insieme in questione è il sostegno della struttura in cui vogliamo interpretare le formule del nostro linguaggio.

2. se $f \in \mathcal{F}_\mathcal{L}$ è un simbolo di funzione m-ario e $t_1, \ldots, t_m$ sono $\mathcal{L}$-termini, anche $f(t_1, \ldots, t_m)$ lo è.

Definizione 2. *Una $\mathcal{L}$-*formula atomica *è una stringa finita di simboli di $\mathcal{L}$ del tipo $t_1 = t_2$ o $R(t_1, \ldots, t_n)$, dove $R \in \mathcal{R}_\mathcal{L}$ è un simbolo relazionale n-ario e $t_1, t_2, \ldots, t_n$ sono $\mathcal{L}$-termini.*

Definizione 3. *L'insieme delle $\mathcal{L}$-*formule *è definito* induttivamente *nella seguente maniera:*

1. ogni $\mathcal{L}$-formula atomica è una $\mathcal{L}$-formula;

2. se φ e ψ sono $\mathcal{L}$-formule, anche $\neg\varphi$ e $(\varphi \wedge \psi)$ lo sono;

3. se φ è una $\mathcal{L}$-formula, anche $(\exists x \varphi)$ lo è (per ogni variabile x del lignuaggio).

Dunque ogni stringa finita di simboli di $\mathcal{L}$ costruita a partire dalle formule atomiche utilizzando le clausole 2. e 3. è una formula.

Quando il liguaggio $\mathcal{L}$ sarà chiaro dal contesto, si ometterà il riferimento ad esso parlando semplicemente di termini, formule atomiche e formule.

Come nel caso della logica proposizionale, ad ogni formula φ si può associare un *albero di costruzione*, che verrà detto albero sintattico[2]. Inoltre, ad ogni formula verrà associato anche un *grado*: generalmente si utilizza come graduazione l'altezza dell'albero sintattico diminuita di 1, ma sarebbero graduazioni anche la lunghezza della formula (vista come stringa di simboli) e il numero di occorrenze di costanti logiche.

Inoltre, sempre analogamente a quanto fatto nel caso della logica proposizionale, ogni formula che compare nell'albero sintattico di φ verrà detta *sottoformula* di φ.

In realtà, quando si scrivono formule nei linguaggi del prim'ordine, si usano anche tutte le altre costanti logiche (cioè $\vee, \rightarrow, \leftrightarrow, \forall$). Di queste, solo $\forall$ resta da definire (gli altri connettivi si definiscono come nel caso della logica proposizionale). Dunque considereremo la formula $(\forall x \varphi)$ come abbreviazione di $\neg(\exists x \neg\varphi)$.

Come al solito, da una formula si possono omettere tutte le parentesi non necessarie, tenendo conto che la priorità tra le costanti logiche è data convenzionalmente dal seguente ordine:

$$\begin{gathered} \neg,\ \exists,\ \forall \\ \wedge,\ \vee \\ \rightarrow \\ \leftrightarrow. \end{gathered}$$

[2] Anche in questo caso si ottiene un albero binario e finito.

Nel caso di $\neg$ e dei quantificatori, l'ordine in cui vanno considerati è dato dall'ordine con cui essi compaiono nella formula (ovvero, ciascuna di queste costanti si applica all'intera sottoformula che la segue).

Se in una formula φ compare un quantificatore Q (dove Q può essere sia $\exists$ che $\forall$), allora φ contiene una sottoformula del tipo $Qx\psi$: la formula ψ viene detta *raggio d'azione del quantificatore* Q. Se una variabile x compare in una formula φ entro il raggio d'azione di un quantificatore Q che è immediatamente seguito da x stessa, si dice che x *occorre vincolata* in φ; in caso contrario, si dice che x *occorre libera* in φ.

Osservazione. Si osservi che ad essere libera o vincolata è la particolare *occorrenza* di una variabile, non la variabile stessa: infatti una stessa variabile può avere sia occorrenze libere che vincolate in una formula φ.

Generalmente, quando in una formula occorrono variabili libere, queste vengono indicate tra parentesi dopo la lettera greca con cui la formula viene denotata. Per esempio, quando scriviamo $\varphi(x, y)$, intendiamo dire che in φ vi sono (al più) occorrenze libere di x e y. In questo modo, potremo denotare semplicemente con $\varphi(t_1, t_2)$ la formula ottenuta da $\varphi(x, y)$ sostituendo alle occorrenze libere di x e y i termini t_1 e t_2.

Definizione 4. *Una $\mathcal{L}$-formula priva di (occorrenze di) variabili libere si dice* $\mathcal{L}$-enunciato *(o anche* $\mathcal{L}$-formula chiusa*).*

Esercizi.

1. Sia $\mathcal{L} = \{+, <, 1, 2, 3\}$, dove $+$ è un simbolo di funzione binario, $<$ è un simbolo relazionale binario e 1, 2 e 3 sono simboli di costante. Scriveremo $(x + y)$ al posto di $+(x, y)$ e $x < y$ al posto di $< (x, y)$. Siano date inoltre le seguenti formule:

   (a) $\forall x \exists y((x + y) = 1)$

   (b) $\forall x \neg(x < 1)$

   (c) $((1 + 1) = 2)$.

   (d) $2 < 1$.

   (e) $\forall x(2 < 1) \rightarrow (x + 2 < x + 1)$

   (f) $\forall x \forall y \exists z(x + y = z)$

   (g) $\forall x \forall y \forall z(((x + 3 = y) \wedge (x + 3 = z)) \rightarrow (y = z))$

   (h) $\forall x \forall y \forall z(((x + y = 3) \wedge (x + z = 3)) \rightarrow (y = z))$

   (i) $\forall x \forall y(((x + 3) < (y + 3)) \rightarrow (x < y))$

   (j) $\forall x \forall y((x < 2) \rightarrow ((x + 3) = 4))$

Per ogni formula, dire se si tratta di un enunciato e, in caso contrario, elencare le variabili che vi occorrono libere. Costruirne l'albero sintattico e determinare il grado della formula. Elencarne le sottoformule (specificando se si tratta di formule atomiche) ed elencare tutti i termini che vi compaiono.

Traduzione dal linguaggio naturale al linguaggio formale

Nei linguaggi naturali spesso si esprime un predicato riferito al soggetto. Nei linguaggi formalizzati del prim'ordine, i predicati si traducono con simboli relazionali unari (si osservi che tale traduzione non era possibile nei linguaggi proposizionali). Per esempio, $P(x)$ traduce "x ha la proprietà P". Per questa ragione i simboli relazionali vengono detti anche *simboli predicativi* (unari, binari e così via) e talvolta ci si riferisce alla logica del prim'ordine con l'espressione *logica predicativa*.

Spesso quando si traduce una frase in una formula di un opportuno linguaggio del prim'ordine, si vorrebbe quantificare una variabile intendendo però che essa possa variare solo su un sottoinsieme dell'universo considerato, per esempio su tutti gli elementi che godono di una certa proprietà P (spesso si usano espressioni del tipo "tutte le persone..." o "esiste una persona...", intendendo con questo che x può variare solo sull'insieme P delle persone). In questo caso si usano i *quantificatori limitati*. Se si vuole scrivere che *tutti* gli x che godono di una certa proprietà P soddisfano una formula $\varphi(x)$, l'enunciato corrispondente sarà

$$\forall x(P(x) \to \varphi(x)).$$

Se invece si vuole scrivere che *esiste* un x che gode di una certa proprietà P e che soddisfa $\varphi(x)$ allora si scriverà

$$\exists x(P(x) \land \varphi(x)).$$

Dunque, come si osserverà facilmente negli esercizi di traduzione seguenti, il quantificatore universale $\forall$ sarà spesso associato ad una implicazione, mentre il quantificatore esistenziale sarà spesso associato ad una congiunzione.

Esercizi.

1. Sia $\mathcal{L} = \{G, D\}$, dove G è un simbolo di relazione binario tale che $G(x, y)$ se e solo se "x è genitore di y", mentre D è un simbolo di relazione unario tale $D(x)$ se e solo se "x è una donna". Scrivere la formula corrispondente a ciascuno dei seguenti predicati:

(a) x è un fratello di y;
(b) x è una zia di y;
(c) x e y sono cugini;
(d) x è figlio unico;
(e) x ha esattamente due fratelli.

2. Sia $\mathcal{L} = \{a, b, V, D\}$, dove a e b sono simboli di costante (che denotano rispettivamente "Angela" e "Bruno"), e V e D sono simboli di relazioni unarie tali che $V(x)$ se e solo se "x va alla festa" e $D(x)$ se e solo se "x si diverte". Tradurre nel linguaggio formale $\mathcal{L}$ le seguenti frasi:

   (a) Angela va alla festa e Bruno no, oppure Angela va alla festa e Bruno si diverte;
   (b) tutti quelli che vanno alla festa si divertono.

   Interpretare nel linguaggio naturale le seguenti $\mathcal{L}$-formule:

   (a) $V(b) \wedge D(b) \rightarrow \neg V(a)$;
   (b) $\exists x(V(x) \wedge \neg D(x))$.

3. Utilizzando il linguaggio $\mathcal{L} = \{M, D, C, S\}$ dove M e D sono simboli relazionali unari ($M(x)$ sta per "x è malato" e $D(x)$ sta per "x è un dottore") C ed S sono simboli relazionali binari ($C(x, y)$ sta per "x cura y" e $S(x, y)$ sta per "x stima y"), tradurre le seguenti frasi:

   (a) ogni malato non stima se stesso;
   (b) ci sono dottori che curano se stessi;
   (c) qualche malato stima tutti i dottori che lo curano;
   (d) tutti i malati stimano almeno un dottore che li cura.

4. Consideriamo il linguaggio $\mathcal{L} = \{S, C, F, U\}$ dove S è un simbolo relazionale unario ("x è uno studente"), C è un simbolo relazionale unario ("x è un computer"), F è un simbolo relazionale unario ("x è funzionante") e U è un simbolo relazionale binario ("x utilizza y"). Consideriamo le seguenti frasi:

   (a) Un computer non è utilizzato da nessun studente.
   (b) Ogni computer funzionante è utilizzato da almeno uno studente.

   (c) Non tutti i computer sono funzionanti.

Quale dei seguenti enunciati è una traduzione corretta di 4a?

$\exists x(C(x) \land \forall y(\neg S(y) \land \neg U(y,x)))$
$\exists x(C(x) \to \forall y(S(y) \to \neg U(y,x)))$
$\exists x(C(x) \land \forall y(S(y) \to \neg U(y,x)))$

Qual'è il significato delle altre formule? Tradurre 4b e 4c in oppurtuni enunciati del linguaggio formale $\mathcal{L}$ introdotto.

5. Introducendo opportuni linguaggi del prim'ordine, tradurre le seguenti frasi:

   (a) Se tutti gli uomini sono mortali e Socrate è un uomo, allora Socrate è mortale.
   (b) Se tutti i cani sono animali e Fido è un cane, allora Fido è un animale.
   (c) Se ogni amico di Mario è amico di Diego e Pietro non è amico di Mario, allora Pietro non è amico di Diego.
   (d) Nessun ladro è onesto.
   (e) Se tutti i filosofi intelligenti sono curiosi, e solo i tedeschi sono filosofi intelligenti, allora, se ci sono filosofi intelligenti, qualche tedesco è curioso.
   (f) Il cervello di un delfino è più grande di quello di un topo.[3].
   (g) Se Carlo è più basso di Diego, allora almeno un amico di Carlo è più basso di tutti gli amici di Diego.
   (h) Se tutti gli studenti sono persone serie, tutti gli studenti sono studiosi e tutte le persone serie e studiose non fanno tardi la sera, allora se esiste qualcuno che fa tardi la sera, non tutti sono studenti.

6. Formalizzare "se ogni sorella di Gianni litiga con almeno una sorella di Fabio, Gianni litiga con Fabio" utilizzando il linguaggio $\mathcal{L} = \{g, f, S, L\}$, dove g ed f sono simboli di costante (che denotano rispettivamente "Gianni" e "Fabio") e S ed L sono simboli relazionali binari ($S(x,y)$ sta per "x è sorella di y" e $L(x,y)$ sta per "x litiga con y").

7. Sia $\mathcal{L} = \{c, P, S\}$ un linguaggio in cui c è un simbolo di costante (interpretato come "Claudio") e P ed S sono due simboli relazionali binari che stanno per "x è parente di y" e "x stima y". Tradurre le seguenti frasi:

[3]Suggerimento: utilizzare il linguaggio contenente i simboli relazionali $G(x,y)$ ("x è più grande di y"), $C(x,y)$ ("x è il cervello di y"), $D(x)$ ("x è un delfino") e $T(x)$ ("x è un topo").

(a) Tutti i parenti di Claudio stimati da Claudio, sono stimati anche da chi stima Claudio.

(b) Claudio stima tutti quelli che stimano qualche suo parente, ma non stima i suoi parenti che non stimano se stessi.

(c) Claudio stima solo quelli che stimano tutti i propri parenti.

8. Utilizzando il linguaggio $\mathcal{L} = \{C, D, A\}$ dove C è un simbolo predicativo unario ($C(x)$ sta per "x è un cantante") e D ed A sono simboli relazionali binari ($D(x, y)$ sta per "x è un cd di y" e $A(x, y)$ sta per "x acquista y"), formalizzare le frasi seguenti:

   (a) Qualcuno acquista tutti i cd di qualche cantante.

   (b) Ogni cd di un cantante è sempre acquistato da qualcuno.

9. Utilizzando il linguaggio $\mathcal{L} = \{S, P, V, G\}$ dove i quattro simboli sono simboli predicativi unari, da interpretarsi come "x è stupido", "x è presuntuoso", "x è vanitoso" e "x è simpatico", ed inoltre considerando "antipatico" come la negazione di "simpatico", formalizzare le frasi seguenti:

   (a) tutti gli stupidi sono presuntuosi o vanitosi;

   (b) i presuntuosi sono antipatici;

   (c) le persone simpatiche non sono vanitose;

   (d) tutti gli stupidi sono antipatici.

10. Consideriamo il linguaggio $\mathcal{L} = \{R, S, I, U, L, C\}$ dove R, S, I, U, L sono simboli relazionali unari ($R(x)$ sta per "x è un ragazzo", $S(x)$ sta per "x è una ragazza", $I(x)$ sta per "x è iscritto all'università", $U(x)$ sta per "x sa usare un computer", $L(x)$ sta per "x troverà lavoro"), mentre C è un simbolo relazionale binario ($C(x, y)$ sta per "x conosce y"). Formalizzare le seguenti frasi:

   (a) Tutti i ragazzi e le ragazze iscritti all'università sanno usare il computer o conoscono qualcuno che lo sa usare.

   (b) Fra quelli che sanno usare un computer c'è qualcuno che troverà lavoro.

   (c) Tutti conoscono qualcuno che troverà lavoro.

Semantica e strutture

Strutture

Una struttura M è composta da un insieme di oggetti U_M (detto *universo* o *sostegno* della struttura), da un certo insieme fissato di funzioni (di varia arietà) che mandano (n-uple di) elementi di U_M in elementi di U_M, da un insieme (fissato) di relazioni di varia arietà tra gli elementi di U_M e da un insieme di costanti (che indicano alcuni particolari elementi di U_M).

Sia $\mathcal{L}$ un linguaggio del prim'ordine. Un'interpretazione di $\mathcal{L}$ in un certo insieme U è una funzione che assegna:

- un elemento di U ad ogni (simbolo di) costante di $\mathcal{L}$;
- una funzione da U^n in U ad ogni simbolo di funzione n-ario di $\mathcal{L}$;
- un sottoinsieme di U^n (ovvero una relazione n-aria) ad ogni simbolo relazionale n-ario di $\mathcal{L}$.

M si dice *$\mathcal{L}$-struttura* se il suo universo U_M è non vuoto ed esiste un'interpretazione biettiva che manda i simboli di $\mathcal{L}$ nelle funzioni, relazioni e costanti di M. Se s è un simbolo di $\mathcal{L}$, scriveremo s^M per indicare l'elemento della struttura M che viene associato a s da tale interpretazione[4]. Inoltre ogni $\mathcal{L}$-termine t privo di variabili (libere) verrà interpretato in un elemento t^M di U_M. Più precisamente:

- se $t = c$, dove c è un simbolo di costante[5], allora poniamo $t^M = c^M$;
- se $t = f(t_1, \dots, t_m)$ con f simbolo di funzione m-ario e $t_1, \dots, t_m$ termini privi di variabili (libere), allora poniamo $t^M = f^M(t_1^M, \dots, t_m^M)$.

Le strutture giocano nella logica del prim'ordine un ruolo analogo a quello degli assegnamenti nel caso della logica proposizionale. Definiremo cosa vuol dire per una $\mathcal{L}$-formula essere vera o no in una data $\mathcal{L}$-struttura, restringendo però la nostra attenzione solamente agli enunciati.

Definizione 5. *Sia $\mathcal{L}$ un linguaggio. Un'*espansione *di $\mathcal{L}$ è un linguaggio $\mathcal{L}'$ tale che $\mathcal{L}' \supseteq \mathcal{L}$.*

*Sia M una $\mathcal{L}$-struttura. Una struttura M' è un'*espansione *di M se e solo se $U_{M'} = U_M$ e M' interpreta i simboli di $\mathcal{L}$ nella stessa maniera di M.*

---

[4] Per esempio, f^M sarà la funzione n-aria di M che interpreta il simbolo funzionale n-ario f di $\mathcal{L}$. Analogamente. R^M sarà la relazione di M che interpreta il simbolo relazionale R di $\mathcal{L}$, e c^M sarà la costante di M in cui viene interpretato il simbolo di costante c di $\mathcal{L}$.

[5] Si osservi che non può presentarsi il caso $t = x$ con x variabile perché abbiamo supposto t *privo* di variabili.

Dunque un'espansione M' della $\mathcal{L}$-struttura M è una $\mathcal{L}'$-struttura (dove $\mathcal{L}'$ è un'espansione di $\mathcal{L}$) che ha lo stesso universo e tale che l'interpretazione di $\mathcal{L}'$ in M' è un'estensione dell'interpretazione di $\mathcal{L}$ in M.

Tra le espansioni di una $\mathcal{L}$-struttura M ve n'è una particolarmente importante. Sia $\mathcal{L}(M)$ il linguaggio $\mathcal{L} \cup \{c_m \,|\, m \in U_M\}$, ovvero l'espansione di $\mathcal{L}$ ottenuta aggiungendo un simbolo di costante per ogni elemento dell'universo U_M di M. Allora possiamo considerare la $\mathcal{L}(M)$-struttura M_C, ovvero l'espansione di M che interpreta ogni nuovo simbolo di costante c_m nell'elemento m di U_M.

Definiamo ora in maniera induttiva cosa vuol dire per una $\mathcal{L}$-struttura M, "modellare" un $\mathcal{L}$-enunciato.

Definizione 6. *1. se φ è atomica, allora o è della forma $t_1 = t_2$, oppure è della forma $R(t_1, \ldots, t_n)$ dove $t_1, t_2, \ldots, t_n$ sono termini e R è un simbolo relazionale n-ario. Poichè φ è un enunciato, ciascuno dei termini t_i ($1 \leq i \leq n$) verrà interpretato in un corrispondente elemento $a_i \in U_M$. Dunque $M \models t_1 = t_2$ (si legge "M modella $t_1 = t_2$) se e solo se a_1 e a_2 sono lo stesso elemento di U_M, e $M \models R(t_1, \ldots, t_n)$ se e solo se gli elementi $a_1, \ldots, a_n$ sono in relazione R^M;*

2. *se φ è della forma $\neg\psi$, allora $M \models \varphi$ se e solo se M* non *modella ψ;*

3. *se φ è della forma $\psi \wedge \theta$, allora $M \models \varphi$ se e solo se $M \models \psi$ e $M \models \theta$;*

4. *se φ è della forma $\exists x \psi(x)$, allora $M \models \varphi$ se e solo se $M_C \models \psi(c)$ per qualche simbolo di costante c del linguaggio $\mathcal{L}(M)$.*

Se $M \models \varphi$ diremo anche che φ *è vera in* M, o che M *soddisfa* φ, o che φ *vale in* M.

Ovviamente, la definizione di $\models$ data precedentemente si estende alle altre costanti logiche nella maniera ovvia (ovvero sfruttando le definizioni di quest'ultime sulla base di $\neg$, $\wedge$ e $\exists$).

Inoltre, la definizione di $\models$ si può estendere anche agli $\mathcal{L}(M)$-enunciati. Infatti, se φ è un $\mathcal{L}(M)$-enunciato, diremo che $M \models \varphi$ se e solo se $M_C \models \varphi$.

Per ogni $\mathcal{L}$-struttura M, chiameremo *parametri* le costanti di $\mathcal{L}(M)$ che non sono già in $\mathcal{L}$.

Definizione 7. *Sia φ un $\mathcal{L}$-enunciato.*

φ si dirà soddisfacibile *se* esiste *una $\mathcal{L}$-struttura M tale che $M \models \varphi$.*

φ si dirà valido *(o che φ è una* tautologia, *o che è* sempre vero*) se* per ogni *$\mathcal{L}$-struttura M si ha che $M \models \varphi$. In questo caso scriveremo $\models \varphi$.*

φ si dirà insoddisfacibile *(o che φ è una* contraddizione*) se non è soddisfacibile, cioè se* per ogni *$\mathcal{L}$ struttura M si ha che M* non *modella φ.*

Osservazione. Mentre il numero di assegnamenti di una formula proposizionale è finito (e dunque si può controllare se essa è una tautologia, o se è insoddisfacibile), non c'è limite al numero di $\mathcal{L}$-strutture che possono modellare o meno un $\mathcal{L}$-enunciato φ. Dunque sarà in generale più facile dimostrare che φ è soddisfacibile (basta mostrare *una* struttura M tale che $M \models \varphi$) piuttosto che dimostrare la sua validità o insoddisfacibilità (bisognerebbe controllare se *tutte* le $\mathcal{L}$-strutture modellano o meno φ).

Definizione 8. *Dati due $\mathcal{L}$-enunciati ψ e φ, diremo che ψ è* conseguenza (logica) *di φ (e si scrive $\varphi \models \psi$) se ogni $\mathcal{L}$-struttura che modella φ modella anche ψ. In simboli, $\varphi \models \psi$ sse per ogni $\mathcal{L}$-struttura M, se $M \models \varphi$ allora $M \models \psi$.*

φ e ψ si dicono (logicamente) equivalenti *(e si scrive $\varphi \equiv \psi$) sse sono una conseguenza logica dell'altra, cioè se $\varphi \models \psi$ e $\psi \models \varphi$.*

Continua a valere la seguente

Proposizione 1. *$\varphi \models \psi$ se e solo se $\models \varphi \to \psi$.*
$\varphi \equiv \psi$ se e solo se $\models \varphi \leftrightarrow \psi$.

Per analogia, se Γ è un insieme di $\mathcal{L}$-formule e ψ è un'altra $\mathcal{L}$-formula, diremo che ψ è conseguenza logica di Γ (e scriveremo $\Gamma \models \psi$) se per ogni $\mathcal{L}$-struttura M che modella *tutte* le formule di Γ si ha $M \models \psi$.

È evidente che se Γ è finito, indicando con $\bigwedge \Gamma$ la congiunzione di tutte le formule che compaiono in Γ, si ha

$$\Gamma \models \psi \text{ sse } \bigwedge \Gamma \models \psi \text{ sse } \models \bigwedge \Gamma \to \psi.$$

Osservazione. Per quanto osservato prima, sarà generalmente più semplice dimostrare che φ *non* è una tautologia, o che ψ *non* è una conseguenza di φ, o che due enunciati φ e ψ *non* sono equivalenti, che non il contrario.
Infatti, nel primo caso basta dimostrare che $\neg\varphi$ è soddisfacibile, nel secondo basta dimostrare che $\varphi \wedge \neg\psi$ è soddisfacibile, e nel terzo è sufficiente dimostrare che almeno una tra $\varphi \wedge \neg\psi$ e $\psi \wedge \neg\varphi$ è soddisfacibile.

Osserviamo che abbiamo definito il concetto di $\models$ solo per enunciati. Tuttavia si usa la seguente convenzione[6]: se $\varphi(x_1, \ldots, x_n)$ è una qualsiasi $\mathcal{L}$-formula (con, eventualmente, occorrenze libere di $x_1, \ldots, x_n$), diremo che la $\mathcal{L}$-struttura M modella $\varphi(x_1, \ldots, x_n)$ se e solo se M modella la sua *chiusura universale* (che è un enunciato), cioè se e solo se $M \models \forall x_1, \ldots \forall x_n \varphi(x_1, \ldots, x_n)$.

[6] Tale convenzione nasce dal fatto che, nel linguaggio naturale, quando non si specifica una quantificazione generalmente si sottintende una quantificazione universale.

È comunque evidente che la formula $\varphi(x_1, \ldots, x_n)$ può valere per alcuni valori di $x_1, \ldots, x_n$ e non per altri. L'insieme delle n-uple per cui vale tale formula viene detto *insieme definito da* φ.

Definizione 9. *Sia $\varphi(x_1, \ldots, x_n)$ una $\mathcal{L}$-formula e sia M una $\mathcal{L}$-struttura. L'insieme delle n-uple $(b_1, \ldots, b_n) \in (U_M)^n$ tali che $M \models \varphi(b_1, \ldots, b_n)$ viene denotato da $\varphi(M)$. L'insieme $\varphi(M)$ è detto* insieme definibile *di M (anche se in realtà è un sottoinsieme di $(U_M)^n$).*

Esempio di struttura.

I database relazionali forniscono un buon esempio di strutture. Siano dati i seguenti database.

Genitore	**Figlio**
Giorgio	Mario
Giorgio	Federica
Federica	Paolo
Valentina	Diego
Pietro	Diego
Pietro	Maria
Diego	Paolo
Federica	Matteo
Diego	Matteo
Anna	Massimo
Matteo	Massimo

Donne
Valentina
Anna
Maria
Federica

Persona	**Compleanno**
Giovanna	5 agosto
Leonardo	8 agosto
Massimo	28 luglio
Matteo	1 agosto
Federica	24 luglio

Ad essi corrisponde una struttura D così definita:

- l'universo U_D è costituito da tutti gli oggetti (nomi, date) che compaiono in qualche database;
- per ogni database, in D vi è una relazione R di arietà corrispondente al numero di colonne del database, tale che un'opportuna n-upla di elementi di U_D è in relazione R se e solo se essa corrisponde ad una delle righe del database prescelto.

Nel nostro esempio, in D vi sarà una relazione binaria G che corrisponderà al primo database, ovvero tale che $D \models G(a, b)$ se e solo se a è genitore di b, ovvero se e solo se, ab è una riga del primo database. Analogamente vi sarà una relazione unaria F ("x è una donna", ovvero x compare in una riga del secondo database), e una relazione binaria B tale che $D \models B(a, b)$ se e solo se b è il compleanno di a, cioè se e solo se ab compaiono in una stessa riga del terzo database.

Partendo da questi simboli, si possono esprimere diverse relazioni legate a D: per esempio, $\neg F(x)$ significherà "x è un uomo", $\exists z(B(x, z) \wedge B(y, z))$

significherà “x e y compiono gli anni lo stesso giorno”, e così via.

Questo tipo di strutture è importante perché vedremo che il Prolog è un linguaggio di programmazione basato su una logica di Horn del prim'ordine che può essere usato per presentare e fare ricerche in ogni database relazionale.

Esercizi.

1. Si considerino nuovamente il linguaggio e le formule dell'esercizio 1 a pag. 4. Quali di esse sono soddisfacibili? Siano $\mathbf{N}$ e $\mathbf{R}$ le strutture con universo, rispettivamente, $\mathbb{N}$ e $\mathbb{R}$ che interpretano i simboli di $\mathcal{L}$ nella maniera usuale. Quali delle formule precedenti sono vere in $\mathbf{N}$ e quali in $\mathbf{R}$?

2. Sia $\mathcal{L} = \{E\}$ (dove E è un simbolo relazionale binario), e siano
$$\begin{aligned}\varphi_1 =& \forall x E(x,x),\\ \varphi_2 =& \forall x \forall y (E(x,y) \rightarrow E(y,x)),\\ \varphi_3 =& \forall x \forall y \forall z ((E(x,y) \wedge E(y,z)) \rightarrow E(x,z))\end{aligned}$$
Mostrare che ciascuna delle φ_i ($1 \leq i \leq 3$) non è conseguenza delle altre due.

3. Sia $\mathcal{L}_{gp}$ il linguaggio $\{+, 0\}$, dove $+$ è una funzione binaria e 0 è una costante. Come al solito, scriveremo $x + y$ al posto di $+(x, y)$. Consideriamo i seguenti $\mathcal{L}$-enunciati.
$$\begin{gathered}\forall x \forall y \forall z (x + (y + z) = (x + y) + z)\\ \forall x ((x + 0 = x) \wedge (0 + x = x))\\ \forall x (\exists y (x + y = 0) \wedge \exists z (z + x = 0))\end{gathered}$$
Sia γ la congiunzione di questi tre enunciati.
   (a) Mostrare che γ è soddisfacibile.
   (b) Mostrare che γ non è una tautologia.
   (c) Sia α l'enunciato $\forall x \forall y (x + y = y + x)$. Mostrare che α non è conseguenza logica di γ.
   (d) Mostrare che γ non è equivalente a nessuna possibile congiunzione di due dei tre enunciati precedenti (ovvero che ciascuno degli enunciati non è conseguenza degli altri due).

4. Una formula del prim'ordine $\varphi(x)$ si dice soddisfacibile se e solo se l'enunciato $\forall x \varphi(x)$ è soddisfacibile. Dimostrare che $\varphi(x)$ è una tautologia se e solo se $\exists x \varphi(x)$ è una tautologia.

5. Sia $\mathcal{L}_N = \{+, \cdot, 1\}$ e sia $\mathbf{N}$ la $\mathcal{L}_N$-struttura che ha come universo $\mathbb{N}$ e che interpreta il linguaggio nella maniera usuale.

   (a) Definire una $\mathcal{L}_N$-formula $\varepsilon(x)$ tale che per ogni $a \in \mathbb{N}$, $\mathbf{N} \models \varepsilon(a)$ se e solo se a è pari.

   (b) Definire una $\mathcal{L}_N$-formula $\pi(x)$ tale che per ogni $a \in \mathbb{N}$, $\mathbf{N} \models \pi(a)$ se e solo se a è primo.

   (c) Definire una $\mathcal{L}_N$-formula $\mu(x, y)$ tale che per ogni a e b in $\mathbb{N}$, $\mathbf{N} \models \mu(a, b)$ se e solo se a e b sono relativamente primi (cioè se e solo se il massimo comun divisore tra a e b è 1).

   (d) Definire una $\mathcal{L}_N$-formula $\nu(x, y, z)$ tale che per ogni a, b e c in $\mathbb{N}$, $\mathbf{N} \models \nu(a, b, c)$ se e solo se c è il più piccolo numero divisibile sia per a che per b (cioè se e solo se c è il minimo comune multiplo di a e b).

6. La congettura di Goldbach afferma che ogni intero pari maggiore (strettamente) di 2 è la somma di due numeri primi. Se la congettura sia vera o no è uno dei problemi di teoria dei numeri ancora aperto. Esprimere la congettura di Goldbach mediante una $\mathcal{L}_{ar}$-formula, dove $\mathcal{L}_{ar} = \{+, \cdot, 0, 1\}$ ($+$ e $\cdot$ funzioni binarie, 0 e 1 costanti).

7. Sia $\mathcal{L}_{ar} = \{+, \cdot, 0, 1\}$ il linguaggio dell'aritmetica. Sia $\mathbf{R}$ la $\mathcal{L}_{ar}$-struttura che ha come universo $\mathbb{R}$ e interpreta il linguaggio nella maniera usuale.

   (a) Definire una $\mathcal{L}_{ar}$-formula $\alpha(x)$ tale che, per ogni $a \in \mathbb{R}$, $\mathbf{R} \models \alpha(a)$ se e solo se a è positivo.

   (b) Definire una $\mathcal{L}_{ar}$-formula $\beta(x, y)$ tale che, per ogni a e b in $\mathbb{R}$, $\mathbf{R} \models \beta(a, b)$ se e solo se $a \leq b$.

   (c) Definire una $\mathcal{L}_{ar}$-formula $\gamma(x)$ tale che, per ogni $a \in \mathbb{R}$, $\mathbf{R} \models \gamma(a)$ se e solo se il valore assoluto di a è minore di 1.

8. Siano $\mathcal{L}_{ar}$ e $\mathbf{R}$ come nell'esercizio precedente. Sia $\mathcal{L}^+ = \mathcal{L}_{ar} \cup \{f\}$ un'espansione di $\mathcal{L}_{ar}$ ottenuta aggiungendo una funzione unaria f. Definire un $\mathcal{L}^+$-enunciato ζ tale che, per ogni espansione $\mathbf{R}^+$ di $\mathbf{R}$ ad una $\mathcal{L}^+$-struttura, $\mathbf{R}^+ \models \zeta$ se e solo se $\mathbf{R}^+$ interpreta f come una funzione continua.

9. Sia $\mathcal{L} = \{\cdot, 2\}$ e sia $\mathbf{N}$ la $\mathcal{L}$-struttura con universo $\mathbb{N}$ che interpreta i simboli di $\mathcal{L}$ nella maniera naturale.

(a) Dire quali sono gli elementi di $A \subseteq \mathbb{N}$, dove A è l'insieme definito dalla $\mathcal{L}$-formula $\exists y(2 \cdot y = x \wedge \exists z(2 \cdot z = y))$.

(b) Dire quali sono gli elementi di $B \subseteq \mathbb{N}$, dove B è l'insieme definito dalla $\mathcal{L}$-formula $\exists y(2 \cdot y = x \wedge \exists y(2 \cdot y = x))$.

(c) Dire quali sono gli elementi di $C \subseteq \mathbb{N}$, dove C è l'insieme definito dalla $\mathcal{L}$-formula $\exists y(x \cdot y = 2 \wedge \exists z(y \cdot z = 2))$.

(d) Dire quali sono gli elementi di $D \subseteq \mathbb{N}$, dove D è l'insieme definito dalla $\mathcal{L}$-formula $\exists y(x \cdot y = 2 \wedge \exists y(x \cdot y = 2))$.

10. Siano A e B due insiemi definibili di una struttura M. Supponiamo che A e B siano entrambi insiemi di n-uple (di elementi di U_M).

    (a) Mostrare che $A \cup B$ è definibile.

    (b) Mostrare che $A \cap B$ è definibile.

    (c) Mostrare che $A \setminus B = \{a \,|\, a \in A \text{ e } a \notin B\}$ è definibile.

11. Sia U_M l'universo di una struttura M. Supponiamo che $A \subseteq (U_M)^3$ e $B \subseteq (U_M)^3$ siano insiemi definibili di M.

    (a) Mostrare che $A \times B \subseteq (U_M)^6$ è definibile.

    (b) Supponiamo di riordinare le n-uple. Consideriamo l'insieme di tutti gli (z, x, y) tali che (x, y, z) è in A. Mostrare che tale insieme è definibile.

    (c) Mostrare che $C \subseteq (U_M)^2$ è definibile, dove C è l'insieme delle coppie ordinate (x, y) tali che $(x, y, z) \in A$ per qualche z.

    (d) Mostrare che $D \subseteq (U_M)^2$ è definibile, dove D è l'insieme delle coppie ordinate (x, y) tali che sia $(x, y, z) \in A$ per qualche z, sia $(x, y, z) \in B$ per qualche z.

    (e) Mostrare che $E \subseteq (U_M)^2$ è definibile, dove E è l'insieme delle coppie ordinate (x, y) tali che, per qualche z, $(x, y, z) \in A$ e $(x, y, z) \in B$.

12. Siano φ e ψ gli enunciati $\forall x(R(x, a) \to R(x, b))$ e $\exists x(R(x, a) \wedge R(x, b))$. Determinare il linguaggio (minimo) $\mathcal{L}$ di tali formule. Definire quattro $\mathcal{L}$-strutture M_1, M_2, M_3 e M_4 tutte con universo $\{0, 1\}$ e tali che:

$$
\begin{array}{lcl}
M_1 \models \varphi & \text{e} & M_1 \models \psi; \\
M_2 \models \varphi & \text{e} & M_2 \not\models \psi; \\
M_3 \not\models \varphi & \text{e} & M_3 \models \psi; \\
M_4 \not\models \varphi & \text{e} & M_4 \not\models \psi.
\end{array}
$$

13. Sia φ l'enunciato

$$\forall x \forall y (R(x,y) \to \neg R(y,x)) \wedge \forall x \neg R(x,x) \wedge \forall x \exists y R(x,y).$$

Dimostrare che non esistono modelli di φ il cui universo contenga solo 1 o 2 elementi, e definire un modello M il cui universo contenga tre elementi e tale che $M \models \varphi$.

14. Per ognuna delle formule seguenti, definire un modello che le renda vere ed uno che le renda false:

    (a) $\forall x \exists y R(x,y) \wedge \neg \forall x P(x)$
    (b) $\forall x P(x) \vee \forall x \neg P(x)$
    (c) $P(c) \to \neg P(c)$
    (d) $(\exists x P(x) \to P(c)) \wedge \neg P(c)$
    (e) $\forall x (P(x) \vee Q(x)) \to \forall x P(x) \vee \forall x Q(x)$
    (f) $P(c) \to P(a)$
    (g) $\forall x \exists y R(x,y) \to \exists y \forall x R(x,y)$

15. Trovare un modello di φ, dove φ è la congiunzione delle tre formule seguenti:

$$\exists u \forall x (P(u,x) \to Q(x));$$
$$\forall x \exists v (P(x,v) \vee P(v,x));$$
$$\forall x \forall y (Q(x) \wedge Q(y) \wedge \forall z (P(x,z) \to P(y,z)) \to \exists z (P(y,z) \wedge \neg P(x,z)).$$