Logică Matematică și Computațională

Anul I, Semestrul I 2022/2023

Laurențiu Leuștean

Pagina web: http://cs.unibuc.ro/~lleustean/

PRELIMINARII

Operații cu mulțimi

Fie A, B, T mulțimi a.î. $A, B \subseteq T$.

$$\begin{aligned} A \cup B &= \{x \in T \mid x \in A \text{ sau } x \in B\} \\ A \cap B &= \{x \in T \mid x \in A \text{ și } x \in B\} \\ A \setminus B &= \{x \in T \mid x \in A \text{ și } x \notin B\} \\ C_T A &= T \setminus A = \{x \in T \mid x \notin A\} \end{aligned}$$

Notații: $\mathbb{N} = \{0, 1, 2, \ldots\}$ este mulțimea numerelor naturale; $\mathbb{N}^* = \mathbb{N} \setminus \{0\}$; $\mathbb{Z}$ este mulțimea numerelor întregi; $\mathbb{R}$ este mulțimea numerelor reale; $\mathbb{Q}$ este mulțimea numerelor raționale.

Mulțimea părților lui T se notează 2^T sau $\mathcal{P}(T)$. Așadar, $2^T = \mathcal{P}(T) = \{A \mid A \subseteq T\}$.

Operații cu mulțimi

Notăm cu (a, b) perechea ordonată formată din a și b (care sunt componentele lui (a, b)).

Observații: dacă $a \neq b$, atunci $(a, b) \neq (b, a)$; $(a, b) \neq \{a, b\}$; $(7, 7)$ este o pereche ordonată validă; două perechi ordonate (a, b) și (c, d) sunt egale ddacă $a = c$ și $b = d$.

Definiție

Produsul cartezian a două mulțimi A și B este definit astfel:

$$A \times B = \{(a, b) \mid a \in A \text{ și } b \in B\}$$

Exercițiu.

$$\begin{aligned} A \times (B \cup C) &= (A \times B) \cup (A \times C) \\ A \times (B \cap C) &= (A \times B) \cap (A \times C) \end{aligned}$$

Funcții

Fie A și B mulțimi și $f : A \to B$ o funcție.
Spunem că $f : A \to B$ este definită pe A cu valori în B, A se numește domeniul de definiție al funcției f și B se numește domeniul valorilor sau codomeniul lui f.

Fie $X \subseteq A$ și $Y \subseteq B$.

- $f(X) = \{f(x) \mid x \in X\}$ este imaginea directă a lui X prin f; $f(A)$ este imaginea lui f.
- $f^{-1}(Y) = \{x \in A \mid f(x) \in Y\}$ este imaginea inversă a lui Y prin f.
- Fie $f|_X : X \to B$, $f|_X(x) = f(x)$ pentru orice $x \in X$. Funcția $f|_X$ este restricția lui f la X.

Mulțimea funcțiilor de la A la B se notează $Fun(A, B)$ sau B^A.

Funcții

Fie $f : A \to B$ o funcție.

- f este injectivă dacă pentru orice $x_1, x_2 \in A$, $x_1 \neq x_2$ implică $f(x_1) \neq f(x_2)$ (sau, echivalent, $f(x_1) = f(x_2)$ implică $x_1 = x_2$).
- f este surjectivă dacă pentru orice $y \in B$ există $x \in A$ a.î. $f(x) = y$ (sau, echivalent, $f(A) = B$).
- f este bijectivă dacă f este injectivă și surjectivă.

Funcția identică a lui A: $1_A : A \to A$, $1_A(x) = x$.

Fie $f : A \to B$ și $g : B \to C$ două funcții. Compunerea lor $g \circ f$ este definită astfel:

$$g \circ f : A \to C, \quad (g \circ f)(x) = g(f(x)) \text{ pentru orice } x \in A.$$

Funcții

$f : A \to B$ este inversabilă dacă există $g : B \to A$ astfel încât $g \circ f = 1_A$ și $f \circ g = 1_B$.

f este bijectivă ddacă f este inversabilă.

Observație

(i) Pentru orice mulțime A, $Fun(\emptyset, A)$ are un singur element, funcția vidă.

(ii) Pentru orice mulțime nevidă A, $Fun(A, \emptyset) = \emptyset$.

Definiția 1.1

Fie A, T mulțimi a.î. $A \subseteq T$. Funcția caracteristică a lui A în raport cu T este definită astfel:

$$\chi_A : T \to \{0, 1\}, \quad \chi_A(x) = \begin{cases} 1, & \text{dacă } x \in A \\ 0, & \text{dacă } x \notin A \end{cases}$$

Echipotență

Definiția 1.2

Spunem că A este echipotentă cu B dacă există o bijecție $f : A \to B$. Notație: $A \sim B$.

Propoziția 1.3

Pentru orice mulțimi A, B, C, avem

(i) $A \sim A$;

(ii) Dacă $A \sim B$, atunci $B \sim A$.

(iii) Dacă $A \sim B$ și $B \sim C$, atunci $A \sim C$.

Dem.: Exercițiu.

Observație

Prin urmare, A este echipotentă cu B ddacă B este echipotentă cu A. De aceea, spunem de obicei că A și B sunt echipotente.

Echipotență

Următorul rezultat este fundamental.

Teorema 1.4 (Teorema Cantor-Schröder-Bernstein)

Fie A și B două mulțimi astfel încât există $f : A \to B$ și $g : B \to A$ funcții injective. Atunci $A \sim B$.

Definiția 1.5

O mulțime A se numește finită dacă $A = \emptyset$ sau dacă există $n \in \mathbb{N}^*$ a.î. A este echipotentă cu $\{1, \ldots, n\}$.
Numărul elementelor unei mulțimi finite A se notează $|A|$ și se mai numește și cardinalul lui A.

Definiția 1.6

O mulțime care nu este finită se numește infinită.

Mulțimi (cel mult) numărabile

Definiția 1.7

O mulțime A este numărabilă dacă este echipotentă cu $\mathbb{N}$.
O mulțime finită sau numărabilă se numește cel mult numărabilă.

Exemple de mulțimi numărabile: $\mathbb{N}$, $\mathbb{N}^*$, $\mathbb{Z}$, $\mathbb{N} \times \mathbb{N}$, $\mathbb{Q}$.

Teorema Cantor

$\mathbb{R}$, $2^{\mathbb{N}}$ nu sunt mulțimi numărabile.

Se poate demonstra că

Propoziția 1.8

$\mathbb{R}$ este echipotentă cu $2^{\mathbb{N}}$.

Mulțimi (cel mult) numărabile

Propoziția 1.9

(i) *Orice mulțime infinită are o submulțime numărabilă.*
(ii) *Orice submulțime a unei mulțimi numărabile este cel mult numărabilă.*
(iii) *O mulțime A este cel mult numărabilă ddacă există o funcție injectivă de la A la o mulțime numărabilă.*
(iv) *Produsul cartezian a două mulțimi cel mult numărabile este cel mult numărabil.*
(v) *Reuniunea a două mulțimi cel mult numărabile este cel mult numărabilă.*

Corolar 1.10

Fie A o mulțime numărabilă și B o mulțime nevidă cel mult numărabilă. Atunci $A \times B$ și $A \cup B$ sunt numărabile.

Cardinale

Numerele cardinale sau cardinalele sunt o generalizare a numerelor naturale, ele fiind folosite pentru a măsura dimensiunea unei mulțimi; au fost introduse de Cantor.

Pentru orice mulțime A, cardinalul lui A (sau numărul cardinal al lui A) este un obiect $|A|$ asociat lui A a.î. sunt satisfăcute următoarele:

- $|A|$ este unic determinat de A.
- pentru orice mulțimi A, B, avem că $|A| = |B|$ ddacă $A \sim B$.

Această definiție nu specifică natura obiectului $|A|$ asociat unei mulțimi A.
Prin urmare, este naturală întrebarea dacă există cardinale.

Cardinale

Un posibil răspuns este:

definim $|A|$ ca fiind clasa tuturor mulțimilor echipotente cu A.

Un alt răspuns este definiția lui von Neumann din teoria axiomatică a mulțimilor. Conform acestei definiții, pentru orice mulțime A, $|A|$ este tot o mulțime.

- Cardinalul unei mulțimi finite este numărul său de elemente. Cardinalele transfinite sunt cardinalele mulțimilor infinite.
- $|\mathbb{N}|$ se notează $\aleph_0$ (se citește alef zero).
- $|\mathbb{R}|$ se notează $\mathfrak{c}$ și se mai numește și puterea continuumului.
- O mulțime A este numărabilă ddacă $|A| = \aleph_0$.
- $|2^{\mathbb{N}}| \neq \aleph_0$.
- $|2^{\mathbb{N}}| = \mathfrak{c}$.

Familii de mulțimi

Fie I o mulțime nevidă.

Definiția 1.11

Fie A o mulțime. O familie de elemente din A indexată de I este o funcție $f : I \to A$. Notăm cu $(a_i)_{i\in I}$ familia $f : I \to A$, $f(i) = a_i$ pentru orice $i \in I$. Vom scrie și $(a_i)_i$ sau (a_i) atunci când I este dedusă din context.

Dacă fiecărui $i \in I$ îi este asociată o mulțime A_i, obținem o familie (indexată) de mulțimi $(A_i)_{i\in I}$.

Fie $(A_i)_{i\in I}$ o familie de submulțimi ale unei mulțimi T. Reuniunea și intersecția familiei $(A_i)_{i\in I}$ sunt definite astfel:

$$\bigcup_{i\in I} A_i = \{x \in T \mid \text{există } i \in I \text{ a.î. } x \in A_i\}$$

$$\bigcap_{i\in I} A_i = \{x \in T \mid x \in A_i \text{ pentru orice } i \in I\}$$

Familii de mulțimi

Fie I o mulțime nevidă și $(A_i)_{i\in I}$ o familie de mulțimi.

Definiția 1.12

Produsul cartezian al familiei $(A_i)_{i\in I}$ se definește astfel:

$$\begin{aligned}\prod_{i\in I} A_i &= \left\{ f : I \to \bigcup_{i\in I} A_i \mid f(i) \in A_i \text{ pentru orice } i \in I \right\} \\ &= \{(x_i)_{i\in I} \mid x_i \in A_i \text{ pentru orice } i \in I\}.\end{aligned}$$

Fie n număr natural, $n \geq 1$, $I = \{1, \ldots, n\}$ și $A_1, \ldots, A_n \subseteq T$.

- $(x_i)_{i\in I} = (x_1, \ldots, x_n)$, un n-tuplu (ordonat)
- $\bigcup_{i\in I} A_i = \bigcup_{i=1}^{n} A_i$ și $\bigcap_{i\in I} A_i = \bigcap_{i=1}^{n} A_i$
- $\prod_{i\in I} A_i = \prod_{i=1}^{n} A_i = A_1 \times \cdots \times A_n$ și $A^n = \underbrace{A \times \cdots \times A}_{n}$

Familii de mulțimi

Propoziția 1.13

(i) Reuniunea unei familii cel mult numărabile de mulțimi cel mult numărabile este mulțime cel mult numărabilă.

(ii) Reuniunea unui număr finit (≥ 2) de mulțimi numărabile este numărabilă.

(iii) Produsul cartezian al unui număr finit (≥ 2) de mulțimi numărabile este numărabil.

Relații

Definiția 1.14

O *relație n-ară* *între* $A_1, \ldots, A_n$ *este o submulțime a produsului cartezian* $\prod_{i=1}^{n} A_i$.
O relație n-ară pe A *este o submulțime a lui* A^n. *Dacă* R *este relație n-ară, spunem că n este* *aritatea* *lui* R.

Definiția 1.15

O *relație binară* *între* A *și* B *este o submulțime a produsului cartezian* $A \times B$.
O relație binară pe A *este o submulțime a lui* $A^2 = A \times A$.

Exemple

- relația de divizibilitate pe $\mathbb{N}$:
  $| = \{(k, n) \in \mathbb{N}^2 \mid \text{ există } m \in \mathbb{N} \text{ a.î. } mk = n\}$
- relația de ordine strictă pe $\mathbb{N}$:
  $< = \{(k, n) \in \mathbb{N}^2 \mid \text{ există } m \in \mathbb{N} \text{ a.î. } m \neq 0 \text{ și } m + k = n\}$

Relații binare

Fie A o mulțime nevidă și R o relație binară pe A.
Notație: Scriem xRy în loc de $(x, y) \in R$ și $\neg(xRy)$ în loc de $(x, y) \notin R$.

Definiția 1.16

- *R este* *reflexivă* *dacă* xRx *pentru orice* $x \in A$.
- *R este* *ireflexivă* *dacă* $\neg(xRx)$ *pentru orice* $x \in A$.
- *R este* *simetrică* *dacă pentru orice* $x, y \in A$, xRy *implică* yRx.
- *R este* *antisimetrică* *dacă pentru orice* $x, y \in A$,
  xRy *și* yRx *implică* $x = y$.
- *R este* *tranzitivă* *dacă pentru orice* $x, y, z \in A$,
  xRy *și* yRz *implică* xRz.
- *R este* *totală* *dacă pentru orice* $x, y \in A$, xRy *sau* yRx.

Relații binare

Fie A o mulțime nevidă și R o relație binară pe A.

Definiția 1.17

R este *relație de echivalență* *dacă este reflexivă, simetrică și tranzitivă.*

Definiția 1.18

R este relație de

- *ordine parțială* *dacă este reflexivă, antisimetrică și tranzitivă.*
- *ordine strictă* *dacă este ireflexivă și tranzitivă.*
- *ordine totală* *dacă este antisimetrică, tranzitivă și totală.*

Notații: Vom nota relațiile de ordine parțială și totală cu $\leq$, iar relațiile de ordine strictă cu $<$.

LOGICA PROPOZIȚIONALĂ

Limbajul logicii propoziționale este bazat pe propoziții sau enunțuri declarative, despre care se poate argumenta în principiu că sunt adevărate sau false.

Propoziții declarative

- Suma numerelor 2 și 4 este 6.
- Mihai Eminescu a fost un scriitor român.
- Maria a reacționat violent la acuzațiile lui Ion.
- Orice număr natural par > 2 este suma a două numere prime. (Conjectura lui Goldbach).
- Andrei este deștept.
- Marțienilor le place pizza.

Propoziții care nu sunt declarative

- Poți să îmi dai, te rog, pâinea?
- Pleacă!

Considerăm anumite propoziții ca find atomice și le notăm $p, q, r, \ldots$ sau $p_1, p_2, p_3, \ldots$.

Exemple: p=Numărul 2 este par. q=Mâine plouă. r=Sunt obosit.

Pornind de la propozițiile atomice, putem crea propoziții complexe (notate φ, ψ, χ, $\cdots$) folosind conectorii logici $\neg$ (negația), $\rightarrow$ (implicația), $\vee$ (disjuncția), $\wedge$ (conjuncția), $\leftrightarrow$ (echivalența).

Exemple:

$\neg p$	$=$	Numărul 2 nu este par.
$p \vee q$	$=$	Numărul 2 este par sau mâine plouă.
$p \wedge q$	$=$	Numărul 2 este par și mâine plouă.
$p \rightarrow q$	$=$	Dacă numărul 2 este par, atunci mâine plouă.
$p \leftrightarrow q$	$=$	Numărul 2 este par dacă și numai dacă mâine plouă.

Putem aplica repetat conectorii pentru a obține propoziții și mai complexe. Pentru a elimina ambiguitățile, folosim parantezele $(,)$.

Exemplu: $\varphi = (p \wedge q) \rightarrow ((\neg r) \vee q)$

Exemplu:

Fie propoziția:

φ=*Azi este vineri, deci avem curs de logică.*

Considerăm propozițiile atomice

p=*Azi este vineri.* q=*Avem curs de logică.*

Atunci $\varphi = p \rightarrow q$. Cine este $\neg\varphi$?

$\neg\varphi = p \wedge (\neg q)$=*Azi este vineri și nu avem curs de logică.*

Exemplu:

Fie propoziția:

φ=*Dacă trenul întârzie și nu sunt taxiuri la gară, atunci Ion întârzie la întâlnire.*

Considerăm propozițiile atomice

p	$=$	*Trenul întârzie.*
q	$=$	*Sunt taxiuri la gară.*
r	$=$	*Ion întârzie la întâlnire.*

Atunci $\varphi = (p \wedge (\neg q)) \rightarrow r$.

Presupunem că φ, p sunt adevărate și r este falsă (deci $\neg r$ este adevărată). Ce putem spune despre q? q este adevărată.

Logica propozițională LP - Limbajul

Definiția 2.1

Limbajul logicii propoziționale LP este format din:

- *o mulțime numărabilă $V = \{v_n \mid n \in \mathbb{N}\}$ de variabile;*
- *conectori logici: $\neg$ (se citește non), $\rightarrow$ (se citește implică)*
- *paranteze: (,).*

- Mulțimea *Sim* a simbolurilor lui *LP* este

$$Sim := V \cup \{\neg, \rightarrow, (,)\}.$$

- Notăm variabilele cu $v, u, w, v_0, v_1, v_2, \ldots$

Logica propozițională LP - Limbajul

Definiția 2.2

Mulțimea Expr a expresiilor lui LP este mulțimea tuturor șirurilor finite de simboluri ale lui LP.

- Expresia vidă se notează λ.
- Lungimea unei expresii θ este numărul simbolurilor din θ. Sim^n este mulțimea șirurilor de simboluri ale lui *LP* de lungime n.
- Prin convenție, $Sim^0 = \{\lambda\}$. Atunci $Expr = \bigcup_{n\in\mathbb{N}} Sim^n$.

Exemple:

$((((v_7,\ v_1\neg \rightarrow (v_2),\ \neg v_1 v_2,\ ((v_1 \rightarrow v_2) \rightarrow (\neg v_1)),\ (\neg(v_1 \rightarrow v_2))$.

Logica propozițională LP - Limbajul

Operația de bază pentru expresii este concatenarea: dacă $\varphi = \varphi_0 \ldots \varphi_{k-1}$ și $\psi = \psi_0 \ldots \psi_{l-1}$ sunt expresii, atunci concatenarea lor, notată $\varphi\psi$, este expresia $\varphi_0 \ldots \varphi_{k-1}\psi_0 \ldots \psi_{l-1}$.

Definiția 2.3

Fie $\theta = \theta_0\theta_1 \ldots \theta_{k-1}$ o expresie a lui LP, unde $\theta_i \in Sim$ pentru orice $i \in \{0, 1, \ldots, k-1\}$.

- *Dacă $0 \leq i \leq j \leq k-1$, atunci expresia $\theta_i \ldots \theta_j$ se numește (i,j)-subexpresia lui θ;*
- *Spunem că o expresie ψ apare în θ dacă există $0 \leq i \leq j \leq k-1$ a.î. ψ este (i,j)-subexpresia lui θ.*

Formule

Definiția formulelor este un exemplu de definiție inductivă.

Definiția 2.4

Formulele lui LP sunt expresiile lui LP definite astfel:

(F0) Orice variabilă propozițională este formulă.

(F1) Dacă φ este formulă, atunci $(\neg\varphi)$ este formulă.

(F2) Daca φ și ψ sunt formule, atunci $(\varphi \rightarrow \psi)$ este formulă.

(F3) Numai expresiile obținute aplicând regulile (F0), (F1), (F2) sunt formule.

Notații: Mulțimea formulelor se notează *Form*. Notăm formulele cu $\varphi, \psi, \chi, \ldots$.

- Orice formulă se obține aplicând regulile (F0), (F1), (F2) de un număr finit de ori.
- $Form \subseteq Expr$. Formulele sunt expresiile "bine formate".

Formule

Exemple:

- $v_1\neg \to (v_2)$, $\neg v_1 v_2$ nu sunt formule.
- $((v_1 \to v_2) \to (\neg v_1))$, $(\neg(v_1 \to v_2))$ sunt formule.

Citire unică (Unique readability)

Dacă φ este o formulă, atunci exact una din următoarele alternative are loc:

- $\varphi = v$, unde $v \in V$;
- $\varphi = (\neg\psi)$, unde ψ este formulă;
- $\varphi = (\psi \to \chi)$, unde ψ, χ sunt formule.

Mai mult, scrierea lui φ sub una din aceste forme este unică.

Propoziția 2.5

Mulțimea Form a formulelor lui LP este numărabilă.

Dem.: Exercițiu.

Principiul inducției pe formule

Propoziția 2.6 (Principiul inducției pe formule)

*Fie **P** o proprietate. Presupunem că:*

(0) *Orice variabilă are proprietatea **P**.*

(1) *Pentru orice formulă φ, dacă φ are proprietatea **P**, atunci și $(\neg\varphi)$ are proprietatea **P**.*

(2) *Pentru orice formule φ, ψ, dacă φ și ψ au proprietatea **P**, atunci $(\varphi \to \psi)$ are proprietatea **P**.*

*Atunci orice formulă φ are proprietatea **P**.*

Dem.: Pentru orice formulă φ, notăm cu $c(\varphi)$ numărul conectorilor logici care apar în φ. Pentru orice $n \in \mathbb{N}$ definim proprietatea $Q(n)$ astfel:

$Q(n)$ e adevărată ddacă orice formulă φ cu $c(\varphi) \leq n$ are proprietatea **P**.

Demonstrăm prin inducție că $Q(n)$ este adevărată pentru orice $n \in \mathbb{N}$.

Principiul inducției pe formule

Pasul inițial. $Q(0)$ este adevărată, deoarece pentru orice formulă φ, $c(\varphi) \leq 0 \iff c(\varphi) = 0 \iff \varphi = v$, cu $v \in V$ și, conform ipotezei (0), v are proprietatea **P**.

Ipoteza de inducție. Fie $n \in \mathbb{N}$. Presupunem că $Q(n)$ este adevărată.

Pasul de inducție. Demonstrăm că $Q(n+1)$ este adevărată. Fie φ o formulă cu $c(\varphi) \leq n+1$. Avem trei cazuri:

- $\varphi = v \in V$. Atunci φ are proprietatea **P**, conform (0).
- $\varphi = (\neg\psi)$, unde ψ este formulă. Atunci $c(\psi) = c(\varphi) - 1 \leq n$, deci, conform ipotezei de inducție, ψ are proprietatea **P**. Aplicînd ipoteza (1), rezultă că φ are proprietatea **P**.
- $\varphi = (\psi \to \chi)$, unde ψ, χ sunt formule. Atunci $c(\psi), c(\chi) \leq c(\varphi) - 1 \leq n$, deci, conform ipotezei de inducție, ψ și χ au proprietatea **P**. Rezultă din (2) că φ are proprietatea **P**.

Așadar, $Q(n)$ este adevărată pentru orice $n \in \mathbb{N}$. Deoarece pentru orice formulă φ există $N \in \mathbb{N}$ a.î. $c(\varphi) \leq N$, rezultă că orice formulă φ are proprietatea **P**. □

Principiul inducției pe formule

Propoziția 2.7 (Principiul inducției pe formule - variantă alternativă)

Fie Γ o mulțime de formule care are următoarele proprietăți:

- $V \subseteq \Gamma$;
- *Γ este închisă la $\neg$, adică $\varphi \in \Gamma$ implică $(\neg\varphi) \in \Gamma$;*
- *Γ este închisă la $\to$, adică $\varphi, \psi \in \Gamma$ implică $(\varphi \to \psi) \in \Gamma$.*

Atunci $\Gamma = Form$.

Dem.: Definim următoarea proprietate **P**: pentru orice formulă φ,

φ are proprietatea **P** ddacă $\varphi \in \Gamma$.

Conform definiției lui Γ, rezultă că sunt satisfăcute ipotezele (0), (1), (2) din Principiul inducției pe formule (Propoziția 2.6), deci îl putem aplica pentru a obține că orice formulă are proprietatea **P**, deci orice formulă φ este în Γ. Așadar, $\Gamma = Form$. □

Subformule

Definiția 2.8

Fie φ o formulă a lui LP. O subformulă a lui φ este orice formulă ψ care apare în φ.
Notație: Mulțimea subformulelor lui φ se notează $SubForm(\varphi)$.

Exemplu:

Fie $\varphi = ((v_1 \to v_2) \to (\neg v_1))$. Atunci

$$SubForm(\varphi) = \{v_1, v_2, (v_1 \to v_2), (\neg v_1), \varphi\}.$$

Formule

Conectorii derivați $\vee$ (se citește sau), $\wedge$ (se citește și), $\leftrightarrow$ (se citește dacă și numai dacă) sunt introduși prin abrevierile:

$$\begin{aligned} (\varphi \vee \psi) &:= ((\neg\varphi) \to \psi) \\ (\varphi \wedge \psi) &:= (\neg(\varphi \to (\neg\psi))) \\ (\varphi \leftrightarrow \psi) &:= ((\varphi \to \psi) \wedge (\psi \to \varphi)). \end{aligned}$$

Convenții

- În practică, renunțăm la parantezele exterioare, le punem numai atunci când sunt necesare. Astfel, scriem $\neg\varphi, \varphi \to \psi$, dar scriem $(\varphi \to \psi) \to \chi$.
- Pentru a mai reduce din folosirea parantezelor, presupunem că
  - $\neg$ are precedența mai mare decât ceilalți conectori;
  - $\wedge, \vee$ au precedență mai mare decât $\to, \leftrightarrow$.

Prin urmare, formula $(((\varphi \to (\psi \vee \chi)) \wedge ((\neg\psi) \leftrightarrow (\psi \vee \chi)))$ va fi scrisă $(\varphi \to \psi \vee \chi) \wedge (\neg\psi \leftrightarrow \psi \vee \chi)$.

Principiul recursiei pe formule

Propoziția 2.9 (Principiul recursiei pe formule)

Fie A o mulțime și funcțiile

$$G_0 : V \to A, \quad G_\neg : A \to A, \quad G_\to : A \times A \to A.$$

Atunci există o unică funcție

$$F : Form \to A$$

care satisface următoarele proprietăți:

(R0) $F(v) = G_0(v)$ *pentru orice variabilă* $v \in V$.

(R1) $F(\neg\varphi) = G_\neg(F(\varphi))$ *pentru orice formulă* φ.

(R2) $F(\varphi \to \psi) = G_\to(F(\varphi), F(\psi))$ *pentru orice formule* φ, ψ.

Principiul recursiei pe formule

Principiul recursiei pe formule se folosește pentru a da definiții recursive ale diverselor funcții asociate formulelor.

Exemplu:

Fie $c : Form \to \mathbb{N}$ definită astfel: pentru orice formulă φ,

$c(\varphi)$ este numărul conectorilor logici care apar în φ.

O definiție recursivă a lui c este următoarea:

$$\begin{aligned} c(v) &= 0 \quad \text{pentru orice variabilă } v \\ c(\neg\varphi) &= c(\varphi) + 1 \quad \text{pentru orice formulă } \varphi \\ c(\varphi \to \psi) &= c(\varphi) + c(\psi) + 1 \quad \text{pentru orice formule } \varphi, \psi. \end{aligned}$$

În acest caz, $A = \mathbb{N}$, $G_0 : V \to A$, $G_0(v) = 0$,

$$G_\neg : \mathbb{N} \to \mathbb{N}, \quad G_\neg(n) = n + 1,$$
$$G_\to : \mathbb{N} \times \mathbb{N} \to \mathbb{N}, \quad G_\to(m, n) = m + n + 1.$$

Principiul recursiei pe formule

Notație:

Pentru orice formulă φ, notăm cu $Var(\varphi)$ mulțimea variabilelor care apar în φ.

Observație

Mulțimea $Var(\varphi)$ poate fi definită și recursiv.

Dem.: Exercițiu.

SEMANTICA LP

Tabele de adevăr

Valori de adevăr

Folosim următoarele notații pentru cele două valori de adevăr: 1 pentru adevărat și 0 pentru fals. Prin urmare, mulțimea valorilor de adevăr este $\{0,1\}$.

Definim următoarele operații pe $\{0,1\}$ folosind tabelele de adevăr.

$\neg : \{0,1\} \to \{0,1\}$,

p	$\neg p$
0	1
1	0

Se observă că $\neg p = 1 \iff p = 0$.

$\to : \{0,1\} \times \{0,1\} \to \{0,1\}$,

p	q	$p \to q$
0	0	1
0	1	1
1	0	0
1	1	1

Se observă că $p \to q = 1 \iff p \leq q$.

Tabele de adevăr

Operațiile $\vee : \{0,1\} \times \{0,1\} \to \{0,1\}$, $\wedge : \{0,1\} \times \{0,1\} \to \{0,1\}$ și $\leftrightarrow : \{0,1\} \times \{0,1\} \to \{0,1\}$ se definesc astfel:

p	q	$p \vee q$
0	0	0
0	1	1
1	0	1
1	1	1

p	q	$p \wedge q$
0	0	0
0	1	0
1	0	0
1	1	1

p	q	$p \leftrightarrow q$
0	0	1
0	1	0
1	0	0
1	1	1

Observație

Pentru orice $p, q \in \{0,1\}$, $p \vee q = \neg p \to q$, $p \wedge q = \neg(p \to \neg q)$ și $p \leftrightarrow q = (p \to q) \wedge (q \to p)$.

Dem.: Exercițiu.

Evaluări

Definiția 2.10

O evaluare (sau interpretare) este o funcție $e : V \to \{0,1\}$.

Teorema 2.11

Pentru orice evaluare $e : V \to \{0,1\}$ există o unică funcție

$$e^+ : Form \to \{0,1\}$$

care verifică următoarele proprietăți:

- $e^+(v) = e(v)$ *pentru orice* $v \in V$.
- $e^+(\neg\varphi) = \neg e^+(\varphi)$ *pentru orice* $\varphi \in Form$,
- $e^+(\varphi \to \psi) = e^+(\varphi) \to e^+(\psi)$ *pentru orice* $\varphi, \psi \in Form$.

Dem.: Aplicăm Principiul recursiei pe formule (Propoziția 2.9) cu $A = \{0,1\}$, $G_0 = e$, $G_\neg : \{0,1\} \to \{0,1\}$, $G_\neg(p) = \neg p$ și $G_\to : \{0,1\} \times \{0,1\} \to \{0,1\}$, $G_\to(p,q) = p \to q$. □

Evaluare (Interpretare)

Propoziția 2.12

Dacă $e : V \to \{0,1\}$ este o evaluare, atunci pentru orice formule φ, ψ,

$$e^+(\varphi \vee \psi) = e^+(\varphi) \vee e^+(\psi),$$
$$e^+(\varphi \wedge \psi) = e^+(\varphi) \wedge e^+(\psi),$$
$$e^+(\varphi \leftrightarrow \psi) = e^+(\varphi) \leftrightarrow e^+(\psi).$$

Dem.: Exercițiu.

Evaluare (Interpretare)

Propoziția 2.13

Pentru orice formulă φ și orice evaluări $e_1, e_2 : V \to \{0,1\}$,

(*) $\quad e_1(v) = e_2(v)$ *pentru orice* $v \in Var(\varphi) \implies e_1^+(\varphi) = e_2^+(\varphi)$.

Dem.: Definim următoarea proprietate $\boldsymbol{P}$: pentru orice formulă φ,

φ are proprietatea $\boldsymbol{P}$ ddacă pentru orice evaluări $e_1, e_2 : V \to \{0,1\}$, φ satisface (*).

Demonstrăm că orice formulă φ are proprietatea $\boldsymbol{P}$ folosind Principiul inducției pe formule. Avem următoarele cazuri:

- $\varphi = v$. Atunci $e_1^+(v) = e_1(v) = e_2(v) = e_2^+(v)$.

Evaluare (Interpretare)

Propoziția 2.13

Pentru orice formulă φ și orice evaluări $e_1, e_2 : V \to \{0,1\}$,

(*) $\quad e_1(v) = e_2(v)$ pentru orice $v \in Var(\varphi) \implies e_1^+(\varphi) = e_2^+(\varphi)$.

Dem.: (continuare)

- $\varphi = \neg\psi$ și ψ satisface $\boldsymbol{P}$. Fie $e_1, e_2 : V \to \{0,1\}$ a.î. $e_1(v) = e_2(v)$ pentru orice $v \in Var(\varphi)$. Deoarece $Var(\varphi) = Var(\psi)$, rezultă că $e_1(v) = e_2(v)$ pentru orice $v \in Var(\psi)$. Așadar, aplicând $\boldsymbol{P}$ pentru ψ, obținem că $e_1^+(\psi) = e_2^+(\psi)$. Rezultă că

$$e_1^+(\varphi) = \neg e_1^+(\psi) = \neg e_2^+(\psi) = e_2^+(\varphi),$$

deci φ satisface $\boldsymbol{P}$.

Evaluare (Interpretare)

Propoziția 2.13

Pentru orice formulă φ și orice evaluări $e_1, e_2 : V \to \{0,1\}$,

(*) $\quad e_1(v) = e_2(v)$ pentru orice $v \in Var(\varphi) \implies e_1^+(\varphi) = e_2^+(\varphi)$.

Dem.: (continuare)

- $\varphi = \psi \to \chi$ și ψ, χ satisfac $\boldsymbol{P}$. Fie $e_1, e_2 : V \to \{0,1\}$ a.î. $e_1(v) = e_2(v)$ pentru orice $v \in Var(\varphi)$. Deoarece $Var(\psi) \subseteq Var(\varphi)$ și $Var(\chi) \subseteq Var(\varphi)$, rezultă că $e_1(v) = e_2(v)$ pentru orice $v \in Var(\psi)$ și pentru orice $v \in Var(\chi)$. Așadar, aplicând $\boldsymbol{P}$ pentru ψ și χ, obținem că $e_1^+(\psi) = e_2^+(\psi)$ și $e_1^+(\chi) = e_2^+(\chi)$. Rezultă că

$$e_1^+(\varphi) = e_1^+(\psi) \rightarrow e_1^+(\chi) = e_2^+(\psi) \rightarrow e_2^+(\chi) = e_2^+(\varphi),$$

deci φ satisface $\boldsymbol{P}$. $\square$

Modele. Satisfiabilitate. Tautologii

Fie φ o formulă.

Definiția 2.14

- *O evaluare $e : V \to \{0,1\}$ este model al lui φ dacă $e^+(\varphi) = 1$. Notație: $e \vDash \varphi$.*
- *φ este satisfiabilă dacă admite un model.*
- *Dacă φ nu este satisfiabilă, spunem și că φ este nesatisfiabilă sau contradictorie.*
- *φ este tautologie dacă orice evaluare este model al lui φ. Notație: $\vDash \varphi$.*

Notație: Mulțimea tuturor modelelor lui φ se notează $Mod(\varphi)$.

Propoziția 2.15

(i) φ este tautologie ddacă $\neg\varphi$ este nesatisfiabilă.

(ii) φ este nesatisfiabilă ddacă $\neg\varphi$ este tautologie.

Dem.: Exercițiu.

Metoda tabelului

Fie φ o formulă arbitrară și $Var(\varphi) = \{x_1, x_2, \ldots, x_k\}$. Pentru orice evaluare $e : V \to \{0,1\}$, $e^+(\varphi)$ depinde doar de $e(x_1), \ldots, e(x_k)$, conform Propoziției 2.13.
Așadar, $e^+(\varphi)$ depinde doar de restricția lui e la $\{x_1, x_2, \ldots, x_k\}$:

$$e' : \{x_1, \ldots, x_k\} \to \{0,1\}, \quad e'(x_i) = e(x_i).$$

Sunt 2^k astfel de funcții posibile $e'_1, e'_2 \ldots, e'_{2^k}$. Asociem fiecăreia o linie într-un tabel:

x_1	x_2	$\ldots$	x_k	$\ldots$ subformule ale lui φ $\ldots$	φ
$e'_1(x_1)$	$e'_1(x_2)$	$\ldots$	$e'_1(x_k)$	$\ldots$	${e'_1}^+(\varphi)$
$e'_2(x_1)$	$e'_2(x_2)$	$\ldots$	$e'_2(x_k)$	$\ldots$	${e'_2}^+(\varphi)$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
$e'_{2^k}(x_1)$	$e'_{2^k}(x_2)$	$\ldots$	$e'_{2^k}(x_k)$	$\ldots$	${e'_{2^k}}^+(\varphi)$

Pentru orice i, ${e'_i}^+(\varphi)$ se definește similar cu Teorema 2.11.

φ este tautologie ddacă ${e'_i}^+(\varphi) = 1$ pentru orice $i \in \{1, \ldots, 2^k\}$.

Metoda tabelului

Exemplu:

Fie

$$\varphi = v_1 \to (v_2 \to (v_1 \wedge v_2)).$$

Vrem să demonstrăm că $\vDash \varphi$.

$$Var(\varphi) = \{v_1, v_2\}.$$

v_1	v_2	$v_1 \wedge v_2$	$v_2 \to (v_1 \wedge v_2)$	φ
0	0	0	1	1
0	1	0	0	1
1	0	0	1	1
1	1	1	1	1

Tautologii

Definiția 2.16

Fie φ, ψ două formule. Spunem că

- *φ este consecință semantică a lui ψ dacă $Mod(\psi) \subseteq Mod(\varphi)$. Notație: $\psi \vDash \varphi$.*
- *φ și ψ sunt (logic) echivalente dacă $Mod(\psi) = Mod(\varphi)$. Notație: $\varphi \sim \psi$.*

Observație

Relația $\sim$ este o relație de echivalență pe mulțimea *Form* a formulelor lui *LP*.

Propoziția 2.17

Fie φ, ψ formule. Atunci

(i) $\psi \vDash \varphi$ ddacă $\vDash \psi \rightarrow \varphi$.

(ii) $\psi \sim \varphi$ ddacă ($\psi \vDash \varphi$ și $\varphi \vDash \psi$) ddacă $\vDash \psi \leftrightarrow \varphi$.

Dem.: Exercițiu.

Tautologii, consecințe semantice și echivalențe

Propoziția 2.18

Pentru orice formule φ, ψ, χ,

terțul exclus	$\vDash \varphi \vee \neg\varphi$	(1)
modus ponens	$\varphi \wedge (\varphi \rightarrow \psi) \vDash \psi$	(2)
afirmarea concluziei	$\psi \vDash \varphi \rightarrow \psi$	(3)
contradicția	$\vDash \neg(\varphi \wedge \neg\varphi)$	(4)
dubla negație	$\varphi \sim \neg\neg\varphi$	(5)
contrapoziția	$\varphi \rightarrow \psi \sim \neg\psi \rightarrow \neg\varphi$	(6)
negarea premizei	$\neg\varphi \vDash \varphi \rightarrow \psi$	(7)
modus tollens	$\neg\psi \wedge (\varphi \rightarrow \psi) \vDash \neg\varphi$	(8)
tranzitivitatea implicației	$(\varphi \rightarrow \psi) \wedge (\psi \rightarrow \chi) \vDash \varphi \rightarrow \chi$	(9)

Tautologii, consecințe semantice și echivalențe

legile lui de Morgan	$\varphi \vee \psi \sim \neg(\neg\varphi \wedge \neg\psi)$	(10)
	$\varphi \wedge \psi \sim \neg(\neg\varphi \vee \neg\psi)$	(11)
exportarea și importarea	$\varphi \rightarrow (\psi \rightarrow \chi) \sim \varphi \wedge \psi \rightarrow \chi$	(12)
idempotența	$\varphi \sim \varphi \wedge \varphi \sim \varphi \vee \varphi$	(13)
slăbirea	$\vDash \varphi \wedge \psi \rightarrow \varphi \qquad \vDash \varphi \rightarrow \varphi \vee \psi$	(14)
comutativitatea	$\varphi \wedge \psi \sim \psi \wedge \varphi \qquad \varphi \vee \psi \sim \psi \vee \varphi$	(15)
asociativitatea	$\varphi \wedge (\psi \wedge \chi) \sim (\varphi \wedge \psi) \wedge \chi$	(16)
	$\varphi \vee (\psi \vee \chi) \sim (\varphi \vee \psi) \vee \chi$	(17)
absorbția	$\varphi \vee (\varphi \wedge \psi) \sim \varphi$	(18)
	$\varphi \wedge (\varphi \vee \psi) \sim \varphi$	(19)
distributivitatea	$\varphi \wedge (\psi \vee \chi) \sim (\varphi \wedge \psi) \vee (\varphi \wedge \chi)$	(20)
	$\varphi \vee (\psi \wedge \chi) \sim (\varphi \vee \psi) \wedge (\varphi \vee \chi)$	(21)

Tautologii, consecințe semantice și echivalențe

$$\varphi \rightarrow \psi \wedge \chi \sim (\varphi \rightarrow \psi) \wedge (\varphi \rightarrow \chi) \quad (22)$$
$$\varphi \rightarrow \psi \vee \chi \sim (\varphi \rightarrow \psi) \vee (\varphi \rightarrow \chi) \quad (23)$$
$$\varphi \wedge \psi \rightarrow \chi \sim (\varphi \rightarrow \chi) \vee (\psi \rightarrow \chi) \quad (24)$$
$$\varphi \vee \psi \rightarrow \chi \sim (\varphi \rightarrow \chi) \wedge (\psi \rightarrow \chi) \quad (25)$$
$$\varphi \rightarrow (\psi \rightarrow \chi) \sim \psi \rightarrow (\varphi \rightarrow \chi) \sim (\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \chi) \quad (26)$$
$$\neg\varphi \sim \varphi \rightarrow \neg\varphi \sim (\varphi \rightarrow \psi) \wedge (\varphi \rightarrow \neg\psi) \quad (27)$$
$$\varphi \rightarrow \psi \sim \neg\varphi \vee \psi \sim \neg(\varphi \wedge \neg\psi) \quad (28)$$
$$\varphi \vee \psi \sim \varphi \vee (\neg\varphi \wedge \psi) \sim (\varphi \rightarrow \psi) \rightarrow \psi \quad (29)$$
$$\varphi \leftrightarrow (\psi \leftrightarrow \chi) \sim (\varphi \leftrightarrow \psi) \leftrightarrow \chi \quad (30)$$
$$\vDash (\varphi \rightarrow \psi) \vee (\neg\varphi \rightarrow \psi) \quad (31)$$
$$\vDash (\varphi \rightarrow \psi) \vee (\varphi \rightarrow \neg\psi) \quad (32)$$
$$\vDash \neg\varphi \rightarrow (\neg\psi \leftrightarrow (\psi \rightarrow \varphi)) \quad (33)$$
$$\vDash (\varphi \rightarrow \psi) \rightarrow (((\varphi \rightarrow \chi) \rightarrow \psi) \rightarrow \psi) \quad (34)$$

Dem.: Exercițiu.

Exemplu de demonstrație

Demonstrăm (1): $\vDash \varphi \vee \neg\varphi$.
Fie $e : V \to \{0,1\}$ o evaluare arbitrară. Trebuie să arătăm că $e^+(\varphi \vee \neg\varphi) = 1$. Observăm că $e^+(\varphi \vee \neg\varphi) = e^+(\varphi) \vee \neg e^+(\varphi)$.
Putem demonstra că $e^+(\varphi) \vee \neg e^+(\varphi) = 1$ în două moduri.

I. Folosim tabelele de adevăr.

$e^+(\varphi)$	$\neg e^+(\varphi)$	$e^+(\varphi) \vee \neg e^+(\varphi)$
0	1	1
1	0	1

II. Raționăm direct.

Avem două cazuri:

- $e^+(\varphi) = 1$. Atunci $\neg e^+(\varphi) = 0$ și, prin urmare, $e^+(\varphi) \vee \neg e^+(\varphi) = 1$.
- $e^+(\varphi) = 0$. Atunci $\neg e^+(\varphi) = 1$ și, prin urmare, $e^+(\varphi) \vee \neg e^+(\varphi) = 1$.

$\top$ și $\bot$

De multe ori este convenabil să avem o tautologie canonică și o formulă nesatisfiabilă canonică.

Observație

$v_0 \to v_0$ este tautologie și $\neg(v_0 \to v_0)$ este nesatisfiabilă.

Dem.: Exercițiu.

Notații

Notăm $v_0 \to v_0$ cu $\top$ și o numim adevărul. Notăm $\neg(v_0 \to v_0)$ cu $\bot$ și o numim falsul.

- φ este tautologie ddacă $\varphi \sim \top$.
- φ este nesatisfiabilă ddacă $\varphi \sim \bot$.

Substituția

Definiția 2.19

Pentru orice formule φ, χ, χ', definim

$$\varphi_\chi(\chi') \quad := \quad \text{expresia obținută din } \varphi \text{ prin înlocuirea tuturor apariițiilor lui } \chi \text{ cu } \chi'.$$

$\varphi_\chi(\chi')$ se numește substituția lui χ cu χ' în φ. Spunem și că $\varphi_\chi(\chi')$ este o instanță de substituție a lui φ.

- $\varphi_\chi(\chi')$ este de asemenea formulă.
- $\varphi_\varphi(\chi') = \chi'$.
- Dacă χ nu este subformulă a lui φ, atunci $\varphi_\chi(\chi') = \varphi$.

Exemple:
Fie $\varphi = (v_1 \to v_2) \to \neg(v_1 \to v_2)$.

- $\chi = v_1 \to v_2$, $\chi' = v_4$. $\quad \varphi_\chi(\chi') = v_4 \to \neg v_4$
- $\chi = v_1$, $\chi' = \neg\neg v_2$. $\varphi_\chi(\chi') = (\neg\neg v_2 \to v_2) \to \neg(\neg\neg v_2 \to v_2)$
- $\chi = v_1 \to v_2$, $\chi' = v_4 \vee v_1$. $\quad \varphi_\chi(\chi') = (v_4 \vee v_1) \to \neg(v_4 \vee v_1)$

Substituția

Propoziția 2.20

Pentru orice formule φ, χ, χ',

$$\chi \sim \chi' \quad \text{implică} \quad \varphi \sim \varphi_\chi(\chi').$$

Propoziția 2.21

Pentru orice formule φ, ψ, χ și orice variabilă $v \in V$,

- $\varphi \sim \psi$ *implică* $\varphi_v(\chi) \sim \psi_v(\chi)$.
- *Dacă φ este tautologie atunci și $\varphi_v(\chi)$ este tautologie.*
- *Dacă φ este nesatisfiabilă, atunci și $\varphi_v(\chi)$ este nesatisfiabilă.*

Conjuncții și disjuncții finite

Notații

Scriem $\varphi \wedge \psi \wedge \chi$ în loc de $(\varphi \wedge \psi) \wedge \chi$. Similar, scriem $\varphi \vee \psi \vee \chi$ în loc de $(\varphi \vee \psi) \vee \chi$.

Fie $\varphi_1, \varphi_2, \ldots, \varphi_n$ formule. Pentru $n \geq 3$, notăm

$$\varphi_1 \wedge \ldots \wedge \varphi_n := ((\ldots(\varphi_1 \wedge \varphi_2) \wedge \varphi_3) \wedge \ldots \wedge \varphi_{n-1}) \wedge \varphi_n$$
$$\varphi_1 \vee \ldots \vee \varphi_n := ((\ldots(\varphi_1 \vee \varphi_2) \vee \varphi_3) \vee \ldots \vee \varphi_{n-1}) \vee \varphi_n.$$

- $\varphi_1 \wedge \ldots \wedge \varphi_n$ se mai scrie și $\bigwedge_{i=1}^n \varphi_i$ sau $\displaystyle\bigwedge_{i=1}^n \varphi_i$.
- $\varphi_1 \vee \ldots \vee \varphi_n$ se mai scrie și $\bigvee_{i=1}^n \varphi_i$ sau $\displaystyle\bigvee_{i=1}^n \varphi_i$.

Conjuncții și disjuncții finite

Propoziția 2.22

Pentru orice evaluare $e : V \to \{0,1\}$,

- $e^+(\varphi_1 \wedge \ldots \wedge \varphi_n) = 1$ *ddacă* $e^+(\varphi_i) = 1$ *pentru orice* $i \in \{1, \ldots, n\}$.
- $e^+(\varphi_1 \vee \ldots \vee \varphi_n) = 1$ *ddacă* $e^+(\varphi_i) = 1$ *pentru un* $i \in \{1, \ldots, n\}$.

Dem.: Exercițiu.

Propoziția 2.23

$$\neg(\varphi_1 \vee \ldots \vee \varphi_n) \quad \sim \quad \neg\varphi_1 \wedge \ldots \wedge \neg\varphi_n$$
$$\neg(\varphi_1 \wedge \ldots \wedge \varphi_n) \quad \sim \quad \neg\varphi_1 \vee \ldots \vee \neg\varphi_n$$

Dem.: Exercițiu.

Mulțimi de formule

Fie Γ o mulțime de formule.

Definiția 2.24

- *O evaluare $e : V \to \{0,1\}$ este model al lui Γ dacă este model al fiecărei formule din Γ (adică $e \vDash \gamma$ pentru orice $\gamma \in \Gamma$).* *Notație:* $e \vDash \Gamma$.
- *Γ este satisfiabilă dacă are un model.*
- *Γ este finit satisfiabilă dacă orice submulțime finită a sa este satisfiabilă.*
- *Dacă Γ nu este satisfiabilă, spunem și că Γ este nesatisfiabilă sau contradictorie.*

Notații: Mulțimea tuturor modelelor lui Γ se notează $Mod(\Gamma)$. Notăm $Mod(\varphi_1, \ldots, \varphi_n)$ în loc de $Mod(\{\varphi_1, \ldots, \varphi_n\})$.

- $Mod(\Gamma) = \bigcap_{\varphi \in \Gamma} Mod(\varphi)$.

Mulțimi de formule

Fie Γ, Δ mulțimi de formule.

Definiția 2.25

O formulă φ este consecință semantică a lui Γ dacă $Mod(\Gamma) \subseteq Mod(\varphi)$. Notație: $\Gamma \vDash \varphi$.
Dacă φ nu este consecință semantică a lui Γ, scriem $\Gamma \nvDash \varphi$.

Notăm cu $Cn(\Gamma)$ mulțimea consecințelor semantice ale lui Γ. Așadar,

$$Cn(\Gamma) = \{\varphi \in Form \mid \Gamma \vDash \varphi\}.$$

Definiția 2.26

- *Δ este consecință semantică a lui Γ dacă $Mod(\Gamma) \subseteq Mod(\Delta)$.* *Notație:* $\Gamma \vDash \Delta$.
- *Γ și Δ sunt (logic) echivalente dacă $Mod(\Gamma) = Mod(\Delta)$.* *Notație:* $\Gamma \sim \Delta$.

Proprietăți

Următoarele rezultate colectează diverse proprietăți utile.

Observație

- $\psi \vDash \varphi$ ddacă $\{\psi\} \vDash \varphi$ ddacă $\{\psi\} \vDash \{\varphi\}$.
- $\psi \sim \varphi$ ddacă $\{\psi\} \sim \{\varphi\}$.

Propoziția 2.27

- *$Mod(\emptyset) = Fun(V, \{0,1\})$, adică orice evaluare $e : V \to \{0,1\}$ este model al mulțimii vide. În particular, mulțimea vidă este satisfiabilă.*
- *$Cn(\emptyset)$ este mulțimea tuturor tautologiilor, adică φ este tautologie ddacă $\emptyset \vDash \varphi$.*

Dem.: Exercițiu ușor.

Proprietăți

Propoziția 2.28

Fie $\Gamma \cup \{\varphi, \psi\} \subseteq Form$.

(i) *Dacă $\Gamma \vDash \varphi$ și $\Gamma \vDash \varphi \to \psi$, atunci $\Gamma \vDash \psi$.*

(ii) *$\Gamma \cup \{\varphi\} \vDash \psi$ ddacă $\Gamma \vDash \varphi \to \psi$.*

(iii) *$\Gamma \vDash \varphi \land \psi$ ddacă $\Gamma \vDash \varphi$ și $\Gamma \vDash \psi$.*

Dem.: Exercițiu.

Propoziția 2.29

Fie Γ o mulțime de formule. Următoarele afirmații sunt echivalente:

(i) *Γ este nesatisfiabilă.*

(ii) *$\Gamma \vDash \varphi$ pentru orice formulă φ.*

(iii) *$\Gamma \vDash \varphi$ pentru orice formulă nesatisfiabilă φ.*

(iv) *$\Gamma \vDash \bot$.*

Dem.: Exercițiu ușor.

Proprietăți

Propoziția 2.30

Fie Γ o mulțime de formule.

(i) *$\Gamma \vDash \varphi$ ddacă $\Gamma \cup \{\neg\varphi\}$ este nesatisfiabilă.*

(ii) *$\Gamma \vDash \neg\varphi$ ddacă $\Gamma \cup \{\varphi\}$ este nesatisfiabilă.*

(iii) *Dacă Γ este satisfiabilă, atunci cel puțin una dintre $\Gamma \cup \{\varphi\}$ și $\Gamma \cup \{\neg\varphi\}$ este satisfiabilă.*

Dem.:

(i) Avem că $\Gamma \not\vDash \varphi \iff$ există o evaluare $e : V \to \{0,1\}$ a.î. $e \vDash \Gamma$ și $e \not\vDash \varphi \iff$ există o evaluare $e : V \to \{0,1\}$ a.î. $e \vDash \Gamma$ și $e \vDash \neg\varphi \iff$ există o evaluare $e : V \to \{0,1\}$ a.î. $e \vDash \Gamma \cup \{\neg\varphi\} \iff \Gamma \cup \{\neg\varphi\}$ este satisfiabilă.

(ii) Similar.

(iii) Fie e un model al lui Γ. Dacă $e \vDash \varphi$, atunci e este model al lui $\Gamma \cup \{\varphi\}$. Dacă $e \not\vDash \varphi$, deci $e \vDash \neg\varphi$, atunci e este model al lui $\Gamma \cup \{\neg\varphi\}$. $\square$

Proprietăți

Propoziția 2.31

Fie $\Gamma = \{\varphi_1, \ldots, \varphi_n\}$ o mulțime finită de formule.

(i) *$\Gamma \sim \{\varphi_1 \land \ldots \land \varphi_n\}$.*

(ii) *$\Gamma \vDash \psi$ ddacă $\vDash \varphi_1 \land \ldots \land \varphi_n \to \psi$.*

(iii) *Γ este nesatisfiabilă ddacă $\neg\varphi_1 \lor \neg\varphi_2 \lor \ldots \lor \neg\varphi_n$ este tautologie.*

(iv) *Dacă $\Delta = \{\psi_1, \ldots, \psi_k\}$ este o altă mulțime finită de formule, atunci următoarele afirmații sunt echivalente:*

  (a) *$\Gamma \sim \Delta$.*

  (b) *$\varphi_1 \land \ldots \land \varphi_n \sim \psi_1 \land \ldots \land \psi_k$.*

Dem.: Exercițiu.

Teorema de compactitate - versiunea 1

Pentru orice mulțime Γ de formule, Γ este satisfiabilă ddacă Γ este finit satisfiabilă.

Teorema de compactitate - versiunea 2

Pentru orice mulțime Γ de formule, Γ este nesatisfiabilă ddacă Γ nu este finit satisfiabilă.

Teorema de compactitate - versiunea 3

Pentru orice mulțime Γ de formule și pentru orice formulă φ, $\Gamma \vDash \varphi$ ddacă există o submulțime finită Δ a lui Γ a.î. $\Delta \vDash \varphi$.

Propoziția 2.32

Cele trei versiuni sunt echivalente.

Dem.: Exercițiu.

Lema 2.33

Fie Γ finit satisfiabilă. Atunci există un șir $(\varepsilon_n)_{n\in\mathbb{N}}$ în $\{0,1\}$ care satisface, pentru orice $n \in \mathbb{N}$:

P_n *Orice submulțime finită Δ a lui Γ are un model $e : V \to \{0,1\}$ cu proprietatea că $e(v_i) = \varepsilon_i$ pentru orice $i \in \{0,1,\dots n\}$.*

Dem.: Exercițiu suplimentar.

Teorema 2.34 (Teorema de compactitate)

Pentru orice mulțime Γ de formule, Γ este satisfiabilă ddacă Γ este finit satisfiabilă.

Dem.: "$\Leftarrow$" Presupunem că Γ este finit satisfiabilă. Definim

$$\overline{e} : V \to \{0,1\}, \quad \overline{e}(v_n) = \varepsilon_n,$$

unde (ε_n) este șirul construit în Lema 2.33. Demonstrăm că $\overline{e}$ este model al lui Γ. Fie $\varphi \in \Gamma$ arbitrară și fie $k \in \mathbb{N}$ a.î. $Var(\varphi) \subseteq \{v_0, v_1, \dots, v_k\}$. Avem că $\{\varphi\} \subseteq \Gamma$ este o submulțime finită a lui Γ.

Teorema 2.34 (Teorema de compactitate)

Pentru orice mulțime Γ de formule, Γ este satisfiabilă ddacă Γ este finit satisfiabilă.

Dem.: (continuare)
Aplicând proprietatea $\boldsymbol{P_k}$, obținem un model e al lui φ a.î. $e(v_i) = \varepsilon_i$ pentru orice $i \in \{0,1,\dots k\}$.
Atunci $\overline{e}(v) = e(v)$ pentru orice variabilă $v \in Var(\varphi)$. Din Propoziția 2.13 rezultă că $\overline{e}^+(\varphi) = e^+(\varphi) = 1$, deci $\overline{e} \vDash \varphi$.
Prin urmare, $\overline{e}$ este model al lui Γ, deci Γ este satisfiabilă.

"$\Rightarrow$" Evident. $\square$

FORMA NORMALĂ CONJUNCTIVĂ / DISJUNCTIVĂ

Definiția 2.35

Un literal este o

- *variabilă (în care caz spunem că este literal pozitiv) sau*
- *negația unei variabile (în care caz spunem că este literal negativ).*

Exemple: v_1, v_2, v_{10} literali pozitivi; $\neg v_0, \neg v_{100}$ literali negativi

Convenție: $\bigvee_{i=1}^{1} \varphi_i = \varphi_1$ și $\bigwedge_{i=1}^{1} \varphi_i = \varphi_1$.

Definiția 2.36

O formulă φ este în formă normală disjunctivă (FND) dacă φ este o disjuncție de conjuncții de literali.

Așadar, φ este în FND ddacă $\varphi = \bigvee_{i=1}^{n} \left(\bigwedge_{j=1}^{k_i} L_{i,j} \right)$, unde fiecare $L_{i,j}$ este literal.

Definiția 2.37

O formulă φ este în formă normală conjunctivă (FNC) dacă φ este o conjuncție de disjuncții de literali.

Așadar, φ este în FNC ddacă $\varphi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{k_i} L_{i,j} \right)$, unde fiecare $L_{i,j}$ este literal.

Exemple

- $(v_0 \vee v_1) \wedge (v_3 \vee v_5) \wedge (\neg v_{20} \vee \neg v_{15} \vee \neg v_{34})$ este în FNC
- $(\neg v_9 \wedge v_1) \vee v_{24} \vee (v_2 \wedge \neg v_1 \wedge v_2)$ este în FND
- $v_1 \wedge \neg v_5 \wedge v_4$ este atât în FND cât și în FNC
- $\neg v_{10} \vee v_{20} \vee v_4$ este atât în FND cât și în FNC
- $(v_1 \vee v_2) \wedge ((v_1 \wedge v_3) \vee (v_4 \wedge v_5))$ nu este nici în FND, nici în FNC

Notație: Dacă L este literal, atunci $L^c := \begin{cases} \neg v & \text{dacă } L = v \in V \\ v & \text{dacă } L = \neg v. \end{cases}$

Propoziția 2.38

(i) Fie φ o formulă în FNC, $\varphi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{k_i} L_{i,j} \right)$. Atunci $\neg\varphi \sim \bigvee_{i=1}^{n} \left(\bigwedge_{j=1}^{k_i} L_{i,j}^c \right)$, o formulă în FND.

(ii) Fie φ o formulă în FND, $\varphi = \bigvee_{i=1}^{n} \left(\bigwedge_{j=1}^{k_i} L_{i,j} \right)$. Atunci $\neg\varphi \sim \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{k_i} L_{i,j}^c \right)$, o formulă în FNC.

Dem.: Exercițiu.

Exemplu: Arătați că $\vDash v_1 \rightarrow (v_2 \rightarrow v_1 \wedge v_2)$.

v_1	v_2	$v_1 \rightarrow (v_2 \rightarrow v_1 \wedge v_2)$
0	0	1
0	1	1
1	0	1
1	1	1

Acest tabel definește o funcție $F : \{0,1\}^2 \rightarrow \{0,1\}$

ε_1	ε_2	$F(\varepsilon_1, \varepsilon_2)$
0	0	1
0	1	1
1	0	1
1	1	1

Funcția asociată unei formule

Fie φ o formulă și $Var(\varphi) = \{v_{i_1}, v_{i_2} \ldots, v_{i_n}\}$, unde $n \geq 1$ și $0 \leq i_1 < i_2 < \ldots < i_n$.

Fie $(\varepsilon_1, \ldots, \varepsilon_n) \in \{0,1\}^n$. Definim $e_{\varepsilon_1,\ldots,\varepsilon_n} : Var(\varphi) \to \{0,1\}$ astfel:

$$e_{\varepsilon_1,\ldots,\varepsilon_n}(v_{i_k}) = \varepsilon_k \quad \text{pentru orice } k \in \{1, \ldots, n\}.$$

Definim $e^+_{\varepsilon_1,\ldots,\varepsilon_n}(\varphi) \in \{0,1\}$ astfel:

$$e^+_{\varepsilon_1,\ldots,\varepsilon_n}(\varphi) := e^+(\varphi),$$

unde $e : V \to \{0,1\}$ este orice evaluare care extinde $e_{\varepsilon_1,\ldots,\varepsilon_n}$, adică, $e(v_{i_k}) = e_{\varepsilon_1,\ldots,\varepsilon_n}(v_{i_k}) = \varepsilon_k$ pentru orice $k \in \{1, \ldots, n\}$.
Conform Propoziției 2.13, definiția nu este ambiguă.

Definiția 2.39

Funcția asociată lui φ este $F_\varphi : \{0,1\}^n \to \{0,1\}$, definită astfel:

$$F_\varphi(\varepsilon_1, \ldots, \varepsilon_n) = e^+_{\varepsilon_1,\ldots,\varepsilon_n}(\varphi) \text{ pentru orice } (\varepsilon_1, \ldots, \varepsilon_n) \in \{0,1\}^n.$$

Așadar, F_φ este funcția definită de tabela de adevăr pentru φ.

Funcția asociată unei formule

Propoziția 2.40

(i) *Fie φ o formulă. Atunci*
  (a) $\models \varphi$ *ddacă F_φ este funcția constantă 1.*
  (b) *φ este nesatisfiabilă ddacă F_φ este funcția constantă 0.*

(ii) *Fie φ, ψ două formule astfel încât $Var(\varphi) = Var(\psi)$. Atunci*
  (a) $\varphi \models \psi$ *ddacă* $F_\varphi \leq F_\psi$.
  (b) $\varphi \sim \psi$ *ddacă* $F_\varphi = F_\psi$.

(iii) *Există formule diferite φ, ψ a.î. $F_\varphi = F_\psi$.*

Caracterizarea funcțiilor booleene

Definiția 2.41

O funcție booleană este o funcție $F : \{0,1\}^n \to \{0,1\}$, unde $n \geq 1$. Spunem că n este numărul variabilelor lui F.

Exemplu: Pentru orice formulă φ, F_φ este funcție Booleană cu n variabile, unde $n = |Var(\varphi)|$.

Teorema 2.42

Fie $n \geq 1$ și $H : \{0,1\}^n \to \{0,1\}$ o funcție booleană arbitrară. Atunci există o formulă φ în FND a.î. $H = F_\varphi$.

Dem.: Dacă $H(\varepsilon_1, \ldots, \varepsilon_n) = 0$ pentru orice $(\varepsilon_1, \ldots, \varepsilon_n) \in \{0,1\}^n$, luăm $\varphi := \bigvee_{i=1}^{n}(v_i \wedge \neg v_i)$. Avem că $Var(\varphi) = \{v_1, \ldots, v_n\}$, așadar, $F_\varphi : \{0,1\}^n \to \{0,1\}$. Cum $v_i \wedge \neg v_i$ este nesatisfiabilă pentru orice i, rezultă că φ este de asemenea nesatisfiabilă. Deci, F_φ este funcția constantă 0.

Caracterizarea funcțiilor booleene

Altcumva, mulțimea

$$T := H^{-1}(1) = \{(\varepsilon_1, \ldots, \varepsilon_n) \in \{0,1\}^n \mid H(\varepsilon_1, \ldots, \varepsilon_n) = 1\}$$

este nevidă.
Considerăm formula

$$\varphi := \bigvee_{(\varepsilon_1,\ldots,\varepsilon_n)\in T} \left(\bigwedge_{\varepsilon_i=1} v_i \wedge \bigwedge_{\varepsilon_i=0} \neg v_i \right).$$

Deoarece $Var(\varphi) = \{v_1, \ldots, v_n\}$, avem că $F_\varphi : \{0,1\}^n \to \{0,1\}$.

Se demonstrează că $H = F_\varphi$ (exercițiu suplimentar). $\square$

Caracterizarea funcțiilor booleene

Teorema 2.43

Fie $n \geq 1$ și $H : \{0,1\}^n \to \{0,1\}$ o funcție booleană arbitrară. Atunci există o formulă ψ în FNC a.î. $H = F_\psi$.

Dem.: Dacă $H(\varepsilon_1, \ldots, \varepsilon_n) = 1$ pentru orice $(\varepsilon_1, \ldots, \varepsilon_n) \in \{0,1\}^n$, atunci luăm

$$\psi := \bigwedge_{i=1}^{n} (v_i \vee \neg v_i).$$

Altcumva, mulțimea

$$F := H^{-1}(0) = \{(\varepsilon_1, \ldots, \varepsilon_n) \in \{0,1\}^n \mid H(\varepsilon_1, \ldots, \varepsilon_n) = 0\}$$

este nevidă.

Considerăm formula $\psi := \displaystyle\bigwedge_{(\varepsilon_1,\ldots,\varepsilon_n)\in F} \left(\bigvee_{\varepsilon_i = 1} \neg v_i \vee \bigvee_{\varepsilon_i=0} v_i \right)$.

Se demonstrează că $H = F_\psi$ (exercițiu suplimentar). $\square$

Caracterizarea funcțiilor Booleene

Exemplu: Fie $H : \{0,1\}^3 \to \{0,1\}$ descrisă prin tabelul:

ε_1	ε_2	ε_3	$H(\varepsilon_1, \varepsilon_2, \varepsilon_3)$	
0	0	0	0	$D_1 = v_1 \vee v_2 \vee v_3$
0	0	1	0	$D_2 = v_1 \vee v_2 \vee \neg v_3$
0	1	0	1	$C_1 = \neg v_1 \wedge v_2 \wedge \neg v_3$
0	1	1	0	$D_3 = v_1 \vee \neg v_2 \vee \neg v_3$
1	0	0	1	$C_2 = v_1 \wedge \neg v_2 \wedge \neg v_3$
1	0	1	1	$C_3 = v_1 \wedge \neg v_2 \wedge v_3$
1	1	0	1	$C_4 = v_1 \wedge v_2 \wedge \neg v_3$
1	1	1	1	$C_5 = v_1 \wedge v_2 \wedge v_3$

$\varphi = C_1 \vee C_2 \vee C_3 \vee C_4 \vee C_5$ în FND a.î. $H = F_\varphi$.

$\psi = D_1 \wedge D_2 \wedge D_3$ în FNC a.î. $H = F_\psi$.

Forma normală conjunctivă / disjunctivă

Teorema 2.44

Orice formulă φ este echivalentă cu o formulă φ^{FND} în FND și cu o formulă φ^{FNC} în FNC.

Dem.:
Fie $Var(\varphi) = \{x_1, \ldots, x_n\}$ și $F_\varphi : \{0,1\}^n \to \{0,1\}$ funcția booleană asociată. Aplicând Teorema 2.42 cu $H := F_\varphi$, obținem o formulă φ^{FND} în FND a.î. $F_\varphi = F_{\varphi^{FND}}$. Așadar, conform Propoziției 2.40.(ii), $\varphi \sim \varphi^{FND}$.

Similar, aplicând Teorema 2.43 cu $H := F_\varphi$, obținem o formulă φ^{FNC} în FNC a.î. $F_\varphi = F_{\varphi^{FNC}}$. Prin urmare, $\varphi \sim \varphi^{FNC}$. $\square$

Forma normală conjunctivă / disjunctivă

Algoritm pentru a aduce o formulă la FNC/FND:

Pasul 1. Se înlocuiesc implicațiile și echivalențele, folosind:

$$\varphi \to \psi \sim \neg\varphi \vee \psi \quad \text{și} \quad \varphi \leftrightarrow \psi \sim (\neg\varphi \vee \psi) \wedge (\neg\psi \vee \varphi).$$

Pasul 2. Se înlocuiesc dublele negații, folosind $\neg\neg\psi \sim \psi$, și se aplică regulile De Morgan pentru a înlocui

$$\neg(\varphi \vee \psi) \text{ cu } \neg\varphi \wedge \neg\psi \quad \text{și} \quad \neg(\varphi \wedge \psi) \text{ cu } \neg\varphi \vee \neg\psi.$$

Pasul 3. Pentru FNC, se aplică distributivitatea lui $\vee$ fața de $\wedge$, pentru a înlocui

$$\varphi \vee (\psi \wedge \chi) \text{ cu } (\varphi \vee \psi) \wedge (\varphi \vee \chi) \quad \text{și} \quad (\psi \wedge \chi) \vee \varphi \text{ cu } (\psi \vee \varphi) \wedge (\chi \vee \varphi).$$

Pentru FND, se aplică distributivitatea lui $\wedge$ fața de $\vee$, pentru a înlocui

$$\varphi \wedge (\psi \vee \chi) \text{ cu } (\varphi \wedge \psi) \vee (\varphi \wedge \chi) \quad \text{și} \quad (\psi \vee \chi) \wedge \varphi \text{ cu } (\psi \wedge \varphi) \vee (\chi \wedge \varphi).$$

Exemplu

Considerăm formula $\varphi := (\neg v_0 \to \neg v_2) \to (v_0 \to v_2)$.

Avem

$$\begin{array}{lll} \varphi & \sim & \neg(\neg v_0 \to \neg v_2) \vee (v_0 \to v_2) \quad \text{Pasul 1} \\ & \sim & \neg(\neg\neg v_0 \vee \neg v_2) \vee (v_0 \to v_2) \quad \text{Pasul 1} \\ & \sim & \neg(\neg\neg v_0 \vee \neg v_2) \vee (\neg v_0 \vee v_2) \quad \text{Pasul 1} \\ & \sim & \neg(v_0 \vee \neg v_2) \vee (\neg v_0 \vee v_2) \quad \text{Pasul 2} \\ & \sim & (\neg v_0 \wedge \neg\neg v_2) \vee (\neg v_0 \vee v_2) \quad \text{Pasul 2} \\ & \sim & (\neg v_0 \wedge v_2) \vee \neg v_0 \vee v_2 \quad \text{Pasul 2} \end{array}$$

Putem lua $\varphi^{FND} := (\neg v_0 \wedge v_2) \vee \neg v_0 \vee v_2$.

Pentru a obține FNC, continuăm cu Pasul 3:

$$\begin{array}{lll} \varphi & \sim & (\neg v_0 \wedge v_2) \vee (\neg v_0 \vee v_2) \\ & \sim & (\neg v_0 \vee \neg v_0 \vee v_2) \wedge (v_2 \vee \neg v_0 \vee v_2). \end{array}$$

Putem lua $\varphi^{FNC} := (\neg v_0 \vee \neg v_0 \vee v_2) \wedge (v_2 \vee \neg v_0 \vee v_2)$. Se observă, folosind idempotența și comutativitatea lui $\vee$, că $\varphi^{FNC} \sim \neg v_0 \vee v_2$.

CLAUZE ȘI REZOLUȚIE

Definiția 2.45

O clauză este o mulțime finită de literali:

$$C = \{L_1, \ldots, L_n\}, \text{ unde } L_1, \ldots, L_n \text{ sunt literali.}$$

Dacă $n = 0$, obținem clauza vidă $\square := \emptyset$.

O clauză nevidă este considerată implicit o disjuncție.

Definiția 2.46

Fie C o clauză și $e : V \to \{0,1\}$. Spunem că e este model al lui C sau că e satisface C și scriem $e \vDash C$ dacă există $L \in C$ a.î. $e \vDash L$.

Definiția 2.47

O clauză C se numește

(i) satisfiabilă dacă are un model.

(ii) validă dacă orice evaluare $e : V \to \{0,1\}$ este model al lui C.

Definiția 2.48

O clauză C este trivială dacă există un literal L a.î. $L \in C$ și $L^c \in C$.

Propoziția 2.49

(i) Orice clauză nevidă este satisfiabilă.

(ii) Clauza vidă $\square$ este nesatisfiabilă.

(iii) O clauză este validă ddacă este trivială.

Dem.: Exercițiu.

Notăm $Var(C) := \{x \in V \mid x \in C \text{ sau } \neg x \in C\}$.
Daca $x \in Var(C)$, spunem ca x apare în C.

- $Var(C) = \emptyset$ ddacă $C = \square$.

Clauze

$\mathcal{S} = \{C_1, \ldots, C_m\}$ este o mulțime finită de clauze.
Dacă $m = 0$, obținem mulțimea vidă de clauze $\emptyset$.

$\mathcal{S}$ este considerată implicit ca o formulă în FNC: conjuncție de disjuncții ale literalilor din fiecare clauză.

Definiția 2.50

Fie $e : V \to \{0,1\}$. Spunem că e este model al lui $\mathcal{S}$ sau că e satisface $\mathcal{S}$ și scriem $e \vDash \mathcal{S}$ dacă $e \vDash C_i$ pentru orice $i \in \{1, \ldots, m\}$.

Definiția 2.51

$\mathcal{S}$ se numește

(i) satisfiabilă dacă are un model.

(ii) validă dacă orice evaluare $e : V \to \{0,1\}$ este model al lui $\mathcal{S}$.

Clauze

Propoziția 2.52

- *Dacă $\mathcal{S}$ conține clauza vidă $\square$, atunci $\mathcal{S}$ este nesatisfiabilă.*
- *$\emptyset$ este validă.*

Dem.: Exercițiu.

Notăm $Var(\mathcal{S}) := \bigcup_{C \in \mathcal{S}} Var(C)$.
Daca $x \in Var(\mathcal{S})$, spunem ca x apare în $\mathcal{S}$.

- $Var(\mathcal{S}) = \emptyset$ ddacă ($\mathcal{S} = \emptyset$ sau $\mathcal{S} = \{\square\}$).

Clauze

Exemplu

$\mathcal{S} = \{\{v_1, \neg v_3\}, \{\neg v_3, v_3\}, \{v_2, v_1\}, \{v_2, \neg v_1, v_3\}\}$ este satisfiabilă.

Dem.: Considerăm $e : V \to \{0,1\}$ a.î. $e(v_1) = e(v_2) = 1$. Atunci $e \vDash \mathcal{S}$. $\square$

Exemplu

$\mathcal{S} = \{\{\neg v_1, v_2\}, \{\neg v_3, \neg v_2\}, \{v_1\}, \{v_3\}\}$ este nesatisfiabilă.

Dem.: Presupunem că $\mathcal{S}$ are un model e. Atunci $e(v_1) = e(v_3) = 1$ și, deoarece $e \vDash \{\neg v_3, \neg v_2\}$, trebuie să avem $e(v_2) = 0$. Rezultă că $e(v_2) = e^+(\neg v_1) = 0$, deci e nu satisface $\{\neg v_1, v_2\}$. Am obținut o contradicție. $\square$

Clauze și FNC

Unei formule φ în FNC îi asociem o mulțime finită de clauze $\mathcal{S}_\varphi$ astfel:

Fie

$$\varphi := \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{k_i} L_{i,j} \right),$$

unde fiecare $L_{i,j}$ este literal. Pentru orice i, fie C_i clauza obținută considerând toți literalii $L_{i,j}, j \in \{1, \ldots, k_i\}$ distincți. Fie $\mathcal{S}_\varphi$ mulțimea tuturor clauzelor $C_i, i \in \{1, \ldots, n\}$ distincte.

$\mathcal{S}_\varphi$ se mai numește și forma clauzală a lui φ.

Propoziția 2.53

Pentru orice evaluare $e : V \to \{0,1\}$, $e \vDash \varphi$ ddacă $e \vDash \mathcal{S}_\varphi$.

Clauze și FNC

Unei mulțimi finite de clauze $\mathcal{S}$ îi asociem o formulă $\varphi_{\mathcal{S}}$ în FNC astfel:

- $C = \{L_1, \ldots, L_n\}, n \geq 1 \longmapsto \varphi_C := L_1 \vee L_2 \vee \ldots \vee L_n$.
- $\square \longmapsto \varphi_\square := v_0 \wedge \neg v_0$.

Fie $\mathcal{S} = \{C_1, \ldots, C_m\}$ o mulțime nevidă de clauze. Formula asociată lui $\mathcal{S}$ este

$$\varphi_{\mathcal{S}} := \bigwedge_{i=1}^{m} \varphi_{C_i}.$$

Formula asociată mulțimii vide de clauze este $\varphi_\emptyset := v_0 \vee \neg v_0$.
Formula $\varphi_{\mathcal{S}}$ nu este unic determinată, depinde de ordinea în care se scriu elementele în clauze și în $\mathcal{S}$, dar se observă imediat că: $\mathcal{S} = \mathcal{S}'$ implică $\varphi_{\mathcal{S}} \sim \varphi_{\mathcal{S}'}$.

Propoziția 2.54

Pentru orice evaluare $e : V \to \{0,1\}$, $e \vDash \mathcal{S}$ ddacă $e \vDash \varphi_{\mathcal{S}}$.

Rezoluția

Definiția 2.55

Fie C_1, C_2 două clauze. O clauză R se numește rezolvent al clauzelor C_1, C_2 dacă există un literal L a.î. $L \in C_1$, $L^c \in C_2$ și

$$R = (C_1 \setminus \{L\}) \cup (C_2 \setminus \{L^c\}).$$

Regula Rezoluției

$$Rez \quad \frac{C_1, C_2}{(C_1 \setminus \{L\}) \cup (C_2 \setminus \{L^c\})}, \quad L \in C_1, L^c \in C_2$$

Notăm cu $Res(C_1, C_2)$ mulțimea rezolvenților clauzelor C_1, C_2.

- Rezoluția a fost introdusă de Blake (1937) și dezvoltată de Davis, Putnam (1960) și Robinson (1965).
- Multe demonstratoare automate de teoreme folosesc rezoluția. Limbajul PROLOG este bazat pe rezoluție.

Rezoluția

Exemplu

$C_1 = \{v_1, v_2, \neg v_5\}$, $C_2 = \{v_1, \neg v_2, v_{100}, v_5\}$.

- Luăm $L := \neg v_5$. Atunci $L \in C_1$ și $L^c = v_5 \in C_2$. Prin urmare, $R = \{v_1, v_2, \neg v_2, v_{100}\}$ este rezolvent al clauzelor C_1, C_2.
- Dacă luăm $L' := v_2$, atunci $L' \in C_1$ și $L'^c = \neg v_2 \in C_2$. Prin urmare, $R' = \{v_1, \neg v_5, v_{100}, v_5\}$ este rezolvent al clauzelor C_1, C_2.

Exemplu

$C_1 = \{v_7\}$, $C_2 = \{\neg v_7\}$. Atunci clauza vidă $\square$ este rezolvent al clauzelor C_1, C_2.

Rezoluția

Fie $\mathcal{S}$ o mulțime finită de clauze.

Definiția 2.56

O derivare prin rezoluție din $\mathcal{S}$ sau o $\mathcal{S}$-derivare prin rezoluție este o secvență $C_1, C_2, \ldots, C_n$ de clauze a.î. pentru fiecare $i \in \{1, \ldots, n\}$, una din următoarele condiții este satisfăcută:

(i) C_i este o clauză din $\mathcal{S}$;

(ii) există $j, k < i$ a.î. C_i este rezolvent al clauzelor C_j, C_k.

Definiția 2.57

Fie C o clauză. O derivare prin rezoluție a lui C din $\mathcal{S}$ este o $\mathcal{S}$-derivare prin rezoluție $C_1, C_2, \ldots, C_n$ a.î. $C_n = C$.

Rezoluția

Exemplu

Fie
$$\mathcal{S} = \{\{\neg v_1, v_2\}, \{\neg v_2, \neg v_3, v_4\}, \{v_1\}, \{v_3\}, \{\neg v_4\}\}.$$

O derivare prin rezoluție a clauzei vide $\square$ din $\mathcal{S}$ este următoarea:

$$\begin{array}{lcll}
C_1 & = & \{\neg v_4\} & C_1 \in \mathcal{S} \\
C_2 & = & \{\neg v_2, \neg v_3, v_4\} & C_2 \in \mathcal{S} \\
C_3 & = & \{\neg v_2, \neg v_3\} & C_3 \text{ rezolvent al clauzelor } C_1, C_2 \\
C_4 & = & \{v_3\} & C_4 \in \mathcal{S} \\
C_5 & = & \{\neg v_2\} & C_5 \text{ rezolvent al clauzelor } C_3, C_4 \\
C_6 & = & \{\neg v_1, v_2\} & C_6 \in \mathcal{S} \\
C_7 & = & \{\neg v_1\} & C_7 \text{ rezolvent al clauzelor } C_5, C_6 \\
C_8 & = & \{v_1\} & C_8 \in \mathcal{S} \\
C_9 & = & \square & C_9 \text{ rezolvent al clauzelor } C_7, C_8.
\end{array}$$

Rezoluția

Notăm $Res(\mathcal{S}) := \bigcup_{C_1, C_2 \in \mathcal{S}} Res(C_1, C_2)$.

Propoziția 2.58

Pentru orice orice evaluare $e : V \to \{0,1\}$,
$$e \vDash \mathcal{S} \quad \Rightarrow \quad e \vDash Res(\mathcal{S}).$$

Dem.: Dacă $Res(\mathcal{S}) = \emptyset$, atunci este validă, deci $e \vDash Res(\mathcal{S})$. Presupunem că $Res(\mathcal{S})$ este nevidă și fie $R \in Res(\mathcal{S})$. Atunci există clauze $C_1, C_2 \in \mathcal{S}$ și un literal L a.î. $L \in C_1, L^c \in C_2$ și $R = (C_1 \setminus \{L\}) \cup (C_2 \setminus \{L^c\})$. Avem două cazuri:

- $e \vDash L$. Atunci $e \nvDash L^c$. Deoarece $e \vDash C_2$, există $U \in C_2, U \neq L^c$ a.î. $e \vDash U$. Deoarece $U \in R$, obținem că $e \vDash R$.
- $e \nvDash L$. Deoarece $e \vDash C_1$, există $U \in C_1, U \neq L$ a.î. $e \vDash U$. Deoarece $U \in R$, obținem că $e \vDash R$. $\square$

Rezoluția

Teorema 2.59 (Teorema de corectitudine a rezoluției)

Dacă $\square$ se derivează prin rezoluție din $\mathcal{S}$, atunci $\mathcal{S}$ este nesatisfiabilă.

Dem.: Fie $C_1, C_2, \ldots, C_n = \square$ o $\mathcal{S}$-derivare prin rezoluție a lui $\square$. Presupunem că $\mathcal{S}$ este satisfiabilă și fie $e \vDash \mathcal{S}$. Demonstrăm prin inducție după i că:
$$\text{pentru orice } 1 \leq i \leq n,\ e \vDash C_i.$$
Pentru $i = n$, obținem că $e \vDash \square$, ceea ce este o contradicție. Cazul $i = 1$ este evident, deoarece $C_1 \in \mathcal{S}$. Presupunem că $e \vDash C_j$ pentru orice $j < i$. Avem două cazuri:

- $C_i \in \mathcal{S}$. Atunci $e \vDash C_i$.
- există $j, k < i$ a.î. $C_i \in Res(C_j, C_k)$. Deoarece, conform ipotezei de inducție, $e \vDash \{C_j, C_k\}$ aplicăm Propoziția 2.58 pentru a conclude că $e \vDash C_i$.

$\square$

Algoritmul Davis-Putnam (DP)

Intrare: $\mathcal{S}$ mulțime finită nevidă de clauze netriviale.
$i := 1$, $\mathcal{S}_1 := \mathcal{S}$.

Pi.1 Fie x_i o variabilă care apare în $\mathcal{S}_i$. Definim
$$\mathcal{T}_i^1 := \{C \in \mathcal{S}_i \mid x_i \in C\}, \quad \mathcal{T}_i^0 := \{C \in \mathcal{S}_i \mid \neg x_i \in C\}.$$

Pi.2 **if** $(\mathcal{T}_i^1 \neq \emptyset$ și $\mathcal{T}_i^0 \neq \emptyset)$ **then**
$$\mathcal{U}_i := \{(C_1 \setminus \{x_i\}) \cup (C_0 \setminus \{\neg x_i\}) \mid C_1 \in \mathcal{T}_i^1, C_0 \in \mathcal{T}_i^0\}.$$
else $\mathcal{U}_i := \emptyset$.

Pi.3 Definim
$$\begin{array}{lcl}
\mathcal{S}'_{i+1} & := & (\mathcal{S}_i \setminus (\mathcal{T}_i^0 \cup \mathcal{T}_i^1)) \cup \mathcal{U}_i; \\
\mathcal{S}_{i+1} & := & \mathcal{S}'_{i+1} \setminus \{C \in \mathcal{S}'_{i+1} \mid C \text{ trivială}\}.
\end{array}$$

Pi.4 **if** $\mathcal{S}_{i+1} = \emptyset$ **then** $\mathcal{S}$ este satisfiabilă.
else if $\square \in \mathcal{S}_{i+1}$ **then** $\mathcal{S}$ este nesatisfiabilă.
else $\{i := i + 1$; go to Pi.1$\}$.

Algoritmul Davis-Putnam (DP)

$\mathcal{S} = \{\{v_1, \neg v_3\}, \{v_2, v_1\}, \{v_2, \neg v_1, v_3\}\}$. $i := 1$, $\mathcal{S}_1 := \mathcal{S}$.

P1.1 $x_1 := v_3$; $\mathcal{T}_1^1 := \{\{v_2, \neg v_1, v_3\}\}$; $\mathcal{T}_1^0 := \{\{v_1, \neg v_3\}\}$.
P1.2 $\mathcal{U}_1 := \{\{v_2, \neg v_1, v_1\}\}$.
P1.3 $\mathcal{S}_2' := \{\{v_2, v_1\}, \{v_2, \neg v_1, v_1\}\}$; $\mathcal{S}_2 := \{\{v_2, v_1\}\}$.
P1.4 $i := 2$ and go to P2.1.

P2.1 $x_2 := v_2$; $\mathcal{T}_2^1 := \{\{v_2, v_1\}\}$; $\mathcal{T}_2^0 := \emptyset$.
P2.2 $\mathcal{U}_2 := \emptyset$.
P2.3 $\mathcal{S}_3 := \emptyset$.
P2.4 $\mathcal{S}$ este satisfiabilă.

Algoritmul Davis-Putnam (DP)

$\mathcal{S} = \{\{\neg v_1, v_2, \neg v_4\}, \{\neg v_3, \neg v_2\}, \{v_1, v_3\}, \{v_1\}, \{v_3\}, \{v_4\}\}$.
$i := 1$, $\mathcal{S}_1 := \mathcal{S}$.

P1.1 $x_1 := v_1$; $\mathcal{T}_1^1 := \{\{v_1, v_3\}, \{v_1\}\}$; $\mathcal{T}_1^0 := \{\{\neg v_1, v_2, \neg v_4\}\}$.
P1.2 $\mathcal{U}_1 := \{\{v_3, v_2, \neg v_4\}, \{v_2, \neg v_4\}\}$.
P1.3 $\mathcal{S}_2 := \{\{\neg v_3, \neg v_2\}, \{v_3\}, \{v_4\}, \{v_3, v_2, \neg v_4\}, \{v_2, \neg v_4\}\}$.
P1.4 $i := 2$ and go to P2.1.

P2.1. $x_2 := v_2$; $\mathcal{T}_2^1 := \{\{v_3, v_2, \neg v_4\}, \{v_2, \neg v_4\}\}$; $\mathcal{T}_2^0 := \{\{\neg v_3, \neg v_2\}\}$.
P2.2 $\mathcal{U}_2 := \{\{v_3, \neg v_4, \neg v_3\}, \{\neg v_4, \neg v_3\}\}$.
P2.3 $\mathcal{S}_3 := \{\{v_3\}, \{v_4\}, \{\neg v_4, \neg v_3\}\}$.
P2.4 $i := 3$ and go to P3.1.

P3.1 $x_3 := v_3$; $\mathcal{T}_3^1 := \{\{v_3\}\}$; $\mathcal{T}_3^0 := \{\{\neg v_4, \neg v_3\}\}$.
P3.2. $\mathcal{U}_3 := \{\{\neg v_4\}\}$. P3.3 $\mathcal{S}_4 := \{\{v_4\}, \{\neg v_4\}\}$.
P3.4 $i := 4$ and go to P4.1.

P4.1 $x_4 := v_4$; $\mathcal{T}_4^1 := \{\{v_4\}\}$; $\mathcal{T}_4^0 := \{\{\neg v_4\}\}$.
P4.2 $\mathcal{U}_4 := \{\square\}$. P4.3 $\mathcal{S}_5 := \{\square\}$.
P4.4 $\mathcal{S}$ nu este satisfiabilă.

Algoritmul DP - terminare

Propoziția 2.60

Fie $n := |Var(\mathcal{S})|$. Atunci algoritmul DP se termină după cel mult n pași.

Dem.: Se observă imediat că pentru orice i,

$$Var(\mathcal{S}_{i+1}) \subseteq Var(\mathcal{S}_i) \setminus \{x_i\} \subsetneq Var(\mathcal{S}_i).$$

Prin urmare, $n = |Var(\mathcal{S}_1)| > |Var(\mathcal{S}_2)| > |Var(\mathcal{S}_3)| > \ldots \geq 0$. $\square$

Fie $N \leq n$ numărul de pași după care se termină DP. Atunci $\mathcal{S}_{N+1} = \emptyset$ sau $\square \in S_{N+1}$.

Algoritmul DP - corectitudine și completitudine

Propoziția 2.61

Pentru orice $i \leq N$,

$$\mathcal{S}_{i+1} \text{ este satisfiabilă} \iff \mathcal{S}_i \text{ este satisfiabilă.}$$

Dem.: Exercițiu suplimentar.

Teorema 2.62

Algoritmul DP este corect și complet, adică,

$$\mathcal{S} \text{ este nesatisfiabilă ddacă } \square \in S_{N+1}.$$

Dem.: Aplicăm Propoziția 2.61. Obținem că $\mathcal{S} = \mathcal{S}_1$ este nesatisfiabilă ddacă S_{N+1} este nesatisfiabilă ddacă $\square \in S_{N+1}$. $\square$

SINTAXA LP

Sistemul deductiv

Folosim un sistem deductiv de tip Hilbert pentru *LP*.

Axiomele logice

Mulțimea *Axm* a axiomelor lui *LP* constă din toate formulele de forma:

(A1) $\varphi \to (\psi \to \varphi)$

(A2) $(\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$

(A3) $(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)$,

unde φ, ψ și χ sunt formule.

Regula de deducție

Pentru orice formule φ, ψ,

din φ și $\varphi \to \psi$ se inferă ψ (modus ponens sau (MP)):

$$\frac{\varphi,\ \varphi \to \psi}{\psi}.$$

Γ-teoreme

Fie Γ o mulțime de formule. Definiția Γ-teoremelor este un nou exemplu de definiție inductivă.

Definiția 2.63

Γ-teoremele sunt formulele lui LP definite astfel:

(T0) *Orice axiomă este Γ-teoremă.*

(T1) *Orice formulă din Γ este Γ-teoremă.*

(T2) *Dacă φ și $\varphi \to \psi$ sunt Γ-teoreme, atunci ψ este Γ-teoremă.*

(T3) *Numai formulele obținute aplicând regulile (T0), (T1), (T2) sunt Γ-teoreme.*

Dacă φ este Γ-teoremă, atunci spunem și că φ este dedusă din ipotezele Γ.

Γ-teoreme

Notații

$Thm(\Gamma)$	$:=$	mulțimea Γ-teoremelor	Thm	$:=$	$Thm(\emptyset)$
$\Gamma \vdash \varphi$	$:\Leftrightarrow$	φ este Γ-teoremă	$\vdash \varphi$	$:\Leftrightarrow$	$\emptyset \vdash \varphi$
$\Gamma \vdash \Delta$	$:\Leftrightarrow$	$\Gamma \vdash \varphi$ pentru orice $\varphi \in \Delta$.			

Definiția 2.64

O formulă φ se numește teoremă a lui LP dacă $\vdash \varphi$.

Reformulând condițiile (T0), (T1), (T2) folosind notația $\vdash$, obținem

Propoziția 2.65

(i) *dacă φ este axiomă, atunci $\Gamma \vdash \varphi$;*

(ii) *dacă $\varphi \in \Gamma$, atunci $\Gamma \vdash \varphi$;*

(iii) *dacă $\Gamma \vdash \varphi$ și $\Gamma \vdash \varphi \to \psi$, atunci $\Gamma \vdash \psi$.*

Γ-teoreme

Definiția Γ-teoremelor dă naștere la metoda de demonstrație prin inducție după Γ-teoreme.

Versiunea 1

Fie $\boldsymbol{P}$ o proprietate a formulelor. Demonstrăm că orice Γ-teoremă satisface $\boldsymbol{P}$ astfel:

(i) Demonstrăm că orice axiomă are proprietatea $\boldsymbol{P}$.

(ii) Demonstrăm că orice formulă din Γ are proprietatea $\boldsymbol{P}$.

(iii) Demonstrăm că dacă φ și $\varphi \to \psi$ au proprietatea $\boldsymbol{P}$, atunci ψ are proprietatea $\boldsymbol{P}$.

Versiunea 2

Fie Σ o mulțime de formule. Demonstrăm că $Thm(\Gamma) \subseteq \Sigma$ astfel:

(i) Demonstrăm că orice axiomă este în Σ.

(ii) Demonstrăm că orice formulă din Γ este în Σ.

(iii) Demonstrăm că dacă $\varphi \in \Sigma$ și $\varphi \to \psi \in \Sigma$, atunci $\psi \in \Sigma$.

Γ-teoreme

Propoziția 2.66

Fie Γ, Δ mulțimi de formule.

(i) Dacă $\Gamma \subseteq \Delta$, atunci $Thm(\Gamma) \subseteq Thm(\Delta)$, adică, pentru orice formulă φ,

$$\Gamma \vdash \varphi \text{ implică } \Delta \vdash \varphi.$$

(ii) $Thm \subseteq Thm(\Gamma)$, adică, pentru orice formulă φ,

$$\vdash \varphi \text{ implică } \Gamma \vdash \varphi.$$

(iii) Dacă $\Gamma \vdash \Delta$, atunci $Thm(\Delta) \subseteq Thm(\Gamma)$, adică, pentru orice formulă φ,

$$\Delta \vdash \varphi \text{ implică } \Gamma \vdash \varphi.$$

(iv) $Thm(Thm(\Gamma)) = Thm(\Gamma)$, adică, pentru orice formulă φ,

$$Thm(\Gamma) \vdash \varphi \;\text{ ddacă } \Gamma \vdash \varphi.$$

Dem.: Exercițiu ușor.

Γ-demonstrații

Definiția 2.67

O Γ-demonstrație (demonstrație din ipotezele Γ) este o secvență de formule $\theta_1, \ldots, \theta_n$ a.î. pentru fiecare $i \in \{1, \ldots, n\}$, una din următoarele condiții este satisfăcută:

(i) θ_i este axiomă;

(ii) $\theta_i \in \Gamma$;

(iii) există $k, j < i$ a.î. $\theta_k = \theta_j \to \theta_i$.

O $\emptyset$-demonstrație se va numi simplu demonstrație.

Lema 2.68

Dacă $\theta_1, \ldots, \theta_n$ este o Γ-demonstrație, atunci

$$\Gamma \vdash \theta_i \;\text{ pentru orice } i \in \{1, \ldots, n\}.$$

Dem.: Exercițiu.

Γ-demonstrații

Definiția 2.69

Fie φ o formulă. O Γ-demonstrație a lui φ sau demonstrație a lui φ din ipotezele Γ este o Γ-demonstrație $\theta_1, \ldots, \theta_n$ a.î. $\theta_n = \varphi$. În acest caz, n se numește lungimea Γ-demonstrației.

Propoziția 2.70

Fie Γ o mulțime de formule și φ o formulă. Atunci $\Gamma \vdash \varphi$ ddacă există o Γ-demonstrație a lui φ.

Propoziția 2.71

Pentru orice mulțime de formule Γ *și orice formulă* φ,

$$\Gamma \vdash \varphi \ \text{ ddacă există o submulțime finită } \Sigma \text{ a lui } \Gamma \ \text{ a.î. } \Sigma \vdash \varphi.$$

Dem.: "$\Leftarrow$" Fie $\Sigma \subseteq \Gamma$, Σ finită a.î. $\Sigma \vdash \varphi$. Aplicând Propoziția 2.66.(i) obținem că $\Gamma \vdash \varphi$.
"$\Rightarrow$" Presupunem că $\Gamma \vdash \varphi$. Conform Propoziției 2.70, φ are o Γ-demonstrație $\theta_1, \ldots, \theta_n = \varphi$. Fie

$$\Sigma := \Gamma \cap \{\theta_1, \ldots, \theta_n\}.$$

Atunci Σ este finită, $\Sigma \subseteq \Gamma$ și $\theta_1, \ldots, \theta_n = \varphi$ este o Σ-demonstrație a lui φ, deci $\Sigma \vdash \varphi$. $\square$