Logică Matematică şi Computaţională
Cursurile XI–XIV

Claudia MUREȘAN
cmuresan@fmi.unibuc.ro, c.muresan@yahoo.com

Universitatea din București
Facultatea de Matematică și Informatică
București

2021–2022, Semestrul I

Cuprinsul acestui curs

Ce este un predicat?

- În a doua parte a acestui curs vom studia **logica clasică a predicatelor**.
- **Predicatele** se mai numesc **propoziții cu variabile**.
- **Exemplu de predicat:** “x este un număr prim“ este un predicat cu variabila x; acest predicat nu are o valoare de adevăr; pentru a obține din el un enunț cu valoare de adevăr, trebuie să–i dăm valori variabilei x.
- Variabilei x i se indică un domeniu al valorilor posibile, de exemplu $\mathbb{N}$. Se dorește ca, prin înlocuirea lui x din acest predicat cu o valoare arbitrară din acest domeniu, să se obțină o propoziție adevărată sau falsă.
- Înlocuind în acest predicat pe $x := 2 \in \mathbb{N}$, se obține propoziția adevărată “2 este un număr prim“, iar înlocuind $x := 4 \in \mathbb{N}$, se obține propoziția falsă “4 este un număr prim“.
- Dacă desemnăm pe $\mathbb{N}$ ca domeniu al valorilor pentru variabila x din predicatul de mai sus, atunci putem să aplicăm un cuantificator acestei variabile, obținând, astfel, tot un enunț cu valoare de adevăr: propoziția $(\forall\, x \in \mathbb{N})\,(x$ este un număr prim$)$ este falsă, în timp ce propoziția $(\exists\, x \in \mathbb{N})\,(x$ este un număr prim$)$ este adevărată.

Cum vom scrie predicatele complet formalizat, i.e. numai în simboluri din alfabetul logicii clasice a predicatelor?

- De fapt, în enunțuri cuantificate, vom considera domeniul valorilor variabilelor stabilit de la început, și nu–l vom mai preciza după variabilele cuantificate ("$\in \mathbb{N}$"), însă vom interpreta (evalua) predicatele în structuri algebrice având ca mulțimi suport (i.e. mulțimi de elemente) mulțimile în care iau valori variabilele.
- Așadar, pentru a exprima matematic modul de lucru cu predicate, avem nevoie nu numai de noțiunea de propoziție cu sau fără variabile și de conectori logici, ci și de un domeniu al valorilor pentru variabilele care apar în predicate (i. e. în propozițiile cu variabile).
- Dorim să scriem predicate precum cele de mai sus complet **formalizat**, adică numai în simboluri din alfabetul logicii clasice a predicatelor. De exemplu, în structura algebrică $\mathcal{N} = (\mathbb{N}, |, 0)$ înzestrată cu relația de ordine $|$ ("divide pe") și constanta $0 \in \mathbb{N}$, vom putea scrie predicatul "x este un număr prim" sub forma:

$$(\forall\, y)\,(\neg\,(y{=}0) \rightarrow y\,|\,x),$$

în care variabila y, de sub incidența acelui cuantificator universal, este "legată" (trebuie să parcurgem cu ea mulțimea suport a structurii algebrice în care evaluăm predicatul), iar variabila x este "liberă" (avem libertatea de a–i da valori în

În ce fel de structuri algebrice pot lua valori variabilele?

mulțimea suport a acelei structuri algebrice). După cum vom vedea, de exemplu, faptul că un anumit număr $p \in \mathbb{N}$ este prim se va scrie sub forma: structura algebrică $\mathcal{N}$ satisface predicatul de mai sus în care variabila x este înlocuită cu numărul p privit drept constantă adăugată în această structură algebrică, scris formalizat astfel:

$$\mathcal{N} \vDash ((\forall\, y)\,(\neg\,(y=0) \rightarrow y \mid x))\,[{}^{x}_{p}]$$

sau, odată introdusă constanta p în această structură algebrică, simplu, astfel:

$$\mathcal{N} \vDash (\forall\, y)\,(\neg\,(y=0) \rightarrow y \mid p)$$

de fapt, cu notația pe care o vom stabili nesimplificată:

$$\mathcal{N} \vDash (\forall\, y)\,(\neg\,(y=0) \rightarrow y \mid p^{ct})$$

- Pentru a descrie sistemul formal al calculului cu predicate clasic, vom avea nevoie de noțiunea de **structură de ordinul I**, reprezentând structuri algebrice generice, în care avem o mulțime suport, operații cu argumente, constante (i.e. operații fără argumente, operații zeroare) și relații. Variabilele vor fi considerate ca luând valori în diverse **structuri de ordinul I**, și **clasa structurilor de ordinul I de un anumit tip** va avea asociată propria ei logică clasică cu predicate (îi vom asocia un limbaj, apoi un sistem logic bazat pe acel limbaj).

Orientativ, ce sunt structurile de ordinul I

- Când ne vom referi la **structuri de ordinul I generice**, ne vor interesa doar **aritățile** (i.e. numerele de argumente ale) **operațiilor** și **relațiilor** și **numărul de constante**, iar **proprietățile aritmetice** ale structurilor algebrice (de exemplu idempotența, comutativitatea, asociativitatea și absorbția operațiilor binare de latici) le vom exprima prin **predicate** pe care le vom evalua în **structuri de ordinul I concrete** (de exemplu, vom spune că laticile sau "această latice" satisfac(e) idempotența, comutativitatea, asociativitatea și absorbția pentru operațiile binare).
- **Structurile de ordinul I** sunt structuri algebrice care posedă o mulțime suport și operații, relații și constante (operații zeroare) pe această mulțime suport, i. e. operații și relații care acționează (numai) asupra elementelor mulțimii suport (de exemplu grupuri, poseturi, latici, algebre Boole, dar, daca, de exemplu, la o algebră Boole adăugăm o operație care îi duce submulțimile în filtrele generate de ele, atunci obținem o **structură de ordinul II**).
- Când, într–o structură algebrică, există operații sau relații care acționează asupra submulțimilor mulțimii suport, i. e. asupra unor mulțimi de elemente din mulțimea suport, atunci spunem că structura respectivă este o **structură de ordinul II**. În același mod (referindu–ne la mulțimi de mulțimi de elemente ș. a. m. d.) pot fi definite **structurile de ordinul III, IV etc.**.

Structuri algebrice înzestrate cu operații și relații

Definiție

O *structură de ordinul I* este o structură de forma

$$\mathcal{A} = (A, (f_i)_{i \in I}, (R_j)_{j \in J}, (c_k)_{k \in K}),$$

unde:

- A este o mulțime nevidă, numită *universul structurii* $\mathcal{A}$
- I, J, K sunt mulțimi oarecare de indici (care pot fi și vide)
- pentru fiecare $i \in I$, există $n_i \in \mathbb{N}^*$, a. î. $f_i : A^{n_i} \to A$ (f_i este o operație n_i–ară pe A)
- pentru fiecare $j \in J$, există $m_j \in \mathbb{N}^*$, a. î. $R_j \subseteq A^{m_j}$ (R_j este o relație m_j–ară pe A)
- pentru fiecare $k \in K$, $c_k \in A$ (c_k este o constantă din A)

În general, operațiilor și relațiilor din componența lui $\mathcal{A}$ li se atașează indicele $\mathcal{A}$, pentru a le deosebi de simbolurile de operații și relații din limbajul pe care îl vom construi în continuare; astfel, structura de ordinul I de mai sus se notează, de regulă, sub forma: $\mathcal{A} = (A, (f_i^{\mathcal{A}})_{i \in I}, (R_j^{\mathcal{A}})_{j \in J}, (c_k^{\mathcal{A}})_{k \in K})$.

Tipuri de astfel de structuri algebrice și limbaje asociate lor

Definiție

Tipul sau *signatura* structurii de ordinul I $\mathcal{A}$ din definiția anterioară este tripletul de familii de numere naturale: $\tau := ((n_i)_{i\in I}; (m_j)_{j\in J}; (0)_{k\in K})$.
Orice structură de forma lui $\mathcal{A}$ de mai sus se numește *structură de ordinul I de tipul (sau signatura)* τ.

Exemplu

- Orice poset nevid este o structură de ordinul I de forma $\mathcal{P} = (P, \leq)$, de tipul (signatura) $\tau_1 = (\emptyset; 2; \emptyset)$ ($\leq$ este o relație binară). **Nu orice** structură de ordinul I de signatura τ_1 este un poset.
- Orice latice nevidă este o structură de ordinul I de forma $\mathcal{L} = (L, \vee, \wedge, \leq)$, de tipul (signatura) $\tau_2 = (2, 2; 2; \emptyset)$ ($\vee$ și $\wedge$ sunt operații binare (i. e. de aritate 2, i. e. cu câte două argumente), iar $\leq$ este o relație binară). **Nu orice** structură de ordinul I de signatura τ_2 este o latice.
- Orice algebră Boole este o structură de ordinul I de forma $\mathcal{B} = (B, \vee, \wedge, \bar{\ }, \leq, 0, 1)$, de tipul (signatura) $\tau_3 = (2, 2, 1; 2; 0, 0)$ ($\vee$ și $\wedge$ sunt operații binare, $\bar{\ }$ este o operație unară, $\leq$ este o relație binară, iar 0 și 1 sunt constante (operații zeroare, i. e. de aritate zero, i. e. fără argumente)). **Nu orice** structură de ordinul I de signatura τ_3 este o algebră Boole.

- Fiecărei signaturi τ a unei structuri de ordinul I (fiecărei clase de structuri de ordinul I de o anumită signatură τ) i se asociază un limbaj, numit **limbaj de ordinul I** și notat, de obicei, cu $\mathcal{L}_\tau$, în care pot fi exprimate proprietățile algebrice ale structurilor de ordinul I de signatura τ.
- Să considerăm o signatură $\tau := ((n_i)_{i \in I}; (m_j)_{j \in J}; (0)_{k \in K})$, pe care o fixăm.
- **Alfabetul** limbajului de ordinul I $\mathcal{L}_\tau$ este format din următoarele **simboluri primitive**:
  1. o mulțime infinită de **variabile**: $Var = \{x, y, z, u, v, \ldots\}$;
  2. **simboluri de operații**: $(f_i)_{i \in I}$; pentru fiecare $i \in I$, numărul natural nenul n_i se numește *ordinul* sau *aritatea* lui f_i;
  3. **simboluri de relații**: $(R_j)_{j \in J}$; pentru fiecare $j \in J$, numărul natural nenul m_j se numește *ordinul* sau *aritatea* lui R_j;
  4. **simboluri de constante**: $(c_k)_{k \in K}$;
  5. **simbolul de egalitate**: $\boldsymbol{=}$ (un semn egal îngroșat) (*a nu se confunda cu egalul simplu!*);
  6. **conectorii logici primitivi**: $\neg$ (*negația*), $\rightarrow$ (*implicația*);
  7. **cuantificatorul universal**: $\forall$ (*oricare ar fi*)
  8. paranteze: $(,), [,]$, precum și virgula.
- Pentru comoditate, vom spune uneori: "operații", "relații" și "constante" în loc de "simboluri de operații/relații/constante", respectiv.

Mulțimile de simboluri de la punctele ①–⑧ se consideră două câte două disjuncte.

Elementele lui *Var* nu sunt variabile propoziționale, ci sunt variabilele care apar în predicate, i.e. în propozițiile cu variabile.

Termenii

$\circledast$ Mulți autori consideră virgula ca având semnificație implicită, subînțeleasă, în scrierea termenilor și a formulelor atomice, și nu includ virgula în limbajul $\mathcal{L}_\tau$.
$\circledast$ La fel putem considera parantezele care încadrează argumentele unui termen sau ale unei formule atomice – a se vedea mai jos.
Dar nu sunt obligatorii distincțiile $\circledast$ și $\circledast$ între simbolurile din acest alfabet.
Iar parantezele din cadrul formulelor (putem include aici toate parantezele, precum și virgula) care stabilesc ordinea aplicării conectorilor logici și a cuantificatorilor pot fi tratate ca în logica propozițională clasică: formulele cu același arbore asociat se consideră a fi egale, indiferent de parantezările acestora – de data aceasta, arborii vor fi oarecare, nu neapărat binari; a se vedea mai jos.

Pentru cele ce urmează **fixăm** o **signatură** arbitrară τ.

Definiție (recursiv: operații imbricate aplicate la variabile și constante)

Termenii limbajului $\mathcal{L}_\tau$ se definesc, recursiv, astfel:

1. variabilele și simbolurile de constante sunt termeni;
2. dacă $n \in \mathbb{N}^*$, f este un simbol de operație n–ară și $t_1, \ldots, t_n$ sunt termeni, atunci $f(t_1, \ldots, t_n)$ este un termen;
3. orice termen se obține prin aplicarea regulilor (1) și (2) de un număr finit de ori.

Formulele (aici, în locul propozițiilor, enunțurile elementare sunt formulele atomice)

Definiție (nerecursiv: **formule atomice**: relații aplicate unor termeni)

Formulele atomice ale limbajului $\mathcal{L}_\tau$ se definesc astfel:

1. dacă t_1 și t_2 sunt termeni, atunci $t_1 = t_2$ este o formulă atomică;
2. dacă $m \in \mathbb{N}^*$, R este un simbol de relație m–ară și $t_1, \ldots, t_m$ sunt termeni, atunci $R(t_1, \ldots, t_m)$ este o formulă atomică.

Definiție (obț. din formule atomice aplicând conectori și cuantificatori)

Formulele limbajului $\mathcal{L}_\tau$ se definesc, recursiv, astfel:

1. formulele atomice sunt formule;
2. dacă φ este o formulă, atunci $\neg\varphi$ este o formulă;
3. dacă φ și ψ sunt formule, atunci $\varphi \to \psi$ este o formulă;
4. dacă φ este o formulă și x este o variabilă, atunci $\forall x\varphi$ este o formulă;
5. orice formulă se obține prin aplicarea regulilor (1), (2), (3) și (4) de un număr finit de ori.

Notație (observați că $Term(\mathcal{L}_\tau) \cap Form(\mathcal{L}_\tau) = \emptyset$)

- $Term(\mathcal{L}_\tau) :=$ mulțimea termenilor limbajului $\mathcal{L}_\tau$ (notație ad–hoc);
- $Form(\mathcal{L}_\tau) :=$ mulțimea formulelor limbajului $\mathcal{L}_\tau$.

Remarcă

Conform definițiilor anterioare:

- $Term(\mathcal{L}_\tau)$ este cea mai mică mulțime de cuvinte finite și nevide peste alfabetul de mai sus care include mulțimea *Var* a variabilelor și mulțimea simbolurilor de constante și este închisă la simbolurile de operații (cu argumente, i.e. de aritate nenulă);
- $Form(\mathcal{L}_\tau)$ este cea mai mică mulțime de cuvinte finite și nevide peste alfabetul de mai sus care include mulțimea formulelor atomice și este închisă la aplicările conectorilor logici $\neg$ și $\rightarrow$ și a cuantificatorului $\forall$.

Finitudinea acestor cuvinte este asigurată de regula (3) din definiția termenilor, definiția formulelor atomice și regula (5) din definiția formulelor. Termenii și formulele pot fi definite și ca fiind cuvinte (apriori presupuse) finite care se obțin prin aplicări succesive ale regulilor din definițiile lor, ceea ce va implica faptul că aplicările acelor reguli se vor face de un număr finit de ori, ca în cazul enunțurilor din logica propozițională clasică.

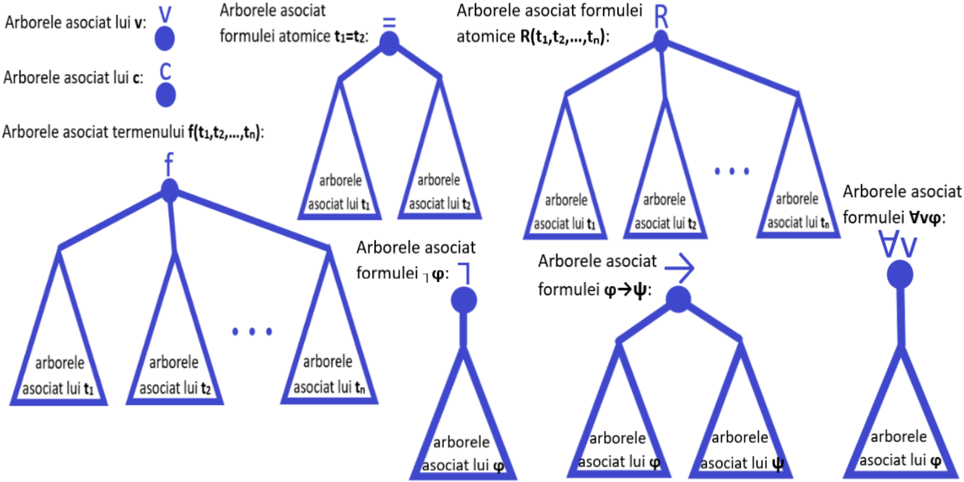
Așadar:

Observație

Și aici ne vom întâlni cu **inducția după un concept**, care va putea fi privită, ca și până acum, atât ca inducție structurală, cât și ca inducție obișnuită după un număr natural (dat de numărul de pași necesari pentru a obține acel concept printr–o recursie).

Arborii oarecare asociați termenilor și formulelor

Fie $v \in Var$, c un simbol de constantă, $n \in \mathbb{N}^*$, $t_1, \ldots, t_n$ termeni, f un simbol de operație n–ară, R un simbol de operație n–ară, $\varphi, \psi \in Form(\mathcal{L}_\tau)$, arbitrare. Arborii asociați termenilor, formulelor atomice, respectiv formulelor se definesc, recursiv, astfel:



Notație

Introducem abrevierile: pentru orice formule φ, ψ și orice variabilă x:

- **conectorii logici derivați** $\vee$ (*disjuncția*), $\wedge$ (*conjuncția*) și $\leftrightarrow$ (*echivalența*) se definesc astfel:
  $\varphi \vee \psi := \neg \varphi \rightarrow \psi$
  $\varphi \wedge \psi := \neg (\varphi \rightarrow \neg \psi)$
  $\varphi \leftrightarrow \psi := (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$
- **cuantificatorul existențial** $\exists$ (*există*) se definește astfel:
  $\exists x \varphi := \neg \forall x \neg \varphi$

(Convenție privind scrierea conectorilor logici, a cuantificatorilor și a simbolului de egalitate, pentru evitarea excesului de paranteze)

- $\neg, \forall, \exists$ vor avea prioritate mai mare;
- $\rightarrow, \vee, \wedge, \leftrightarrow, =$ vor avea prioritate mai mică.

Exemplu

Considerăm signatura $(2,1;2,1;0)$ și limbajul de ordinul I de această signatură conținând simbolul de operație binară f, simbolul de operație unară g, simbolul de relație binară R, simbolul de relație unară S și simbolul de constantă c (astfel că o algebră de această signatură este de forma $\mathcal{A} = (A, f^{\mathcal{A}}, g^{\mathcal{A}}, R^{\mathcal{A}}, S^{\mathcal{A}}, c^{\mathcal{A}})$, cu: $f^{\mathcal{A}} : A^2 \rightarrow A$, $g^{\mathcal{A}} : A \rightarrow A$, $R^{\mathcal{A}} \subseteq A^2$, $S^{\mathcal{A}} \subseteq A$ și $c^{\mathcal{A}} \in A$). Fie $x, y \in Var$ și:

$$\varphi = \exists x \forall y [f(x, g(y)) = f(f(x, y), c) \wedge (R(g(x), f(g(x), c)) \vee S(g(g(c))))].$$

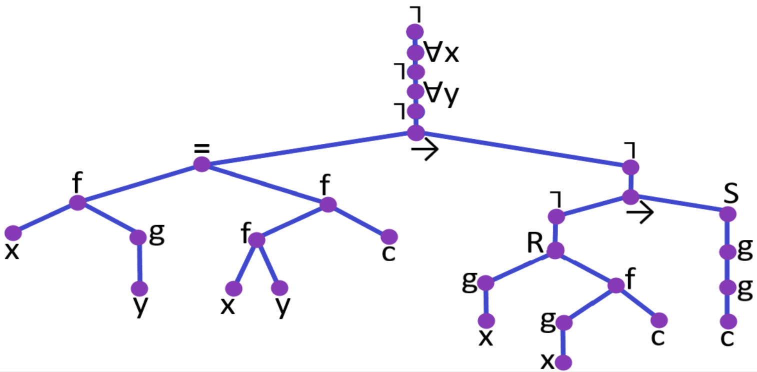
Vom vedea că φ este un tip de formulă numit **enunț**. φ este o abreviere pentru:

$$\varphi = \exists x \forall y [f(x, g(y)) = f(f(x, y), c) \wedge (\neg R(g(x), f(g(x), c)) \rightarrow S(g(g(c))))] =$$

$$\exists x \forall y \neg [f(x, g(y)) = f(f(x, y), c) \rightarrow \neg(\neg R(g(x), f(g(x), c)) \rightarrow S(g(g(c))))] =$$

$$\neg \forall x \neg \forall y \neg [f(x, g(y)) = f(f(x, y), c) \rightarrow \neg(\neg R(g(x), f(g(x), c)) \rightarrow S(g(g(c))))],$$

așadar φ are următorul arbore binar asociat:



Notație (mulțimile din această notație vor fi definite recursiv mai jos)

Pentru orice termen t și orice formulă φ, notăm:

- $V(t) :=$ mulțimea variabilelor termenului t
- $V(\varphi) :=$ mulțimea variabilelor formulei φ
- $FV(\varphi) :=$ mulțimea variabilelor *libere* ale formulei φ

Definiție (variabilele dintr–un termen)

Pentru orice termen t:

- dacă $t = x$, unde x este o variabilă, atunci $V(t) = V(x) := \{x\}$
- dacă $t = c$, unde c este o constantă, atunci $V(t) = V(c) := \emptyset$
- dacă $t = f(t_1, \ldots, t_n)$, unde $n \in \mathbb{N}^*$, f este un simbol de operație n–ară și $t_1, \ldots, t_n$ sunt termeni, atunci $V(t) := \bigcup\limits_{i=1}^{n} V(t_i)$

Definiție (variabilele și variabilele libere dintr–o formulă)

Pentru orice formulă φ:

- dacă $\varphi = (t_1 {=} t_2)$, unde t_1 și t_2 sunt termeni, atunci $V(\varphi) = FV(\varphi) := V(t_1) \cup V(t_2)$

Definiție (variabilele și variabilele libere dintr–o formulă – continuare)

- dacă $\varphi = R(t_1, \ldots, t_m)$, unde $m \in \mathbb{N}^*$, R este un simbol de relație m–ară și $t_1, \ldots, t_m$ sunt termeni, atunci $V(\varphi) = FV(\varphi) := \bigcup_{i=1}^{m} V(t_i)$
- dacă $\varphi = \neg\psi$, pentru o formulă ψ, atunci $V(\varphi) := V(\psi)$ și $FV(\varphi) := FV(\psi)$
- dacă $\varphi = \psi \rightarrow \chi$, pentru două formule ψ, χ, atunci $V(\varphi) := V(\psi) \cup V(\chi)$ și $FV(\varphi) := FV(\psi) \cup FV(\chi)$
- dacă $\varphi = \forall x\psi$, pentru o formulă ψ și o variabilă x, atunci $V(\varphi) := V(\psi) \cup \{x\}$ și $FV(\varphi) := FV(\psi) \setminus \{x\}$

Remarcă

Este imediat, din definiția anterioară și definiția conectorilor logici derivați și a cuantificatorului existențial, că, pentru orice formule ψ, χ și orice variabilă x:

- $V(\psi \vee \chi) = V(\psi \wedge \chi) = V(\psi \leftrightarrow \chi) = V(\psi) \cup V(\chi)$ și $FV(\psi \vee \chi) = FV(\psi \wedge \chi) = FV(\psi \leftrightarrow \chi) = FV(\psi) \cup FV(\chi)$
- $V(\exists x\psi) = V(\psi) \cup \{x\}$ și $FV(\exists x\psi) = FV(\psi) \setminus \{x\}$

Enunțurile sunt formulele fără variabile libere, deci cu toate variabilele **legate**:

Definiție (variabile libere și variabile legate)

Pentru orice variabilă x și orice formulă φ:

- dacă $x \in FV(\varphi)$, atunci x se numește *variabilă liberă a lui* φ;
- dacă $x \in V(\varphi) \setminus FV(\varphi)$ (prin extensie, chiar dacă $x \in Var \setminus FV(\varphi)$), atunci x se numește *variabilă legată a lui* φ;
- dacă $FV(\varphi) = \emptyset$ (i. e. φ nu are variabile libere), atunci φ se numește *enunț*.

Exemplu

- În formula $\exists x(x^2 \mathbf{=} x)$, x este variabilă legată. Această formulă este un enunț.
- În formula $\forall y \forall z(z \cdot x \leq y \cdot z)$, x este variabilă liberă.

Notație (cum specificăm că nu avem alte variabile (libere))

Fie $n \in \mathbb{N}^*$ și $x_1, \ldots, x_n$ variabile.
Dacă t este un termen cu $V(t) \subseteq \{x_1, \ldots, x_n\}$, atunci vom nota $t(x_1, \ldots, x_n)$.
Dacă φ este o formulă cu $FV(\varphi) \subseteq \{x_1, \ldots, x_n\}$, atunci vom nota $\varphi(x_1, \ldots, x_n)$.

Observație (diferența dintre tipurile de variabile, intuitiv)

- **Variabilele libere** sunt variabilele care nu intră sub incidența unui cuantificator, variabilele cărora "avem libertatea de a le da valori".
- **Variabilele legate** sunt variabilele care intră sub incidența unui cuantificator, deci sunt destinate parcurgerii unei întregi mulțimi de valori.

Definiție (în secțiunea privind rezoluția, vom vedea că $\varphi(t, x_1, \ldots, x_n) = \sigma(\varphi)$, unde σ este substituția: $\sigma : Var \to Term(\mathcal{L}_\tau)$, $\sigma(v) = \begin{cases} t, & v = x, \\ v, & v \in Var \setminus \{x\} \end{cases}$)

Fie $n \in \mathbb{N}$, $\varphi(x, x_1, \ldots, x_n) \in Form(\mathcal{L}_\tau)$ cu $FV(\varphi) = \{x, x_1, \ldots, x_n\} \subset Var$ și $t \in Term(\mathcal{L}_\tau)$. Formula obținută din φ prin *substituția lui x cu t* se notează cu $\varphi(t, x_1, \ldots, x_n)$ și se definește astfel:

- fiecare $y \in (V(\varphi) \setminus FV(\varphi)) \cap V(t) = (V(\varphi) \setminus \{x, x_1, \ldots, x_n\}) \cap V(t)$ (variabilă legată a lui φ care apare și în t) se înlocuiește cu o variabilă $v \notin V(t) \cup V(\varphi)$;
- apoi se înlocuiește x cu t.

Exemplu

Fie variabilele x, y, z, formula $\varphi(x) := \exists\, y(x < y)$ și termenul $t := y + z$. Atunci $\varphi(t)$ se obține astfel:

- $\varphi(x) = \exists\, y(x < y)$ se înlocuiește cu $\exists\, v(x < v)$;
- prin urmare, $\varphi(t) = \exists\, v(y + z < v)$.

Axiomele logicii clasice a predicatelor (pentru limbajul $\mathcal{L}_\tau$)

Pentru orice $\varphi, \psi, \chi \in Form(\mathcal{L}_\tau)$, $t \in Term(\mathcal{L}_\tau)$, $n, i \in \mathbb{N}^*$ și $x, y, v_1, \ldots, v_n \in Var$:

- axiomele logicii propoziționale clasice:
  (G_1) $\varphi \to (\psi \to \varphi)$
  (G_2) $(\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$
  (G_3) $(\neg\varphi \to \neg\psi) \to (\psi \to \varphi)$
- *regula* $(\to\forall)$:
  (G_4) dacă $x \notin FV(\varphi)$, atunci: $\forall x(\varphi \to \psi) \to (\varphi \to \forall x\psi)$
- o regulă privind substituțiile (o numim ad–hoc *particularizarea*):
  (G_5) $\forall x\varphi(x, v_1, \ldots, v_n) \to \varphi(t, v_1, \ldots, v_n)$
- *axiomele egalității*, pe care le denumim ad–hoc:
  reflexivitatea: (G_6) $x{=}x$
  substitutivitatea în termeni: (G_7)
  $(x{=}y) \to (t(v_1, \ldots, v_{i-1}, x, v_{i+1}, \ldots, v_n){=}t(v_1, \ldots, v_{i-1}, y, v_{i+1}, \ldots, v_n))$
  substitutivitatea în formule: (G_8)
  $(x{=}y) \to (\varphi(v_1, \ldots, v_{i-1}, x, v_{i+1}, \ldots, v_n){=}\varphi(v_1, \ldots, v_{i-1}, y, v_{i+1}, \ldots, v_n))$

Notație (ad–hoc pentru mulțimea acestor axiome)

$Ax(\mathcal{L}_\tau) := \{\varphi \to (\psi \to \varphi),\ (\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi)),\ (\neg\varphi \to \neg\psi) \to (\psi \to \varphi),\ \forall v(\alpha \to \psi) \to (\alpha \to \forall v\psi),\ \forall x\varphi(x, v_1, \ldots, v_n) \to \varphi(t, v_1, \ldots, v_n),\ x{=}x,\ (x{=}y) \to (t(v_1, \ldots, v_{i-1}, x, v_{i+1}, \ldots, v_n){=}$

$t(v_1, \ldots, v_{i-1}, y, v_{i+1}, \ldots, v_n))$, $(x{=}y) \to (\varphi(v_1, \ldots, v_{i-1}, x, v_{i+1}, \ldots, v_n){=}$ $\varphi(v_1, \ldots, v_{i-1}, y, v_{i+1}, \ldots, v_n)) \mid \alpha, \varphi, \psi, \chi \in Form(\mathcal{L}_\tau), t \in Term(\mathcal{L}_\tau), n, i \in \mathbb{N}^*, x, y, v_1, \ldots, v_n \in Var, v \in Var \setminus FV(\alpha)\}$.

Notație

Faptul că o formulă φ este *teoremă (formală) (adevăr sintactic) a(l) lui* $\mathcal{L}_\tau$ se notează cu $\vdash \varphi$ și se definește, recursiv, ca mai jos.
Notăm (ad–hoc) cu $Teor(\mathcal{L}_\tau)$ mulțimea teoremelor formale ale lui $\mathcal{L}_\tau$.

Ca în logica propozițională, folosim notația $\dfrac{\text{Premise}}{\text{Consecință}}$ pentru regulile de deducție.

Definiție (teoremele logicii clasice a predicatelor (pentru limbajul $\mathcal{L}_\tau$))

1. Orice axiomă e teoremă formală a lui $\mathcal{L}_\tau$.
2. Pentru orice formule φ, ψ, $\dfrac{\vdash \psi, \vdash \psi \to \varphi}{\vdash \varphi}$ (regula de deducție *modus ponens* (**MP**)).
3. Pentru orice formulă φ și orice variabilă x, $\dfrac{\vdash \varphi}{\vdash \forall x \varphi}$ (regula de deducție numită *principiul generalizării* (**PG**)).
4. Orice teoremă formală se obține prin aplicarea regulilor ①, ② și ③ de un număr finit de ori.

Notație

Fie Σ o mulțime de formule ale lui $\mathcal{L}_\tau$. Faptul că o formulă φ *se deduce (formal) din ipotezele* Σ (φ *este consecință sintactică a mulțimii de ipoteze* Σ) se notează cu $\Sigma \vdash \varphi$ și se definește, recursiv, ca mai jos.

Definiție (deducția sintactică din ipoteze)

Fie Σ o mulțime de formule ale lui $\mathcal{L}_\tau$.

1. Orice axiomă a lui $\mathcal{L}_\tau$ se deduce formal din Σ.
2. Pentru orice $\sigma \in \Sigma$, $\Sigma \vdash \sigma$.
3. Pentru orice formule φ, ψ, $\dfrac{\Sigma \vdash \psi, \Sigma \vdash \psi \to \varphi}{\Sigma \vdash \varphi}$ (regula de deducție *modus ponens* (MP)).
4. Pentru orice formulă φ și orice variabilă x, $\dfrac{\Sigma \vdash \varphi}{\Sigma \vdash \forall x \varphi}$ (regula de deducție numită *principiul generalizării* (PG)).
5. Orice consecință sintactică a lui Σ se obține prin aplicarea regulilor ①, ②, ③ și ④ de un număr finit de ori.

Avem și principiul inducției după teoreme formale și după consecințele sintactice ale unei mulțimi de ipoteze, întrucât, conform definițiilor anterioare:

Remarcă

- $Teor(\mathcal{L}_\tau)$ este cea mai mică submulțime a lui $Form(\mathcal{L}_\tau)$ care include pe $Ax(\mathcal{L}_\tau)$ și este închisă la **MP** și **PG**.
- Pentru orice $\Sigma \subseteq Form(\mathcal{L}_\tau)$, mulțimea $\{\varphi \in Form(\mathcal{L}_\tau) \mid \Sigma \vdash \varphi\}$ este cea mai mică submulțime a lui $Form(\mathcal{L}_\tau)$ care include pe $Ax(\mathcal{L}_\tau) \cup \Sigma$ și este închisă la **MP** și **PG**.

Definiție

Fie $\Sigma \subseteq Form(\mathcal{L}_\tau)$, iar $\varphi \in Form(\mathcal{L}_\tau)$. Se numește *demonstrație formală*, respectiv *demonstrație formală din mulțimea de ipoteze* Σ pentru φ un șir $\varphi_1, \ldots, \varphi_n$, unde $n \in \mathbb{N}^*$, numit *lungimea demonstrației formale*, $\varphi_1, \ldots, \varphi_n \in Form(\mathcal{L}_\tau)$ cu $\varphi_n = \varphi$ și, pentru fiecare $i \in \overline{1, n}$, are loc una dintre proprietățile:

1. $\varphi_i \in Ax(\mathcal{L}_\tau)$, respectiv $\varphi_i \in Ax(\mathcal{L}_\tau) \cup \Sigma$;
2. φ_i se obține prin **MP** din enunțuri dintre $\varphi_1, \ldots, \varphi_{i-1}$, i.e. există $j, k \in \overline{1, i-1}$ astfel încât $\varphi_k = \varphi_j \to \varphi_i$;
3. φ_i se obține prin **PG** dintr–un enunț dintre $\varphi_1, \ldots, \varphi_{i-1}$, i.e. există $j \in \overline{1, i-1}$ și $x \in Var$ astfel încât $\varphi_i = \forall x \, \varphi_j$.

Din definițiile anterioare rezultă:

Remarcă

Pentru orice $\varphi \in Form(\mathcal{L}_\tau)$ și orice $\Sigma \subseteq Form(\mathcal{L}_\tau)$:

- $\vdash \varphi$ ddacă $\emptyset \vdash \varphi$ ddacă φ admite o demonstrație formală;
- $\Sigma \vdash \varphi$ ddacă φ admite o demonstrație formală din mulțimea de ipoteze Σ.

Întrucât axiomele logicii propoziționale clasice și regula de deducție **MP** au loc și în logica clasică a predicatelor, rezultă că:

Remarcă

Toate teoremele formale și regulile de deducție derivate din logica propozițională clasică sunt valabile și în logica clasică a predicatelor dacă înlocuim în acestea mulțimea E a enunțurilor logicii propoziționale clasice cu mulțimea $Form(\mathcal{L}_\tau)$ a formulelor limbajului de ordinul I $\mathcal{L}_\tau$.

Remarcă

Pentru orice $n \in \mathbb{N}^*$, orice $x_1, \ldots, x_n \in Var$ și orice $\varphi \in Form(\mathcal{L}_\tau)$ cu $FV(\varphi) = \{x_1, \ldots, x_n\}$, $FV(\forall x_1 \ldots \forall x_n \varphi) = FV(\varphi) \setminus \{x_1, \ldots, x_n\} = \emptyset$, așadar $\forall x_1 \ldots \forall x_n \varphi$ este un enunț.

Definiție

Pentru orice $n \in \mathbb{N}^*$, orice $x_1, \ldots, x_n \in Var$ și orice $\varphi \in Form(\mathcal{L}_\tau)$ cu $FV(\varphi) = \{x_1, \ldots, x_n\}$, enunțul $\forall x_1 \ldots \forall x_n \varphi$ se numește *închiderea universală a formulei* φ.

Propoziție

Pentru orice $n \in \mathbb{N}^*$, orice $x_1, \dots, x_n \in Var$, orice $\varphi \in Form(\mathcal{L}_\tau)$ cu $FV(\varphi) = \{x_1, \dots, x_n\}$ și orice $\Sigma \subseteq Form(\mathcal{L}_\tau)$: $\Sigma \vdash \varphi$ ddacă $\Sigma \vdash \forall x_1 \dots \forall x_n \varphi$.

Demonstrație: "$\Rightarrow$:"

$\Sigma \vdash \varphi$	ipoteza acestei implicații
$\Sigma \vdash \forall x_n \varphi$	**PG**
$\Sigma \vdash \forall x_{n-1} \forall x_n \varphi$	**PG**
$\vdots$	
$\Sigma \vdash \forall x_1 \dots \forall x_n \varphi$	**PG**

"$\Leftarrow$:" Cum $x_1, \dots, x_n \in FV(\varphi)$, rezultă că, pentru fiecare $i \in \overline{1, n}$, $x_i \in FV(\forall x_{i+1} \dots \forall x_n \varphi)$.
Pentru orice $\alpha \in Form(\mathcal{L}_\tau)$ și orice $x \in FV(\alpha)$, putem înlocui termenul t cu x în axioma particularizării (G_5), și obținem: $\vdash \forall x \alpha \to \alpha$, așadar: $\Sigma \vdash \forall x \alpha \to \alpha$.
Prin urmare:

$\Sigma \vdash \forall x_1 \dots \forall x_n \varphi$	ipoteza acestei implicații
$\Sigma \vdash \forall x_1 \dots \forall x_n \varphi \to \forall x_2 \dots \forall x_n \varphi$	observația de mai sus
$\Sigma \vdash \forall x_2 \dots \forall x_n \varphi$	**MP**
$\Sigma \vdash \forall x_2 \dots \forall x_n \varphi \to \forall x_3 \dots \forall x_n \varphi$	observația de mai sus
$\Sigma \vdash \forall x_3 \dots \forall x_n \varphi$	**MP**

$\vdots$

$\Sigma \vdash \forall x_n \varphi$	**MP**
$\Sigma \vdash \forall x_n \varphi \to \varphi$	observația de mai sus
$\Sigma \vdash \varphi$	**MP**

Propoziție ($\star$)

Pentru orice $\varphi \in Form(\mathcal{L}_\tau)$ și orice $\Sigma, \Delta \in \mathcal{P}(Form(\mathcal{L}_\tau))$:

- *dacă $\Sigma \subseteq \Delta$ și $\Sigma \vdash \varphi$, atunci $\Delta \vdash \varphi$;*
- *dacă $\Sigma \vdash \varphi$, atunci există $\Gamma \subseteq \Sigma$ astfel încât Γ este finită și $\Gamma \vdash \varphi$.*

Demonstrație: "$\Rightarrow$:" Prin inducție după formule, ca în logica propozițională clasică.

Teoremă (Teorema deducției sintactice)

Fie $\Sigma \subseteq Form(\mathcal{L}_\tau)$ și $\varphi, \psi \in Form(\mathcal{L}_\tau)$.
Dacă φ este enunț, atunci are loc echivalența:

$$\Sigma \vdash \varphi \to \psi \Leftrightarrow \Sigma \cup \{\varphi\} \vdash \psi.$$

Implicația directă are loc și dacă φ nu este enunț.

Demonstrație: "$\Rightarrow$:" Procedăm ca în logica propozițională clasică.
Conform Propoziției $\star$, dacă $\Sigma \vdash \varphi \rightarrow \psi$, atunci $\Sigma \cup \{\varphi\} \vdash \varphi \rightarrow \psi$.
Dar $\varphi \in \Sigma \cup \{\varphi\}$, așadar $\Sigma \cup \{\varphi\} \vdash \varphi$.
Conform **MP**, rezultă că $\Sigma \cup \{\varphi\} \vdash \psi$.
"$\Leftarrow$:" Prin inducție asupra consecințelor sintactice ale mulțimii de ipoteze $\Sigma \cup \{\varphi\}$, ca în logica propozițională clasică, rămânând de tratat doar cazul în care, pentru $x \in Var$ și $\alpha \in Form(\mathcal{L}_\tau)$, au loc $\Sigma \cup \{\varphi\} \vdash \alpha$ și $\psi = \forall x\, \alpha$. Cum φ este un enunț, avem $x \notin \emptyset = FV(\varphi)$, așadar putem aplica (G_4) (regula $\rightarrow \forall$) în forma de mai jos:

$\Sigma \vdash \varphi \rightarrow \alpha$	ipoteza de inducție
$\Sigma \vdash \forall x\, (\varphi \rightarrow \alpha)$	**PG**
$\Sigma \vdash \forall x\, (\varphi \rightarrow \alpha) \rightarrow (\varphi \rightarrow \forall x\, \alpha)$	(G_4)
$\Sigma \vdash \varphi \rightarrow \forall x\, \alpha$	**MP**
$\Sigma \vdash \varphi \rightarrow \psi$	$\psi = \forall x\, \alpha$

O interpretare/evaluare/semantică dă variabilelor valori într–o structură de ordinul I

- Fie $\mathcal{A}$ o structură de ordinul I de signatură τ.
- Fixăm pe τ și pe $\mathcal{A}$ pentru cele ce urmează.
- A va fi universul structurii $\mathcal{A}$ (mulțimea ei suport).
- Pentru fiecare simbol de operație f, fiecare simbol de relație R și fiecare simbol de constantă c din signatura τ, notăm cu $f^{\mathcal{A}}$, respectiv $R^{\mathcal{A}}$, respectiv $c^{\mathcal{A}}$ operația, respectiv relația, respectiv constanta corespunzătoare din $\mathcal{A}$.

Definiție

O *interpretare* (sau *evaluare*, sau *semantică*) a limbajului $\mathcal{L}_\tau$ în structura $\mathcal{A}$ este o funcție $s : Var \to A$.
Fiecare variabilă $x \in Var$ este "interpretată" prin elementul $s(x) \in A$.

Pentru cele ce urmează, fixăm o interpretare arbitrară $s : Var \to A$.
Vom prelungi pe s la mulțimea tuturor termenilor lui $\mathcal{L}_\tau$, $Term(\mathcal{L}_\tau) \supseteq Var$, apoi o vom defini și pe formulele lui $\mathcal{L}_\tau$, prelungindu–i și codomeniul la $A \cup \mathcal{L}_2$. Vom nota aceste prelungiri tot cu s.

Valorile din A asociate termenilor de o interpretare

Prelungim pe s la mulțimea tuturor termenilor, obținând o funcție $s : Term(\mathcal{L}_\tau) \to A$.

Definiție (valorile asociate termenilor de o interpretare)

Pentru orice termen $t \in Term(\mathcal{L}_\tau)$, definim recursiv elementul $s(t) \in A$:

- dacă $t = x \in Var$, atunci $s(t) = s(x)$;
- dacă $t = c$, unde c este un simbol de constantă, atunci $s(t) = c^{\mathcal{A}}$;
- dacă $t = f(t_1, \dots, t_n)$, unde $n \in \mathbb{N}^*$, f este un simbol de operație n–ară, iar $t_1, \dots, t_n \in Term(\mathcal{L}_\tau)$, atunci $s(t) = f^{\mathcal{A}}(s(t_1), \dots, s(t_n))$.

Prin inducție după termeni se arată că, prin recursia de mai sus, se definește s pe toți termenii din $Term(\mathcal{L}_\tau)$ și că valoarea lui s în fiecare termen $t \in Term(\mathcal{L}_\tau)$ este unic determinată de valorile lui $s : Var \to A$ în variabilele $x \in V(t)$, astfel că funcția $s : Term(\mathcal{L}_\tau) \to A$ este complet și corect definită ca mai sus și este unic determinată de restricția sa $s\,|_{Var}: Var \to A$.

Valorile din $\mathcal{L}_2$ asociate formulelor de o interpretare

Notație

Pentru orice variabilă $x \in Var$ și orice element $a \in A$, notăm cu $s[{}^{x}_{a}] : Var \to A$ interpretarea definită prin: oricare ar fi $v \in Var$,

$$s[{}^{x}_{a}](v) := \begin{cases} a, & \text{dacă } v = x, \\ s(v), & \text{dacă } v \neq x. \end{cases}$$

Acum definim valorile unei interpretări s în formule; acestea vor fi valori din **algebra Boole standard** (**algebra Boole a valorilor de adevăr**) $\mathcal{L}_2 = (\{0, 1\}, \vee, \wedge, \bar{\cdot}, 0, 1)$. Ca în cazul valorilor enunțurilor din logica propozițională clasică într–o interpretare, folosim notațiile $\rightarrow$ pentru implicația booleană și $\leftrightarrow$ pentru echivalența booleană în $\mathcal{L}_2$.

Astfel, vom obține o funcție $s : Term(\mathcal{L}_\tau) \cup Form(\mathcal{L}_\tau) \to A \cup \mathcal{L}_2$.

Prin inducție după formule, se arată că această funcție este corect și complet definită ca mai jos, iar valoarea ei într–o formulă $\varphi \in Form(\mathcal{L}_\tau)$ este unic determinată de valorile lui $s \mid_{Var}: Var \to A$ în fiecare $x \in V(\varphi)$ (de fapt, după cum vom vedea, chiar de cele în fiecare $x \in FV(\varphi)$).

Definiție (valorile booleene din $\mathcal{L}_2$ asociate formulelor de o interpretare)

Pentru orice formulă $\varphi \in Form(\mathcal{L}_\tau)$, *valoarea de adevăr a lui* φ *în interpretarea* s este un element din algebra Boole standard $\mathcal{L}_2 = \{0, 1\}$, notat cu $s(\varphi)$, definit, recursiv, astfel:

- dacă $\varphi = (t_1 = t_2)$, pentru doi termeni $t_1, t_2 \in Term(\mathcal{L}_\tau)$, atunci
  $$s(\varphi) = \begin{cases} 1, & \text{dacă } s(t_1) = s(t_2), \\ 0, & \text{dacă } s(t_1) \neq s(t_2); \end{cases}$$
- dacă $\varphi = R(t_1, \ldots, t_m)$, unde $m \in \mathbb{N}^*$, R este un simbol de relație m–ară, iar $t_1, \ldots, t_m \in Term(\mathcal{L}_\tau)$ sunt termeni, atunci
  $$s(\varphi) = \begin{cases} 1, & \text{dacă } (s(t_1), \ldots, s(t_m)) \in R^{\mathcal{A}}, \\ 0, & \text{dacă } (s(t_1), \ldots, s(t_m)) \notin R^{\mathcal{A}}; \end{cases}$$
- dacă $\varphi = \neg\, \psi$, pentru o formulă $\psi \in Form(\mathcal{L}_\tau)$, atunci $s(\varphi) = \overline{s(\psi)}$;
- dacă $\varphi = \psi \to \chi$, pentru două formule $\psi, \chi \in Form(\mathcal{L}_\tau)$, atunci $s(\varphi) = s(\psi) \to s(\chi)$;
- dacă $\varphi = \forall\, x\psi$, pentru o formulă $\psi \in Form(\mathcal{L}_\tau)$ și o variabilă $x \in Var$, atunci $s(\varphi) = \bigwedge\limits_{a \in A} s[{}^x_a](\psi)$.

Remarcă

Este imediat că, pentru orice formule $\psi, \chi \in Form(\mathcal{L}_\tau)$ și orice variabilă $x \in Var$, au loc egalitățile:

- $s(\psi \vee \chi) = s(\psi) \vee s(\chi)$;
- $s(\psi \wedge \chi) = s(\psi) \wedge s(\chi)$;
- $s(\psi \leftrightarrow \chi) = s(\psi) \leftrightarrow s(\chi)$;
- $s(\exists\, x\psi) = \bigvee\limits_{a \in A} s[{}^{x}_{a}](\psi)$.

Ca în logica propozițională clasică, interpretările transformă conectorii logici în operații booleene.

Remarcă (semnificația valorilor unei interpretări în formule cuantificate)

Fie $x \in Var$ și $\psi \in Form(\mathcal{L}_\tau)$. Cum $s(\alpha) \in \mathcal{L}_2 = \{0, 1\}$ pentru orice $\alpha \in Form(\mathcal{L}_\tau)$, din cele de mai sus rezultă că:

- $s(\forall\, x\psi) = 1$ ddacă $\bigwedge\limits_{a \in A} s[{}^{x}_{a}](\psi) = 1$ ddacă $s[{}^{x}_{a}](\psi) = 1$ pentru toți $a \in A$;
- $s(\exists\, x\psi) = 1$ ddacă $\bigvee\limits_{a \in A} s[{}^{x}_{a}](\psi) = 1$ ddacă există un $a \in A$ a.î. $s[{}^{x}_{a}](\psi) = 1$.

Notație (adăugarea de constante într–o signatură)

Fie $a \in A$. Adăugăm un simbol de constantă a^{ct} la signatura τ, obținând o signatură $\tau[a]$. Organizăm pe $\mathcal{A}$ ca structură de ordinul I de signatură $\tau[a]$ adăugându–i constanta: $(a^{ct})^{\mathcal{A}} = a$. Putem nota $a := a^{ct}$.
Fie $x \in Var$.
Pentru orice termen $t \in Term(\mathcal{L}_\tau)$, definim recursiv termenul $t[\frac{x}{a}] \in Term(\mathcal{L}_{\tau[a]})$, astfel:

- dacă $t = x$, atunci $t[\frac{x}{a}] = a^{ct}$;
- dacă $t \in Var \setminus \{x\}$ sau t este un simbol de constantă din $\mathcal{L}_\tau$, atunci $t[\frac{x}{a}] = t$;
- dacă $n \in \mathbb{N}^*$, $t_1, \ldots, t_n \in Term(\mathcal{L}_\tau)$, f este un simbol de operație n–ară, iar $t = f(t_1, \ldots, t_n)$, atunci $t[\frac{x}{a}] = f(t_1[\frac{x}{a}], \ldots, t_n[\frac{x}{a}])$.

Pentru orice formulă $\varphi \in Form(\mathcal{L}_\tau)$, definim recursiv formula $\varphi[\frac{x}{a}] \in Form(\mathcal{L}_{\tau[a]})$, astfel:

- dacă $\varphi = t_1 \mathbf{=} t_2$ pentru doi termeni $t_1, t_2 \in Term(\mathcal{L}_\tau)$, atunci $\varphi[\frac{x}{a}] = t_1[\frac{x}{a}] \mathbf{=} t_2[\frac{x}{a}]$;
- dacă $\varphi = R(t_1, \ldots, t_m)$ pentru un $m \in \mathbb{N}*$, un simbol de relație m–ară R și m termeni $t_1, \ldots, t_m \in Term(\mathcal{L}_\tau)$, atunci $\varphi[\frac{x}{a}] = R(t_1[\frac{x}{a}], \ldots, t_m[\frac{x}{a}])$;
- dacă $\varphi = \neg\, \psi$ pentru o formulă $\psi \in Form(\mathcal{L}_\tau)$, atunci $\varphi[\frac{x}{a}] = \neg\, \psi[\frac{x}{a}]$;
- dacă $\varphi = \psi \rightarrow \chi$ pentru două formule $\psi, \chi \in Form(\mathcal{L}_\tau)$, atunci $\varphi[\frac{x}{a}] = \psi[\frac{x}{a}] \rightarrow \chi[\frac{x}{a}]$;

- dacă $\varphi = \forall\, v\, \psi$ pentru o formulă $\psi \in Form(\mathcal{L}_\tau)$ și o variabilă $v \in Var$, atunci $\varphi[\substack{x\\a}] = \forall\, w\, (\psi[\substack{x\\a}])$, unde $w = v$, dacă $v \neq x$, și $w \in Var \setminus (\{x\} \cup FV(\psi))$, altfel.

La fel pentru un număr $k \in \mathbb{N}^*$ de simboluri de constante corespunzătoare unor elemente $a_1, \ldots a_k \in A$: $\tau[a_1, \ldots a_k] := \tau[a_1] \ldots [a_k]$, $(a_i^{ct})^{\mathcal{A}} = a_i$ pentru fiecare $i \in \overline{1, k}$, și, pentru orice variabile $x_1, \ldots, x_k \in Var$, orice termen $t \in Term(\mathcal{L}_\tau)$ și orice formulă $\varphi \in Form(\mathcal{L}_\tau)$, avem termenul $t[\substack{x_1\\a_1}] \ldots [\substack{x_k\\a_k}] := (\ldots(t[\substack{x_1\\a_1}])\ldots)[\substack{x_k\\a_k}] \in Term(\mathcal{L}_{\tau[a_1,\ldots a_k]})$, formula $\varphi[\substack{x_1\\a_1}] \ldots [\substack{x_k\\a_k}] := (\ldots(\varphi[\substack{x_1\\a_1}] \ldots [\substack{x_k\\a_k}])\ldots) \in Form(\mathcal{L}_{\tau[a_1,\ldots a_k]})$ și substituția $s[\substack{x_1\\a_1}] \ldots [\substack{x_k\\a_k}] := (\ldots(s[\substack{x_1\\a_1}] \ldots [\substack{x_k\\a_k}])\ldots) : Var \to Term(\mathcal{L}_{\tau[a_1,\ldots a_k]})$.

Propoziție

Pentru orice $k \in \mathbb{N}^*$, $x, x_1, \ldots, x_k \in Var$, $a, a_1, \ldots a_k \in A$, orice termen $t \in Term(\mathcal{L}_\tau)$ și orice formulă $\varphi \in Form(\mathcal{L}_\tau)$:

- $s[\substack{x\\a}](t) = s(t[\substack{x\\a}])$; *mai general:* $s[\substack{x_1\\a_1}] \ldots [\substack{x_k\\a_k}](t) = s(t[\substack{x_1\\a_1}] \ldots [\substack{x_k\\a_k}])$;
- $s[\substack{x\\a}](\varphi) = s(\varphi[\substack{x\\a}])$; *mai general:* $s[\substack{x_1\\a_1}] \ldots [\substack{x_k\\a_k}](\varphi) = s(\varphi[\substack{x_1\\a_1}] \ldots [\substack{x_k\\a_k}])$.

Demonstrație: Proprietatea mai generală rezultă prin inducție după $k \in \mathbb{N}*$ din cea pentru x și a, așa că o vom demonstra pe aceasta, prin inducție după termenii din $Term(\mathcal{L}_\tau)$, apoi după formulele din $Form(\mathcal{L}_\tau)$, considerând proprietățile asupra unui termen $t \in Term(\mathcal{L}_\tau)$, respectiv unei formule $\varphi \in Form(\mathcal{L}_\tau)$:

$P(t): \quad s[{}^x_a](t) = s(t[{}^x_a]); \qquad Q(\varphi): \quad (\forall\, r : Var \to A)\,(r[{}^x_a](\varphi) = r(\varphi[{}^x_a])).$

Pentru orice variabilă $v \in Var$, avem: $v[{}^x_a] = \begin{cases} v, & \text{dacă } v \neq x, \\ a^{ct}, & \text{dacă } v = x, \end{cases}$ prin urmare:

$s[{}^x_a](v) = \begin{cases} s(v), & \text{dacă } v \neq x, \\ a = s(a^{ct}), & \text{dacă } v = x, \end{cases} = s(v[{}^x_a])$, așadar v satisface proprietatea P.

Pentru orice simbol de constantă c din $\mathcal{L}_\tau$, $c[{}^x_a] = c$, așadar $s[{}^x_a](c) = c^{\mathcal{A}} = s(c) = s(c[{}^x_a])$, așadar c satisface proprietatea P.

Acum fie $n \in \mathbb{N}^*$, f un simbol de operație n–ară din $\mathcal{L}_\tau$ și $t_1, \dots, t_n \in Term(\mathcal{L}_\tau)$ astfel încât termenii $t_1, \dots, t_n$ satisfac proprietatea P. Atunci $f(t_1, \dots, t_n)[{}^x_a] = f(t_1[{}^x_a], \dots, t_n[{}^x_a])$, așadar $s(f(t_1, \dots, t_n)[{}^x_a]) = s(f(t_1[{}^x_a], \dots, t_n[{}^x_a])) = f^{\mathcal{A}}(s(t_1[{}^x_a]), \dots, s(t_n[{}^x_a])) = f^{\mathcal{A}}(s[{}^x_a](t_1), \dots, s[{}^x_a](t_n)) = s[{}^x_a](f(t_1, \dots, t_n))$, așadar mulțimea termenilor lui $\mathcal{L}_\tau$ care satisfac proprietatea P este închisă la aplicarea simbolurilor de operații (de arități nenule).

Așadar mulțimea $\{t \in Term(\mathcal{L}_\tau) \mid P(t)\}$ a termenilor lui $\mathcal{L}_\tau$ care satisfac proprietatea P include mulțimea Var a variabilelor și mulțimea simbolurilor de constantă din $\mathcal{L}_\tau$ și este închisă la aplicarea simbolurilor de operații, prin urmare $\{t \in Term(\mathcal{L}_\tau) \mid P(t)\}$ include cea mai mică mulțime care include mulțimea Var a variabilelor și mulțimea simbolurilor de constantă din $\mathcal{L}_\tau$ și este închisă la aplicarea simbolurilor de operații, anume mulțimea $Term(\mathcal{L}_\tau)$ a tuturor termenilor lui $\mathcal{L}_\tau$, așadar $\{t \in Term(\mathcal{L}_\tau) \mid P(t)\} = Term(\mathcal{L}_\tau)$, adică toți termenii din

$Term(\mathcal{L}_\tau)$ satisfac proprietatea P.
Pentru orice termeni $t_1, t_2 \in Term(\mathcal{L}_\tau)$, $(t_1{=}t_2)[\frac{x}{a}] = t_1[\frac{x}{a}]{=}t_2[\frac{x}{a}]$, așadar, întrucât t_1 și t_2 satisfac proprietatea P conform celor de mai sus:
$s((t_1{=}t_2)[\frac{x}{a}]) = s(t_1[\frac{x}{a}]{=}t_2[\frac{x}{a}]) = \begin{cases} 1, & \text{dacă } s(t_1[\frac{x}{a}]) = s(t_2[\frac{x}{a}]), \\ 0, & \text{altfel} \end{cases} =$
$\begin{cases} 1, & \text{dacă } s[\frac{x}{a}](t_1) = s[\frac{x}{a}](t_2), \\ 0, & \text{altfel} \end{cases} = s[\frac{x}{a}](t_1{=}t_2)$, așadar formula $t_1{=}t_2$ satisface proprietatea Q.
Pentru orice $m \in \mathbb{N}^*$, orice simbol R de relație m–ară și orice termeni $t_1, \ldots, t_m \in Term(\mathcal{L}_\tau)$, avem, întrucât $t_1, \ldots, t_m$ satisfac proprietatea P conform celor de mai sus: $s(R(t_1, \ldots, t_m)[\frac{x}{a}]) = s(R(t_1[\frac{x}{a}], \ldots, t_m[\frac{x}{a}])) =$
$\begin{cases} 1, & \text{dacă } (s(t_1[\frac{x}{a}]), \ldots, s(t_m[\frac{x}{a}])) \in R^{\mathcal{A}}, \\ 0, & \text{altfel} \end{cases} =$
$\begin{cases} 1, & \text{dacă } (s[\frac{x}{a}](t_1), \ldots, s[\frac{x}{a}](t_m)) \in R^{\mathcal{A}}, \\ 0, & \text{altfel} \end{cases} = s[\frac{x}{a}](R(t_1, \ldots, t_m))$, așadar formula $R(t_1, \ldots, t_m)$ satisface proprietatea Q.
Fie $\psi, \chi \in Form(\mathcal{L}_\tau)$ astfel încât ψ și χ satisfac proprietatea Q, și $v \in Var$, iar
$w \begin{cases} = v, & \text{dacă } v \neq x, \\ \in Var \setminus (\{x\} \cup FV)(\psi)), & \text{altfel.} \end{cases}$

Atunci, cu operațiile booleene definite punctual pe funcții de la $Form(\mathcal{L}_\tau)$ la $\mathcal{L}_2$:
$s((\neg\psi)[^x_a]) = s(\neg\psi[^x_a]) = \overline{s(\psi[^x_a])} = \overline{s[^x_a](\psi)} = s[^x_a](\neg\psi)$, așadar $\neg\psi$ satisface proprietatea Q;
$s((\psi \to \chi)[^x_a]) = s(\psi[^x_a] \to \chi[^x_a]) = s(\psi[^x_a]) \to s(\chi[^x_a]) = s[^x_a](\psi) \to s[^x_a](\chi) = s[^x_a](\psi \to \chi)$, așadar $\psi \to \chi$ satisface proprietatea Q;
$s((\forall v\,\psi)[^x_a]) = s(\forall w\,\psi[^x_a]) = \bigwedge\limits_{b \in A} s[^w_b](\psi[^x_a]) = \bigwedge\limits_{b \in A} s[^w_b][^x_a](\psi) = \bigwedge\limits_{b \in A} s[^x_a][^w_b](\psi) =$
$s[^x_a](\forall v\,\psi)$, așadar $\forall v\,\psi$ satisface proprietatea Q.
Prin urmare mulțimea $\{\varphi \in Form(\mathcal{L}_\tau) \mid Q(\varphi)\}$ a formulelor lui $\mathcal{L}_\tau$ care satisfac proprietatea Q include mulțimea formulelor atomice ale lui $\mathcal{L}_\tau$ și este închisă la negație, implicație și cuantificatori universali, așadar $\{\varphi \in Form(\mathcal{L}_\tau) \mid Q(\varphi)\}$ include cea mai mică mulțime de formule ale lui $\mathcal{L}_\tau$ care satisfac proprietatea Q include mulțimea formulelor atomice ale lui $\mathcal{L}_\tau$ și este închisă la negație, implicație și cuantificatori universali, anume mulțimea $Form(\mathcal{L}_\tau)$ a tuturor formulelor lui $\mathcal{L}_\tau$, așadar $\{\varphi \in Form(\mathcal{L}_\tau) \mid Q(\varphi)\} = Form(\mathcal{L}_\tau)$, adică orice formulă din $\mathcal{L}_\tau$ satisface proprietatea Q, în particular orice $\varphi \in \mathcal{L}_\tau$ satisface proprietatea: $s(\varphi[^x_a]) = s[^x_a](\varphi)$.

Notație

Extindem notația de mai sus la mulțimi arbitrare de simboluri de constante: dacă C e o mulțime de simboluri de constante, notăm cu $\tau[C]$ signatura obținută din τ prin adăugarea elementelor lui C la mulțimea simbolurilor de constante ale lui τ.

Remarcă

În mod trivial, oricare două interpretări cu valori în A coincid pe $\emptyset$.
Într–adevăr, pentru orice interpretări $s_1, s_2 : Var \to A$,
$(\forall x)\,(x \in \emptyset \Rightarrow s_1(x) = s_2(x))$ e adevărată, pentru că $x \in \emptyset$ e falsă pentru orice x, așadar $x \in \emptyset \Rightarrow s_1(x) = s_2(x)$ e adevărată pentru orice x.

După cum am menționat mai sus:

Lemă (dacă două interpretări coincid pe variabilele dintr–un termen, atunci ele coincid în acel termen)

Fie $s_1, s_2 : Var \to A$ două interpretări. Atunci, pentru orice termen $t \in Term(\mathcal{L}_\tau)$, are loc implicația: $s_1 \mid_{V(t)} = s_2 \mid_{V(t)} \Rightarrow s_1(t) = s_2(t)$.

Demonstrație: Procedăm prin inducție după termeni, considerând proprietatea asupra unui termen t:

$$P(t): \quad s_1 \mid_{V(t)} = s_2 \mid_{V(t)} \Rightarrow s_1(t) = s_2(t)$$

Considerăm mulțimea $\{t \in Term(\mathcal{L}_\tau) \mid P(t)\} = \{t \in Term(\mathcal{L}_\tau) \mid s_1 \mid_{V(t)} = s_2 \mid_{V(t)} \Rightarrow s_1(t) = s_2(t)\}$ a termenilor care satisfac proprietatea P.
Dacă $t = x \in Var$, atunci $V(t) = V(x) = \{x\}$, așadar $s_1 \mid_{V(t)} = s_2 \mid_{V(t)}$ ddacă $s_1 \mid_{\{x\}} = s_2 \mid_{\{x\}}$ ddacă $s_1(x) = s_2(x)$ ddacă $s_1(t) = s_2(t)$, deci $t = x$ satisface proprietatea P. Prin urmare $Var \subseteq \{t \in Term(\mathcal{L}_\tau) \mid P(t)\}$.

Dacă $t = c$, unde c este un simbol de constantă, atunci $s_1(t) = s_1(c) = c^{\mathcal{A}} = s_2(c) = s_2(t)$, așadar implicația $s_1 \mid_{V(t)} = s_2 \mid_{V(t)} \Rightarrow s_1(t) = s_2(t)$ este adevărată, deci $t = c$ satisface proprietatea P (putem observa că, pentru simbolurile de constantă c, conform remarcii anterioare, proprietatea $P(c)$ este chiar echivalentă cu egalitatea $s_1(c) = s_2(c)$ – a se vedea și proprietatea de mai jos pentru enunțuri). Așadar mulțimea simbolurilor de constantă este inclusă în $\{t \in Term(\mathcal{L}_\tau) \mid P(t)\}$.

Dacă $n \in \mathbb{N}^*$, f este un simbol de operație n–ară și $t_1, \ldots, t_n \in Term(\mathcal{L}_\tau)$ termeni care satisfac proprietatea P, iar $t = f(t_1, \ldots, t_n)$, atunci: dacă $s_1 \mid_{V(t)} = s_2 \mid_{V(t)}$, adică $s_1(v) = s_2(v)$ pentru orice $v \in V(t) = \displaystyle\bigcup_{i=1}^{n} V(t_i)$, adică $s_1 \mid_{V(t_i)} = s_2 \mid_{V(t_i)}$ pentru fiecare $i \in \overline{1, n}$, atunci, cum $P(t_i)$ e satisfăcută pentru fiecare $i \in \overline{1, n}$, rezultă că $s_1(t_i) = s_2(t_i)$ pentru fiecare $i \in \overline{1, n}$, așadar $s_1(t) = s_1(f(t_1, \ldots, t_n)) = f^{\mathcal{A}}(s_1(t_1), \ldots, s_1(t_n)) = f^{\mathcal{A}}(s_2(t_1), \ldots, s_2(t_n)) = s_2(f(t_1, \ldots, t_n)) = s_2(t)$, deci $t = f(t_1, \ldots, t_n)$ satisface proprietatea P. Așadar mulțimea $\{t \in Term(\mathcal{L}_\tau) \mid P(t)\}$ este închisă la aplicarea simbolurilor de operații (cu arități nenule).

Prin urmare mulțimea $\{t \in Term(\mathcal{L}_\tau) \mid P(t)\}$ include cea mai mică mulțime care include mulțimea Var a variabilelor și mulțimea simbolurilor de constantă și este închisă la aplicarea simbolurilor de operații, anume mulțimea $Term(\mathcal{L}_\tau)$ a tuturor termenilor, așadar $\{t \in Term(\mathcal{L}_\tau) \mid P(t)\} = Term(\mathcal{L}_\tau)$, adică toți termenii $t \in Term(\mathcal{L}_\tau)$ satisfac proprietatea $P(t)$: $s_1 \mid_{V(t)} = s_2 \mid_{V(t)} \Rightarrow s_1(t) = s_2(t)$.

Propoziție (dacă două interpretări coincid pe variabilele libere dintr–o formulă, atunci ele coincid în acea formulă)

Fie $s_1, s_2 : Var \to A$ două interpretări. Atunci, pentru orice formulă $\varphi \in Form(\mathcal{L}_\tau)$, are loc implicația: $s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)} \Rightarrow s_1(\varphi) = s_2(\varphi)$.

Demonstrație: Procedăm prin inducție după formule, considerând proprietatea asupra unei formule φ:

$$P(\varphi): \quad s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)} \Rightarrow s_1(\varphi) = s_2(\varphi)$$

Considerăm mulțimea $\{\varphi \in Form(\mathcal{L}_\tau) \mid P(\varphi)\} = \{\varphi \in Form(\mathcal{L}_\tau) \mid s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)} \Rightarrow s_1(\varphi) = s_2(\varphi)\}$ a formulelor care satisfac proprietatea P.
Dacă $\varphi = t_1 \mathbf{=} t_2$, unde $t_1, t_2 \in Term(\mathcal{L}_\tau)$, atunci $FV(\varphi) = V(t_1) \cup V(t_2)$, așadar, dacă $s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)}$, atunci $s_1 \mid_{V(t_1)} = s_2 \mid_{V(t_1)}$ și $s_1 \mid_{V(t_2)} = s_2 \mid_{V(t_2)}$, ceea ce, conform lemei anterioare, implică $s_1(t_1) = s_2(t_1)$ și $s_1(t_2) = s_2(t_2)$, de unde rezultă: $s_1(\varphi) = 1$ ddacă $s_1(t_1) = s_1(t_2)$ ddacă $s_2(t_1) = s_2(t_2)$ ddacă $s_2(\varphi) = 1$; cum $s_1(\varphi), s_2(\varphi) \in \mathcal{L}_2 = \{0, 1\}$, rezultă și: $s_1(\varphi) = 0$ ddacă $s_1(\varphi) \neq 1$ ddacă $s_2(\varphi) \neq 1$ ddacă $s_2(\varphi) = 0$; așadar $s_1(\varphi) = s_2(\varphi)$.
Dacă $\varphi = R(t_1, \ldots, t_m)$, unde $m \in \mathbb{N}^*$, $t_1, \ldots, t_m \in Term(\mathcal{L}_\tau)$, iar R este un simbol de relație m–ară, atunci $FV(\varphi) = \bigcup_{j=1}^{m} V(t_j)$, așadar, dacă

$s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)}$, atunci $s_1 \mid_{V(t_j)} = s_2 \mid_{V(t_j)}$ pentru fiecare $j \in \overline{1, m}$, ceea ce, conform lemei anterioare, implică $s_1(t_j) = s_2(t_j)$ pentru fiecare $j \in \overline{1, m}$, de unde rezultă: $s_1(\varphi) = 1$ ddacă $(s_1(t_1), \ldots, s_1(t_m)) \in R^{\mathcal{A}}$ ddacă $(s_2(t_1), \ldots, s_2(t_m)) \in R^{\mathcal{A}}$ ddacă $s_2(\varphi) = 1$; ca mai sus, de aici rezultă că $s_1(\varphi) = s_2(\varphi)$.

Prin urmare $\{\varphi \in Form(\mathcal{L}_\tau) \mid P(\varphi)\}$ include mulțimea formulelor atomice.

Dacă $\varphi = \neg\, \psi$, unde $\psi \in Form(\mathcal{L}_\tau)$ a.î. proprietatea $P(\psi)$ e satisfăcută, atunci, întrucât $FV(\varphi) = FV(\psi)$, dacă $s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)}$, atunci $s_1 \mid_{FV(\psi)} = s_2 \mid_{FV(\psi)}$, prin urmare $s_1(\psi) = s_2(\psi)$, așadar $s_1(\varphi) = \overline{s_1(\psi)} = \overline{s_2(\psi)} = s_2(\varphi)$.

Așadar mulțimea $\{\varphi \in Form(\mathcal{L}_\tau) \mid P(\varphi)\}$ este închisă la negație.

Dacă $\varphi = \psi \rightarrow \chi$, unde $\psi, \chi \in Form(\mathcal{L}_\tau)$ a.î. proprietățile $P(\psi)$ și $P(\chi)$ sunt satisfăcute, atunci, întrucât $FV(\varphi) = FV(\psi) \cup FV(\chi)$, dacă $s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)}$, atunci $s_1 \mid_{FV(\psi)} = s_2 \mid_{FV(\psi)}$ și $s_1 \mid_{FV(\chi)} = s_2 \mid_{FV(\chi)}$, prin urmare $s_1(\psi) = s_2(\psi)$ și $s_1(\chi) = s_2(\chi)$, așadar $s_1(\varphi) = s_1(\psi) \rightarrow s_1(\chi) = s_2(\psi) \rightarrow s_2(\chi) = s_2(\varphi)$.

Așadar mulțimea $\{\varphi \in Form(\mathcal{L}_\tau) \mid P(\varphi)\}$ este închisă la implicație.

Dacă $\varphi = \forall\, x\psi$, unde $x \in Var$, iar $\psi \in Form(\mathcal{L}_\tau)$ a.î. proprietatea $P(\psi)$ e satisfăcută, atunci, dacă $s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)}$, atunci, pentru orice $a \in A$, $s_1[{}^x_a] \mid_{FV(\varphi)} = s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)} = s_2[{}^x_a]_{FV(\varphi)}$; dar $s_1[{}^x_a](x) = a = s_2[{}^x_a](x)$; așadar $s_1[{}^x_a] \mid_{FV(\varphi) \cup \{x\}} = s_2[{}^x_a]_{FV(\varphi) \cup \{x\}}$; cum $FV(\varphi) = FV(\forall\, x\psi) = FV(\psi) \setminus \{x\}$, deci $FV(\psi) \subseteq FV(\varphi) \cup \{x\}$, rezultă că $s_1[{}^x_a] \mid_{FV(\psi)} = s_2[{}^x_a] \mid_{FV(\psi)}$ pentru fiecare $a \in A$;

prin urmare $s_1(\varphi) = \bigwedge_{a \in A} s_1[{}^x_a](\psi) = \bigwedge_{a \in A} s_2[{}^x_a](\psi) = s_2(\varphi)$.

Așadar mulțimea $\{\varphi \in Form(\mathcal{L}_\tau) \mid P(\varphi)\}$ este închisă la aplicarea cuantificatorului universal.

Prin urmare $\{\varphi \in Form(\mathcal{L}_\tau) \mid P(\varphi)\}$ include cea mai mică mulțime care include mulțimea formulelor atomice și este închisă la negație, implicație și cuantificatori universali, anume întreaga mulțime $Form(\mathcal{L}_\tau)$, așadar $\{\varphi \in Form(\mathcal{L}_\tau) \mid P(\varphi)\} = Form(\mathcal{L}_\tau)$, adică orice formulă $\varphi \in Form(\mathcal{L}_\tau)$ satisface proprietatea $P(\varphi)$: $s_1 \mid_{FV(\varphi)} = s_2 \mid_{FV(\varphi)} \Rightarrow s_1(\varphi) = s_2(\varphi)$.

Corolar (cum enunțurile nu au variabile libere (i.e. $FV(\varphi) = \emptyset$ pt. orice enunț φ), rezultă că, într–un enunț, toate interpretările au aceeași valoare)

Dacă φ este un enunț, atunci $s(\varphi)$ nu depinde de interpretarea $s : Var \to A$.

Demonstrație: Fie $s_1, s_2 : Var \to A$. Atunci, conform remarcii precedente, $s_1 \mid_{FV(\varphi)} = s_1 \mid_\emptyset = s_2 \mid_\emptyset = s_2 \mid_{FV(\varphi)}$, de unde, conform propoziției anterioare, rezultă că $s_1(\varphi) = s_2(\varphi)$.

Notație

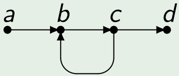
Corolarul anterior ne permite să notăm, pentru orice enunț φ și orice interpretare $s : Var \to A$, pe $s(\varphi)$ cu $||\varphi||_\mathcal{A}$ sau $||\varphi||$.

Exercițiu (renunțăm temporar la fixarea lui τ, $\mathcal{A}$ și s)

Fie signatura $\tau = (1; 2; \emptyset)$ și structura de ordinul I de această signatură $\mathcal{A} = (A; f^{\mathcal{A}}; R^{\mathcal{A}}; \emptyset)$, unde $A = \{a, b, c, d\}$ este o mulțime cu 4 elemente, iar funcția $f^{\mathcal{A}} : A \to A$ și relația binară $R^{\mathcal{A}}$ pe A vor fi notate respectiv cu f și R, și sunt definite prin: $f(a) = b$, $f(b) = c$, $f(c) = d$, $f(d) = a$ și $R = \{(a, b), (b, c), (c, b), (c, d)\} \subset A^2$:

x	a	b	c	d
$f^{\mathcal{A}}(x)$	b	c	d	a

$R^{\mathcal{A}}$:



Să se calculeze valorile de adevăr ale enunțurilor: $\exists\, x\, (R(x, f(x)) \wedge R(f(x), x))$ și $\exists\, x\, \forall\, y\, (R(y, f(f(x))) \vee R(f(x), y))$ în structura algebrică $\mathcal{A}$.

Rezolvare: Fixăm o interpretare arbitrară $s : Var \to A$. Conform definiției extinderii acesteia $s : Term(\mathcal{L}_\tau) \cup Form(\mathcal{L}_\tau) \to A \cup \mathcal{L}_2$, pentru orice termeni $t, u \in Term(\mathcal{L}_\tau)$:

$$s(R(t, u)) = \begin{cases} 1, & \text{dacă } (s(t), s(u)) \in R^{\mathcal{A}}, \\ 0, & \text{dacă } (s(t), s(u)) \notin R^{\mathcal{A}}. \end{cases}$$

Așadar, notând ca mai sus, având în vedere că valoarea de adevăr a unui enunț într–o interpretare nu depinde de acea interpretare și observând că, de exemplu, $s(R(a^{ct}, f(b^{ct})) = s(R(a^{ct}, f^{\mathcal{A}}(b)^{ct})) = s(R(a^{ct}, c^{ct}))$:

$||\exists x\,(R(x,f(x))\wedge R(f(x),x))||_{\mathcal{A}}=\bigvee_{e\in A} s[{}^{x}_{e}](R(x,f(x))\wedge R(f(x),x))=$
$s[{}^{x}_{a}](R(x,f(x))\wedge R(f(x),x))\vee s[{}^{x}_{b}](R(x,f(x))\wedge R(f(x),x))\vee$
$||\exists x\,(R(x,f(x))\wedge R(f(x),x))||=\bigvee_{e\in A} s[{}^{x}_{e}](R(x,f(x))\wedge R(f(x),x))=$
$s[{}^{x}_{a}](R(x,f(x))\wedge R(f(x),x))\vee s[{}^{x}_{b}](R(x,f(x))\wedge R(f(x),x))\vee$
$s[{}^{x}_{c}](R(x,f(x))\wedge R(f(x),x))\vee s[{}^{x}_{d}](R(x,f(x))\wedge R(f(x),x))=$
$(s[{}^{x}_{a}](R(x,f(x)))\wedge s[{}^{x}_{a}](R(f(x),x)))\vee(s[{}^{x}_{b}](R(x,f(x)))\wedge s[{}^{x}_{b}](R(f(x),x)))\vee(s[{}^{x}_{c}$
$](R(x,f(x)))\wedge s[{}^{x}_{c}](R(f(x),x)))\vee(s[{}^{x}_{d}](R(x,f(x)))\wedge s[{}^{x}_{d}](R(f(x),x)))=$
$(s(R(x,f(x))[{}^{x}_{a}])\wedge s(R(f(x),x)[{}^{x}_{a}]))\vee(s(R(x,f(x))[{}^{x}_{b}])\wedge s(R(f(x),x)[{}^{x}_{b}$
$]))\vee(s(R(x,f(x))[{}^{x}_{c}])\wedge s(R(f(x),x)[{}^{x}_{c}]))\vee(s(R(x,f(x))[{}^{x}_{d}])\wedge s(R(f(x),x)[{}^{x}_{d}]))=$
$(s(R(a^{ct},f(a^{ct}))\wedge s(R(f(a^{ct}),a^{ct}))\vee(s(R(b^{ct},f(b^{ct}))\wedge s(R(f(b^{ct}),b^{ct}))\vee$
$(s(R(c^{ct},f(c^{ct}))\wedge s(R(f(c^{ct}),c^{ct}))\vee(s(R(d^{ct},f(d^{ct}))\wedge s(R(f(d^{ct}),d^{ct}))=$
$(s(R(a^{ct},b^{ct})\wedge s(R(b^{ct},a^{ct}))\vee(s(R(b^{ct},c^{ct})\wedge s(R(c^{ct},b^{ct}))\vee(s(R(c^{ct},d^{ct})\wedge$
$s(R(d^{ct},c^{ct}))\vee(s(R(d^{ct},a^{ct})\wedge s(R(a^{ct},d^{ct}))=$
$(s(R(a^{ct},f^{\mathcal{A}}(a)^{ct})\wedge s(R(f^{\mathcal{A}}(a)^{ct},a^{ct}))\vee(s(R(b^{ct},f^{\mathcal{A}}(b)^{ct})\wedge s(R(f^{\mathcal{A}}(b)^{ct},b^{ct}))\vee$
$(s(R(c^{ct},f^{\mathcal{A}}(c)^{ct})\wedge s(R(f^{\mathcal{A}}(c)^{ct}),c^{ct}))\vee(s(R(d^{ct},f^{\mathcal{A}}(d)^{ct})\wedge s(R(f^{\mathcal{A}}(d)^{ct},d^{ct}))=$
$(s(R(a^{ct},b^{ct})\wedge s(R(b^{ct},a^{ct}))\vee(s(R(b^{ct},c^{ct})\wedge s(R(c^{ct},b^{ct}))\vee(s(R(c^{ct},d^{ct})\wedge$
$s(R(d^{ct},c^{ct}))\vee(s(R(d^{ct},a^{ct})\wedge s(R(a^{ct},d^{ct}))=1$, întrucât
$s(R(b^{ct},c^{ct})\wedge s(R(c^{ct},b^{ct})=1\wedge 1=1$;

pentru a doua formulă nu mai explicităm în etape, ca mai sus:

$$||\exists x \forall y\,(R(y,f(f(x))) \vee R(f(x),y))||_{\mathcal{A}} = \bigvee_{e\in A}\bigwedge_{g\in A} s[^x_e][^y_g](R(y,f(f(x))) \vee R(f(x),y)) = \bigvee_{e\in A}\bigwedge_{g\in A} s[^x_e][^y_g](R(y,f(f(x)))) \vee s[^x_e][^y_g](R(f(x),y))) = \bigvee_{e\in A}\bigwedge_{g\in A} s(R(y,f(f(x)))[^x_e][^y_g]) \vee s(R(f(x),y)[^x_e][^y_g])) =$$

$$\bigvee_{e\in A}\bigwedge_{g\in A}(s(R(g^{ct},f(f(e^{ct})))) \vee s(R(f(e^{ct}),g^{ct}))) =$$

$$\left(\bigwedge_{g\in A}(s(R(g^{ct},f(f(a^{ct})))) \vee s(R(f(a^{ct}),g^{ct})))\right) \vee$$

$$\left(\bigwedge_{g\in A}(s(R(g^{ct},f(f(b^{ct})))) \vee s(R(f(b^{ct}),g^{ct})))\right) \vee$$

$$\left(\bigwedge_{g\in A}(s(R(g^{ct},f(f(c^{ct})))) \vee s(R(f(c^{ct}),g^{ct})))\right) \vee$$

$$\left(\bigwedge_{g\in A}(s(R(g^{ct},f(f(d^{ct})))) \vee s(R(f(d^{ct}),g^{ct})))\right) = 0 \vee 0 \vee 0 \vee 0 = 0,$$ pentru că:

$(a,c),(b,a)\notin R^{\mathcal{A}}$, așadar
$s(R(a^{ct},f(f(a^{ct}))))\vee s(R(f(a^{ct}),a^{ct}))=s(R(a^{ct},c^{ct}))\vee s(R(b^{ct},a^{ct}))=0\vee 0=0$,
deci $\bigwedge\limits_{g\in A}(s(R(g^{ct},f(f(a^{ct}))))\vee s(R(f(a^{ct}),g^{ct})))=0$;
$(a,d),(c,a)\notin R^{\mathcal{A}}$, așadar
$s(R(a^{ct},f(f(b^{ct}))))\vee s(R(f(b^{ct}),a^{ct}))=s(R(a^{ct},d^{ct}))\vee s(R(c^{ct},a^{ct}))=0\vee 0=0$,
deci $\bigwedge\limits_{g\in A}(s(R(g^{ct},f(f(b^{ct}))))\vee s(R(f(b^{ct}),g^{ct})))=0$;
$(a,a),(d,a)\notin R^{\mathcal{A}}$, așadar
$s(R(a^{ct},f(f(c^{ct}))))\vee s(R(f(c^{ct}),a^{ct}))=s(R(a^{ct},a^{ct}))\vee s(R(d^{ct},a^{ct}))=0\vee 0=0$,
deci $\bigwedge\limits_{g\in A}(s(R(g^{ct},f(f(c^{ct}))))\vee s(R(f(c^{ct}),g^{ct})))=0$;
$(d,b),(a,d)\notin R^{\mathcal{A}}$, așadar $s(R(d^{ct},f(f(d^{ct}))))\vee s(R(f(d^{ct}),d^{ct}))=$
$s(R(d^{ct},b^{ct}))\vee s(R(a^{ct},d^{ct}))=0\vee 0=0$, deci
$\bigwedge\limits_{g\in A}(s(R(g^{ct},f(f(d^{ct}))))\vee s(R(f(d^{ct}),g^{ct})))=0$.

Definiție

Pentru orice enunț φ, notăm:

$$\mathcal{A}\vDash\varphi \text{ ddacă } ||\varphi||_{\mathcal{A}}=1.$$

Spunem că $\mathcal{A}$ *satisface enunțul* φ sau φ *este adevărat în* $\mathcal{A}$ sau $\mathcal{A}$ *este model pentru* φ ddacă $\mathcal{A} \vDash \varphi$.
Pentru orice mulțime Γ de enunțuri, spunem că $\mathcal{A}$ *satisface* Γ sau că $\mathcal{A}$ *este model pentru* Γ ddacă $\mathcal{A} \vDash \gamma$ pentru orice $\gamma \in \Gamma$; notăm acest lucru cu $\mathcal{A} \vDash \Gamma$.

Exemplu

Cu notațiile din exercițiul anterior: $\mathcal{A} \vDash \exists\, x\, (R(x, f(x)) \wedge R(f(x), x))$ și $\mathcal{A} \nvDash \exists\, x\, \forall\, y\, (R(y, f(f(x))) \vee R(f(x), y))$.

Exemplu (renunțăm temporar la fixarea lui τ și a lui $\mathcal{A}$)

Considerăm signatura $(2, 1; 2; 0)$, un simbol de operație binară f, unul de operație binară g, unul de relație binară R și unul de constantă c în limbajul de ordinul I de această signatură.
Fie $\mathcal{A} = (A, f^{\mathcal{A}}, g^{\mathcal{A}}, R^{\mathcal{A}}, c^{\mathcal{A}})$ o algebră de această signatură, astfel că: $f^{\mathcal{A}} : A^2 \to A$, $g^{\mathcal{A}} : A \to A$, $R^{\mathcal{A}} \subseteq A^2$ și $c^{\mathcal{A}} \in A$. Fie $x, y, z \in Var$.
Atunci operația binară $f^{\mathcal{A}}$ pe A:

- este idempotentă ddacă $\mathcal{A} \vDash \forall\, x\, (f(x, x) = x)$,
- este comutativă ddacă $\mathcal{A} \vDash \forall\, x\, \forall\, y\, (f(x, y) = f(y, x))$,
- este asociativă ddacă $\mathcal{A} \vDash \forall\, x\, \forall\, y\, \forall\, z\, (f(x, f(y, z)) = f(f(x, y), z))$,
- are elementul neutru la stânga $c^{\mathcal{A}}$ ddacă $\mathcal{A} \vDash \forall\, x\, (f(c, x) = x)$,
- are elementul neutru la dreapta $c^{\mathcal{A}}$ ddacă $\mathcal{A} \vDash \forall\, x\, (f(x, c) = x)$,

- are elementul neutru $c^{\mathcal{A}}$ ddacă $\mathcal{A}$ satisface cele două formule precedente sau, echivalent, ddacă $\mathcal{A} \vDash \forall x \, [(f(x, c)=x) \wedge (f(c, x)=x)]$,
- are un element neutru ddacă $\mathcal{A} \vDash \exists y \, \forall x \, [(f(x, y)=x) \wedge (f(y, x)=x)]$,

operația unară $g^{\mathcal{A}}$ pe A:

- este idempotentă ddacă $\mathcal{A} \vDash \forall x \, (g(g(x))=g(x))$,
- are punct fix ddacă $\mathcal{A} \vDash \exists x \, (g(x)=x)$,
- are punctul fix $c^{\mathcal{A}}$ ddacă $\mathcal{A} \vDash g(c)=c$,

iar, conform definiției valorii unei formule atomice într–o interpretare (care arată că, intuitiv, $R(x, y)$ semnifică faptul că perechea (x, y) se află în relația R), relația binară $R^{\mathcal{A}}$ pe A este:

- reflexivă ddacă $\mathcal{A} \vDash \forall x \, R(x, x)$,
- ireflexivă ddacă $\mathcal{A} \vDash \forall x \, \forall y \, [R(x, y) \rightarrow \neg \, (x=y)]$,
- simetrică ddacă $\mathcal{A} \vDash \forall x \, \forall y \, (R(x, y) \rightarrow R(y, x))$,
- antisimetrică ddacă $\mathcal{A} \vDash \forall x \, \forall y \, [(R(x, y) \wedge R(y, x)) \rightarrow x=y]$,
- asimetrică ddacă $\mathcal{A} \vDash \forall x \, \forall y \, (R(x, y) \rightarrow \neg \, R(y, x)]$,
- tranzitivă ddacă $\mathcal{A} \vDash \forall x \, \forall y \, \forall z \, [(R(x, y) \wedge R(y, z)) \rightarrow R(x, z)]$.

Definiție

Pentru orice $n \in \mathbb{N}^*$, orice variabile $x_1, \ldots, x_n$ și orice formulă $\varphi \in Form(\mathcal{L}_\tau)$ cu $FV(\varphi) \subseteq \{x_1, \ldots, x_n\}$, notăm:

$$\mathcal{A} \vDash \varphi \text{ ddacă } \mathcal{A} \vDash \forall x_1 \ldots \forall x_n \varphi.$$

Spunem că $\mathcal{A}$ *satisface formula* φ sau φ *este adevărat în* $\mathcal{A}$ sau $\mathcal{A}$ *este model pentru* φ ddacă $\mathcal{A} \vDash \varphi$.
Pentru orice mulțime $\Sigma \subseteq Form(\mathcal{L}_\tau)$ de formule, spunem că $\mathcal{A}$ *satisface* Σ sau că $\mathcal{A}$ *este model pentru* Σ ddacă $\mathcal{A}$ este model pentru fiecare formulă din Σ; notăm acest lucru cu $\mathcal{A} \vDash \Sigma$.

Exemplu

Cu notațiile din exemplul anterior:

- operația binară $f^{\mathcal{A}}$ pe A este asociativă ddacă $\mathcal{A} \vDash f(x, f(y, z)) = f(f(x, y), z)$;
- operația unară $g^{\mathcal{A}}$ pe A este idempotentă ddacă $\mathcal{A} \vDash g(g(x)) = g(x)$;
- relația binară $R^{\mathcal{A}}$ pe A este antisimetrică ddacă $\mathcal{A} \vDash (R(x, y) \wedge R(y, x)) \rightarrow x = y$.

Remarcă

$\mathcal{A} \vDash \emptyset$, întrucât, pentru orice φ, proprietatea $\varphi \in \emptyset$ este falsă, prin urmare implicația $\varphi \in \emptyset \rightarrow \mathcal{A} \vDash \varphi$ este adevărată, așadar proprietatea $(\forall \varphi)(\varphi \in \emptyset \rightarrow \mathcal{A} \vDash \varphi)$ este adevărată.

Adevărurile semantice și deducția semantică în logica clasică a predicatelor (pentru $\mathcal{L}_\tau$)

- Renunțăm la fixarea structurii $\mathcal{A}$ (**nu** și la fixarea signaturii τ).

Definiție

Dacă $\varphi \in Form(\mathcal{L}_\tau)$, atunci spunem că formula φ este *universal adevărată (adevăr semantic, tautologie)* ddacă, pentru orice structură de ordinul I $\mathcal{A}$ de signatură τ, $\mathcal{A} \vDash \varphi$; notăm acest lucru cu $\vDash \varphi$.

Definiție

Pentru orice mulțime $\Sigma \subseteq Form(\mathcal{L}_\tau)$ și orice $\varphi \in Form(\mathcal{L}_\tau)$, spunem că φ *se deduce semantic din ipotezele* Σ sau că φ *este consecință semantică a mulțimii de ipoteze* Σ ddacă φ este adevărată în orice model $\mathcal{A}$ al lui Σ, i. e., pentru orice structură de ordinul I $\mathcal{A}$ de signatură τ, are loc implicația: $\mathcal{A} \vDash \Sigma \Rightarrow \mathcal{A} \vDash \varphi$; notăm acest lucru prin: $\Sigma \vDash \varphi$.

Remarcă

Conform remarcii anterioare, pentru orice $\varphi \in Form(\mathcal{L}_\tau)$ și orice structură de ordinul I $\mathcal{A}$ de signatură τ, implicația $\mathcal{A} \vDash \emptyset \Rightarrow \mathcal{A} \vDash \varphi$ este echivalentă cu $\mathcal{A} \vDash \varphi$, așadar:

$$\emptyset \vDash \varphi \text{ ddacă } \vDash \varphi.$$

Teoremă (Teorema de completitudine tare/extinsă (abreviată TCT))

Pentru orice formulă $\varphi \in Form(\mathcal{L}_\tau)$ și orice mulțime de formule $\Sigma \subseteq Form(\mathcal{L}_\tau)$, are loc echivalența:

$$\Sigma \vdash \varphi \quad \Leftrightarrow \quad \Sigma \vDash \varphi.$$

Ideea demonstrației: $\Rightarrow$: Se procedează prin inducție după consecințele sintactice ale lui Σ, adică se arată că:

- axiomele au valoarea de adevăr 1 în orice interpretare, așadar sunt tautologii, implicit consecințe semantice ale lui Σ;
- în mod trivial, dacă $\varphi \in \Sigma$, atunci orice model pentru Σ este model pentru φ;
- dacă φ se obține prin **MP** sau **PG** din formule satisfăcute de orice model al lui Σ, atunci φ e satisfăcută de orice model al lui Σ;

prin urmare toate consecințele sintactice ale lui Σ sunt consecințe semantice ale lui Σ.

$\Leftarrow$: Presupunând prin absurd că $\Sigma \nvdash \varphi$, se demonstrează că acest lucru implică faptul $\Sigma \cup \{\neg\,\varphi\}$ este o mulțime de formule *consistentă*, adică având proprietatea că există $\psi \in Form(\mathcal{L}_\tau)$ astfel încât $\Sigma \cup \{\neg\,\varphi\} \nvdash \psi$, la fel ca în logica propozițională clasică.

Din acest fapt rezultă că există o mulțime de constante C și o mulțime

$\Lambda \subseteq Form(\mathcal{L}_{\tau[C]})$ astfel încât $\Sigma \cup \{\neg\varphi\} \subseteq \Lambda$, Λ e consistentă în $\mathcal{L}_{\tau[C]}$, pentru orice $\alpha \in Form(\mathcal{L}_\tau)$ și orice $x \in Var$ cu $FV(\alpha) \subseteq \{x\}$, există un simbol de constantă $c \in C$ astfel încât $\Lambda \vdash \exists x\alpha \rightarrow \alpha[\frac{x}{c}]$, și Λ este maximală între mulțimile de formule ale lui $\mathcal{L}_{\tau[C]}$ cu aceste proprietăți.
Pentru o astfel de mulțime Λ se poate construi o structură de ordinul I $\mathcal{B}$ de signatură $\tau[C]$ care satisface pe Λ. Atunci structura de ordinul I $\mathcal{A}$ de signatură τ subiacentă lui $\mathcal{B}$ satisface pe $\Sigma \cup \{\neg\varphi\}$, adică $\mathcal{A} \vDash \Sigma$ și $\mathcal{A} \vDash \neg\varphi$, prin urmare $\mathcal{A} \nvDash \varphi$, ceea ce contrazice ipoteza că $\Sigma \vDash \varphi$ a acestei implicații.
Așadar, dacă $\Sigma \vDash \varphi$, atunci $\Sigma \vdash \varphi$.

Corolar (Teorema de completitudine (abreviată **TC**))

Pentru orice formulă $\varphi \in Form(\mathcal{L}_\tau)$, are loc echivalența:

$$\vdash \varphi \quad \Leftrightarrow \quad \vDash \varphi.$$

Demonstrație: Luăm $\Sigma = \emptyset$ în **TCT**.

Corolar (Teorema deducției semantice)

Pentru orice mulțime de **formule** Σ, *orice* **enunț** φ *și orice* **formulă** ψ *din* $\mathcal{L}_\tau$, *are loc echivalența:*

$$\Sigma \vDash \varphi \rightarrow \psi \quad \Leftrightarrow \quad \Sigma \cup \{\varphi\} \vDash \psi.$$

Demonstrație: Din **TCT** și Teorema deducției sintactice.

Fie τ o signatură și $\mathcal{A}$ o structură de ordinul I de signatură τ, având mulțimea de elemente A.
Amintesc că mulțimea enunțurilor limbajului de ordinul I $\mathcal{L}_\tau$ este $\{\varphi \in Form(\mathcal{L}_\tau) \mid FV(\varphi) = \emptyset\}$.

Definiție (echivalența semantică)

Considerăm pe pe mulțimea enunțurilor din limbajul $\mathcal{L}_\tau$ o relație binară notată $\vDash\!\!\!\!\dashv$, definită astfel: oricare ar fi enunțurile φ și ψ, $\varphi \vDash\!\!\!\!\dashv \psi$ ddacă, oricare ar fi structura algebrică $\mathcal{M}$ de signatură τ, avem: $\mathcal{M} \vDash \varphi$ ddacă $\mathcal{M} \vDash \psi$.
Adică o pereche de enunțuri $(\varphi, \psi) \in \vDash\!\!\!\!\dashv$ ddacă φ și ψ au aceleași modele.
Relația binară $\vDash\!\!\!\!\dashv$ se numește *echivalența semantică* în limbajul $\mathcal{L}_\tau$. Pentru orice enunțuri φ, ψ, spunem că φ și ψ sunt *echivalente semantic* ddacă $(\varphi, \psi) \in \vDash\!\!\!\!\dashv$.

Remarcă

$\vDash\!\!\!\!\dashv$ este o relație de echivalență pe mulțimea enunțurilor lui $\mathcal{L}_\tau$. Într–adevăr, reflexivitatea, simetria și tranzitivitatea lui $\vDash\!\!\!\!\dashv$ sunt imediate.

Remarcă (două enunțuri sunt echivalente semantic ddacă echivalența lor logică este tautologie)

Pentru orice enunțuri φ, ψ, avem:

$$\varphi \models\!\mid \psi \text{ ddacă } \models \varphi \leftrightarrow \psi.$$

Într–adevăr, în primul rând să observăm că $\varphi \leftrightarrow \psi$ este un enunț, întrucât $FV(\varphi \leftrightarrow \psi) = FV(\varphi) \cup FV(\psi) = \emptyset \cup \emptyset = \emptyset$.
Acum, considerând o interpretare arbitrară $s : Var \to A$, avem: $||\varphi \leftrightarrow \psi||_{\mathcal{A}} = s(\varphi \leftrightarrow \psi) = s(\varphi) \leftrightarrow s(\psi) = ||\varphi||_{\mathcal{A}} \leftrightarrow ||\psi||_{\mathcal{A}}$.
Prin urmare: $\models \varphi \leftrightarrow \psi$ ddacă, pentru orice algebră $\mathcal{M}$ de tip τ, $\mathcal{M} \models \varphi \leftrightarrow \psi$, dacă, pentru orice algebră $\mathcal{M}$ de tip τ, $||\varphi \leftrightarrow \psi||_{\mathcal{M}} = 1$, dacă, pentru orice algebră $\mathcal{M}$ de tip τ, $||\varphi||_{\mathcal{M}} \leftrightarrow ||\psi||_{\mathcal{M}} = 1$, dacă, pentru orice algebră $\mathcal{M}$ de tip τ, $||\varphi||_{\mathcal{M}} = ||\psi||_{\mathcal{M}}$, dacă, pentru orice algebră $\mathcal{M}$ de tip τ, avem echivalența: $||\varphi||_{\mathcal{M}} = 1 \Leftrightarrow ||\psi||_{\mathcal{M}} = 1$, dacă, pentru orice algebră $\mathcal{M}$ de tip τ, avem echivalența: $\mathcal{M} \models \varphi \Leftrightarrow \mathcal{M} \models \psi$, ddacă $\varphi \models\!\mid \psi$.
Am folosit următoarele fapte:

- pentru orice elemente a, b ale unei algebre Boole, avem: $a \leftrightarrow b = 1$ ddacă $a = b$;
- oricare ar fi algebra $\mathcal{M}$ de tip τ: $||\varphi||_{\mathcal{M}}, ||\psi||_{\mathcal{M}} \in \mathcal{L}_2 = \{0, 1\}$, așadar $||\varphi||_{\mathcal{M}} = ||\psi||_{\mathcal{M}}$ ddacă fie ambele sunt egale cu 1, fie ambele sunt egale cu 0.

Remarcă

Conform remarcii anterioare, pentru orice enunțuri φ, ψ, avem: $\varphi \vDash\!\dashv \psi$ ddacă, pentru orice algebră $\mathcal{M}$ de tip τ, $||\varphi||_{\mathcal{M}} = ||\psi||_{\mathcal{M}}$.
Din această exprimare a relației $\vDash\!\dashv$ de echivalență semantică pentru $\mathcal{L}_\tau$ rezultă că $\vDash\!\dashv$ are proprietățile relației $\vDash\!\dashv$ de echivalență semantică din logica propozițională clasică, anume cele provenite din proprietățile booleene *(a se vedea mai jos)*.
În plus, au loc proprietăți de tipul următor *(a se vedea o listă extinsă mai jos)*, pentru orice $x, y \in Var$ și orice $\alpha, \beta \in Form(\mathcal{L}_\tau)$:

- dacă $FV(\alpha) \subseteq \{x, y\}$, astfel că următoarele formule sunt enunțuri, atunci: $\forall x \forall y \alpha \ \vDash\!\dashv \ \forall y \forall x \alpha$;
- dacă α este un enunț, $x \notin V(\alpha)$ și $FV(\beta) \subseteq \{x\}$, astfel că următoarele formule sunt enunțuri, atunci: $\alpha \ \vDash\!\dashv \ \forall x \alpha \ \vDash\!\dashv \ \exists x \alpha$ și $\forall x(\alpha \vee \beta) \ \vDash\!\dashv \ \alpha \vee \forall x \beta \ \vDash\!\dashv \ \forall x \alpha \vee \forall x \beta$.

Într–adevăr, considerând $x, y \in Var$, două **formule arbitrare** α, β, o algebră $\mathcal{M}$ de tip τ, cu mulțimea elementelor M, și o interpretare $s : Var \to M$, avem: notând, pentru orice $a, b \in M$, cu $t_{a,b} : Var \to M$ interpretarea definită prin: oricare ar fi $v \in Var$, $t_{a,b}(v) = \begin{cases} a, & \text{dacă } v = x, \\ b, & \text{dacă } v = y, \\ s(v), & \text{dacă } v \notin \{x, y\}, \end{cases}$ observăm că $t_{a,b} = (s[{}^{x}_{a}])[{}^{y}_{b}] = (s[{}^{y}_{b}])[{}^{x}_{a}]$, așadar:

$$s(\forall x\forall y\alpha) = \bigwedge_{a\in A} s[^x_a](\forall y\alpha) = \bigwedge_{a\in A}\bigwedge_{b\in A}(s[^x_a])[^y_b](\alpha) = \bigwedge_{a\in A}\bigwedge_{b\in A} t_{a,b}(\alpha) =$$

$$\bigwedge_{b\in A}\bigwedge_{a\in A} t_{a,b}(\alpha) = \bigwedge_{b\in A}\bigwedge_{a\in A}(s[^y_b])[^x_a](\alpha) = \bigwedge_{b\in A} s[^y_b](\forall x\alpha) = s(\forall y\forall x\alpha);$$

iar, dacă $\forall x\forall y\alpha$ este un **enunț**, adică $FV(\alpha) \subseteq \{x, y\}$, astfel că și $\forall y\forall x\alpha$ este un enunț, atunci, conform calculului anterior,
$||\forall x\forall y\alpha||_{\mathcal{M}} = s(\forall x\forall y\alpha) = s(\forall y\forall x\alpha) = ||\forall x\forall y\alpha||_{\mathcal{M}}$, **și la fel mai jos**;

- dacă $x \notin FV(\alpha)$, în particular dacă $x \notin V(\alpha)$, atunci, pentru orice $a \in A$, $s\mid_{FV(\alpha)} = s[^x_a]\mid_{FV(\alpha)}$, așadar, conform propoziției precedente, pentru orice $a \in A$, $s(\alpha) = s[^x_a](\alpha)$, prin urmare:

$$s(\forall x\alpha) = \bigwedge_{a\in A} s[^x_a](\alpha) = s(\alpha) = \bigvee_{a\in A} s[^x_a](\alpha) = s(\exists x\alpha);$$

întrucât algebra Boole $\mathcal{L}_2$ este finită și, în particular, completă, fapt pe care l–am folosit deja când am admis scrieri de tipul $\bigwedge_{a\in A} s[^x_a](\alpha)$ sau $\bigvee_{a\in A} s[^x_a](\alpha)$, deci am folosit existența conjuncțiilor și disjuncțiilor arbitrare, avem:

$$s(\forall x(\alpha\vee\beta)) = \bigwedge_{a\in A} s[^x_a](\alpha\vee\beta) = \bigwedge_{a\in A}(s[^x_a](\alpha)\vee s[^x_a](\beta)) = \bigwedge_{a\in A}(s(\alpha)\vee s[^x_a](\beta)) =$$

$s(\alpha)\vee\bigwedge_{a\in A} s[^x_a](\beta) = s(\alpha)\vee s(\forall x\beta) = s(\alpha\vee\forall x\beta)$, dar, conform celor de mai sus, avem și: $s(\alpha)\vee s(\forall x\beta) = s(\forall x\alpha)\vee s(\forall x\beta) = s(\forall x\alpha\vee\forall x\beta)$.

Formule fără cuantificatori

Definiție (obținute din formule atomice aplicând conectori logici)

Formulele fără cuantificatori ale limbajului $\mathcal{L}_\tau$ sunt cuvintele finite peste alfabetul de mai sus definite, recursiv, astfel:

1. formulele atomice sunt formule fără cuantificatori;
2. dacă φ este o formulă fără cuantificatori, atunci $\neg\varphi$ este o formulă fără cuantificatori;
3. dacă φ și ψ sunt formule fără cuantificatori, atunci $\varphi \rightarrow \psi$ este o formulă fără cuantificatori.

Cum formulele au lungimi finite, rezultă că orice formulă fără cuantificatori se obține prin aplicarea regulilor ①, ② și ③ de un număr finit de ori.

Remarcă

Formulele fără cuantificatori sunt exact formulele φ cu $V(\varphi) = FV(\varphi)$, adică exact formulele fără variabile legate, i.e. formulele în care toate variabilele sunt libere (adică nu sunt cuantificate, nu li se aplică niciun cuantificator).

Notație (ad–hoc)

Notăm cu $Form_{\not\forall\,\not\exists}(\mathcal{L}_\tau)$ mulțimea formulelor fără cuantificatori.

Substituții

Definiție (substituțiile sunt funcțiile de la variabile la termeni)

O *substituție* este o funcție $\sigma : Var \to Term(\mathcal{L}_\tau)$.

Definiție (extinderile substituțiilor la termeni sunt funcțiile de la termeni la termeni care păstrează simbolurile de operații)

Fie $\sigma : Var \to Term(\mathcal{L}_\tau)$. Următoarea extindere a lui σ la $Term(\mathcal{L}_\tau)$ se numește tot *substituție* (și, de obicei, se notează tot cu σ): $\widetilde{\sigma} : Term(\mathcal{L}_\tau) \to Term(\mathcal{L}_\tau)$, definită, recursiv, astfel:

- $\widetilde{\sigma} \mid_{Var} = \sigma$;
- oricare ar fi $c \in \mathcal{C}$, $\widetilde{\sigma}(c) = c$;
- pentru orice $n \in \mathbb{N}^*$, orice $f \in \mathcal{F}$ având aritatea n și orice $t_1, \ldots, t_{n_i} \in Term(\mathcal{L}_\tau)$, $\widetilde{\sigma}(f(t_1, \ldots, t_{n_i})) = f(\widetilde{\sigma}(t_1), \ldots, \widetilde{\sigma}(t_n))$.

Ca în cazul interpretărilor, extinderea $\widetilde{\sigma}$ din definiția anterioară este:

- **complet definită**, pentru că toți termenii se scriu ca mai sus, i.e. sunt obținuți prin acea recursie;
- **corect definită**, pentru că orice termen are o unică scriere ca mai sus.

Definiție (substituțiile pe formulele fără cuantificatori sunt funcțiile de la $Form_{\not\forall\not\exists}(\mathcal{L}_\tau)$ la $Form_{\not\forall\not\exists}(\mathcal{L}_\tau)$ care păstrează simbolurile de relații și conectorii logici)

Fie $\sigma : Term(\mathcal{L}_\tau) \to Term(\mathcal{L}_\tau)$ o substituție (adică o extindere ca în definiția anterioară a unei funcții de la Var la $Term(\mathcal{L}_\tau)$).
Notăm tot cu $\sigma : Form_{\not\forall\not\exists}(\mathcal{L}_\tau) \to Form_{\not\forall\not\exists}(\mathcal{L}_\tau)$ funcția, numită tot substituție, definită, recursiv, astfel:

- pentru orice $t_1, t_2 \in Term(\mathcal{L}_\tau)$, $\sigma(t_1 = t_2) \ = \ \sigma(t_1) = \sigma(t_2)$;
- pentru orice $m \in \mathbb{N}^*$, orice $R \in \mathcal{R}$ de aritate m și orice $t_1, \ldots, t_m \in Term(\mathcal{L}_\tau)$, $\sigma(R(t_1, \ldots, t_m)) = R(\sigma(t_1), \ldots, \sigma(t_m))$;
- pentru orice formulă fără cuantificatori φ, $\sigma(\neg\varphi) = \neg\sigma(\varphi)$;
- pentru orice formule fără cuantificatori φ, ψ, $\sigma(\varphi \to \psi) = \sigma(\varphi) \to \sigma(\psi)$.

Prin inducție după formulele fără cuantificatori, întrucât, conform definiției formulelor fără cuantificatori, $Form_{\not\forall\not\exists}(\mathcal{L}_\tau)$ este cea mai mică mulțime de formule care conține toate formulele atomice și este închisă la negație și implicație, obținem:

Remarcă

Funcția $\sigma : Form_{\not\forall\not\exists}(\mathcal{L}_\tau) \to Form_{\not\forall\not\exists}(\mathcal{L}_\tau)$ din definiția anterioară este corect și complet definită pe $Form_{\not\forall\not\exists}(\mathcal{L}_\tau)$.

Remarcă

Pentru orice substituție $\sigma : Form_{\not\forall,\not\exists}(\mathcal{L}_\tau) \rightarrow Form_{\not\forall,\not\exists}(\mathcal{L}_\tau)$ și orice $\varphi, \psi \in Form_{\not\forall,\not\exists}(\mathcal{L}_\tau)$, au loc:

- $\sigma(\varphi \vee \psi) = \sigma(\varphi) \vee \sigma(\psi)$;
- $\sigma(\varphi \wedge \psi) = \sigma(\varphi) \wedge \sigma(\psi)$;
- $\sigma(\varphi \leftrightarrow \psi) = \sigma(\varphi) \leftrightarrow \sigma(\psi)$.

Formă prenex și formă normală conjunctivă prenex

Fie τ și $\mathcal{A}$ ca în secțiunea anterioară a cursului.

Definiție

Se numește *formulă prenex* sau *formulă în formă prenex* o formulă de tipul

$$\mathcal{Q}_1 x_1 \ldots \mathcal{Q}_n x_n \varphi,$$

unde $n \in \mathbb{N}^*$, $x_1, \ldots, x_n \in Var$, $\mathcal{Q}_1, \ldots, \mathcal{Q}_n \in \{\forall, \exists\}$ și φ este o formulă fără cuantificatori.

Remarcă (orice enunț poate fi pus în formă prenex)

Pentru orice enunț ε, există o formulă prenex ψ astfel încât $\varepsilon \vDash\!\!\!\!\Dashv \psi$.

Putem obține o formulă prenex echivalentă semantic cu ε folosind următoarele echivalențe semantice valabile pentru orice enunțuri α, β, γ, orice formule φ, ψ, χ și orice variabile x, y a.î. $FV(\varphi) \cup FV(\psi) \subseteq \{x\}$, iar $FV(\chi) \subseteq \{x, y\}$, astfel că următoarele formule sunt enunțuri:

(proprietăți pentru mutarea cuantificatorilor în față)

- $\forall x \forall y \chi \;\vDash\!\!\!\dashv\; \forall y \forall x \chi$ și $\exists x \exists y \chi \;\vDash\!\!\!\dashv\; \exists y \exists x \chi$
- $\neg \forall x \chi \;\vDash\!\!\!\dashv\; \exists x \neg \chi$ și $\neg \exists x \chi \;\vDash\!\!\!\dashv\; \forall x \neg \chi$
- $\forall x(\varphi \wedge \psi) \;\vDash\!\!\!\dashv\; \forall x \varphi \wedge \forall x \psi$ și $\exists x(\varphi \vee \psi) \;\vDash\!\!\!\dashv\; \exists x \varphi \vee \exists x \psi$
- dacă $x \notin V(\varphi)$, atunci: $\begin{cases} \varphi \;\vDash\!\!\!\dashv\; \forall x \varphi \;\vDash\!\!\!\dashv\; \exists x \varphi \\ \varphi \vee \forall x \psi \;\vDash\!\!\!\dashv\; \forall x(\varphi \vee \psi) \;\vDash\!\!\!\dashv\; \forall x \varphi \vee \forall x \psi \\ \varphi \wedge \exists x \psi \;\vDash\!\!\!\dashv\; \exists x(\varphi \wedge \psi) \;\vDash\!\!\!\dashv\; \exists x \varphi \wedge \exists x \psi \end{cases}$

(proprietăți din calculul propozițional)

1. $\alpha \rightarrow \beta \;\vDash\!\!\!\dashv\; \neg \alpha \vee \beta$ și $\alpha \leftrightarrow \beta \;\vDash\!\!\!\dashv\; (\neg \alpha \vee \beta) \wedge (\neg \beta \vee \alpha)$
2. $\alpha \vee \alpha \;\vDash\!\!\!\dashv\; \alpha \wedge \alpha \;\vDash\!\!\!\dashv\; \alpha$
3. $\alpha \vee \beta \;\vDash\!\!\!\dashv\; \beta \vee \alpha$ și $\alpha \wedge \beta \;\vDash\!\!\!\dashv\; \beta \wedge \alpha$
4. $(\alpha \vee \beta) \vee \gamma \;\vDash\!\!\!\dashv\; \alpha \vee (\beta \vee \gamma)$ și $(\alpha \wedge \beta) \wedge \gamma \;\vDash\!\!\!\dashv\; \alpha \wedge (\beta \wedge \gamma)$
5. $\alpha \vee (\alpha \wedge \beta) \;\vDash\!\!\!\dashv\; \alpha \wedge (\alpha \vee \beta) \;\vDash\!\!\!\dashv\; \alpha$
6. $\alpha \vee (\beta \wedge \gamma) \;\vDash\!\!\!\dashv\; (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$ și $\alpha \wedge (\beta \vee \gamma) \;\vDash\!\!\!\dashv\; (\alpha \wedge \beta) \vee (\alpha \wedge \gamma)$
7. $\neg \neg \alpha \;\vDash\!\!\!\dashv\; \alpha$
8. $\neg (\alpha \vee \beta) \;\vDash\!\!\!\dashv\; \neg \alpha \wedge \neg \beta$ și $\neg (\alpha \wedge \beta) \;\vDash\!\!\!\dashv\; \neg \alpha \vee \neg \beta$
9. $\alpha \vee (\beta \wedge \neg \beta) \;\vDash\!\!\!\dashv\; \alpha \vee (\beta \wedge \neg \beta \wedge \gamma) \;\vDash\!\!\!\dashv\; \alpha \wedge (\beta \vee \neg \beta) \;\vDash\!\!\!\dashv\; \alpha \wedge (\beta \vee \neg \beta \vee \gamma) \;\vDash\!\!\!\dashv\; \alpha$

Remarcă

Cum, în ultima remarcă din secțiunea anterioară a cursului, α și β sunt **formule arbitrare**, rezultă că **echivalențele semantice** de mai sus au loc și în **subformule** care nu sunt neapărat enunțuri, adică putem aplica faptul că, pentru orice $\varphi, \psi \in Form(\mathcal{L}_\tau)$, orice algebră $\mathcal{M}$ de signatură τ, cu mulțimea elementelor M, și orice interpretare $s : Var \to M$, au loc, de exemplu:

- $s(\neg(\alpha \vee \beta)) = s(\neg\alpha \wedge \neg\beta)$, $s(\exists x \exists y \alpha) = s(\exists y \exists x \alpha)$;
- dacă $x \notin FV(\alpha)$, atunci $s(\alpha) = s(\forall x \alpha) = s(\exists x \alpha)$ și $s(\alpha \wedge \exists x \beta) = s(\exists x(\alpha \wedge \beta)) = s(\exists x \alpha \wedge \exists x \beta)$ etc..

Definiție (FNC prenex)

- Un *literal* este o formulă atomică sau negația unei formule atomice.
- O *clauză* este o disjuncție de literali.

Orice clauză se identifică cu mulțimea literalilor care o compun.

- O *formă normală conjunctivă prenex* sau un *enunț în formă normală conjunctivă prenex* (*FNC prenex*) este un enunț de forma:

$$\forall x_1 \ldots \forall x_n \varphi,$$

unde $n \in \mathbb{N}^*$, $x_1, \ldots, x_n \in Var$ și φ este o conjuncție de clauze, i. e. o

Definiție (continuare)

conjuncție de disjuncții de literali, implicit φ este o formulă fără cuantificatori, așadar $x_1, \ldots, x_n \in Var$ este o formulă prenex.

Orice conjuncție de clauze se identifică, cu mulțimea acelor clauze.
Orice FNC prenex $\forall x_1 \ldots \forall x_n \varphi$ se identifică cu mulțimea clauzelor care îl compun pe φ.

Observație

Întrucât toate enunțurile au lungime finită, conjuncțiile și disjuncțiile la care face referire definiția de mai sus sunt finite.

Remarcă

O mulțime finită și nevidă de enunțuri este satisfiabilă ddacă, conjuncția enunțurilor din acea mulțime e satisfiabilă.
Într–adevăr, dacă $n \in \mathbb{N}^*$ și $\gamma_1, \ldots, \gamma_n$ sunt enunțuri, iar $\mathcal{M}$ este o algebră de signatură τ, atunci: $||\gamma_1 \wedge \ldots \wedge \gamma_n||_{\mathcal{M}} = ||\gamma_1||_{\mathcal{M}} \wedge \ldots \wedge ||\gamma_n||_{\mathcal{M}}$, așadar:
$\mathcal{M} \vDash \gamma_1 \wedge \ldots \wedge \gamma_n$ ddacă $||\gamma_1 \wedge \ldots \wedge \gamma_n||_{\mathcal{M}} = 1$ ddacă $||\gamma_1||_{\mathcal{M}} \wedge \ldots \wedge ||\gamma_n||_{\mathcal{M}} = 1$ ddacă $||\gamma_1||_{\mathcal{M}} = \ldots = ||\gamma_n||_{\mathcal{M}} = 1$ ddacă, pentru fiecare $i \in \overline{1, n}$, $\mathcal{M} \vDash \gamma_i$, ddacă $\mathcal{M} \vDash \{\gamma_1, \ldots, \gamma_n\}$.

Remarcă

În mod trivial, mulțimea vidă de enunțuri este satisfiabilă, întrucât următoarea afirmație este adevărată: pentru orice algebră $\mathcal{M}$ de tip τ și orice γ, $\gamma \in \emptyset \Rightarrow \mathcal{M} \vDash \gamma$.

Definiție

Dacă ε este un enunț, iar ψ este o FNC prenex cu $\varepsilon \;\reflectbox{$\vDash$}\!\vDash\; \psi$, atunci mulțimea de clauze care îl compun pe ψ se numește *formă clauzală* pentru ε.
Dacă $n \in \mathbb{N}^*$, iar $\varepsilon_1, \ldots, \varepsilon_n$ sunt enunțuri, atunci reuniunea unor forme clauzale pentru $\varepsilon_1, \ldots, \varepsilon_n$ se numește *formă clauzală* pentru $\{\varepsilon_1, \ldots, \varepsilon_n\}$.

Definiție

- **Clauza vidă** (i. e. clauza fără literali, clauza fără elemente) se notează cu $\square$ (pentru a o deosebi de **mulțimea vidă de clauze**, $\emptyset$).
- O clauză C se zice *trivială* ddacă există o formulă atomică α astfel încât $\alpha, \neg\,\alpha \in C$.
- O clauză nevidă $C = \{L_1, \ldots, L_n\}$ (cu $n \in \mathbb{N}^*$ și $L_1, \ldots, L_n$ literali) se zice *satisfiabilă* ddacă enunțul $L_1 \vee \ldots \vee L_n$ corespunzător lui C e satisfiabil.
- O mulțime finită de clauze se zice *satisfiabilă* ddacă enunțul în FNC corespunzător acelei mulțimi de clauze e satisfiabil.

Remarcă

Fie $n \in \mathbb{N}^*$, iar $\varphi_1, \ldots, \varphi_n$ și φ enunțuri.
Să observăm că are loc echivalența:

$$\vDash \varphi \text{ ddacă } \neg\,\varphi \text{ e nesatisfiabil (adică nu are model).}$$

Într–adevăr: $\vDash \varphi$ (i.e. φ e adevăr semantic) ddacă, pentru orice algebră $\mathcal{A}$ de signatură τ, avem $||\varphi||_{\mathcal{A}} = 1$ ddacă, pentru orice algebră $\mathcal{A}$ de signatură τ, avem $||\neg\,\varphi||_{\mathcal{A}} = \overline{||\varphi||_{\mathcal{A}}} = \overline{1} = 0$ ddacă $\neg\,\varphi$ e nesatisfiabil.
Mai mult, ca o consecință a Teoremei Deducției și a observației de mai sus, avem următoarea caracterizare a deducției semantice $\{\varphi_1, \ldots, \varphi_n\} \vDash \varphi$:

$$\{\varphi_1, \ldots, \varphi_n\} \vDash \varphi \text{ ddacă } \{\varphi_1 \wedge \ldots \wedge \varphi_n\} \vDash \varphi$$

$$\text{ddacă } \vDash (\varphi_1 \wedge \ldots \wedge \varphi_n) \rightarrow \varphi$$

$$\text{ddacă enunțul } \neg\,[(\varphi_1 \wedge \ldots \wedge \varphi_n) \rightarrow \varphi] \text{ e nesatisfiabil.}$$

Definiție (definim substituțiile pe clauze – la fel putem proceda pentru FNC fără cuantificatori reprezentate ca mulțimi de clauze)

Fie $\sigma : Term(\mathcal{L}_\tau) \to Term(\mathcal{L}_\tau)$ o substituție.
Pentru orice clauză $C = \{L_1, \ldots, L_p\}$, unde $p \in \mathbb{N}^*$, iar $L_1, \ldots, L_p$ sunt literali, definim $\sigma(C) = \{\sigma(L_1), \ldots, \sigma(L_p)\}$.

Remarcă (la fel va fi în cazul FNC fără cuantificatori)

Cu notațiile din definiția anterioară, clauza $\sigma(C) = \{\sigma(L_1), \ldots, \sigma(L_p)\}$ corespunde enunțului $\sigma(L_1) \vee \ldots \vee \sigma(L_p) = \sigma(L_1 \vee \ldots \vee L_p)$ dat de valoarea lui σ în enunțul $L_1 \vee \ldots \vee L_p$ corespunzător clauzei C.

Notație

Fie $n \in \mathbb{N}^*$, $x_1, \ldots, x_n \in Var$, două câte două distincte, și $t_1, \ldots, t_n \in Term(\mathcal{L}_\tau)$.
Notăm cu

$$\{x_1/t_1, \ldots, x_n/t_n\} : Var \to Term(\mathcal{L}_\tau)$$

substituția definită prin:

$$\begin{cases} (\forall\, i \in \overline{1,n})\,(\{x_1/t_1, \ldots, x_n/t_n\}(x_i) = t_i); \\ (\forall\, x \in Var \setminus \{x_1, \ldots, x_n\})\,(\{x_1/t_1, \ldots, x_n/t_n\}(x) = x). \end{cases}$$

Notație

Fie $n, k \in \mathbb{N}^*$, $x_1, \ldots, x_n \in Var$, două câte două distincte, $i_1, \ldots, i_k \in \overline{1, n}$, astfel încât $1 \leq i_1 < i_2 < \ldots < i_k \leq n$, $t_{i_1}, \ldots, t_{i_k} \in Term(\mathcal{L}_\tau)$ și $\varphi(x_1, \ldots, x_n)$ o formulă fără cuantificatori cu $V(\varphi) \subseteq \{x_1, \ldots, x_n\}$.
Atunci notăm cu

$$\varphi(x_1, \ldots, x_{i_1-1}, t_{i_1}, x_{i_1+1}, \ldots, x_{i_2-1}, t_{i_2}, x_{i_2+1}, \ldots, x_{i_k-1}, t_{i_k}, x_{i_k+1}, \ldots, x_n)$$

formula $\{x_{i_1}/t_{i_1}, \ldots, x_{i_k}/t_{i_k}\}(\varphi)$.

Exemplu

Dacă v, x, y, z sunt variabile distincte, s, t, u sunt termeni, ρ este un simbol de relație binară, f e un simbol de operație binară, g e un simbol de operație unară și c este un simbol de constantă, atunci rezultatul aplicării substituției $\{x/s, y/t, z/u\}$ asupra formulei fără cuantificatori:

$$\varphi(v, x, y, z) = \neg\, [\rho(g(x), f(v, y)) \rightarrow \neg\, \rho(f(g(y), g(z)), f(v, g(c))]$$

este formula fără cuantificatori: $\{x/s, y/t, z/u\}(\varphi(v, x, y, z)) = \varphi(v, s, t, u) =$

$$\neg\, [\rho(g(s), f(v, t)) \rightarrow \neg\, \rho(f(g(t), g(u)), f(v, g(c))].$$

Amintesc că și **constantele** sunt **operații**, anume *operații fără argumente*, *operații cu 0 argumente*, *operații de aritate 0*, numite și *operații zeroare* sau *nulare*, așadar nu trebuie tratate ca un caz separat în cele ce urmează.

Într–un anumit context, **relațiile** vor fi identificate cu OPERAŢII AVÂND REZULTATUL BOOLEAN: pentru un $m \in \mathbb{N}^*$ și un simbol de relație m–ară R, putem identifica pe R cu un simbol de operație m–ară dacă extindem domeniul lui $\mathcal{A}$ la $A \cup \mathcal{L}_2$, și definim **funcția parțială** $R^{\mathcal{A}} : (A \cup \mathcal{L}_2)^m \rightarrow\!\!\!\circ\; A \cup \mathcal{L}_2$ doar pe A^m, astfel: pentru orice $(a_1, \ldots, a_m) \in A^m$,

$$R^{\mathcal{A}}(a_1, \ldots, a_m) = \begin{cases} 1, & \text{dacă } (a_1, \ldots, a_m) \in R^{\mathcal{A}} \text{ ca relație,} \\ 0, & \text{altfel.} \end{cases}$$

Cu această identificare, **formulele atomice** devin (sunt identificate cu) **termeni**.

(În esență, **algoritmul de unificare** de mai jos funcționează astfel)

Pentru orice $n, k \in \mathbb{N}$, orice operații (incluzând aici relațiile) f și g de arități n, respectiv k și orice termeni (care pot avea operația dominantă de rezultat boolean, i.e. predicat, relație) $t_1, \ldots, t_n, u_1, \ldots, u_k$, termenii $f(t_1, \ldots, t_n)$ și $g(u_1, \ldots, u_k)$ unifică ddacă:

$$\begin{cases} n = k, \\ t_1 \text{ și } u_1 \text{ unifică,} \\ \vdots \\ t_n \text{ și } u_n \text{ unifică,} \end{cases}$$

și această recursie continuă până la unificări de termeni cu termeni care sunt:

- fie variabile, iar acestea unifică, cu orice termen care nu le conține,
- fie constante, iar acestea sunt operații fără argumente, așadar, conform regulii de mai sus, nu unifică decât cu ele însele.

Definiție (ce înseamnă a **unifica** mai mulți **termeni**)

Fie $k \in \mathbb{N} \setminus \{0, 1\}$ și $t_1, \ldots, t_k \in Term(\mathcal{L}_\tau)$. Spunem că termenii $t_1, \ldots, t_k$ *unifică* ddacă există o substituție $\sigma : Term(\mathcal{L}_\tau) \rightarrow Term(\mathcal{L}_\tau)$, numită *unificator pentru* $t_1, \ldots, t_k$, cu proprietatea că $\sigma(t_1) = \ldots = \sigma(t_k)$.

Fixăm un număr natural $k \geq 2$ și k termeni $t_1, \ldots, t_k \in Term(\mathcal{L}_\tau)$.
Cerința de a determina dacă termenii $t_1, \ldots, t_k$ unifică și, în caz afirmativ, a determina și un unificator pentru acești termeni este o *problemă de unificare*.
În cazul în care $t_1, \ldots, t_k$ unifică, următorul algoritm obține, într–un număr finit de pași, *un cel mai general unificator pentru* $t_1, \ldots, t_k$, adică un unificator $\mu : Term(\mathcal{L}_\tau) \rightarrow Term(\mathcal{L}_\tau)$ cu proprietatea că orice unificator $\sigma : Term(\mathcal{L}_\tau) \rightarrow Term(\mathcal{L}_\tau)$ pentru $t_1, \ldots, t_k$ este de forma $\sigma = \lambda \circ \mu$ pentru o substituție $\lambda : Term(\mathcal{L}_\tau) \rightarrow Term(\mathcal{L}_\tau)$.
În cazul în care $t_1, \ldots, t_k$ nu unifică, algoritmul următor se termină într–un număr finit de pași cu EŞEC, i.e. fără a găsi un unificator.

Vom reține două liste (mulțimi) de *ecuații* (*egalități*, *probleme de unificare între câte doi termeni*):

o *listă soluție* S și o *listă de rezolvat* R.

Inițial:

- $S = \emptyset$;
- $R = \{t_1 = t_2, t_2 = t_3, \ldots, t_{k-1} = t_k\}$.

Se aplică, pe rând, următorii pași, în orice ordine posibilă:

- SCOATERE (ELIMINARE, ŞTERGERE): orice ecuație de forma $t = t$ (i.e. între un termen și el însuși) este eliminată din lista R;
- DESCOMPUNERE: orice ecuație de forma $\varphi_i(u_1, \ldots, u_{n_i}) = \varphi_i(w_1, \ldots, w_{n_i})$, unde $i \in I$ astfel încât $n_i > 0$ și $u_1, \ldots, u_{n_i}, w_1, \ldots, w_{n_i} \in Term(\mathcal{L}_\tau)$ din lista R este înlocuită cu următoarele n_i ecuații: $u_1 = w_1, \ldots, u_{n_i} = w_{n_i}$;
- REZOLVARE: orice ecuație din R de forma $v = t$ sau $t = v$, unde $v \in Var$ și $t \in Term(\mathcal{L}_\tau)$, este scoasă din R și introdusă în lista soluție S sub forma $v = t$ (dacă și $t \in Var$, atunci nu contează cum orientăm ecuația: putem alege pe oricare dintre variabile ca membru stâng), apoi toate aparițiile lui v în termenii care apar în ecuațiile din lista de rezolvat R și din lista soluție S se înlocuiesc cu t, adică toți acești termeni u se înlocuiesc cu $\{v/t\}(u)$.

Algoritmul se ÎNCHEIE în oricare dintre cazurile următoare:

1. dacă, după execuția unui pas de SCOATERE sau REZOLVARE, obținem $R = \emptyset$, i.e. lista de rezolvat devine vidă, iar lista soluție curentă este $S = \{v_1 = u_1, \ldots, v_n = u_n\}$, atunci **un (cel mai general) unificator** pentru $t_1, \ldots, t_k$ este substituția $\{v_1/u_1, \ldots, v_n/u_n\}$: IEŞIRE CU SUCCES;
2. dacă, în cursul execuției pașilor de mai sus, apare în lista de rezolvat R:
   - fie o ecuație de forma $v = t$, cu $t \in Term(\mathcal{L}_\tau)$ și $v \in V(t)$ (i.e. cu variabila v apărând în t),
   - fie o ecuație de forma $\varphi_i(u_1, \ldots, u_{n_i}) = \varphi_j(w_1, \ldots, w_{n_j})$, cu $i, j \in I$, $n_i, n_j \in \mathbb{N}$ (nu neapărat nenule), $u_1, \ldots, u_{n_i}, w_1, \ldots, w_{n_j} \in Term(\mathcal{L}_\tau)$ și $i \neq j$, așadar cu simbolurile de operații dominante ale termenilor din cei doi membri diferite: $\varphi_i \neq \varphi_j$,

   atunci se IESE CU EŞEC: nu s–a găsit niciun unificator, așadar **termenii** $t_1, \ldots, t_k$ **nu unifică**.

Exercițiu (temă pentru seminar)

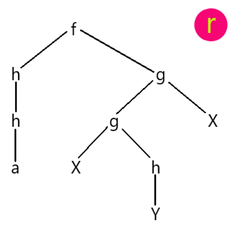
Să considerăm două simboluri de operații binare distincte f și g, unul de operație unară h, două simboluri de constante diferite a și b și patru variabile $V, X, Y, Z \in Var$, două câte două distincte.
Considerăm următorii termeni formați cu simbolurile de operații și variabilele de mai sus: $r, s, t, u, w \in Term(\mathcal{L}_\tau)$,

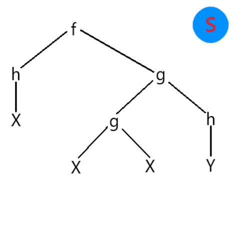
Exercițiu (continuare: termenii de unificat, dați prin expresiile lor și prin arborii asociați acestor expresii)

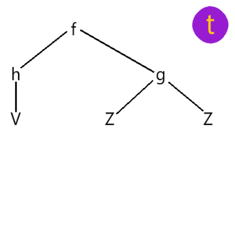
$r = f(h(h(a)), g(g(X, h(Y)), X))$,
$s = f(h(X), g(g(X, X), h(Y)))$,
$t = f(h(V), g(Z, Z))$,
$u = f(h(h(Z)), g(g(X, h(b)), h(Z)))$,
$w = f(h(b), g(g(V, Z), f(V, Z)))$.

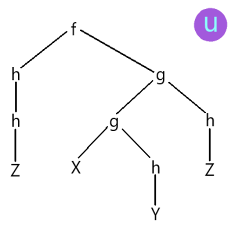
Să se unifice acești termeni doi câte doi, adică să se rezolve, pe rând, problemele de unificare:

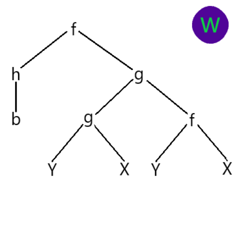
$r = s, r = t$,
$r = u, r = w$,
$s = t, s = u$,
$s = w, t = u$,
$t = w, u = w$.











Pentru perechile $\{p, q\}$ de termeni $p, q \in \{r, s, t, u, w\}$ care unifică, să se unifice și reuniunea tuturor acestor perechi.

Pentru **rezolvare**, a se vedea SEMINARUL.

Pentru a aplica rezoluția unui enunț, enunțul trebuie pus într–un anumit tip de FNC, numit **formă Skolem**, despre care se poate demonstra că există pentru orice enunț și satisfiabilă ddacă acel enunț e satisfiabil.
Fie ε un enunț. Folosind proprietățile de mai sus și, eventual, redenumind variabilele pentru a nu avea variabile cuantificate și universal, și existențial (mai multe detalii în SEMINAR), se determină o formă prenex pentru ε:

$$\varepsilon \;\;\vDash\!\!\!\!\Dash\;\; \mathcal{Q}_1 x_1 \ldots \mathcal{Q}_n x_n \varphi(x_1, \ldots, x_n),$$

unde $n \in \mathbb{N}^*$, $x_1, \ldots, x_n \in Var$, $\mathcal{Q}_1, \ldots, \mathcal{Q}_n \in \{\forall, \exists\}$ și $\varphi(x_1, \ldots, x_n)$ este o formulă fără cuantificatori. Apoi, folosind proprietățile 1–9 de mai sus, folosite în calculul propozițional pentru a pune enunțurile în FNC, se pune $\varphi(x_1, \ldots, x_n)$ într–o FNC ψ, astfel obținându–se o altă formă prenex pentru ε:

$$\varepsilon \;\;\vDash\!\!\!\!\Dash\;\; \mathcal{Q}_1 x_1 \ldots \mathcal{Q}_n x_n \psi(x_1, \ldots, x_n),$$

cu formula fără cuantificatori ψ în FNC.
Pentru fiecare cuantificator existențial $\mathcal{Q}_i = \exists$, cu $i \in \overline{1, n}$, se adaugă la signatura τ câte o operație "fictivă" h_i, numită *funcție Skolem*, de aritate $i - |\{j \in \overline{1, i} \mid \mathcal{Q}_j = \exists\}|$ (astfel că, în cazul în care $\mathcal{Q}_1 = \exists$, h_1 este o constantă), și se înlocuiește fiecare apariție a variabilei x_i în $\psi(x_1, \ldots, x_n)$ cu termenul h_i având $V(h_i) = \{x_1, \ldots, x_i\} \setminus \{x_j \mid j \in \overline{1, i}\}$, iar cuantificatorii existențiali sunt eliminați din forma prenex anterioară.

Astfel obținem următoarea FNC: $\chi = \forall x_1, \ldots \forall x_{i_1-1} \forall x_{i_1+1} \ldots \forall x_{i_2-1} \forall x_{i_2+1} \ldots \forall x_{i_k-1} \forall x_{i_k+1} \ldots \forall x_n \varphi(x_1, \ldots, x_{i_1-1}, h_{i_1}(x_1, \ldots, x_{i_1-1}), x_{i_1+1}, \ldots, x_{i_2-1}, h_{i_2}(x_1, \ldots, x_{i_1-1}, x_{i_1+1}, \ldots, x_{i_2-1}), x_{i_2+1}, \ldots, x_{i_k-1}, h_{i_k}(x_1, \ldots, x_{i_1-1}, x_{i_1+1}, \ldots, x_{i_2-1}, x_{i_2+1}, \ldots, x_{i_k-1}), x_{i_k+1}, \ldots, x_n)$.

Fie $p = n - k$ și (doar pentru a simplifica următoarea scriere) să redenumim variabilele $x_1, \ldots, x_{i_1-1}, x_{i_1+1}, \ldots, x_{i_2-1}, x_{i_2+1}, \ldots, x_{i_k-1}, x_{i_k+1}, \ldots, x_n$ în $y_1, \ldots, y_p$, respectiv. Dacă $C_1(y_1, \ldots, y_p), \ldots, C_r(y_1, \ldots, y_p)$ sunt clauzele formulei fără cuantificatori în FNC $\varphi(x_1, \ldots, x_{i_1-1}, h_{i_1}(x_1, \ldots, x_{i_1-1}), x_{i_1+1}, \ldots, x_{i_2-1}, h_{i_2}(x_1, \ldots, x_{i_1-1}, x_{i_1+1}, \ldots, x_{i_2-1}), x_{i_2+1}, \ldots, x_{i_k-1}, h_{i_k}(x_1, \ldots, x_{i_1-1}, x_{i_1+1}, \ldots, x_{i_2-1}, x_{i_2+1}, \ldots, x_{i_k-1}), x_{i_k+1}, \ldots, x_n)$, atunci:

$$\chi = \forall y_1 \ldots \forall y_p\, C_1(y_1, \ldots, y_p) \wedge \ldots \wedge C_r(y_1, \ldots, y_p) \;\; \models\!\!\!\!\!\;\dashv$$

$$(\forall y_1 \ldots \forall y_p\, C_1(y_1, \ldots, y_p)) \wedge \ldots \wedge (\forall y_1 \ldots \forall y_p\, C_r(y_1, \ldots, y_p)),$$

iar această din urmă formulă este satisfiabilă ddacă fiecare dintre formulele $\forall y_1 \ldots \forall y_p\, C_1(y_1, \ldots, y_p)$, ..., $\forall y_1 \ldots \forall y_p\, C_r(y_1, \ldots, y_p)$ este satisfiabilă.

Acum redenumim variabilele $\{y_1, \ldots, y_p\}$ în fiecare dintre clauzele $C_1, \ldots, C_r$ astfel încât mulțimile $V(C_1), \ldots, V(C_r)$ să devină două câte două disjuncte: $V(C_1) = \{z_{11}, \ldots, z_{1p}\}$, ..., $V(C_r) = \{z_{r1}, \ldots, z_{rp}\}$, și, folosind, ca și mai sus, distributivitatea cuantificatorului universal ($\forall$) față de conjuncție ($\wedge$), obținem:

$$\xi \models\!\!\!\!\!\;\dashv \gamma = \forall z_{11} \ldots \forall z_{1p} \ldots \forall z_{r1} \ldots \forall z_{rp}(C_1(z_{11}, \ldots, z_{1p}) \wedge \ldots \wedge C_r(z_{r1}, \ldots, z_{rp})).$$

FNC γ se numește *formă Skolem* pentru ε, și este satisfiabilă ddacă ε e satisfiabil. Nu putem spune că $\varepsilon \models\!\!\mid \gamma$, întrucât ε și γ nu sunt satisfăcute de aceleași algebre: la signatura τ trebuie să adăugăm simboluri de operații/constante corespunzătoare funcțiilor Skolem pentru a obține signatura algebrelor în care este evaluată valoarea de adevăr a enunțului γ.

Definiție (**unificare** între **literali**)

Prin *unificare* între **formule atomice** înțelegem unificarea acelor formule privite ca termeni cu operația dominantă de rezultat boolean asociată relației din acele formule atomice.

Dacă φ și ψ sunt formule atomice, vom spune că $\neg\varphi$ și $\neg\psi$ *unifică* ddacă φ și ψ unifică. Un *(cel mai general) unificator* pentru $\neg\varphi$ și $\neg\psi$ este un *(cel mai general) unificator* pentru φ și ψ, adică un (cel mai general) unificator pentru termenii obținuți din φ și ψ prin înlocuirea relațiilor din aceste formule atomice cu operațiile de rezultat boolean asociate acestor relații.

Definiție (**clauze triviale**)

Numim *clauză trivială* o clauză care conține doi literali φ și $\neg\psi$, unde φ și ψ sunt formule atomice care **unifică**.

(Rezoluția (regulă de deducție pentru logica clasică a predicatelor))

Pentru orice clauze C și D cu $V(C) \cap V(D) = \emptyset$, dacă φ și ψ sunt formule atomice astfel încât $\varphi, \neg\varphi$ nu unifică, cu niciun literal din C și $\psi, \neg\psi$ nu unifică, cu niciun literal din D, iar $\sigma : Term(\mathcal{L}_\tau) \to Term(\mathcal{L}_\tau)$ este o substituție cu proprietatea că $\sigma(\varphi) = \sigma(\psi)$, atunci:

$$\frac{C \cup \{\varphi\}, D \cup \{\neg\psi\}}{\sigma(C) \cup \sigma(D)}.$$

Definiție (derivări prin rezoluție)

Fie o mulțime finită de clauze $\{D_1, \ldots, D_k\}$.
Dacă $i, j \in \overline{1, k}$ a. î. $i \neq j$ și există două formule atomice φ și ψ astfel încât $\varphi, \neg\varphi$ nu unifică, cu niciun literal din D_i, $\psi, \neg\psi$ nu unifică, cu niciun literal din D_j, iar $\sigma : Term(\mathcal{L}_\tau) \to Term(\mathcal{L}_\tau)$ este o substituție cu proprietatea că $\sigma(\varphi) = \sigma(\psi)$, atunci mulțimea de clauze $R = \{\sigma(D_1), \ldots, \sigma(D_k)\}$ se numește *rezolvent* al mulțimii de clauze $Q = \{D_i \cup \{\varphi\}, D_j \cup \{\neg\psi\}\} \cup \{D_t \mid t \in \overline{1, k} \setminus \{i, j\}\}$.
Deducția $\dfrac{Q}{R}$ se numește *derivare prin rezoluție* a mulțimii Q. Vom numi orice succesiune de derivări prin rezoluție tot *derivare prin rezoluție*. O succesiune de derivări prin rezoluție care începe cu o FNC/mulțime de clauze μ și se termină cu o FNC/mulțime de clauze ν se numește *derivare prin rezoluție a lui ν din μ*.

INPUT: $m \in \mathbb{N}^*$ și o mulțime $S = \{C_1, \ldots, C_m\}$ de clauze netriviale având $V(C) \cap V(D) = \emptyset$ pentru orice clauze $C, D \in S$ cu $C \neq D$;

$i := 1;$ $S_1 := S;$

PASUL 1: considerăm o formulă atomică α_i care apare în măcar una dintre clauzele din S_i;

$T_i^0 := \{j \in \overline{1, m} \mid C_j \in S_i, (\exists\, \beta_i)(\neg\, \beta_{i,j} \in C_j$ și $\beta_{i,j}$ unifică, cu $\alpha_i)\};$

$T_i^1 := \{j \in \overline{1, m} \mid C_j \in S_i, (\exists\, \beta_i)(\beta_{i,j} \in C_j$ și $\beta_{i,j}$ unifică, cu $\alpha_i)\};$

$T_i := T_i^0 \cup T_i^1;$

fie σ_i un unificator pentru elementele mulțimii $\{\beta_{i,j} \mid j \in T_i\}$;

PASUL 2: dacă $T_i^0 \neq \emptyset$ și $T_i^1 \neq \emptyset$,

atunci $U_i := \{\sigma_i(C_j \setminus \{\neg\, \beta_{i,j}\}) \cup \sigma_i(C_k \setminus \{\beta_{i,k}\}) \mid j \in T_i^0, k \in T_i^1\};$

altfel $U_i := \emptyset;$

PASUL 3: $S_{i+1} := (S_i \setminus \{C_j \mid j \in T_i\}) \cup U_i;$

$S_{i+1} := S_{i+1} \setminus \{C \in S_{i+1} \mid (\exists\, \beta \in V)\,(\beta$ formulă atomică și $\beta, \neg\, \beta \in C)\}$ (eliminăm din S_{i+1} clauzele triviale);

PASUL 4: dacă $S_{i+1} = \emptyset$,

atunci OUTPUT: *S e satisfiabilă*;

altfel, dacă $\square \in S_{i+1}$,

atunci OUTPUT: *S nu e satisfiabilă*;

altfel $i := i + 1$ și mergi la PASUL 1.

În derivările prin rezoluție, inclusiv în unificările efectuate în cadrul acestor derivări, deci și în aplicarea algoritmului DP, funcțiile Skolem sunt tratate ca orice funcții.

Rezultatul algoritmului DP e influențat de FORMA SKOLEM obținută pentru enunțul despre care dorim să aflăm dacă e satisfiabil sau nu. A se vedea un exemplu mai jos.
Spre deosebire de cazul logicii propozițiilor, rezultatul aplicării algoritmului DP unui enunț cu sau fără cuantificatori existențiali nu este neapărat corect.

Revenind la o **formă Skolem** pentru un enunț ε: ce rol au funcțiile Skolem?

Dacă $x, y \in Var$, iar φ este o formulă cu $FV(\varphi) \subseteq \{x, y\}$, de ce avem: $\exists x \forall y \varphi \vDash \forall y \exists x \varphi$, dar nu și $\forall y \exists x \varphi \nvDash \exists x \forall y \varphi$? Pentru că, în enunțul $\forall y \exists x \varphi$, o valoare a a lui x care satisface acest enunț poate să depindă de valoarea b a lui y pentru care perechea (a, b) satisface acest enunț.

Exemplu

- $(\exists x \in \mathbb{R})(\forall y \in \mathbb{R})(x + y = 0)$: **fals**: nu există un astfel de x comun tuturor valorilor lui $y \in \mathbb{R}$;
- $(\forall y \in \mathbb{R})(\exists x \in \mathbb{R})(x + y = 0)$: **adevărat**;

Exemplu (continuare)

- $(\exists\, x \in \mathbb{N})\,(\forall\, y \in \mathbb{Z})\,(x \in \{y, -y\})$: **fals**: nu există un astfel de x comun tuturor valorilor lui $y \in \mathbb{Z}$;
- $(\forall\, y \in \mathbb{Z})\,(\exists\, x \in \mathbb{N})\,(x \in \{y, -y\})$: **adevărat**.

Fie $f : \mathbb{R} \to \mathbb{R}$, având cel puțin 3 valori distincte, i.e. cardinalul imaginii sale $f(\mathbb{R})$ mai mare sau egal cu 3.

- $(\exists\, x \in [0, +\infty))\,(\forall\, y \in \mathbb{R})\,(x \in \{f(y), -f(y)\})$: **fals**: nu există un astfel de x comun tuturor valorilor lui $y \in \mathbb{R}$, pentru că alegerea lui f ne asigură de faptul că există $y_0, y_1 \in \mathbb{R}$ cu proprietatea că $f(y_1) \notin \{f(y_0), -f(y_0)\}$, așadar perechea $(x_1 = f(y_1), y_0)$ nu satisface proprietatea $x_1 \in \{f(y_0), -f(y_0)\}$;
- $(\forall\, y \in \mathbb{R})\,(\exists\, x \in [0, +\infty))\,(x \in \{f(y), -f(y)\})$: **adevărat**.

O **formă Skolem** pentru enunțul $(\forall\, y)\,(\exists\, x)\,(x + y = 0)$ este: $(\forall\, y)\,(g(y) + y = 0)$, unde g este o funcție Skolem unară, iar acest enunț în formă Skolem, pentru domeniul valorilor $\mathbb{R}$ pentru ambele variabile, x și y, semnifică faptul că există o funcție $g : \mathbb{R} \to \mathbb{R}$ astfel încât $(\forall\, y \in \mathbb{R})\,(g(y) + y = 0)$.
O **formă Skolem** pentru enunțul $(\forall\, y)\,(\exists\, x)\,(x \in \{y, -y\})$ este: $(\forall\, y)\,(h(y) \in \{y, -y\})$, unde h este o funcție Skolem unară, iar acest enunț în formă Skolem, pentru domeniile valorilor $\mathbb{N}$, respectiv $\mathbb{Z}$ pentru variabilele x, respectiv y, semnifică faptul că există o funcție $h : \mathbb{Z} \to \mathbb{N}$ astfel încât $(\forall\, y \in \mathbb{Z})\,(h(y) \in \{y, -y\})$.

O **formă Skolem** pentru enunțul $(\forall\, y)\,(\exists\, x)\,(x \in \{f(y), -f(y)\})$ este: $(\forall\, y)\,(k(y) \in \{f(y), -f(y)\})$, unde k este o funcție Skolem unară, iar acest enunț în formă Skolem, pentru pentru domeniile valorilor $[0, +\infty)$, respectiv $\mathbb{R}$ pentru variabilele x, respectiv y, semnifică faptul că există o funcție $k : \mathbb{R} \to [0, +\infty)$ astfel încât $(\forall\, y \in \mathbb{R})\,(k(y) \in \{f(y), -f(y)\})$. Într–adevăr, acest enunț este satisfăcut de funcția $k : \mathbb{R} \to [0, +\infty)$, definită prin:

$$(\forall\, y \in \mathbb{R}) \left(k(y) = |f(y)| = \begin{cases} f(y), & \text{dacă } f(y) \geq 0, \\ -f(y), & \text{altfel.} \end{cases} \right).$$

Deci faptul că funcția k nu coincide nici cu f, nici cu $-$ nu influențează satisfiabilitatea enunțului anterior.

Exemplu

Să considerăm un limbaj de ordinul I conținând două simboluri de relații unare p și q. Fie x, y variabile distincte. Considerăm enunțurile:

- $\varphi = \forall\, x\, p(x) \wedge \exists\, y\, q(y)$;
- $\psi = \exists\, x\, [(p(x) \vee q(x)) \wedge \forall\, y\, ((p(x) \vee \neg\, q(y)) \wedge \neg\, p(x))]$.

$\varphi \vDash\!\!\dashv \forall\, x\, \exists\, y\, (p(x) \wedge q(y))$, dar avem și
$\varphi \vDash\!\!\dashv \exists\, y\, q(y) \wedge \forall\, x\, p(x) \vDash\!\!\dashv \exists\, y\, \forall\, x\, (q(y) \wedge p(x)) \vDash\!\!\dashv \exists\, y\, \forall\, x\, (p(x) \wedge q(y))$.

Așadar două forme Skolem diferite pentru enunțul φ sunt:

- $\forall x\,(p(x) \wedge q(c))$, unde c este o constantă Skolem; această formă Skolem corespunde mulțimii de clauze $\{\{p(x)\}, \{q(c)\}\}$, care nu are derivări prin rezoluție, deci algoritmul DP determină satisfiabilitatea acestei mulțimi de clauze;
- $\forall x\,(p(x) \wedge q(f(x)))$, unde f este o funcție Skolem unară; această formă Skolem corespunde mulțimii de clauze $\{\{p(x)\}, \{q(f(x))\}\}$, care nu are derivări prin rezoluție, deci algoritmul DP determină și satisfiabilitatea acestei mulțimi de clauze.

La fel ca mai sus:

- $\psi \mathrel{\models\!\!\!\dashv} \exists x\,\forall y\,[(p(x) \vee q(x)) \wedge (p(x) \vee \neg\, q(y)) \wedge \neg\, p(x)]$, având drept formă Skolem enunțul $\forall y\,[(p(c) \vee q(c)) \wedge (p(c) \vee \neg\, q(y)) \wedge \neg\, p(c)]$, unde c este o constantă Skolem, corespunzător mulțimii de clauze $\{\{p(c), q(c)\}, \{p(c), \neg\, q(y)\}, \{\neg\, p(c)\}\}$;
- $\psi \mathrel{\models\!\!\!\dashv} \forall y\,\exists x\,[(p(x) \vee q(x)) \wedge (p(x) \vee \neg\, q(y)) \wedge \neg\, p(x)]$, având drept formă Skolem enunțul $\forall y\,[(p(f(y)) \vee q(f(y))) \wedge (p(f(y)) \vee \neg\, q(y)) \wedge \neg\, p(f(y))]$, unde f este o funcție Skolem unară, corespunzător mulțimii de clauze $\{\{p(f(v)), q(f(v))\}, \{p(f(y)), \neg\, q(y)\}, \{\neg\, p(f(z))\}\}$.

Pentru prima dintre mulțimile de clauze de mai sus avem următoarea derivare prin rezoluție a clauzei vide, prin urmare algoritmul DP determină nesatisfiabilitatea acestei mulțimi de clauze:

$$
\begin{array}{c}
\{p(c), \cancel{q(c)}\}, \{p(c), \neg \cancel{q(y)}\} \text{ (cu unificatorul } \{y/c\}), \{\neg p(c)\} \\
\hline
\{\cancel{p(c)}\}, \{\neg \cancel{p(c)}\} \\
\hline
\square
\end{array}
$$

Pentru a doua dintre mulțimile de clauze de mai sus avem următoarele derivări prin rezoluție, dintre care niciuna nu ajunge la clauza vidă, prin urmare algoritmul DP determină satisfiabilitatea acestei mulțimi de clauze:

$$
\begin{array}{c}
\{p(f(v)), \cancel{q(f(v))}\}, \{p(f(y)), \neg \cancel{q(y)}\} \text{ (cu unificatorul } \{y/f(v)\}), \{\neg p(f(z))\} \\
\hline
\{\cancel{p(f(v))}, p(f(f(v)))\}, \{\neg \cancel{p(f(z))}\} \text{ (cu unificatorul } \{z/v\}) \\
\hline
\{p(f(f(v)))\}
\end{array}
$$

$$
\begin{array}{c}
\{p(f(v)), \cancel{q(f(v))}\}, \{p(f(y)), \neg \cancel{q(y)}\} \text{ (cu unificatorul } \{y/f(v)\}), \{\neg p(f(z))\} \\
\{p(f(v)), \cancel{p(f(f(v)))}\}, \{\neg \cancel{p(f(z))}\} \text{ (cu unificatorul } \{z/f(v)\}) \\
\hline
\{p(f(v))\}
\end{array}
$$

$$
\begin{array}{c}
\{\cancel{p(f(v))}, q(f(v))\}, \{p(f(y)), \neg q(y)\}, \{\neg \cancel{p(f(z))}\} \text{ (cu unificatorul } \{z/v\}) \\
\hline
\{\cancel{q(f(v))}\}, \{p(f(y)), \neg \cancel{q(y)}\} \text{ (cu unificatorul } \{y/f(v)\}) \\
\hline
\{p(f(f(v)))\}
\end{array}
$$

Algoritmul DP pentru logica predicatelor dă rezultatul corect, adică determină corect satisfiabilitatea, în anumite cazuri particulare, cum este cazul clauzelor definite, care stă la baza limbajului Prolog.