

Logică pentru informatică - note de curs

Universitatea Alexandru Ioan Cuza, Iași

Facultatea de Informatică

Anul universitar 2021-2022

Ștefan Ciobâcă

Andrei Arusoaie

Rodica Condurache

Cristian Masalagiu

Cuprins

1	Motivație și introducere	5
2	Structuri și semnături	7
2.1	Fișă de exerciții	10
3	Sintaxa logicii de ordinul I	13
3.1	Alfabetul	13
3.2	Termen	14
3.3	Formule atomice	15
3.4	Formule de ordinul I	16
3.5	Modelarea în LPI a afirmațiilor din limba română	18
3.6	Modelarea în LPI a afirmațiilor despre aritmetică	19
3.7	Fișă de exerciții	20
4	Semantica formulelor logicii de ordinul I	23
4.1	Atribuire	24
4.2	Valoarea de adevăr a unei formule de ordinul I	26
4.3	Satisfiabilitate într-o structură fixată	29
4.4	Validitate într-o structură fixată	29
4.5	Satisfiabilitate	30
4.6	Validitate	30
4.7	Consecință semantică	30
4.8	Mulțime consistentă de formule	31
4.9	Fișă de exerciții	31

Capitolul 1

Motivație și introducere

Logica de ordinul I, pe care o vom studia în continuare, este o extensie a logicii propoziționale, extensie care aduce un plus de expresivitate. Expresivitatea adițională este necesară pentru a putea modela anumite afirmații care nu pot fi exprimate în logica propozițională.

În logica propozițională, nu putem exprima într-un mod natural următoarea afirmație: *Orice om este muritor*.

Pentru a modela o afirmație în logica propozițională, identificăm întâi propozițiile atomice. Apoi asociem fiecărei propoziții atomice o variabilă propozițională. Propozițiile atomice sunt propozițiile care nu pot fi împărțite în alte propoziții mai mici, care să fie conectate între ele prin conectorii logici \neg , \wedge , \vee , \rightarrow și respectiv \leftrightarrow .

Observăm că afirmația *Orice om este muritor* nu poate fi descompusă în afirmații indivizibile legate între ele prin conectorii logicii propoziționale, după cum este descris mai sus. Așadar, în logica propozițională, afirmația este atomică. Asociem întregii afirmații o variabilă propozițională $p \in A$.

Acum să modelăm afirmația *Socrate este om*. Evident, acestei a doua afirmații trebuie să îi asociem o altă variabilă propozițională $q \in A$. Să presupunem că știm că p și q sunt adevărate. Formal, știm că lucrăm cu o atribuire $\tau : A \rightarrow B$ astfel încât $\tau(p) = 1$ și $\tau(q) = 1$. Putem trage concluzia ca afirmația *Socrate este muritor* este adevărată în atribuirea τ ?

Nu, deoarece afirmației *Socrate este muritor* ar trebui să îi asociem o a treia variabilă propozițională r și nu putem trage nicio concluzie asupra lui $\tau(r)$ din faptul că $\tau(p) = 1$ și $\tau(q) = 1$. Deci, din semantica logicii propoziționale, nu putem trage concluzia că r este adevărată în orice atribuire în care p și q sunt adevărate, în ciuda faptului că, dacă *orice om este muritor* și *Socrate este om* atunci sigur *Socrate este muritor*. Această diferență între realitate și modelarea noastră ne indică faptul că modelarea nu este suficient

de bună.

Logica de ordinul I aduce, în plus față de logica propozițională, noțiunea de *cuantificator* (existențial sau universal) și noțiunea de *predicat*. Cuantificatorul universal este notat cu \forall (de la litera *A* întoarsă – *all* în limba engleză), iar cuantificatorul existențial este notat cu \exists (de la litera *E* întoarsă – *exists* în limba engleză).

Un predicat este o afirmație a cărei valoare de adevăr depinde de zero sau mai mulți parametri. De exemplu, pentru afirmația de mai sus, vom folosi două predicate: 0 și M . Predicatul 0 va fi definit astfel: $0(x)$ va fi adevărat când x este om. Predicatul $M(x)$ este adevărat când x este muritor. Deoarece predicatele de mai sus au fiecare câte un singur argument/parametru, ele se numesc predicate *unare*. Predicatele generalizează variabilele propoziționale prin faptul că pot primi argumente. De fapt, variabilele propoziționale pot fi văzute ca predicate fără argumente.

Astfel, afirmația *orice om este muritor* va fi modelată prin formula

$$(\forall x.(0(x) \rightarrow M(x))),$$

care este citită astfel: *pentru orice x , dacă 0 de x , atunci M de x* . Afirmația *Socrate este om* va fi modelată prin formula $0(s)$, unde s este o *constantă* prin care înțelegem Socrate, la fel cum prin constanta 0 ne referim la numărul natural *zero*. De exemplu, $0(s)$ este adevărat (deoarece s denotă un om), dar $0(l)$ este fals dacă, spre exemplu, l este o constantă care ține locul cățelului *Lăbuș*.

Afirmația *Socrate este muritor* va fi reprezentată prin $M(s)$ (deoarece constanta s se referă la Socrate). Afirmația $M(s)$ este adevărată deoarece Socrate este muritor; la fel și afirmația $M(l)$ este adevărată.

Vom vedea că în logica de ordinul I, formula $M(s)$ este consecință a formulelor $(\forall x.(0(x) \rightarrow M(x)))$ și respectiv $0(s)$. În acest sens, logica de ordinul I este suficient de expresivă pentru a explica din punct de vedere teoretic raționamentul prin care putem deduce că *Socrate este muritor* din faptul că *Orice om este muritor* și din faptul că *Socrate este om*.

Capitolul 2

Structuri și semnături

Cu siguranță ați întâlnit deja mai multe formule din logica de ordinul I, fără să știți neapărat că aveți de a face cu logica de ordinul I. Fie următoare formulă:

$$\varphi = \left(\forall x. (\forall y. (x < y \rightarrow \exists z. (x < z \wedge z < y))) \right).$$

Formula folosește un simbol $<$ căruia îi corespunde un predicat binar $<$ (adică o relație binară) definit astfel: $<(x, y)$ este adevărat dacă x este mai mic strict decât y . Pentru multe predicate binare (inclusiv pentru $<$), pentru a simplifica scrierea, folosim notația infixată ($x < y$) în loc de notația prefixată ($<(x, y)$).

Este formula φ de mai sus adevărată? Formula afirmă că între orice două valori ale variabilelor x, y există o a treia valoare, a variabilei z . Formula este adevărată dacă domeniul variabilelor x, y, z este \mathbb{R} , dar este falsă dacă domeniul este \mathbb{N} (între orice două numere reale există un al treilea, dar între două numere naturale consecutive nu există niciun alt număr natural).

În general, formulele de ordinul I se referă la o anumită *structură matematică*.

Definiția 1 (Structură matematică). *O structură matematică este un triplet $S = (D, Pred, Fun)$, unde:*

- D este o mulțime nevidă numită domeniu;
- fiecare $P \in Pred$ este predicat (de o aritate oarecare) peste mulțimea D ;
- fiecare $f \in Fun$ este funcție (de o aritate oarecare) peste mulțimea D .

Iată câteva exemple de structuri matematice:

1. $(\mathbb{N}, \{<, =\}, \{+, 0, 1\})$;

Domeniul structurii este mulțimea numerelor naturale. Structura conține două predicate: $<$ și $=$, ambele de aritate 2. Predicatul $<$ este predicatul *mai mic* pe numere naturale, iar predicatul $=$ este predicatul de *egalitate* a numerelor naturale.

Funcția binară $+$: $\mathbb{N}^2 \rightarrow \mathbb{N}$ este funcția de adunare a numerelor naturale, iar structura conține și constantele $0 \in \mathbb{N}$ și $1 \in \mathbb{N}$.

2. $(\mathbb{R}, \{<, =\}, \{+, -, 0, 1\})$;

Această structură conține două predicate binare, $<$ și $=$, precum și patru funcții peste \mathbb{R} : funcția binară $+$, funcția unară $-$ și constantele $0, 1 \in \mathbb{R}$.

3. $(\mathbb{Z}, \{<, =\}, \{+, -, 0, 1\})$;

Această structură este similară cu structura precedentă, dar domeniul este mulțimea numerelor întregi.

4. $(B, \emptyset, \{., +, \bar{\cdot}\})$;

Această structură este o algebră booleană, unde domeniul este mulțimea valorilor de adevăr, iar funcțiile sunt cele cunoscute din prima jumătate a semestrului. Astfel de structuri, fără niciun predicat, se numesc *structuri algebrice*.

5. $(\mathbb{R}, \{<\}, \emptyset)$.

Această structură conține doar un predicat de aritate 2 (relația *mai mic* peste \mathbb{R}) și nicio funcție. Structurile care nu conțin funcții se numesc structuri relaționale. Structurile relaționale cu domeniul finit se mai numesc baze de date relaționale și se studiază în anul 2.

Când avem o formulă de ordinul I și dorim să îi evaluăm valoarea de adevăr, trebuie să fixăm structura în care lucrăm. Revenind la formula de mai devreme:

$$\varphi = \left(\forall x. (\forall y. (x < y \rightarrow \exists z. (x < z \wedge z < y))) \right),$$

avem că această formulă este adevărată în structura $(\mathbb{R}, \{<, =\}, \{+, -, 0, 1\})$ (între orice două numere reale distincte există cel puțin un număr real) dar este falsă în structura $(\mathbb{Z}, \{<, =\}, \{+, -, 0, 1\})$ (deoarece nu între orice două numere întregi putem găsi un alt număr întreg – de exemplu între două numere întregi consecutive nu există niciun întreg). În primul caz, domeniul variabilelor x, y, z este \mathbb{R} și simbolului $<$ îi corespunde predicatul $<_{\subseteq \mathbb{R}^2}$. În al doilea caz, domeniul variabilelor x, y, z este \mathbb{Z} și simbolului $<$ îi corespunde predicatul $<_{\subseteq \mathbb{Z}^2}$.

Este posibil ca două structuri diferite să aibă un set de predicate și de funcții cu același nume. De exemplu, chiar structurile de mai devreme, $(\mathbb{R}, \{<, =\}, \{+, -, 0, 1\})$ și respectiv $(\mathbb{Z}, \{<, =\}, \{+, -, 0, 1\})$. Deși predicatul $< \subseteq \mathbb{R}^2$ este diferit de predicatul $< \subseteq \mathbb{Z}^2$, ele au același nume: $<$.

În general, în Matematică și în Informatică, nu facem diferența între un predicat și numele lui, respectiv între o funcție și numele funcției, dar în Logică diferența este extrem de importantă. În particular, dacă ne referim la numele unei funcții vom folosi sintagma *simbol funcțional*, iar dacă ne referim la numele unui predicat vom folosi sintagma *simbol predicativ*. De ce este importantă diferența dintre un simbol predicativ și un predicat? Deoarece vom avea (ne)voie să asociem simbolului predicativ diverse predicate, analog modului în care unei variabile într-un limbaj de programare imperativ îi putem asocia diverse valori.

Când ne interesează doar numele funcțiilor și predicatelor (nu și funcțiile și respectiv predicatele în sine), vom utiliza semnături:

Definiția 2 (Signatură). *O semnatură Σ este un tuplu $\Sigma = (\mathcal{P}, \mathcal{F})$ unde \mathcal{P} este o mulțime de simboluri predicative și \mathcal{F} este o mulțime de simboluri funcționale. Fiecare simbol s (predicativ sau funcțional) are asociat un număr natural pe care îl vom numi aritatea simbolului și îl vom nota cu $ar(s)$.*

Unei semnături îi putem asocia mai multe structuri:

Definiția 3 (Σ -structuri). *Dacă $\Sigma = (\mathcal{P}, \mathcal{F})$ este o semnatură, o Σ -structură este orice structură $S = (D, Pred, Fun)$ astfel încât fiecărui simbol predicativ (sau funcțional) îi corespunde în mod unic un predicat (respectiv, o funcție).*

Exemplul 4. *Fie $\Sigma = (\{\mathbf{P}, \mathbf{Q}\}, \{\mathbf{f}, \mathbf{i}, \mathbf{a}, \mathbf{b}\})$ unde \mathbf{P}, \mathbf{Q} sunt simboluri predicative de aritate $ar(\mathbf{P}) = ar(\mathbf{Q}) = 2$ și $\mathbf{f}, \mathbf{i}, \mathbf{a}, \mathbf{b}$ sunt simboluri funcționale cu aritățile: $ar(\mathbf{f}) = 2$, $ar(\mathbf{i}) = 1$ și $ar(\mathbf{a}) = ar(\mathbf{b}) = 0$.*

Avem că $(\mathbb{R}, \{<, =\}, \{+, -, 0, 1\})$ și respectiv $(\mathbb{Z}, \{<, =\}, \{+, -, 0, 1\})$ sunt Σ -structuri.

Observație. *Dupa cum se poate observa și în Exemplul 4, pentru simboluri predicative (e.g., \mathbf{P}, \mathbf{Q}) vom utiliza o culoare diferită față de culoarea simbolurilor funcționale (e.g., $\mathbf{f}, \mathbf{i}, \mathbf{a}, \mathbf{b}$). Pentru predicatele și funcțiile din structuri vom utiliza fontul obișnuit pentru formule matematice.*

De reținut!

Structură = domeniu + predicate + funcții

Signatură = simboluri predicative + simboluri funcționale

Unei semnături Σ îi putem asocia mai multe structuri, numite Σ -structuri.

Notăție. Mulțimea simbolurilor predicative dintr-o Σ -structură de aritate n este notată cu $\mathcal{P}_n = \{P \mid \text{ar}(P) = n\}$, iar mulțimea simbolurilor funcționale de aritate n este notată cu $\mathcal{F}_n = \{f \mid \text{ar}(f) = n\}$. Pentru cazul particular $n = 0$, \mathcal{F}_0 reprezintă mulțimea simbolurilor constante (simboluri funcționale de aritate 0).

2.1 Fișă de exerciții

Exercițiul 5. *Identificați predicatele și funcțiile din textul de mai jos și stabiliți aritatea acestora. Care este domeniul acestora?*

Ion este student. Orice student învață la Logică. Oricine învață la Logică trece examenul. Orice student este om. Există un om care nu a trecut examenul. Deci nu toți oamenii sunt studenți.

Exercițiul 6. *Identificați predicatele și funcțiile din textul de mai jos și stabiliți aritatea acestora. Care este domeniul acestora?*

Suma a două numere pare este un număr par.

Exercițiul 7. *Identificați predicatele și funcțiile din textul de mai jos și stabiliți aritatea acestora. Care este domeniul acestora?*

În jocul de șah, regina poate efectua o mutare dintr-o căsuță într-alta ddacă tura sau nebunul poate efectua aceeași mutare.

Exercițiul 8. *Identificați predicatele și funcțiile din textul de mai jos și stabiliți aritatea acestora. Care este domeniul acestora?*

Suma a două numere mai mari decât zero este mai mare decât zero.

Exercițiul 9. *Identificați predicatele și funcțiile din textul de mai jos și stabiliți aritatea acestora. Care este domeniul acestora?*

Numărul 7 este prim.

Exercițiul 10. *Identificați predicatele și funcțiile din textul de mai jos și stabiliți aritatea acestora. Care este domeniul acestora?*

Orice număr par mai mare decât 2 este suma a două numere prime.

Exercițiul 11. *Identificați predicatele și funcțiile din textul de mai jos și stabiliți aritatea acestora. Care este domeniul acestora?*

Dacă Pământul este plat, atunci $2 + 2 = 5$.

Exercițiul 12. *Identificați predicatele și funcțiile din textul de mai jos și stabiliți aritatea acestora. Care este domeniul acestora?*

Pentru orice $\epsilon \in (0, \infty)$, există $\delta_\epsilon \in (0, \infty)$ astfel încât pentru orice $x \in \mathbb{R}$ cu $d(x_0, x) < \delta_\epsilon$, avem $d(f(x_0), f(x)) < \epsilon$.

Capitolul 3

Sintaxa logicii de ordinul I

În acest capitol vom prezenta sintaxa formulelor din logica cu predicate de ordinul I. Pentru logica de ordinul I limbajul (mulțimea de șiruri de simboluri) este determinat de alegerea unei semnături Σ . Practic, există mai multe limbaaje de ordinul I, câte un limbaj pentru fiecare semnătură Σ .

În continuare, vom presupune fixată o semnătură Σ cu simboluri predicative \mathcal{P} și simboluri funcționale \mathcal{F} .

3.1 Alfabetul

Ca și formulele din logica propozițională, formulele din logica de ordinul I sunt șiruri de simboluri peste un anumit alfabet. Spre deosebire de logica propozițională, alfabetul este mai bogat și conține următoarele simboluri:

1. *conectori logici* deja cunoscuți: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \perp$, precum și doi *cuantificatori*: \forall, \exists ;
2. *variabile*: vom fixa o mulțime infinit numărabilă de variabile, notată $\mathcal{X} = \{x, y, z, x', y', x_1, z'', \dots\}$ (a nu se confunda cu mulțimea variabilelor propoziționale din logica propozițională; sunt două noțiuni fundamentale diferite și din acest motiv utilizăm altă culoare pentru a le reprezenta);
3. simboluri auxiliare: $(,), \cdot, \cdot\cdot, (,),$ și $;$;
4. simboluri suplimentare, care sunt specifice fiecărei semnături $\Sigma = (\mathcal{P}, \mathcal{F})$ în parte: simbolurile funcționale din mulțimea \mathcal{F} și respectiv simbolurile predicative din mulțimea \mathcal{P} .

3.2 Termen

Definiția 13. Mulțimea termenilor, \mathcal{T} , este cea mai mică mulțime care satisface următoarele proprietăți:

1. $\mathcal{F}_0 \subseteq \mathcal{T}$ (orice simbol constant este termen);
2. $\mathcal{X} \subseteq \mathcal{T}$ (orice variabilă este termen);
3. dacă $f \in \mathcal{F}_n$ (cu $n > 0$) și $t_1, \dots, t_n \in \mathcal{T}$, atunci $f(t_1, \dots, t_n) \in \mathcal{T}$ (un simbol funcțional de aritate n aplicat unui număr de exact n termeni este termen).

Observație. Deoarece definiția mulțimii termenilor depinde de $\Sigma = (\mathcal{P}, \mathcal{F})$, elementele mulțimii \mathcal{T} se mai numesc Σ -termeni.

Practic, termenii se construiesc aplicând simboluri funcționale peste simboluri constante și variabile.

Exemplul 14. Fie signatura $\Sigma = (\{\mathbf{P}, \mathbf{Q}\}, \{\mathbf{f}, \mathbf{i}, \mathbf{a}, \mathbf{b}\})$ definită în Exemplul 4, unde $ar(\mathbf{P}) = ar(\mathbf{Q}) = 2$, $ar(\mathbf{f}) = 2$, $ar(\mathbf{i}) = 1$, $ar(\mathbf{a}) = ar(\mathbf{b}) = 0$. Iată câteva exemple de termeni: \mathbf{a} , \mathbf{b} , \mathbf{x} , \mathbf{y} , \mathbf{x}_1 , \mathbf{y}' , $\mathbf{i}(\mathbf{a})$, $\mathbf{i}(\mathbf{x})$, $\mathbf{i}(\mathbf{i}(\mathbf{a}))$, $\mathbf{i}(\mathbf{i}(\mathbf{x}))$, $\mathbf{f}(\mathbf{a}, \mathbf{b})$, $\mathbf{i}(\mathbf{f}(\mathbf{a}, \mathbf{b}))$, $\mathbf{f}(\mathbf{f}(\mathbf{x}, \mathbf{a}), \mathbf{f}(\mathbf{y}, \mathbf{y}))$.

Exercițiul 15. Identificați în lista de mai jos Σ -termenii:

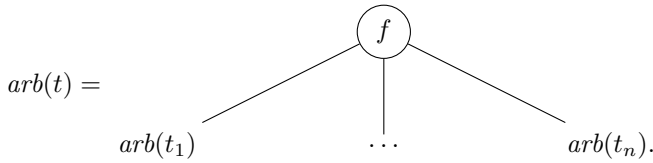
1. $\mathbf{i}(\mathbf{i}(\mathbf{x}))$;
2. \mathbf{i} ;
3. $\mathbf{f}(\mathbf{x}, \mathbf{x})$;
4. $\mathbf{P}(\mathbf{a}, \mathbf{b})$;
5. $\mathbf{i}(\mathbf{a}, \mathbf{a})$;
6. $\mathbf{f}(\mathbf{i}(\mathbf{x}), \mathbf{i}(\mathbf{x}))$;
7. $\mathbf{f}(\mathbf{i}(\mathbf{x}, \mathbf{x}))$;
8. $\mathbf{a}(\mathbf{i}(\mathbf{x}))$.

Termenii (sau, în mod echivalent, *termii*), sunt notați cu t, s, t_1, t_2, s_1, t' , etc. Deși termenii sunt scriși în mod uzual ca un șir de simboluri, ei au asociat un arbore de sintaxă abstractă definit după cum urmează:

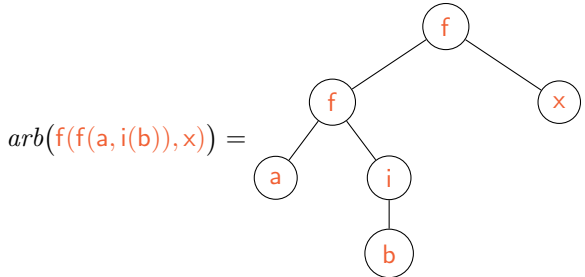
1. dacă $t = c$ și $c \in \mathcal{F}_0$, atunci $arb(t) = \bigcirc(c)$

2. dacă $t = x$ și $x \in \mathcal{X}$, atunci $arb(t) = (x)$

3. dacă $t = f(t_1, \dots, t_n)$ și $f \in \mathcal{F}_n$ ($n > 0$), $t_1, \dots, t_n \in \mathcal{T}$, atunci



Observație. Deși formal termenii sunt definiți ca fiind șiruri de simboluri peste alfabetul descris mai sus, aceștia trebuie înțeleși ca fiind arbori. De altfel, în orice software care lucrează cu termeni, aceștia sunt memorati sub formă de arbori cu rădăcină. Iată arborele atașat termenului $f(f(a, i(b)), x)$:



Exercițiul 16. Calculați arborii de sintaxă pentru termenii din Exemplul 14.

3.3 Formule atomice

Definiția 17 (Formulă atomică). O formulă atomică este orice șir de simboluri de forma $P(t_1, \dots, t_n)$, unde $P \in \mathcal{P}_n$ este un simbol predicativ de aritate $n \geq 0$, iar $t_1, \dots, t_n \in \mathcal{T}$ sunt termeni. Dacă $n = 0$, scriem P în loc de $P()$.

Exemplul 18. Continuând Exemplul 14, folosim signatura

$$\Sigma = (\{P, Q\}, \{f, i, a, b\}),$$

unde $ar(P) = ar(Q) = 2$, $ar(f) = 2$, $ar(i) = 1$, $ar(a) = ar(b) = 0$.

Iată câteva exemple de formule atomice: $P(a, b)$, $P(x, y)$, $Q(i(i(x)), f(x, x))$, $Q(a, b)$, $P(f(f(a, i(x)), b), i(x))$.

Exercițiul 19. Explicați de ce $P(a)$, $P, i(i(x))$ nu sunt formule atomice peste signatura din Exemplul 18.

3.4 Formule de ordinul I

Definiția 20 (Formule de ordinul I). *Mulțimea formulelor de ordinul I, notată \mathbb{LPI} , este cea mai mică mulțime astfel încât:*

1. (cazul de bază) orice formulă atomică este formulă (adică $P(t_1, \dots, t_n) \in \mathbb{LPI}$ pentru orice simbol predicativ $P \in \mathcal{P}_n$ și orice termeni $t_1, \dots, t_n \in \mathcal{T}$;
2. (cazurile inductive) pentru orice formule $\varphi, \varphi_1, \varphi_2 \in \mathbb{LPI}$, pentru orice variabilă $x \in \mathcal{X}$, avem că:

- (a) $\neg \varphi_1 \in \mathbb{LPI}$;
- (b) $(\varphi_1 \wedge \varphi_2) \in \mathbb{LPI}$;
- (c) $(\varphi_1 \vee \varphi_2) \in \mathbb{LPI}$;
- (d) $(\varphi_1 \rightarrow \varphi_2) \in \mathbb{LPI}$;
- (e) $(\varphi_1 \leftrightarrow \varphi_2) \in \mathbb{LPI}$;
- (f) $(\forall x. \varphi) \in \mathbb{LPI}$;
- (g) $(\exists x. \varphi) \in \mathbb{LPI}$.

Observație. În Definiția 20, regăsim conectorii logici $\neg, \wedge, \vee, \rightarrow$ și respectiv \leftrightarrow din logica propozițională. Locul variabilelor propoziționale (deocamdată, la nivel sintactic) este luat de simbolurile predicative de aritate 0. Construcțiile $(\forall x. \varphi)$ și $(\exists x. \varphi)$ sunt noi.

Exemplul 21. Continuând Exemplul 14, folosim signatură $\Sigma = (\{\mathbf{P}, \mathbf{Q}\}, \{\mathbf{f}, \mathbf{i}, \mathbf{a}, \mathbf{b}\})$, unde $ar(\mathbf{P}) = ar(\mathbf{Q}) = 2$, $ar(\mathbf{f}) = 2$, $ar(\mathbf{i}) = 1$ și $ar(\mathbf{a}) = ar(\mathbf{b}) = 0$.

Iată câteva exemple de formule din logica de ordinul I:

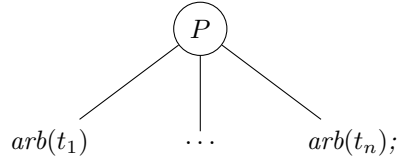
1. $\mathbf{P}(\mathbf{a}, \mathbf{b})$;
2. $\mathbf{Q}(\mathbf{a}, \mathbf{b})$;
3. $\mathbf{P}(\mathbf{a}, \mathbf{x})$;
4. $\neg \mathbf{P}(\mathbf{a}, \mathbf{b})$;
5. $(\mathbf{P}(\mathbf{a}, \mathbf{b}) \wedge \neg \mathbf{Q}(\mathbf{a}, \mathbf{b}))$;
6. $(\mathbf{P}(\mathbf{a}, \mathbf{b}) \vee \neg \mathbf{Q}(\mathbf{x}, \mathbf{y}))$;
7. $(\mathbf{P}(\mathbf{a}, \mathbf{b}) \rightarrow \mathbf{P}(\mathbf{a}, \mathbf{b}))$;
8. $((\mathbf{P}(\mathbf{a}, \mathbf{b}) \rightarrow \mathbf{P}(\mathbf{a}, \mathbf{b})) \leftrightarrow (\mathbf{P}(\mathbf{a}, \mathbf{b}) \rightarrow \mathbf{P}(\mathbf{a}, \mathbf{b})))$;

9. $(\forall x.P(a, x));$

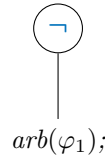
10. $(\exists x.\neg Q(x, y)).$

Definiția 22 (Arborele de sintaxă abstractă asociat formulelor din \mathbb{LPI}).
Formulele au asociat un arbore de sintaxă abstractă definit în cele ce urmează:

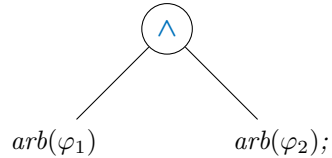
1. dacă $\varphi = P(t_1, \dots, t_n)$, atunci $arb(\varphi) =$



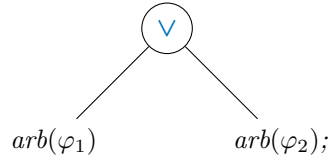
2. dacă $\varphi = \neg\varphi_1$, atunci $arb(\varphi) =$



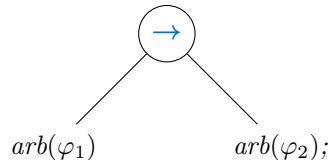
3. dacă $\varphi = (\varphi_1 \wedge \varphi_2)$, atunci $arb(\varphi) =$



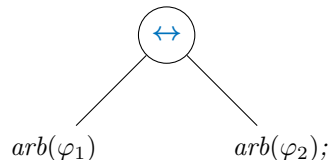
4. dacă $\varphi = (\varphi_1 \vee \varphi_2)$, atunci $arb(\varphi) =$



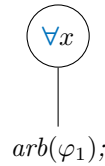
5. dacă $\varphi = (\varphi_1 \rightarrow \varphi_2)$, atunci $arb(\varphi) =$



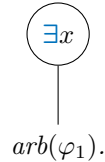
6. dacă $\varphi = (\varphi_1 \leftrightarrow \varphi_2)$, atunci $arb(\varphi) =$



7. dacă $\varphi = (\forall x.\varphi_1)$, atunci $arb(\varphi) =$



8. dacă $\varphi = (\exists x.\varphi_1)$, atunci $arb(\varphi) =$



Exercițiul 23. *Calculați arborele de sintaxă asociat formulelor din Exemplul 21.*

3.5 Modelarea în LPI a afirmațiilor din limba română

În această secțiune vom explica care este signatura folosită pentru a modela în logica de ordinul întâi afirmațiile: *Orice om este muritor*, *Socrate este om* și respectiv *Socrate este muritor*.

În primul rând, identificăm predicatele din text. Avem două predicate unare *este om* și respectiv *este muritor*. Alegem simbolul predicativ **O** pentru primul predicat și simbolul predicativ **M** pentru al doilea predicat. De asemenea, în text avem și o constantă: Socrate. Alegem simbolul funcțional **s** de aritate 0 pentru această constantă. Așadar, pentru a modela afirmațiile de mai sus, vom lucra cu signatura

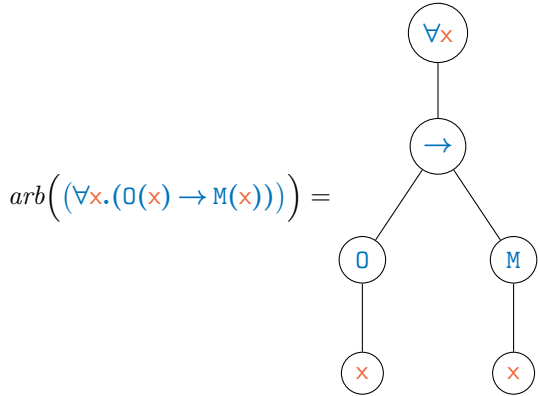
$$\Sigma = (\{\mathbf{O}, \mathbf{M}\}, \{\mathbf{s}\}),$$

unde **O** și **M** sunt simboluri predicative de aritate $ar(\mathbf{O}) = ar(\mathbf{M}) = 1$, iar **s** este un simbol funcțional de aritate $ar(\mathbf{s}) = 0$, adică un simbol constant.

Afirmația *orice om este muritor* va fi modelată prin formula de ordinul I

$$(\forall x.(\mathbf{O}(x) \rightarrow \mathbf{M}(x))),$$

al cărei arbore de sintaxă abstractă este:



Afirmația *Socrate este om* o vom modela prin formula atomică $O(s)$, iar afirmația *Socrate este muritor* o vom modela prin formula atomică $M(s)$.

Pentru signatura $\Sigma = (\{O, M\}, \{s\})$ stabilită mai sus, există mai multe Σ -structuri posibile. Un exemplu este structura $S = (D, \{O^S, M^S\}, \{s^S\})$ definită astfel:

1. D este mulțimea tuturor ființelor de pe Pământ;
2. $O^S(x)$ este adevărat pentru orice ființă x care este și om;
3. $M^S(x)$ este adevărat pentru orice ființă x (toate elementele domeniului sunt muritoare);
4. s^S este Socrate (Socrate, fiind o ființă, aparține mulțimii D).

Anticipând puțin (vom discuta despre semantica formulelor de ordinul I în Capitolul 4), toate cele trei formule discutate în această secțiune, adică $(\forall x. (O(x) \rightarrow M(x)))$, $O(s)$ și respectiv $M(s)$, sunt adevărate în structura S definită mai sus. De fapt, calitatea raționamentului *orice om este muritor; Socrate este om; deci: Socrate este muritor* este dată de faptul că formula $M(s)$ este în mod necesar adevărată în *orice* structură în care formulele $O(s)$ și $(\forall x. (O(x) \rightarrow M(x)))$ sunt adevărate, nu doar în structura S de mai sus.

3.6 Modelarea în LPI a afirmațiilor despre aritmetică

Fie signatura $\Sigma = (\{<, =\}, \{+, -, 0, 1\})$, unde $<$ și $=$ sunt simboluri predicative de aritate 2, $+$ este simbol funcțional de aritate 2, $-$ este simbol funcțional de aritate 1, iar 0 și 1 sunt simboluri constante. Iată câteva formule care fac parte din limbajul de ordinul I asociat signaturii Σ :

1. $(\forall x.(\forall y.(<(x, y) \rightarrow \exists z.(<(x, z) \wedge <(z, y)))));$
2. $(\forall x.(\forall y.(\exists z.(= (+ (x, y), z)))));$
3. $(\forall x.(<(0, x) \vee = (0, x)));$
4. $(\forall x.(\exists y.(= (x, -(y)))));$
5. $= (+ (x, y), z).$

De multe ori, în cazul simbolurilor predicative și simbolurilor funcționale binare, se folosește notația infixată (e.g., $x < y$ în loc de $<(x, y)$). În acest caz, putem scrie formulele de mai sus în felul următor:

1. $(\forall x.(\forall y.(x < y \rightarrow \exists z.(x < z \wedge z < y)))));$
2. $(\forall x.(\forall y.(\exists z.(x + y = z)))));$
3. $(\forall x.(0 < x \vee 0 = x));$
4. $(\forall x.(\exists y.(x = -(y)))));$
5. $x + y = z.$

Două dintre Σ -structurile posibile sunt $S_1 = (\mathbb{R}, \{<, =\}, \{+, -, 0, 1\})$ și $S_2 = (\mathbb{Z}, \{<, =\}, \{+, -, 0, 1\})$, unde predicatele și funcțiile sunt cele cunoscute de la matematică (cu precizarea că $-$ este funcția minus unar).

Anticipând cursul următor referitor la semantica formulelor de ordinul I, prima formulă este falsă în S_2 și adevărată în S_1 . A doua formulă și a patra formulă sunt adevărate atât în S_1 cât și în S_2 . A treia formulă este falsă atât în S_1 cât și în S_2 . Valoarea de adevăr a celei de-a cincea formule nu depinde doar de structura în care evaluăm formula, ci și de valorile variabilelor x, y, z . Deoarece variabilele x, y, z nu apar cuantificate în formula numărul 5, acestea se numesc *libere*. Formula 5 este *satisfiabilă* atât în structura S_1 cât și în structura S_2 , deoarece în ambele cazuri există valori pentru variabilele x, y, z care să facă formula adevărată (e.g. valorile 1, 2, 3 pentru x, y și respectiv z).

3.7 Fișă de exerciții

Exercițiul 24. *Identificați o semnătură pentru afirmațiile de mai jos și apoi modelați aceste afirmații ca formule în logica de ordinul I:*

Ion este student. Orice student învață la Logică. Oricine învață la Logică trece examenul. Orice student este om. Există un om care nu a trecut examenul. Deci nu toți oamenii sunt studenți.

Exercițiul 25. Fie structura $S = (\mathbb{R}, \{Nat, Int, Prim, Par, >\}, \{+, 0, 1, 2\})$, unde $Nat, Int, Prim, Par$ sunt predicate unare cu următoarea semnificație:

- $Nat(u) = u$ este număr natural;
- $Int(u) = u$ este număr întreg;
- $Prim(u) = u$ este număr prim;
- $Par(u) = u$ este număr par.

Predicatul binar $>$ este relația “mai mare” peste numere reale. Funcția $+$ este funcția de adunare a numerelor reale. Constantele $0, 1, 2$ sunt chiar numerele $0, 1, 2$.

1. Propuneți o semnătură Σ pentru structura S de mai sus.
2. Modelați următoarele afirmații ca formule de ordinul I în semnătura asociată structurii S de mai sus:
 - (a) Orice număr natural este și număr întreg.
 - (b) Suma oricăror două numere naturale este număr natural.
 - (c) Oricum am alege un număr natural, există un număr prim care este mai mare decât numărul respectiv.
 - (d) Dacă orice număr natural este număr prim, atunci zero este număr prim.
 - (e) Oricum am alege un număr prim, există un număr prim mai mare decât el.
 - (f) Suma a două numere pare este un număr par.
 - (g) Orice număr prim mai mare decât 2 este impar.
 - (h) Orice număr prim poate fi scris ca suma a patru numere prime.
 - (i) Suma a două numere pare este un număr impar.

Exercițiul 26. Dați exemplu de 5 termeni peste semnăturile de la Exercițiul 25 și calculați arborele de sintaxă abstractă al acestor termeni.

Exercițiul 27. Dați exemplu de 5 formule peste semnătura de la Exercițiul 25 și calculați arborele de sintaxă abstractă al acestora.

Exercițiul 28. *Calculați arborele de sintaxă abstractă al următoarelor formule:*

1. $(P(x) \vee (P(y) \wedge \neg P(z)))$;
2. $((\neg \neg P(x) \vee P(y)) \rightarrow (P(x) \wedge \neg P(z)))$;
3. $(\forall x. (\forall y. ((\neg \neg P(x) \vee P(y)) \rightarrow (P(x) \wedge \neg P(z)))))$;
4. $(\forall x. (\forall y. ((\neg \neg P(x) \vee P(y)) \rightarrow (\exists x. (P(x) \wedge \neg P(x)))))$;
5. $(\forall x'. \neg (\forall x. (P(x) \wedge (\exists y. ((Q(x, y) \vee \neg Q(z, z)) \rightarrow (\exists z'. P(z'))))))$.

Capitolul 4

Semantica formulelor logicii de ordinul I

Sintaxa logicii de ordinul I explică care sunt, din punct de vedere sintactic, formulele logicii de ordinul I. Semantica logicii de ordinul I se referă la *înțelesul* formulelor. Semantica unei formule (sau înțelesul formulei) va fi o valoare de adevăr. Ca și la logica propozițională, în general, valoarea de adevăr a unei formule depinde nu doar de formulă, ci și de *structura* în care evaluăm formula.

Reamintim că o semnătură $\Sigma = (\mathcal{P}, \mathcal{F})$ este o pereche formată dintr-o mulțime de simboluri predicative \mathcal{P} și o mulțime de simboluri funcționale \mathcal{F} . Fiecare simbol are atașat un număr natural numit aritatea simbolului.

În acest capitol vom utiliza semnătura $\Sigma = (\{\mathbf{P}\}, \{\mathbf{f}, \mathbf{i}, \mathbf{e}\})$, unde \mathbf{P} este simbol predicativ de aritate 2, iar \mathbf{f} , \mathbf{i} și \mathbf{e} sunt simboluri funcționale de aritate 2, 1 și respectiv 0. Altfel spus, $\mathcal{P}_2 = \{\mathbf{P}\}$, $\mathcal{P}_1 = \emptyset$, $\mathcal{P}_0 = \emptyset$, $\mathcal{F}_2 = \{\mathbf{f}\}$, $\mathcal{F}_1 = \{\mathbf{i}\}$ și $\mathcal{F}_0 = \{\mathbf{e}\}$.

Reamintim și că, dacă $\Sigma = (\mathcal{P}, \mathcal{F})$ este o semnătură, prin Σ -structură înțelegem orice tuplu $S = (D, Pred, Fun)$ cu proprietatea că:

1. D este o mulțime nevidă numită domeniul structurii S ;
2. pentru fiecare simbol predicativ $P \in \mathcal{P}$ există un predicat $P^S \in Pred$ de aritate corespunzătoare;
3. pentru fiecare simbol funcțional $f \in \mathcal{F}$ există o funcție $f^S \in Fun$ de aritate corespunzătoare.

Exemplul 29. *Mai jos avem câteva exemple de Σ -structuri:*

1. $S_1 = (\mathbb{Z}, \{=\}, \{+, -, 0\})$;

$$2. S_2 = (\mathbb{R}^*, \{=\}, \{\times, \cdot^{-1}, 1\});$$

$$3. S_3 = (\mathbb{N}, \{=\}, \{+, s, 0\});$$

$$4. S_4 = (\mathbb{N}, \{<\}, \{+, s, 0\});$$

$$5. S_5 = (\mathbb{Z}, \{<\}, \{+, -, 0\}).$$

Structura S_1 are domeniul \mathbb{Z} (mulțimea numerelor întregi), predicatul asociat simbolului predicativ \mathbf{P} este = (predicatul de egalitate pentru numere întregi), funcția + este funcția de adunare a numerelor întregi asociată simbolului funcțional \mathbf{f} , - este funcția minus unar asociată simbolului funcțional \mathbf{i} , iar simbolul constant \mathbf{e} are asociată constanta 0.

Structura S_2 are domeniul \mathbb{R}^* (mulțimea numerelor reale strict pozitive), predicatul asociat simbolului predicativ \mathbf{P} este = (predicatul de egalitate pentru numere reale pozitive), funcția \times este funcția de înmulțire a numerelor reale asociată simbolului funcțional \mathbf{f} , \cdot^{-1} este funcția unară asociată simbolului funcțional \mathbf{i} care calculează inversul unui număr real (e.g. $5^{-1} = \frac{1}{5}$, iar $\frac{1}{10}^{-1} = 10$), iar simbolul constant \mathbf{e} are asociată constanta 1.

Structura S_3 are domeniul \mathbb{N} (mulțimea numerelor naturale), predicatul asociat simbolului predicativ \mathbf{P} este = (predicatul de egalitate pentru numere naturale), funcția + este funcția de adunare a numerelor naturale asociată simbolului funcțional \mathbf{f} , s este funcția succesor (care asociază unui număr natural următorul număr natural - e.g., $s(7) = 8$) asociată simbolului funcțional \mathbf{i} , iar simbolul constant \mathbf{e} are asociată constanta 0.

Structura S_4 are domeniul \mathbb{N} (mulțimea numerelor naturale), predicatul asociat simbolului predicativ \mathbf{P} este $<$ (relația mai mic peste numere naturale), funcția + este funcția de adunare a numerelor naturale asociată simbolului funcțional \mathbf{f} , s este funcția succesor (care asociază unui număr natural următorul număr natural - e.g., $s(7) = 8$) asociată simbolului funcțional \mathbf{i} , iar simbolul constant \mathbf{e} are asociată constanta 0.

Structura S_5 este similară cu S_1 , doar că simbolul predicativ \mathbf{P} are asociată relația mai mic în loc de egal.

Folosind notațiile de mai sus, avem că $\mathbf{P}^{S_4} = <$, $\mathbf{f}^{S_2} = \times$, iar $\mathbf{e}^{S_1} = 0$.

4.1 Atribuiri

Asemănător cu logica propozițională, pentru a obține valoarea de adevăr a unei formule într-o structură, trebuie să pornim cu fixarea unor valori concrete pentru simbolurile sintactice din alfabetul peste care este construită formula. În cazul de față, începem cu variabilele.

Definiția 30 (Atribuire). *Fie Σ o semnătură și S o Σ -structură cu domeniul D . Se numește S -atribuire orice funcție*

$$\alpha : \mathcal{X} \rightarrow D.$$

Exemplul 31. *Funcția $\alpha_1 : \mathcal{X} \rightarrow \mathbb{Z}$, definită ca mai jos, este o S_1 -atribuire:*

1. $\alpha_1(\mathbf{x}_1) = 5$;
2. $\alpha_1(\mathbf{x}_2) = 5$;
3. $\alpha_1(\mathbf{x}_3) = 6$;
4. $\alpha_1(x) = 0$ pentru orice $x \in \mathcal{X} \setminus \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$.

Exemplul 32. *Funcția $\alpha_2 : \mathcal{X} \rightarrow \mathbb{Z}$, definită ca mai jos, este o S_1 -atribuire:*

1. $\alpha_2(\mathbf{x}_1) = 6$;
2. $\alpha_2(\mathbf{x}_2) = 5$;
3. $\alpha_2(\mathbf{x}_3) = 6$;
4. $\alpha_2(x) = 0$ pentru orice $x \in \mathcal{X} \setminus \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$.

Acum, având la dispoziție o atribuire α , putem calcula valoarea unui termen într-o asemenea atribuire. Pentru aceasta, vom folosi de fapt extensia lui α , notată $\bar{\alpha}$,

$$\bar{\alpha} : \mathcal{T} \rightarrow D,$$

dată în definiția care urmează.

Definiția 33 (Valoarea unui termen într-o atribuire). *Dându-se o S -atribuire α și un termen $t \in \mathcal{T}$ peste semnatura Σ , valoarea termenului t în atribuirea α este un element al domeniului D notat cu $\bar{\alpha}(t)$ și calculat recursiv astfel:*

1. $\bar{\alpha}(c) = c^S$ dacă $c \in \mathcal{F}_0$ (i.e., c este un simbol constant);
2. $\bar{\alpha}(x) = \alpha(x)$ dacă $x \in \mathcal{X}$ (i.e., x este o variabilă);
3. $\bar{\alpha}(f(t_1, \dots, t_n)) = f^S(\bar{\alpha}(t_1), \dots, \bar{\alpha}(t_n))$ dacă $f \in \mathcal{F}_n$ este un simbol funcțional de aritate n , iar t_1, \dots, t_n sunt termeni.

Exemplul 34. Continuând Exemplul 31, unde α_1 este o S_1 -atribuire, avem:

$$\begin{aligned} \overline{\alpha_1}(\mathbf{f}(\mathbf{i}(\mathbf{x}_1), \mathbf{e})) &= \overline{\alpha_1}(\mathbf{i}(\mathbf{x}_1)) + \overline{\alpha_1}(\mathbf{e}) \\ &= -(\overline{\alpha_1}(\mathbf{x}_1)) + \mathbf{e}^{S_1} \\ &= -(\alpha_1(\mathbf{x}_1)) + 0 \\ &= -5 + 0 \\ &= -5. \end{aligned}$$

Așadar, valoarea termenului $\mathbf{f}(\mathbf{i}(\mathbf{x}_1), \mathbf{e})$ în atribuirea α_1 este -5 .

Definiția 35 (Actualizarea unei atribuii). Dându-se o atribuire α , o variabilă $x \in \mathcal{X}$ și un element $u \in D$, notăm cu $\alpha[x \mapsto u]$ o nouă atribuire, care coincide cu α , exceptând valoarea variabilei x , care devine acum u :

$$\alpha[x \mapsto u] : \mathcal{X} \rightarrow D, \text{ a.î.}$$

1. $(\alpha[x \mapsto u])(x) = u$;
2. $(\alpha[x \mapsto u])(y) = \alpha(y)$, pentru orice $y \in \mathcal{X} \setminus \{x\}$.

Exemplul 36. De exemplu, atribuirea $\alpha_1[x_1 \mapsto 6]$ este exact atribuirea α_2 definită în exemplele de mai sus. Valoarea termenului $\mathbf{f}(\mathbf{i}(\mathbf{x}_1), \mathbf{e})$ în atribuirea $\alpha_1[x_1 \mapsto 6]$, notată cu $\overline{\alpha_1[x_1 \mapsto 6]}(\mathbf{f}(\mathbf{i}(\mathbf{x}_1), \mathbf{e}))$, este -6 .

Exercițiul 37. Calculați valorile de mai jos:

1. $\overline{\alpha_1[x_1 \mapsto 10]}(\mathbf{f}(\mathbf{i}(\mathbf{x}_1), \mathbf{e}))$;
2. $\overline{\alpha_1[x_2 \mapsto 10]}(\mathbf{f}(\mathbf{i}(\mathbf{x}_1), \mathbf{e}))$;
3. $\overline{\alpha_1[x_2 \mapsto 10][x_1 \mapsto 10]}(\mathbf{f}(\mathbf{i}(\mathbf{x}_1), \mathbf{e}))$.

4.2 Valoarea de adevăr a unei formule de ordinul I

În acest moment avem ingredientele pentru a defini formal valoarea de adevăr a unei formule de ordinul I, construită peste o semnătură Σ . Această valoare se poate calcula doar într-o Σ -structură S , cu ajutorul unei S -atribuii α .

Notațiile folosite sunt similare cu cele pentru logica propozițională. Astfel, notăm faptul că o formulă φ este adevărată într-o structură S cu o atribuire α prin $S, \alpha \models \varphi$. Faptul că o formulă φ nu este adevărată într-o structură S cu o atribuire α se notează cu $S, \alpha \not\models \varphi$.

Notația $S, \alpha \models \varphi$ se mai citește *S satisface φ cu atribuirea α* , iar $S, \alpha \not\models \varphi$ se mai citește *S nu satisface φ cu atribuirea α* .

Definiția 38. *Faptul că o structură S satisface o formulă φ cu o anumită atribuire α (echivalent, φ este adevărată în structura S cu atribuirea α) se definește inductiv astfel (prima linie din enumerarea care urmează desemnează cazul de bază, restul reprezentând cazurile inductive):*

1. $S, \alpha \models P(t_1, \dots, t_n)$ ddacă $P^S(\bar{\alpha}(t_1), \dots, \bar{\alpha}(t_n))$;
2. $S, \alpha \models \neg\varphi$ ddacă $S, \alpha \not\models \varphi$;
3. $S, \alpha \models (\varphi_1 \wedge \varphi_2)$ ddacă $S, \alpha \models \varphi_1$ și $S, \alpha \models \varphi_2$;
4. $S, \alpha \models (\varphi_1 \vee \varphi_2)$ ddacă $S, \alpha \models \varphi_1$ sau $S, \alpha \models \varphi_2$;
5. $S, \alpha \models (\varphi_1 \rightarrow \varphi_2)$ ddacă $S, \alpha \not\models \varphi_1$ sau $S, \alpha \models \varphi_2$;
6. $S, \alpha \models (\varphi_1 \leftrightarrow \varphi_2)$ ddacă (1) atât $S, \alpha \models \varphi_1$, cât și $S, \alpha \models \varphi_2$, sau (2) $S, \alpha \not\models \varphi_1$ și $S, \alpha \not\models \varphi_2$;
7. $S, \alpha \models (\exists x.\varphi)$ ddacă există $u \in D$ astfel încât $S, \alpha[x \mapsto u] \models \varphi$;
8. $S, \alpha \models (\forall x.\varphi)$ ddacă pentru orice $u \in D$, avem că $S, \alpha[x \mapsto u] \models \varphi$.

Exemplul 39. *Vom lucra în continuare peste signatura $\Sigma = (\{P\}, \{f, i, e\})$, Σ -structura $S_1 = (\mathbb{Z}, \{=\}, \{+, -, 0\})$ definită la începutul capitoului și S_1 -atribuirile α_1, α_2 .*

Avem că

$$\begin{array}{ll}
 S_1, \alpha_1 \models P(\mathbf{x}_1, \mathbf{x}_1) & \text{ddacă} \quad P^{S_1}(\bar{\alpha}_1(\mathbf{x}_1), \bar{\alpha}_1(\mathbf{x}_1)) \\
 & \text{ddacă} \quad \bar{\alpha}_1(\mathbf{x}_1) = \bar{\alpha}_1(\mathbf{x}_1) \\
 & \text{ddacă} \quad \alpha_1(\mathbf{x}_1) = \alpha_1(\mathbf{x}_1) \\
 & \text{ddacă} \quad 5 = 5.
 \end{array}$$

Din moment ce $5 = 5$, rezultă că $S_1, \alpha_1 \models P(\mathbf{x}_1, \mathbf{x}_1)$, adică formula $P(\mathbf{x}_1, \mathbf{x}_1)$ este adevărată în structura S_1 cu atribuirea α_1 . Altfel spus, S_1 satisface $P(\mathbf{x}_1, \mathbf{x}_1)$ cu atribuirea α_1 .

Exemplul 40. *Continuând exemplul anterior, avem*

$$\begin{array}{ll}
 S_1, \alpha_1 \models P(\mathbf{x}_1, \mathbf{x}_3) & \text{ddacă} \quad P^{S_1}(\bar{\alpha}_1(\mathbf{x}_1), \bar{\alpha}_1(\mathbf{x}_3)) \\
 & \text{ddacă} \quad \bar{\alpha}_1(\mathbf{x}_1) = \bar{\alpha}_1(\mathbf{x}_3) \\
 & \text{ddacă} \quad \alpha_1(\mathbf{x}_1) = \alpha_1(\mathbf{x}_3) \\
 & \text{ddacă} \quad 5 = 6.
 \end{array}$$

Din moment ce $5 \neq 6$, rezultă că $S_1, \alpha_1 \not\models P(x_1, x_3)$, adică formula $P(x_1, x_3)$ este falsă în structura S_1 cu atribuirea α_1 . Altfel spus S_1 nu satisface $P(x_1, x_3)$ cu atribuirea α_1 .

Exemplul 41. Continuând exemplul anterior, avem

$S_1, \alpha_1 \models \neg P(x_1, x_3)$	ddacă	$S_1, \alpha_1 \not\models P(x_1, x_3)$
	ddacă	nu $P^{S_1}(\overline{\alpha_1}(x_1), \overline{\alpha_1}(x_3))$
	ddacă	nu $\overline{\alpha_1}(x_1) = \overline{\alpha_1}(x_3)$
	ddacă	$\overline{\alpha_1}(x_1) \neq \overline{\alpha_1}(x_3)$
	ddacă	$\alpha_1(x_1) \neq \alpha_1(x_3)$
	ddacă	$5 \neq 6$.

Din moment ce $5 \neq 6$, rezultă că $S_1, \alpha_1 \models \neg P(x_1, x_3)$, adică formula $\neg P(x_1, x_3)$ este adevărată în structura S_1 cu atribuirea α_1 . Altfel spus, S_1 satisface $\neg P(x_1, x_3)$ cu atribuirea α_1 .

Exemplul 42. Continuând exemplul anterior, avem

$S_1, \alpha_1 \models P(x_1, x_1) \wedge \neg P(x_1, x_3)$	ddacă
$S_1, \alpha_1 \models P(x_1, x_1)$ și $S_1, \alpha_1 \models \neg P(x_1, x_3)$	ddacă
... și ...	ddacă
$5 = 5$ și $5 \neq 6$.	

Din moment ce $5 = 5$ și $5 \neq 6$, rezultă că $S_1, \alpha_1 \models P(x_1, x_1) \wedge \neg P(x_1, x_3)$.

Exemplul 43. Continuând exemplul anterior, avem

$S_1, \alpha_1 \models P(x_1, x_3) \vee P(x_1, x_1)$ dacă $S_1, \alpha_1 \models P(x_1, x_3)$ sau $S_1, \alpha_1 \models P(x_1, x_1)$.

Am stabilit deja că $S_1, \alpha_1 \models P(x_1, x_3)$, deci $S_1, \alpha_1 \models P(x_1, x_3) \vee P(x_1, x_1)$ (chiar dacă $S_1, \alpha_1 \not\models P(x_1, x_1)$).

Exemplul 44. Continuând exemplul anterior, avem

$S_1, \alpha_1 \models \exists x_1. P(x_1, x_3)$	ddacă
există $u \in D$ a.î. $S_1, \alpha_1[x_1 \mapsto u] \models P(x_1, x_3)$	ddacă
există $u \in D$ a.î. $P^{S_1}(\overline{\alpha_1[x_1 \mapsto u]}(x_1), \overline{\alpha_1[x_1 \mapsto u]}(x_3))$	ddacă
există $u \in D$ a.î. $\overline{\alpha_1[x_1 \mapsto u]}(x_1) = \overline{\alpha_1[x_1 \mapsto u]}(x_3)$	ddacă
există $u \in D$ a.î. $\alpha_1[x_1 \mapsto u](x_1) = \alpha_1[x_1 \mapsto u](x_3)$	ddacă
există $u \in D$ a.î. $u = \alpha_1(x_3)$	ddacă
există $u \in D$ a.î. $u = 6$.	

Din moment ce există u (putem alege $u = 6$) a.î. $u = 6$, avem că $S_1, \alpha_1 \models \exists x_1. P(x_1, x_3)$.

Exemplul 45. Continuând exemplul anterior, avem

$S_1, \alpha_1 \models \forall x_1. \exists x_3. P(x_1, x_3)$ ddacă
 pentru orice $u \in D$, avem că $S_1, \alpha_1[x_1 \mapsto u] \models \exists x_3. P(x_1, x_3)$ ddacă
 pt. orice $u \in D$, există $v \in D$ a.î. $S_1, \alpha_1[x_1 \mapsto u][x_3 \mapsto v] \models P(x_1, x_3)$ ddacă
 ... ddacă
 pentru orice $u \in D$, avem că există $v \in D$ a.î. $u = v$.

Din moment ce pentru orice număr întreg u , există un număr întreg v a.î. $u = v$, avem că $S_1, \alpha_1 \models \forall x_1. \exists x_3. P(x_1, x_3)$.

Exercițiul 46. Arătați că $S_1, \alpha_1 \models \forall x_1. \exists x_3. P(x_1, i(x_3))$.

4.3 Satisfiabilitate într-o structură fixată

Definiția 47 (Satisfiabilitate într-o structură fixată). O formulă φ este satisfiabilă într-o structură S dacă există o S -atribuire α cu proprietatea că

$$S, \alpha \models \varphi.$$

Exemplul 48. Formula $P(x_1, x_3)$ este satisfiabilă în structura S_1 , deoarece există o atribuire, de exemplu α_2 , cu proprietatea că $S_1, \alpha_2 \models P(x_1, x_3)$.

Exercițiul 49. Arătați că formula $\neg P(x_1, x_1)$ nu este satisfiabilă în structura S_1 (deoarece, pentru orice atribuire α aleasă, avem că $S_1, \alpha \not\models \neg P(x_1, x_1)$).

4.4 Validitate într-o structură fixată

Definiția 50 (Validitate într-o structură fixată). O formulă φ este validă într-o structură S dacă pentru orice S -atribuire α , avem că

$$S, \alpha \models \varphi.$$

Exemplul 51. Formula $P(x_1, x_3)$ nu este validă în structura S_1 , deoarece există o atribuire, și anume α_1 , cu proprietatea că $S_1, \alpha_1 \not\models P(x_1, x_3)$.

Exercițiul 52. Arătați că formula $P(x_1, x_1)$ este validă în structura S_1 (deoarece orice atribuire α am alege, $S_1, \alpha \models P(x_1, x_1)$).

4.5 Satisfiabilitate

Definiția 53 (Satisfiabilitate). *O formulă φ este satisfiabilă dacă există o structură S și o S -atribuire α cu proprietatea că*

$$S, \alpha \models \varphi.$$

Exemplul 54. Formula $\neg P(x_1, x_1)$ este satisfiabilă, deoarece există o structură (de exemplu S_5) și o S_5 -atribuire (de exemplu α_1) astfel încât $S_5, \alpha_1 \models \neg P(x_1, x_1)$ (deoarece $5 \not\leq 5$).

Să subliniem faptul că, deoarece S_5 și S_1 au același domeniu, atribuirea α_1 este atât o S_1 -atribuire cât și o S_5 -atribuire.

Observație. O formulă poate să nu fie satisfiabilă într-o structură fixată (de exemplu $\neg P(x_1, x_1)$ nu este satisfiabilă în structura S_1) și totuși să fie satisfiabilă (vezi Exemplul 54 unde aceeași formulă $\neg P(x_1, x_1)$ este satisfiabilă).

4.6 Validitate

Definiția 55 (Validitate). *O formulă φ este validă dacă, pentru orice structură S și pentru orice S -atribuire α , avem*

$$S, \alpha \models \varphi.$$

Exemplul 56. Formula $P(x_1, x_1)$ nu este validă, deoarece $S_5, \alpha_1 \not\models P(x_1, x_1)$. Pe de altă parte, formula $P(x_1, x_1) \rightarrow P(x_1, x_1)$ este validă.

Observație. O formulă poate să fie validă într-o structură fixată (de exemplu $P(x_1, x_1)$ este validă în structura S_1) și totuși să nu fie validă (de exemplu, $P(x_1, x_1)$ nu este validă, deoarece $S_5, \alpha_1 \not\models P(x_1, x_1)$).

4.7 Consecință semantică

Definiția 57. *O formulă φ este consecință semantică a formulelor $\varphi_1, \dots, \varphi_n$ într-o structură fixată S , notat $\varphi_1, \dots, \varphi_n \models_S \varphi$, dacă, pentru orice S -atribuire α pentru care $S, \alpha \models \varphi_1$, $S, \alpha \models \varphi_2$, ..., $S, \alpha \models \varphi_n$, avem că $S, \alpha \models \varphi$.*

Exemplul 58. Avem că $P(x, y) \models_{S_1} P(y, x)$, deoarece, pentru orice S_1 -atribuire α cu proprietatea că $S_1, \alpha \models P(x, y)$ (adică $\alpha(x) = \alpha(y)$), avem și că $S_1, \alpha \models P(y, x)$ (adică $\alpha(y) = \alpha(x)$).

Avem că $P(x, y) \not\models_{S_5} P(y, x)$, deoarece, pentru atribuirea $\alpha(x) = 5$, $\alpha(y) = 6$, avem că $S_5, \alpha \models P(x, y)$ (adică $5 < 6$), dar $S_5, \alpha \not\models P(y, x)$ ($6 \not< 5$).

Definiția 59. *O formulă φ este consecință semantică a formulelor $\varphi_1, \dots, \varphi_n$, notat $\varphi_1, \dots, \varphi_n \models \varphi$, dacă*

$$\varphi_1, \dots, \varphi_n \models_S \varphi$$

pentru orice structură S .

Exemplul 60. *Avem că $P(x, y) \not\models P(y, x)$, deoarece există o structură (și anume S_5) astfel încât $P(x, y) \models_{S_5} P(y, x)$.*

Exercițiul 61. *Arătați că:*

$$\forall x. \forall y. \forall z. (P(x, y) \wedge P(y, z) \rightarrow P(x, z)), P(x_1, x_2), P(x_2, x_3) \models P(x_1, x_3).$$

Desigur că, în cele de mai sus (similar cu logica propozițională), lista $\varphi_1, \varphi_2, \dots, \varphi_n$ denotă de fapt o mulțime având respectivele elemente.

4.8 Mulțime consistentă de formule

Definiția 62. *O mulțime de formule Γ este consistentă într-o structură S dacă, prin definiție,*

$$\text{există } \alpha \text{ a.î. } S, \alpha \models \varphi \text{ pentru orice } \varphi \in \Gamma.$$

Exemplul 63. *Avem că $\{P(x, y), P(y, x)\}$ este consistentă în S_1 , dar nu în S_4 .*

Definiția 64. *O mulțime de formule Γ este consistentă dacă, prin definiție, există o structură în care este consistentă.*

Exemplul 65. *Avem că $\{P(x, y), P(y, x)\}$ este consistentă (deoarece este consistentă în măcar o structură, de exemplu S_1).*

Exemplul 66. *Avem că $\{P(x, y), \neg P(x, y)\}$ nu este consistentă.*

4.9 Fișă de exerciții

Amintim mai jos structurile din Exemplul 29:

1. $S_1 = (\mathbb{Z}, \{=\}, \{+, -, 0\})$;
2. $S_2 = (\mathbb{R}^*, \{=\}, \{\times, \cdot^{-1}, 1\})$;
3. $S_3 = (\mathbb{N}, \{=\}, \{+, s, 0\})$;
4. $S_4 = (\mathbb{N}, \{<\}, \{+, s, 0\})$.

$$5. S_5 = (\mathbb{Z}, \{<\}, \{+, -, 0\}).$$

Aceste structuri vor fi utilizate în exercițiile de mai jos.

Exercițiul 67. *Stabiliți dacă:*

1. $S_1, \alpha_1 \models P(x_2, x_3)$;
2. $S_1, \alpha_1 \models \neg P(x_2, x_3)$;
3. $S_1, \alpha_1 \models \neg P(x_2, x_3) \wedge P(x_1, x_1)$;
4. $S_1, \alpha_1 \models \exists x_3. P(x_2, x_3)$;
5. $S_1, \alpha_1 \models \forall x_2. \exists x_3. P(x_2, x_3)$;
6. $S_1, \alpha_1 \models \exists x_3. \forall x_2. P(x_2, x_3)$;
7. $S_1, \alpha_2 \models \forall x_2. \exists x_3. P(x_2, i(x_3))$;

Exercițiul 68. *Găsiți pentru fiecare dintre itemii de mai jos câte o S_2 -atribuire α_3 astfel încât:*

1. $S_2, \alpha_3 \models P(x_1, x_2)$;
2. $S_2, \alpha_3 \models P(f(x_1, x_2), x_3)$;
3. $S_2, \alpha_3 \models P(f(x_1, x_2), i(x_3))$;
4. $S_2, \alpha_3 \models P(x, e)$;
5. $S_2, \alpha_3 \models \exists y. P(x, i(y))$;
6. $S_2, \alpha_3 \models \forall y. \exists x. P(x, i(y))$.

Exercițiul 69. *Arătați că următoarele formule sunt valide în S_2 :*

1. $\forall x. \exists y. P(x, i(y))$;
2. $\forall x. P(f(x, e), x)$;
3. $\forall x. P(x, i(i(x)))$.

Exercițiul 70. *Arătați că formula $\forall x. \exists y. P(x, i(y))$ nu este validă în S_3 .*

Exercițiul 71. *Găsiți o formulă care să fie satisfiabilă în S_1 dar nu în S_3 .*

Exercițiul 72. *Arătați că formula $\forall x. \exists y. P(x, y)$ nu este validă.*

Exercițiul 73. *Arătați că formula $(\forall x. P(x, x)) \rightarrow \exists x_2. P(x_1, x_2)$ este validă.*

Exercițiul 74. *Arătați că formula $\forall x. \exists y. P(y, x)$ nu este validă.*

Exercițiul 75. *Arătați că formula $\forall x. \neg P(x, x)$ este satisfiabilă.*

Exercițiul 76. *Arătați că formula $\forall x. \neg P(x, x) \wedge \exists x. P(x, x)$ nu este satisfiabilă.*