Anexă

Rezolvarea exercițiilor şi exerciții propuse

După cum se va observa, unele rezolvări nu sunt complete, fiind suficiente câteodată doar unele indicații sau chiar doar unele comentarii asupra modului de rezolvare.

§1. Rezolvări Capitolul 1

V.1.1. Chiar fără a utiliza principiul dualității, demonstrațiile nu prezintă dificultăți majore.

V.1.2. Toate relațiile corespunzătoare se regăsesc demonstrate într-un manual de algebră de clasa a IX-a ([DID]). Egalitatea de mulțimi se defineşte ca o dublă incluziune.

V.1.3. Folosirea tabelelor de adevăr nu credem că prezintă dificultăți. Presupunând acum că am demonstrat deja $x + \overline{x} = 1$ şi $x \bullet 1 = x$ (se pot utiliza tabelele de adevăr în cel mai rău caz), precum şi dualele lor, vom construi un raționament pentru a deduce doar relația:

$$8)\ x \bullet x = x.$$

Astfel, găsim imediat:

$$x = x \bullet 1 = x \bullet (x + \overline{x}) = \qquad \text{distributivitate}$$
$$= x \bullet x + x \bullet \overline{x} = x \bullet x + 0 = x \bullet x.$$

V.1.4. Fie două mulțimi oarecare, finite, nevide, A_1 şi A_2, având $|A_1| = n$ şi $|A_2| = m$, $n \in \mathbf{N}^*$. Direct din definiții, se deduce că $|A_1 \times A_2| = m \bullet n$. Mai mult, se ştie că ([DID]) numărul total de funcții de la A_1 la A_2 este m^n (de altfel demonstrația este foarte simplă dacă se face prin inducție după n). Cum |**B**| = 2 și $|\mathbf{B}^n| = 2^n$, trebuie să mai calculăm numărul total de funcții de la $A_1 = \mathbf{B}^n$ la $A_2 = \mathbf{B}$, ceea ce este imediat.

V.1.5. Faptul că g este bijectivă este o consecință imediată a definiției unui termen peste mulțimea $X = \{x_1, x_2, \dots, x_n\}$. Să arătăm că un n-uplu $e = \langle e_1, e_2, \dots, e_n \rangle$ în care, pentru fiecare $i \in [n]$, avem $e_i \in \{0, 1, 2\}$, poate fi interpretat într-un mod unic ca o funcție $f_e : [n] \to \{0, 1, 2\}$. Astfel, definim $f_e(i) = e_i$, pentru fiecare $i \in [n]$. Se aplică apoi un raționament similar cu cel din exercițiul precedent.

V.1.6. Presupunem că X = {x, y, z}, în această ordine. Se aplică direct **Teorema 1.3** și respectiv **1.4**, găsind, următoarea **FNDP**:

$f(x, y, z) = \bar{x} \bullet \bar{y} \bullet z + \bar{x} \bullet y \bullet \bar{z} + x \bullet \bar{y} \bullet \bar{z} + x \bullet \bar{y} \bullet z + x \bullet y \bullet z$.

V.1.7. Fiecare dintre mulțimile considerate conțin pe **E,** care este o mulțime infinită (acest lucru rezultând ușor din definițiile respective). Pentru a arăta că $\mathbf{T_0}$ este o mulțime închisă, este suficient să arătăm (**Teorema 1.5**) că $\mathbf{T_0}$ conține **E** și că *este închisă la superpoziție*. Într-adevăr, după cum deja am observat, $\mathbf{E} \subseteq \mathbf{T_0}$ pentru că $i^n_p(0, 0, \mathbf{K}, \underset{p}{0}, \mathbf{K}, 0) = 0$ indiferent de valorile lui $n \in \mathbf{N^*}$ și $p \in [n]$. Fie acum $n \in \mathbf{N^*}$, $t \in \mathbf{N}$, f, $h_1, h_2, \dots, h_t \in FB^{(n)}$, $g \in FB^{(t)}$, astfel încât $f = SUP(g, h_1, h_2, \dots, h_t)$ și $g, h_1, h_2, \dots, h_t \in \mathbf{T_0}$. Rămâne să arătăm că $f \in \mathbf{T_0}$, adică f(0, 0, ... 0) = 0. Dar, știm că pentru fiecare $x = \langle x_1, x_2, \dots, x_n \rangle$ avem $f(x) = g(h_1(x), h_2(x), \dots, h_t(x))$, de unde, luând $x = \langle 0, 0, \dots, 0 \rangle$, găsim $f(0, 0, \dots 0) = g(h_1(0, 0, \dots, 0), h_2(0, 0, \dots, 0), \dots$

, $h_t(0, 0, ... , 0)) = g(0, 0, ... , 0) = 0$, folosind exact faptul că $h_1, h_2, ..., h_t$ şi g sunt elemente din $\mathbf{T_0}$.

V.1.8. Fie ([CAZ1]) M mulţimea considerată. Trebuie să arătăm că $\overline{M}$ = FB, adică să arătăm că orice funcţie booleană este fie o funcţie elementară, fie o funcţie din M, fie se obţine printr-un număr finit de superpoziţii din funcţiile elementare şi/sau elementele lui M. Desigur că ne vor fi de folos teoremele de reprezentare ale funcţiilor booleene. Pentru a simplifica notaţiile, vom desemna prin n, s şi p funcţiile din M ($\bar{\ }$, + şi respectiv •, acestea neavând o notaţie prefixată). Considerăm funcţiile:

(i) $\mathbf{f_p^\alpha}$**, α ∈ B, date prin** $\mathbf{f_p^\alpha(x_1, ... , x_p, ... , x_m) = x_p^\alpha}$ (oricare ar fi numărul de argumente $m \in$ **N*** şi $p \in [m]$). Avem $f_p^1 = i_p^m$ şi cum $f_p^0(x_1, x_2, ... , x_m) = \overline{i_p^m(x_1,...,x_m)}$, oricare ar fi elementele $x_1, x_2, ... , x_m \in$ **B**, putem scrie şi f_p^0 = SUP(n, i_p^m).

(ii) Fie acum funcţiile $\mathbf{p_k}$**, date prin** $\mathbf{p_k(x_1, x_2, ... , x_m) = x_1 \bullet x_2 \bullet ... \bullet x_k}$, oricare ar fi $m \in$ **N*** şi $k \in [m]$. Observăm acum că $p_1 = i_1^m$, p_2 = SUP(p, i_1^m, i_2^m), ş. a. m. d., adică pentru fiecare $k > 2$ găsim p_k = SUP(p, p_{k-1}, i_k^m).

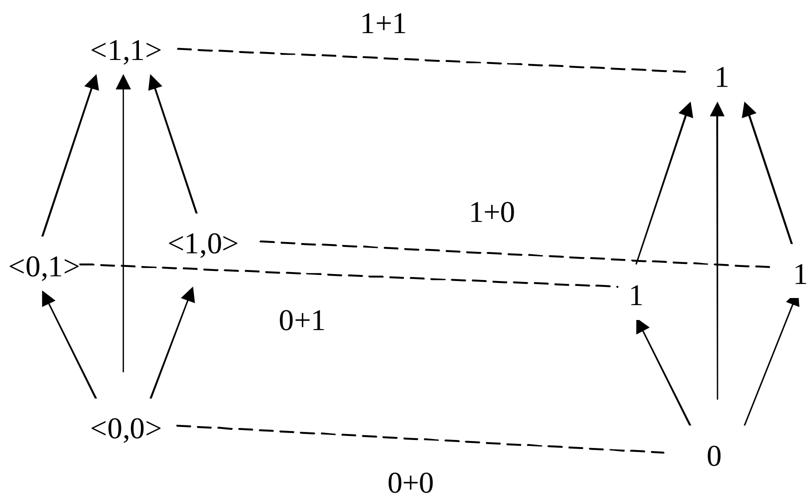
(iii) Considerăm în continuare funcţiile $\mathbf{s_r}$**, date prin** $\mathbf{s_r(x_1, x_2, ... , x_m) = x_1 + x_2 + ... + x_r}$ (pentru fiecare $m \in$ **N*** şi $r \in [m]$), despre care se arată în modul recursiv similar cazului precedent că sunt elemente din $\overline{M}$, ($s_1 = i_1^m$, s_2 = SUP(s, i_1^m, i_2^m) şi, pentru fiecare $r > 2$, s_r = SUP(s, s_{r-1}, i_r^m)).

(iv) Putem trage acum concluzia că şi **funcțiile de forma** $\mathbf{g(x_1, x_2, \ldots, x_m) = x_1^{\alpha_1} \bullet x_2^{\alpha_2} \bullet \ldots \bullet x_m^{\alpha_m}}$ ($m \in$ **N***, $\alpha_1, \alpha_2, \ldots \alpha_m \in$ **B**) sunt în $\overline{\mathrm{M}}$, deoarece $g = \mathrm{SUP}(p_m, f_1^{\alpha_1}, f_2^{\alpha_2}, \ldots, f_m^{\alpha_m})$.

(v) Fie, în sfârşit, **orice funcție booleană f Î FB$^{(m)}$, m Î N***.

- Dacă $f(x_1, x_2, \ldots, x_m) = 0$, pentru fiecare $x_1, x_2, \ldots, x_m \in B$, atunci $f = \mathrm{SUP}(p, f_1^0, f_1^1)$.
- În caz contrar, conform **Teoremei 1.3**, f admite o reprezentare unică în **FNDP**, fiind o sumă de $r \in$ **N*** maxtermeni, adică de funcții $g_1, g_2, \ldots, g_r$ de tipul dat în cazul (iv), adică $f = \mathrm{SUP}(s_r, g_1, g_2, \ldots g_r)$.

V.1.9. Avem **B** × **B** = {<0, 0>, <0, 1>, <1, 0>, <1, 1>}, valorile corespunzătoare ale funcției + (pentru perechile de argumente de mai sus) fiind respectiv 0, 1, 1, 1. Pentru că am pus $0 \leq 1$ (ştim deja că $0 \neq 1$) şi extensia acestei relații la produsul cartezian este *pe componente*, putem spune că doar elementele <0, 1> şi <1, 0> sunt incomparabile, celelalte fiind în relație, după cum este indicat în figura de mai jos prin săgeată continuă (astfel, <0, 1> ≤ <1, 1>, <1, 0> ≤ <1, 1>, etc.). Se observă imediat că valorile funcției păstrează relația de ordine dintre (perechile de) argumente:



V.1.10. Conform [DID], cunoscându-se şi tabelele de definiție ale operațiilor implicate, nu este nici o dificultate de a se arăta ceea ce se cere (dacă **B** şi Z_2 se identifică, atunci izomorfismul este dat chiar de funcția identică).

V.1.11. Să arătăm că funcția specificată nu este liniară (conform celor arătate în lucrare şi exercițiului precedent, există o unicitate a reprezentării funcției ca polinom):

$x \cdot \overline{y} + y \cdot \overline{z} =$

conform $a + b = \overline{\overline{a} \cdot \overline{b}}$

$= \overline{\overline{x \cdot \overline{y}} \cdot \overline{y \cdot \overline{z}}} =$

conform $\overline{\overline{a} \cdot \overline{b}} = (a \oplus 1) \cdot (b \oplus 1) \oplus 1$

$= (x \cdot \overline{y} \oplus 1) \cdot (y \cdot \overline{z} \oplus 1) \oplus 1 =$

conform $\bar{a} = a \oplus 1$

$= (x \bullet (y \oplus 1) \oplus 1) \bullet (y \bullet (z \oplus 1) \oplus 1) \oplus 1 =$

distributivitate, asociativitate

$= (x \bullet y \oplus x \oplus 1) \bullet (y \bullet z \oplus y \oplus 1) \oplus 1 =$

distributivitate, $a \bullet a = a$

$x \bullet y \bullet z \oplus x \bullet y \oplus x \bullet y \oplus x \bullet y \bullet z \oplus x \bullet y \oplus x \oplus y \bullet z \oplus y \oplus 1 \oplus 1 =$

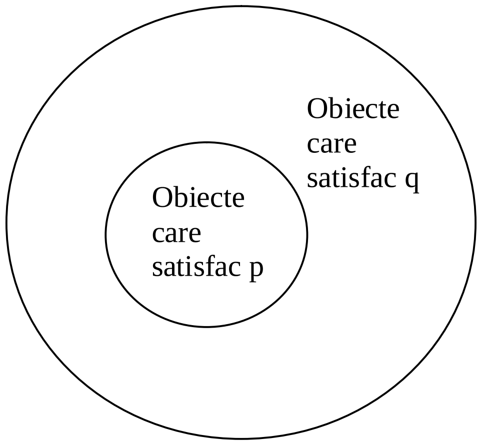
comutativitate, $a \oplus a = 0, a \oplus 0 = a$

$= x \bullet y \oplus y \bullet z \oplus x \oplus y.$

Evident, aceasta nu este o reprezentare liniară.

§2. Rezolvări Capitolul 2

V.2.1. *Teorema directă* (TD) are forma *ipoteze* implică *concluzii*, adică TD: $p \rightarrow q$ (sau $p \Rightarrow q$ dacă este vorba chiar de un raționament sintactic). Atunci avem: TR (*teorema reciprocă*) este $q \Rightarrow p$; TCD (*contrara directei*) este $\rceil p \Rightarrow \rceil q$, iar TCR (*contrara reciprocei*) este $\rceil q \Rightarrow \rceil p$. Se poate arăta că $p \Rightarrow q$ este logic echivalentă (în metalimbaj) cu $\rceil q \Rightarrow \rceil p$ (adică *teorema directă este echivalentă cu contrara reciprocei, de unde se deduce* ***metoda reducerii la absurd***, ca metodă generală de demonstrație). Se schimbă ceva în cele de mai sus dacă avem mai multe ipoteze sau/şi mai multe concluzii (p şi q pot fi formule compuse, între ipotezele sau/şi concluziile elementare putând exista conectori de tipul $\wedge$ sau $\vee$)? O echivalență sintactică $p \Leftrightarrow q$ se mai exprimă sub forma *„Pentru a avea q adevărată este necesar şi suficient să avem p adevărată”*. În această exprimare, $p \Rightarrow q$ este *condiția suficientă* şi $q \Rightarrow p$ este *condiția necesară*. Grafic:



Cu alte cuvinte, pentru a avea q adevărată este suficient ca p să fie adevărată, iar pentru a avea p adevărată, este necesar mai întâi ca formula q să fie adevărată. Ceea ce este reprezentat mai sus sunt sferele noțiunilor caracterizate prin condițiile p şi q, echivalența logică (fie ea de natură semantică, fie sintactică) „spunând” că cele două sfere coincid.

V.2.2. Avem $\lim\limits_{n\to\infty} a_n = a \in R$ dacă şi numai dacă (scrierea este conform manualelor din liceu):

$$(\forall\varepsilon > 0)(\exists n \in N)(\forall m \in N)(m > n \Rightarrow |a_m - a| < \varepsilon)$$

Negația formulei anterioare (nu este adevărat că şirul $(a_n)_{n\in \mathbf{N}}$ converge la $a \in R$; ceea ce nu înseamnă că $(a_n)_{n\in N}$ nu ar fi convergent) este:

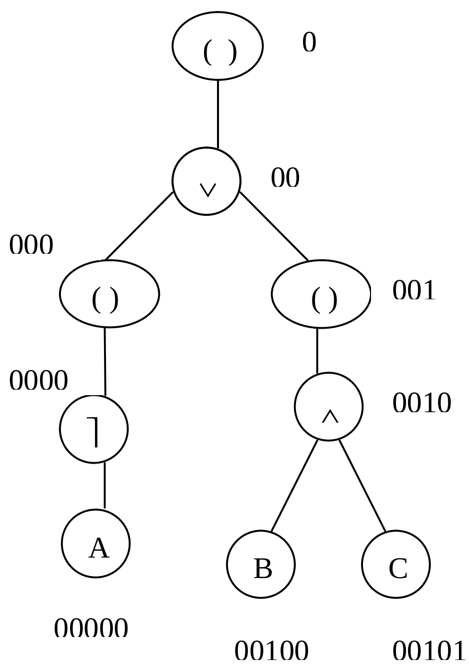
$$(\exists\varepsilon > 0)(\forall n \in \mathbf{N})(\exists m \in \mathbf{N})(\rceil(m > n \Rightarrow |a_m - a| < \varepsilon)) \equiv$$

$$\equiv (\exists\varepsilon > 0)(\forall n \in N)(\exists m \in N)(m > n \wedge |a_m - a| \geq \varepsilon)$$

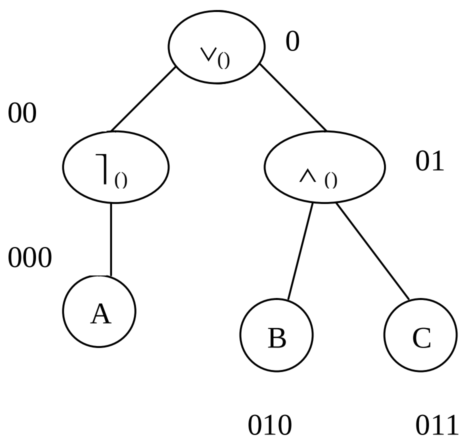
În cele de mai sus s-a folosit faptul că $p \to q \equiv \rceil p \vee q$ şi $\rceil(\rceil p \vee q) \equiv p \wedge \rceil q$.

V.2.3. Notăm de exemplu p: *Mi-e sete* şi q: *Beau apă*. Avem $p \to q \equiv \rceil p \vee q$ şi $\rceil(\rceil p \to q) \equiv \rceil\rceil p \wedge \rceil q \equiv p \wedge \rceil q$. Prin urmare, propoziția în limbaj natural: *„Mi-e sete şi nu beau apă”* exprimă acelaşi lucru cu *„Nu este adevărat că dacă mi-e sete beau apă” (care este negația lui „Dacă mi-e sete, beau apă”)*.

V.2.4. Fie $F = ((\rceil A) \vee (B \wedge C))$. Atunci Arb(F), etichetat (adică, arborele *orientat, ordonat* care reprezintă F), este:



Dacă am fi considerat că notațiile tuturor operatorilor care intervin sunt *prefixate* (nu *infixate*, cum sunt cele pentru $\wedge$ şi $\vee$), parantezele ar fi devenit uşor un accesoriu direct al oricărui operator (aşa cum de altfel mai sugerat). Arborele, în acest caz, ar putea fi uşor simplificat:



Cu ajutorul unei astfel de *structuri de informație* s-ar putea defini simplu, formal (structural), noțiuni cum ar fi: apariția unui simbol într-o formulă F, pe poziția i; subformula lui F care începe la poziția i, etc.

V.2.5. Din demonstrația **Teoremei 2.1** au mai rămas cazurile $F = (F_1 \vee F_2)$ şi $F = (F_1 \wedge F_2)$. Sugerăm cititorului, datorită simplității calculelor, să trateze simultan aceste cazuri, folosind acelaşi simbol atât pentru $\wedge$ şi $\vee$, cât şi, similar , pentru + şi •.

V.2.6. Procedăm prin inducție structurală.

Baza. $F = A \in \mathcal{A}$. Atunci prop(F) = {A}.

Pas inductiv.

(i) $F = (\rceil F_1)$. Evident, prop(F) = prop(F_1). Acelaşi lucru se întâmplă şi atunci când $F = (F_1)$.

(ii) $F = (F_1 \circ F_2)$, $\circ \in \{\wedge , \vee\}$. Atunci prop(F) = prop (F_1) $\cup$ prop (F_2).

Folosind o definiție formală a unei formule considerate ca un arbore binar etichetat (**V.2.4.**), **puteți** defini formal şi *numărul de apariții ale unei variabile*, *împreună cu poziția* la care există aceste apariții?

V.2.7. Se verifică uşor că orice $A \in \mathcal{A}$ este o formulă satisfiabilă, dar nevalidă, că $F = A \vee \rceil A$ este o formulă validă şi că $F = A \wedge \rceil A$ este o contradicție.

V.2.8.

(a) Faptul că relațiile implicate sunt reflexive, simetrice şi tranzitive rezultă imediat, deoarece acestea sunt definite cu ajutorul relației de egalitate.

(b) Mai trebuie arătat că:

- Dacă $F_1 \equiv F_2$, atunci $\rceil F_1 \equiv \rceil F_2$, pentru fiecare $F_1, F_2 \in$ **LP** (compatibilitatea lui $\equiv$ față de $\rceil$), şi că
- Dacă $F_1 \equiv F_2$, atunci $F_1 \vee F \equiv F_2 \vee F$, pentru fiecare $F, F_1, F_2 \in$ **LP** (analog pentru $\wedge$; F poate fi „adăugată” şi la stânga), ceea ce exprimă compatibilitatea la dreapta (respectiv la stânga) a lui $\equiv$ față de $\vee$ (sau $\wedge$).

Să arătăm de exemplu că dacă $F_1, F_2 \in$ **LP** sunt formule oarecare, astfel încât $F_1 \equiv F_2$, atunci pentru oricare $F \in$ **LP** avem: $F \vee F_1 \equiv F \vee F_2$. Fie $\mathcal{S}$ o structură oarecare, corectă pentru F, F_1, F_2. Avem succesiv: $\mathcal{S}(F \vee F_1) = \mathcal{S}(F) + \mathcal{S}(F_1) = \mathcal{S}(F) + \mathcal{S}(F_2) = \mathcal{S}(F \vee F_2)$.

(c) Deoarece $\equiv \subseteq$ **LP** $\times$ **LP** este o echivalență şi $\equiv$ este compatibilă cu $\wedge$, $\vee$ şi $\rceil$, rezultă că pe *mulțimea cât* **LP**/º = {[F] | $F \in$ **LP**}, unde $[F] = \{F' \in$ **LP** $| F' \equiv F\}$, se pot defini *noile operații* Ù, Ú, ù (preferăm să nu folosim simboluri mult diferite pentru **LP**/º):

$$([F_1] \text{ Ù } [F_2]) = [(F_1 \wedge F_2)]$$

$$([F_1] \text{ Ú } [F_2]) = [(F_1 \vee F_2)]$$

(ù $[F_1]) = [(\rceil F_1)]$.

Operațiile (considerate la nivel semantic) sunt într-adevăr funcții tocmai datorită compatibilității lor cu echivalența.

(d) Axiomele unei algebre booleene se arată simplu. **0** reprezintă clasa tuturor tautologiilor, iar **1** este clasa contradicțiilor. Nu există doar un singur homomorfism între mulțimile considerate, ci o infinitate. Astfel, pentru fiecare $\mathcal{S}$: **LP** $\rightarrow$ **B**, putem lua h : **LP**/º $\rightarrow$ **B**, dat prin h([F]) = $\mathcal{S}$(F).

V.2.9. Se pot folosi şi tabelele de adevăr, nu doar definiția.

V.2.10. Avem succesiv:

F ≡

prioritate operatori, distributivitate

$\equiv (\rceil B \vee \rceil D) \wedge \rceil E \wedge \rceil C \wedge B \wedge (\rceil B \vee D) \wedge (\rceil B \vee B) \equiv$

conform ù ***B Ú B º 1 şi A Ù 1 º 1***

$\equiv (\rceil B \vee \rceil D) \wedge \rceil E \wedge \rceil C \wedge B \wedge (\rceil B \vee D)$.

Aplicăm acum **Algoritmul Horn** formulei anterioare, găsind:

$\{B_{*1} \wedge D_{*2} \rightarrow 0,\ E \rightarrow 0,\ C \rightarrow 0,\ 1 \rightarrow B_{*1},\ B_{*1} \rightarrow D_{*2}\}$.

Formula este prin urmare nesatisfiabilă. Vă sugerăm să aplicați şi altă metodă de testare a satisfiabilității unei formule.

V.2.11. Putem considera următoarele variabile propoziționale:

D: *Vom câştiga alegerile.*

E: *Popescu va fi ales liderul partidului.*

G: *Ionescu părăseşte partidul.*

H: *Rădulescu părăseşte partidul.*

Atunci, prima parte a afirmației ar putea fi exprimată prin $E \to D$, iar doua prin $\rceil E \to (G \vee H) \wedge \rceil D$. Întreaga afirmație ar putea fi conjuncția afirmațiilor precedente sau, mai degrabă, ***disjuncția exclusivă*** a lor. După cum ştim, acest operator logic are de fapt acelaşi tabel de adevăr ca şi ***suma modulo 2*** şi va fi notat la fel (tot cu $\oplus$). Deci, $F = (E \to D) \oplus (\rceil E \to (G \vee H) \wedge \rceil D)$. Lăsăm pe seama cititorului finalizarea exercițiului.

V.2.12. Avem succesiv, fără a mai indica explicit toate proprietățile utilizate:

$F \equiv ((\rceil A_1 \vee \rceil A_2) \vee (A_2 \to A_4)) \to ((\rceil A_1 \vee A_2) \wedge (\rceil A_2 \vee A_4)) \equiv$

$\equiv ((\rceil A_1 \vee \rceil A_2) \wedge (\rceil A_2 \vee A_4)) \to ((\rceil A_1 \vee A_2) \wedge (\rceil A_2 \vee A_4)) \equiv$

$\equiv (A_1 \wedge A_1 \wedge \rceil A_4) \vee ((\rceil A_1 \vee A_2) \wedge (\rceil A_2 \vee A_4)) \equiv$

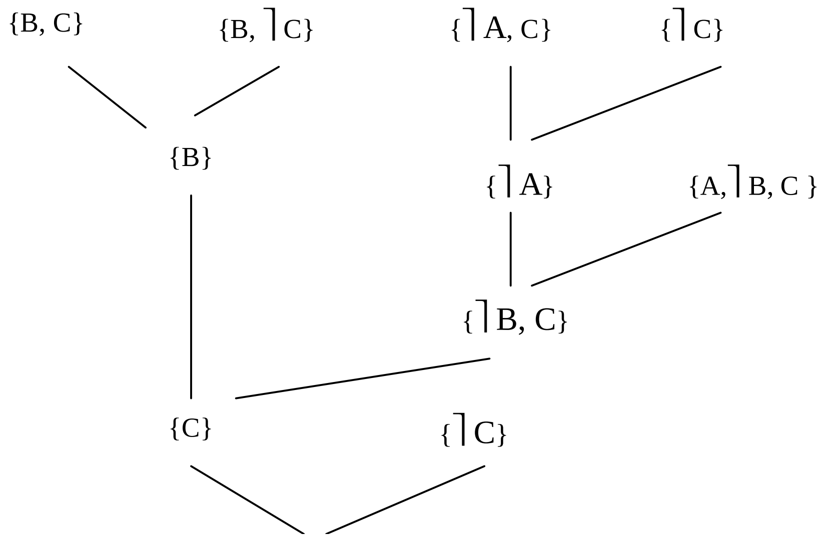
$\equiv (A_1 \wedge A_2 \wedge \rceil A_4) \vee ((\rceil A_1 \vee A_2) \wedge \rceil A_2 \vee (\rceil A_1 \vee A_2) \wedge A_4) \equiv$

$\equiv (A_1 \wedge A_2 \wedge \rceil A_4) \vee ((\rceil A_1 \wedge \rceil A_2) \vee (A_2 \wedge \rceil A_2) \vee (\rceil A_1 \wedge A_4) \vee (A_2 \wedge A_4)) \equiv$

$\equiv A_1 \wedge A_2 \wedge A_4 \vee \rceil A_1 \wedge \rceil A_2 \vee \rceil A_1 \wedge A_4 \vee A_2 \wedge A_4.$

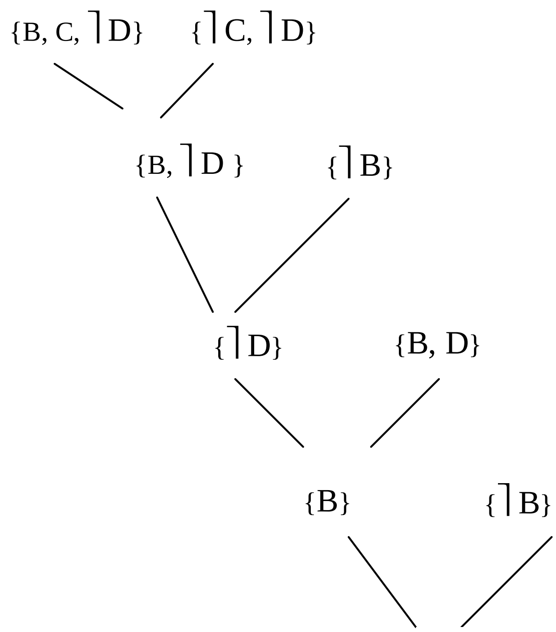
În cele de mai sus, am ținut cont de faptul că $\rceil$ are prioritatea 0, $\wedge$ - 1 şi $\vee$ - 2. În cadrul unei aceleaşi priorități, gruparea am făcut-o însă „la stânga”.

V.2.13. Deşi nu am mai precizat literalii care „se reduc” din fiecare clauză, sperăm ca ei să fie uşor de identificat:

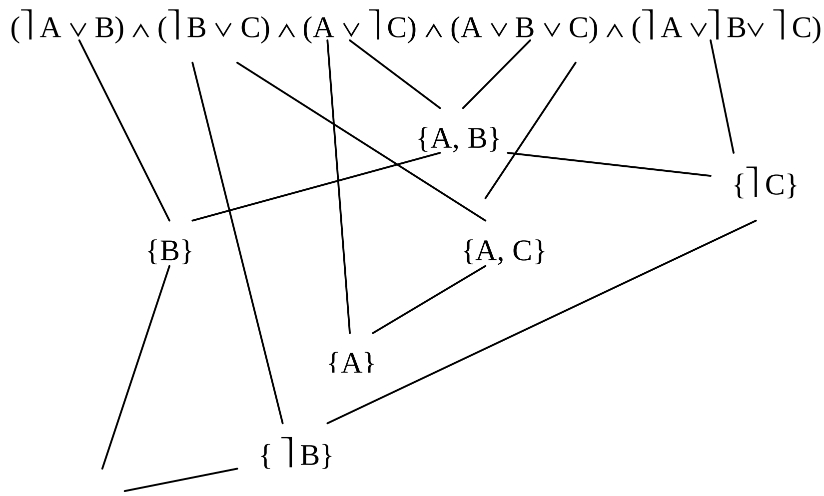


V.2.14. O formulă F este tautologie dacă şi numai dacă $\rceil$ F este contradicţie. Ca urmare, avem: $\rceil F \equiv (B \vee C \vee \rceil D) \wedge (B \vee D) \wedge (\rceil C \vee \rceil D) \wedge \rceil B$, care, în reprezentarea cu mulţimi, înseamnă

$\{\{B, C, \neg D\}, \{B, D\}, \{\neg C, \neg D\}, \{\neg B\}\}$. O respingere ar putea fi:



V.2.15. Ştim că $\mathcal{G} \models G$ dacă şi numai dacă (**Teorema 2.3**) $(\bigwedge_{i=1}^{n} G_i) \wedge \neg G$ este nesatisfiabilă. O respingere poate fi descrisă mai pe scurt şi prin ***graful***:



De fapt, în cele de mai sus, obținerea lui {$\neg$C} trebuia făcută altfel, clauzele alese, {A, B} şi {$\neg$A, $\neg$B, $\neg$C} având practic două posibilități de rezolvare (ceea ce este incorect datorită convențiilor făcute, printre care şi aceea de a nu lua în considerare tautologiile găsite prin rezoluție). Astfel, putem lua {$\neg$A, B} cu {A, $\neg$C} rezultând {B, $\neg$C}. Ultima se rezolvă cu {$\neg$A, $\neg$B, $\neg$C} şi rezultă {$\neg$A, $\neg$C}. În sfârşit, aceasta se rezolvă cu {A, $\neg$C}.

V.2.16. Se observă că definind $S(A_k) = \begin{cases} 1, k = 2i+1, i \in \mathbf{N} \\ 0, k = 2i, i \in \mathbf{N}^* \end{cases}$, avem $S \models M$. Într-adevăr, mulțimile $M' = \{A_1 \vee A_2, A_3 \vee A_4, \ldots, A_{2n+1} \vee A_{2n+2}, \ldots\}$ şi $M'' = \{\neg A_2 \vee \neg A_3, \neg A_4 \vee \neg A_5, \ldots, \neg A_{2n+2} \vee \neg A_{2n+3}, \ldots\}$ *formează o partiție a lui M.* S este model pentru M', pentru că avem

$S(A_{2n+1} \vee A_{2n+2}) = 1$ pentru fiecare $n \in \mathbf{N}$ (deoarece $S(A_{2n+1}) = 1$). Analog, $S(\rceil A_{2n+2} \vee \rceil A_{2n+3}) = \overline{S(A_{2n+2})} + \overline{S(A_{2n+3})} = \overline{0} + \overline{1} = 1$, pentru fiecare $n \in \mathbf{N}$, adică S ‘ M’’.

V.2.17. Demonstrăm doar punctul (b), afirmația corespunzătoare nefiind adevărată. Vom căuta astfel să găsim un contraexemplu. Presupunem astfel că $F \rightarrow G$ este satisfiabilă şi F este satisfiabilă. Este posibil ca G să nu fie satisfiabilă, după cum se observă din următoarea tabelă de adevăr ($F = A \in A$, $G = B \wedge \rceil B$, $B \in A$):

$A = F$	B	$B \wedge \rceil B = G$	$F \rightarrow G$
0	0	0	1
0	1	0	1
1	0	0	0
1	1	0	0

V.2.18. Informațiile avute sunt suficiente pentru ca cititorul să rezolve singur exercițiul.

V.2.19. Din nou, cititorul este invitat „să-şi încerce singur puterile”, rezolvarea neprezentând nici un „truc”.

§3. Rezolvări Capitolul 3

V.3.1. Pentru formule atomice F avem:

(i) Dacă $F = P \in P_0$ atunci free(P) = Ø.

(ii) Dacă $F = P(t_1, t_2,..., t_n)$, unde $n \in \mathbf{N}^*$, $t_1, t_2,..., t_n \in T$, $P \in P_n$, atunci free(P($t_{1,}$ t_2,..., t_n)) = $\bigcup_{i=1}^{n}$ free(t_i) .

Prin urmare, este necesar să definim acum structural free(t), oricare ar fi t ∈ $\mathcal{T}$.

Baza. Dacă t = c ∈ $\mathcal{F}_0$, atunci free(t) = Ø, iar dacă t = x ∈ $\mathcal{X}$, atunci free(t) = {x}.

Pas inductiv. Să presupunem acum că t = f($t_{1,}$ t_2,..., t_n), unde n ∈ $\mathbf{N}^*$, t_1, t_2,..., t_n ∈ $\mathcal{T}$, f ∈ $\mathcal{F}_n$. Atunci free(t) = $\bigcup_{i=1}^{n}$ free(t_i) .

În sfârşit, putem trece la formule:

Baza. Acest pas a fost descris mai sus (F este formulă atomică).

Pas constructiv.

(i) Dacă F = ($\rceil$ F_1), atunci free(F) = free(F_1). La fel, pentru F = (F_1).

(ii) Dacă F = ($F_1 \wedge F_2$), atunci free(F) = free(F_1) ∪ free(F_2).

(iii) Dacă F = ($F_1 \vee F_2$), atunci free(F) = free(F_1) ∪ free(F_2).

(iv) Dacă F = (∀x)(F_1) sau F = (∃x)(F_1), atunci free(F) = free(F_1) \ {x}, doar dacă x ∈ free(F_1). Altfel, free(F) = free(F_1) (desigur că puteam lăsa aceeaşi mulţime de mai sus).

V.3.2. subf(F) = {R(u, f(v)), Q(z), P(x, g(a)), (P(x, g(a)) ∧ Q(z)), ((P(x, g(a)) ∧ Q(z)) ∨ R(u, f(v))), (((P(x, g(a)) ∧Q(z)) ∨ R(u, f(v)))), F}. Puteţi deduce un algoritm imperativ pentru a calcula sub(F), din cel recursiv sugerat de definiţie?

V.3.3. Definirea constructivă a mulţimii leg(F), F ∈ **LP1**, urmează *aceiaşi paşi* ca la definirea mulţimii free(F). Lăsăm în seama cititorului analiza diferenţelor care intervin.

V.3.4. Să presupunem că avem o substituție $s = [x_1/t_1]\bullet[x_2/t_2]\bullet\ldots\bullet[x_n/t_n]$, $n \geq 2$, nenormalizată. Ideea este ca, pentru fiecare $i \in [n - 1]$, termii t_{i+1}, t_{i+2}, ... , t_n să nu mai conțină pe x_i. Atunci putem transforma pe s în felul următor: toate aparițiile variabilei x_n în t_1, t_2, ... , t_{n-1} se înlocuiesc cu t_n, apoi toate aparițiile lui x_{n-1} în (noii) t_1, t_2, ... , t_{n-2} se înlocuiesc cu (noul) t_{n-1}, ş. a. m. d.

V.3.5. $(F)s = (\forall x)(P(x, f(x)) \wedge Q(g(a, h(x))))$. Să remarcăm însă că substituția în cauză, s, nu este permisă pentru F.

V.3.6. A se vedea **V.3.11**.

V.3.7. Formula F este satisfiabilă dar nevalidă. Considerând orice structură $S = \langle U_S, I_S \rangle$, corectă pentru F, avem $F^S = 1$ dacă şi numai dacă $S_{[x/u][y/v]}(P(x, y, f(z))) = 1$, adică dacă şi numai dacă pentru fiecare $u \in U_S$, există $v \in U_S$, astfel încât

$$P^{S_{[x/u][y/v]}}(x^{S_{[x/u][y/v]}}, y^{S_{[x/u][y/v]}}, f^{S_{[x/u][y/v]}}(z^{S_{[x/u][y/v]}}))) = 1.$$

Acest lucru are loc dacă şi numai dacă pentru fiecare $u \in U_S$, există $v \in U_S$, astfel încât $P^S(u, v, f^S(z^S)) = 1$. Continuarea (şi finalizarea) exercițiului este de acum simplă.

V.3.8. $F = (\forall x)(\forall y)(P(x, y) \rightarrow \neg P(y, x))$. Dar dacă P ar fi şi reflexivă şi formula ar aparține lui **LP1**$_=$?

V.3.9. $F = (\forall x)(\forall y)(\forall z)(P(x, x) \wedge (P(x, y) \rightarrow P(y, x)) \wedge ((P(x, y) \wedge P(y, z)) \rightarrow P(x, z))$.

V.3.10. $F^S = 1$ dacă şi numai dacă există $u, v, w \in U_S$ astfel încât avem

$P^{S_{[x/u][y/v][z/w]}}(x, y) = 1$ şi $P^{S_{[x/u][y/v][z/w]}}(z, y) = 1$ şi $P^{S_{[x/u][y/v][z/w]}}(x, z) = 1$ şi $P^{S_{[x/u][y/v][z/w]}}(z, x) = 0$, adică dacă şi numai dacă există u, v, w $\in U_S$ astfel încât:

$P^S(u, v) = 1$ şi $P^S(w, v) = 1$ şi $P^S(u, v) = 1$ şi $P^S(w, u) = 0$.

Finalizarea rezolvării exercițiului este lăsată pe seama cititorului.

V.3.11. Afirmațiile se demonstrează a fi adevărate prin aplicarea directă a definițiilor notiunilor care sunt implicate.

V.3.12. Cu acest exercițiu vă puteți verifica singuri abilitatea de a manipula definițiile constructive.

V.3.13. Se cunoaşte faptul că un grup poate fi definit şi în alte moduri, folosind relații în care nu apare explicit simbolul elementului neutru, e (a se vedea, de exemplu, [DID]). Putem „construi” formula F, pornind de la aceste relații. În acest fel, nu vom mai avea apariții ale elementului neutru (simbolului de constantă) e. Absența simbolului de egalitate poate provoca alte inconveniente. A se vedea şi **Capitolul 4**.

V.3.14. A se vedea **V.3.8**.

V.3.15. Rezolvarea nu poate prezenta dificultăți dacă materia anterioară a fost receptată corect.

V.3.16. Aceeaşi indicație ca la exercițiul precedent.

V.3.17. Procedăm algoritmic după cum urmează:

(i) **Pasul 1.** Obținere **FNPR**.

$F \equiv (\forall u)(\exists y)((P(u, g(y), z) \vee \rceil(\forall x)Q(x)) \wedge (\forall z)(\exists x)\rceil R(f(x, z), z))$

$\equiv (\forall u)(\exists y)((P(u, g(y), z) \vee \rceil(\forall x)Q(x)) \wedge (\forall w)(\exists x)\rceil R(f(x, w), w))$

$\equiv (\forall u)(\exists y)((P(u, g(y), z) \vee \rceil(\forall v)Q(v)) \wedge (\forall w)(\exists x)\rceil R(f(x, w), w))$

$\equiv (\forall u)(\exists y)((P(u, g(y), z) \vee (\exists v)\rceil Q(v)) \wedge (\forall w)(\exists x)\rceil R(f(x, w), w))$

$\equiv (\forall u)(\exists y)(\exists v)(\forall w)(\exists x)((P(u, g(y), z) \vee \rceil Q(v)) \wedge \rceil R(f(x, w), w))$.

(ii) **Pasul 2.** Cum z este unica variabilă cu apariții libere, putem obține ușor o **FNPR** închisă, slab echivalentă cu F:

$F \equiv_s (\exists z)(\forall u)(\exists y)(\exists v)(\forall w)(\exists x)((P(u, g(y), z) \vee \rceil Q(v)) \wedge \rceil R(f(x, w), w))$.

(iii) **Pasul 3.** Aplicăm **Algoritmul Skolem** ultimei forme a lui F, pentru a obține o **FNS** (închisă). Separând execuțiile corpului buclei, obținem succesiv:

- Alege „b" constantă și elimină $(\exists z)$:
  $F \equiv_s (\forall u)(\exists y)(\exists v)(\forall w)(\exists x)((P(u,g(y),b) \vee \rceil Q(v)) \wedge \rceil R(f(x,w),w))$.
- Alege h, un simbol funcțional de aritate 1 și elimină $(\exists y)$:
  $F \equiv_s (\forall u)(\exists v)(\forall w)(\exists x)((P(u,g(h(u)),b) \vee \rceil Q(v)) \wedge \rceil R(f(x, w), w))$.
- Alege t, un simbol funcțional de aritate 1 și elimină $(\exists v)$:
  $F \equiv_s (\forall u)(\forall w)(\exists x) ((P(u, g(h(u)), b) \vee \rceil Q(t(u))) \wedge \rceil R(f(x, w), w))$.
- Alege s, un simbol funcțional de aritate 2 și elimină $(\exists x)$:
  $F \equiv_s (\forall u)(\forall w)((P(u,g(h(u)),b) \vee \rceil Q(t(u))) \wedge \rceil R(f(s(u,w), w), w))$.

(iv) **Pasul 4.** Formula este deja în **FNSC** (închisă).

V.3.18. Fie formula:

$F = (\forall x)(P(x, f(x))) \wedge (\forall y)(\rceil P(y, y)) \wedge (\forall u)(\forall v)(\forall w)((P(u, v) \wedge P(v, w)) \rightarrow P(u, w))$.

F „ne spune" că P este o relație binară, tranzitivă și nereflexivă, având o proprietate suplimentară, exprimată prin subformula $F_1 = (\forall x)(P(x, f(x)))$. Arătăm că, deși F este satisfiabilă, ea nu admite nici un model finit.

(i) **F este satisfiabilă**. Fie structura $S = <U_S, I_S>$, unde U_S = **N**, $P^S = \{<m, n>| m, n \in \mathbf{N}$ cu $m<n\}$, $f^S: U_S \rightarrow U_S$, dată prin $f^S(n) = n + 1$. Pentru că formula nu conține nici constante şi nici variabile libere, structura este corectă şi avem imediat că S ‘ F.

(ii) **F nu poate avea model finit**. Folosind metoda reducerii la absurd, să presupunem că există $S = <U_S, I_S>$ corectă, finită (card(U_S) < χ_o) şi S ‘ F. Deoarece orice univers este presupus a fi nevid, să fixăm un element oarecare $u \in U_S$. Putem acum considera următoarea secvență, infinită, de elemente din U_S, $u_0, u_1, u_2, \ldots$ unde:

$$(1) \qquad \begin{cases} u_0 = u \\ u_{i+1} = f^S(u_i), \ \textit{pentru orice } i \in \mathbf{N} \end{cases}$$

Cum U_S este finită, nu toate elementele din secvență pot fi distincte. Prin urmare, trebuie să existe i, j ∈ **N**, i < j astfel încât $u_i = u_j$. Din S ‘ F, subformula F_1 trebuie să fie adevărată şi de aici rezultă că trebuie să avem:

$<u_0, u_1> \in P^S$,

$<u_1, u_2> \in P^S$,

………………

$<u_{i-1}, u_i> \in P^S$,

$<u_i, u_{i+1}> \in P^S$,

………………

$<u_{j-1}, u_j> \in P^S$,

………………

Considerând subformula lui F,

$F_3 = (\forall u)(\forall v)(\forall w)((P(u,v) \wedge P(v, w)) \rightarrow P(u, w))$

şi ea trebuie să fie adevărată în structura dată. Rezultă că P^S trebuie să fie o relație tranzitivă pe domeniul în cauză şi deci trebuie să avem $<u_i, u_j> \in P^S$. Dar $u_i = u_j$ şi atunci $<u_i, u_i> \in P^S$. Prin urmare, există măcar un element – şi anume u_i – în U_S astfel încât $<u_i, u_i> \in P^S$ şi asta înseamnă că subformula lui F, $F_2 = (\forall y)(\rceil P(y, y))$ este falsă în structura dată. Deducem că F este falsă în S (contradicție).

V.3.19. Lăsăm rezolvarea pe seama cititorului.

§4. Rezolvări Capitolul 4

V.4.1. Avem succesiv, pentru (i), (ii), (iii) (cifrele din secvență sunt locale rezolvării punctelor respective):

Pentru (i):

1. $A\rightarrow B, A \vdash A\rightarrow B, A$	*evident*
2. $A\rightarrow B, A \vdash B$	1., **(MP)**
3. $A\rightarrow B, B\rightarrow C, A \vdash B, B\rightarrow C$	2., *de două ori*
4. $A\rightarrow B, B\rightarrow C, A \vdash C$	3., **(MP)**
5. $A\rightarrow B, B\rightarrow C \vdash A\rightarrow C$	4., **(TD)**
6. $A\rightarrow B \vdash (B\rightarrow C) \rightarrow (A\rightarrow C)$	5., **(TD)**
7. $\vdash (A\rightarrow B)\rightarrow ((B\rightarrow C) \rightarrow (A\rightarrow C))$	6., **(TD)**.

Pentru (ii):

1. $A\to(B\to C), A \bullet A\to(B\to C), A$	*evident*
2. $A\to(B\to C), A \bullet B\to C$	1., **(MP)**
3. $A\to(B\to C), A, B \bullet B, B\to C$	2.
4. $A\to(B\to C), A, B \bullet C$	3., **(MP)**
5. $A\to(B\to C), B \bullet A\to C$	4., **(TD)**
6. $A\to(B\to C) \bullet B\to(A\to C)$	5., **(TD)**
7. $\bullet (A\to(B\to C))\to (B\to(A\to C))$	6., **(TD)**.

Pentru (iii):

1. $\bullet (\rceil A \to \rceil\rceil A)\to((\rceil A \to\rceil A) \to A)$	*axioma 3. din* **SD3**
2. $\bullet \rceil A\to\rceil A$	*am arătat deja*
3. $\bullet (\rceil A \to \rceil A)\to((\rceil A \to\rceil\rceil A) \to A)$	1., (ii), **(MP)**
4. $\bullet (\rceil A \to \rceil\rceil A)\to A$	2., 3., **(MP)**
5. $\bullet \rceil\rceil A \to (\rceil A\to\rceil\rceil A)$	*axioma 1. din* **SD3**
6. $\bullet \rceil\rceil A \to A$	4., 5., (i), **(MP)**.

V.4.2. Similar cu rezoluția propozițională, vom avea câte un sistem pentru fiecare $F \in$ **LP1**, schimbându-se doar mulțimea de formule suplimentare, care este sugerată de E'(F).

V.4.3. Lăsăm rezolvarea pe seama cititorului, ea neimplicând nici un „truc".

V.4.4. Avem succesiv:

1. *I* • A→B	*evident*
2. *I*, A • A, A→B	1.
3. *I*, A • B	2., **(EI)**
4. *I*, A, ⌉B • B, ⌉B	3.
5. *I*, ⌉B • ⌉A	4., **(IN)**
6. *I* • ⌉B → ⌉A	5., **(TD)**, **(II)**.

Apoi:

1. *I* • ⌉B→⌉A	*evident*
2. *I*, ⌉B • ⌉B, ⌉B→⌉A	1.
3. *I* • ⌉B→⌉A	2. **(EI)**
4. *I*, A, ⌉B • A, ⌉A	3.
5. *I*, A • B	4., **(EN)**
6. *I* • A → B	5., **(TD)**, **(II)**.

În sfârşit:

1. *I* • A∨B, A→C, B→C	*evident*
2. *I* • A∨B, ⌉C→⌉A, B→C	1., Ax3
3. *I* • A∨B, ⌉C→⌉A, ⌉C→⌉B	2., Ax3
4. *I*, ⌉C • A∨B, ⌉C, ⌉C→⌉A, ⌉C→⌉B	3.
5. *I*, ⌉C • ⌉A, A∨B, ⌉C, ⌉C→⌉B	4., **(EI)**

6. /, ⌉C • ⌉B, ⌉C, ⌉C→⌉B 5., **(ED)**

7. /, ⌉C • B, ⌉B 6., **(EI)**

8. / • C 7., **(EN)**.

V.4.5.

$$A \vee C, B \Rightarrow B, D, C \qquad A \Rightarrow B \vee D, A, C \qquad C \Rightarrow B \vee D, A, C$$

$$A \vee C, B \Rightarrow B \vee D, C \qquad A \vee C \Rightarrow B \vee D, A, C$$

$$(A \vee C) \wedge (A \rightarrow B), D \Rightarrow B, D \qquad A \vee C, A \rightarrow B \Rightarrow B \vee D, C$$

$$(A \vee C) \wedge (A \rightarrow B), D \Rightarrow B \vee D \qquad (A \vee C) \wedge (A \rightarrow B) \Rightarrow B \vee D, C$$

$$(A \vee C) \wedge (A \rightarrow B), C \rightarrow D \Rightarrow B \vee D$$

$$(A \vee C) \wedge (A \rightarrow B) \wedge (C \rightarrow D) \Rightarrow B \vee D$$

$$\Rightarrow (A \vee C) \wedge (A \rightarrow B) \wedge (C \rightarrow D) \rightarrow (B \vee D)$$

V.4.6. Avem succesiv:

1. $(\forall x)(\forall y)(x = y \rightarrow y = x)$ *conform 7. (din enunț)*
2. $(\forall y)(\forall z)(z = y \rightarrow y = z)$ 1., **(E")**

3. $(\forall z)(\forall v)(z=v \rightarrow v=z)$ 2., **(E'')**
4. $(\forall z)(\forall v)(z=v \rightarrow v=z)$ 3., **(I'')**
5. $(\forall u)(\forall v)(u=v \rightarrow v=u)$ 4., **(E'')**.

V.4.7. Lăsăm rezolvarea pe seama cititorului (din nou, nu există „trucuri”, deşi rezolvarea este cumva plictisitoare prin lungimea ei).

V.4.8. Conform **Exemplului 4.3**, trebuie să adăugăm:

$(\forall x)(\forall y)(\forall z)((x\bullet y)\bullet z = x\bullet(y\bullet z))$ şi

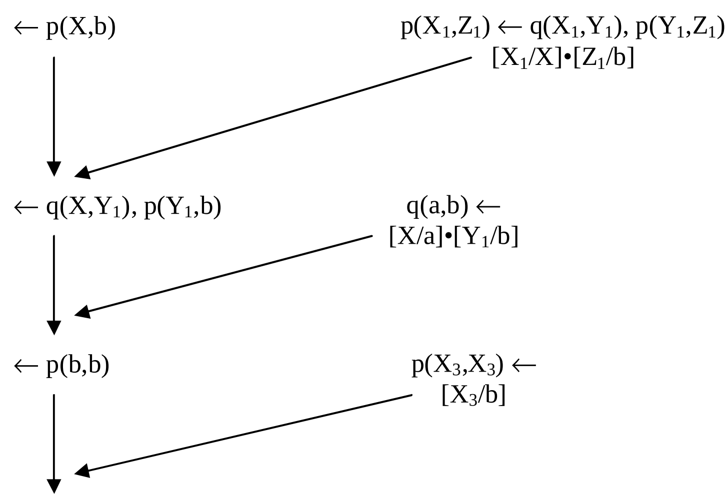
$(\forall y)(\exists x)(x\bullet y = y) \wedge (\forall x)(\exists y)(x\bullet y = x)$.

§5. Rezolvări Capitolul 5

V.5.1. Putem nota clauzele cu $C_1 = \{A(x, 0, x)\}$, $C_2 = \{\rceil A(x, y, z), A(x, s(y), s(z)\}$, $C_3 = \{\rceil A(s(s(s(0))), s(s(0)), u)\}$. Găsim uşor E'(F) după ce s-a calculat D(F). O respingere folosind rezoluția de bază este descrisă prin:

- Substituind $x = s(s(s(0)))$, $y = s(0)$, $z = s(s(s(s(0))))$, $u = s(s(s(s(s(0)))))$ în C_2 şi C_3, rezolvăm C_2 cu C_3 pentru a obține $C_4 = \{\rceil A(s(s(s(0))), s(0), s(s(s(s(0)))))\}$.
- Substituind $x = s(s(s(0)))$, $y = 0$, $z = s(s(s(0)))$ în C_2, rezolvăm C_4 cu C_2 pentru a obține $C_5 = \{\rceil A(s(s(s(0))), 0, s(s(s(0))))\}$.
- În sfârşit, substituind $x = s(s(s(0)))$, obținem prin rezolvarea lui C_5 cu C_1.

V.5.2. Substituția finală va fi s = [X/U]•[Z/b]•[U/c]•[Y/b]•[X/b]. Ca o concluzie, $(\exists U)p(U, b)$ este consecință semantică din $CP1 \wedge CP2 \wedge CP3$, valoarea cerută a lui U fiind c. În arborele de mai jos, nodurile suplimentare (care se rezolvă cu clauza scop curentă, precedată şi de simbolul ←) sunt chiar clauzele de program, nu doar capul acestora (pentru a ne familiariza şi cu alte notații folosite în literatura de specialitate). Numele unor variabile sunt de asemenea schimbate (aşa cum am mai amintit, sunt necesare cunoştințe suplimentare pentru a ne convinge de ce anumite redenumiri nu pot fi evitate pe parcurs).



V.5.3. Posibile definiții sunt (semnificația intuitivă a predicatelor folosite a fost deja sugerată; de exemplu, faptele din minus1 şi minus2 ne „spun” că X – 0 = X şi respectiv X – X = 0):

```
minus1a(X, Y, Z):- plus(Y, Z, X).
minus1b(X, Y, Z):- plus(Z, Y, X).
minus2a(X, Y, Z):- plus(Y, Z, X).
minus2b(X, Y, Z):- plus(Z, Y, X).
```

Am putea lua în considerare şi o definiție recursivă directă de genul

minus1(X, 0, X).

minus1(s(X), s(Y), Z):-minus1(X, Y, Z).,

sau

minus2(X, X, 0).

minus2(X, Y, s(Z)):-minus2(X, s(Y), Z).

Înmulțirea poate fi reprezentată de predicatul:

inmulțire(0, X, 0).

inmulțire(s(X), Y, Z):-inmulțire(X, Y, W), plus1(W, Y, Z).

La rândul său, ridicarea la putere se defineşte sub forma unei înmulțiri repetate (să precizăm încă o dată că predicatul exp(N,X,Y) este adevărat dacă si numai dacă $Y = X^N$):

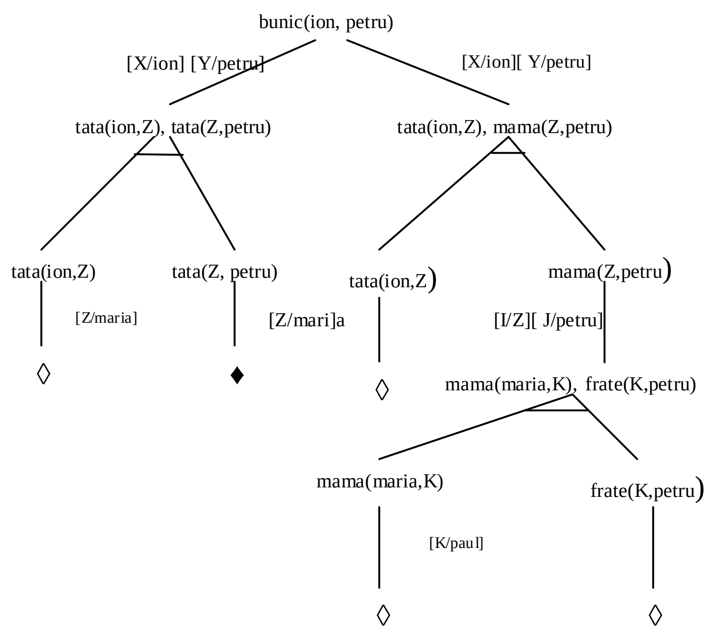
exp(U, 0, 0).

exp(0, V, s(0)).

exp(s(N), X, Y):-exp(N, X, Z), inmulțire(Z, X, Y).

V.5.4. Execuția programului începe cu scopul curent, dat de interogarea inițială. Aceasta conține un singur literal, sora_lui(eliza, andrei). Există o singură clauză program cu al cărui cap literalul ales se poate unifica, şi anume clauza suplimentară (ceea ce se află în dreapta simbolului se mai numeşte şi ***corpul regulii***). Substituția de unificare este [X/eliza]•[Y/andrei], rezultând noul scop curent (***scopul derivat***) ? :- femeie(eliza), parinti(eliza, M, T), parinti(andrei, M, T), care are trei ***subscopuri***. Primul este femeie(eliza), care se unifică cu faptul corespunzător, rezultând ? :- parinti(eliza, M, T), parinti(andrei, M, T). Subscopul parinti(eliza, M, T) nu se poate unifica decât cu faptul parinti(eliza, maria, paul), unde M este instanțiat de valoarea „maria”, iar T, de valoarea „paul” (un cel mai general unificator fiind evident dat de [M/maria]•[T/paul]. Astfel, scopul curent devine ? :- parinti(andrei, maria, paul). Acesta se unifică cu faptul corespunzător din baza de date. Prin urmare, **PROLOG**-ul a reuşit să obțină o respingere, răspunzând „DA” şi găsind valorile cerute (chiar dacă nu explicit).

V.5.5. Arborele **AND-OR** este reprezentat în figura de mai jos. Pe arce nu am mai trecut clauze, ci substituțiile corespunzătoare:



V.5.6. L = {P(x,y), P(f(b), g(x)), P(f(z), g(f(z)))}. Începem cu: sub_0 = [], (L)sub_0 = L. Alegem pentru o primă posibilă unificare literalii:

P(**x**, y) şi

P(**f**(b), g(x)).

Găsim sub_1 = [x/f(b)] şi

$(\mathcal{L})sub_1 = \{P(f(b), y), P(f(b), g(f(b))), P(f(z), g(f(z)))\}$.

Acum pot fi aleşi tot primii doi literali:

$$P(f(b), \mathbf{y})$$
$$P(f(b), \mathbf{g}(f(b))).$$

Găsim $sub_2 = [y/g(f(b))]$ şi

$(\mathcal{L})sub_1 \bullet sub_2 = \{P(f(b), g(f(b))), P(f(z), g(f(z)))\}$.

Nu putem alege acum decât singurii literali rămaşi în mulțime:

$$P(f(\mathbf{b}), g(f(b)))$$
$$P(f(\mathbf{z}), g(f(z))).$$

Punând $sub_3 = [z/b]$, găsim în final

$(\mathcal{L})sub_1 \bullet sub_2 \bullet sub_3 = \{P(f(b), g(f(b)))\}$, substituția cerută (finală) fiind evident $sub = sub_1 \bullet sub_2 \bullet sub_3$.

V.5.7. Putem considera următoarele relații (predicate):

P(X): X este un dragon.

Q(X): X este fericit.

R(X): X este verde.

S(X): X poate zbura.

T(X,Y): X este copilul lui Y.

Lumea în cauza se poate reprezenta prin formulele:

- $F_1 = (\forall Y)((P(Y) \wedge (\forall X)((T(X,Y) \wedge P(X) \rightarrow S(X))) \rightarrow Q(Y))$.
- $F_2 = (\forall Y)((P(Y) \wedge R(Y)) \rightarrow S(Y))$.
- $F_3 = (\forall X)((P(X) \wedge (\exists Y)(T(X,Y) \wedge P(Y) \wedge R(Y))) \rightarrow R(X))$.

Trebuie demonstrat că formula:

$G = (\forall Y)((P(Y) \wedge R(Y)) \rightarrow Q(Y))$

este o consecință semantică din $\{F_1, F_2, F_3\}$, utilizând rezoluția. Urmăm pașii cunoscuți:

(i) **Găsim mai întâi** o formulă F' (aflată în **FNS** închisă şi cu matricea în **FNC**), slab echivalentă cu $\neg F = \neg(F_1 \wedge F_2 \wedge F_3 \rightarrow G)$. Obținem reprezentarea clauzală sub formă de mulțimi a lui F' ca fiind:

$F' = \{\{\neg P(Y), Q(Y), T(f(Y,Y))\}, \{\neg P(Y), Q(Y), P(f(Y))\},$

$\{\neg P(Y), Q(Y), \neg S(f(Y))\}, \{\neg P(Y), \neg R(Y), S(Y)\},$

$\{\neg P(X), \neg T(X,Y), \neg P(Y), \neg R(Y), R(X)\}, \{P(b)\}, \{R(b)\}, \{\neg Q(b)\}\}.$

În cele de mai sus, primele trei clauze sunt din F_1, cea de-a patra este din F_2, cea de-a cincea din F_3, iar ultimele sunt din $\neg G$. Le putem atunci denota prin $C_1, \ldots, C_8$, în această ordine. Mai mult, din procesul de Skolemizare, rezultă două (noi) simboluri funcționale, f – de aritate 1 şi b – de aritate 0.

(ii) **O posibilă respingere** este:

(1) C_1 C_2

$L = (\{\neg P(y),$
$\neg P(a)\})[y/a]$

$C_1' = \{Q(a),\ T(f(a),\ a))\}$

(2) C_2 C_6

$L = (\{\neg P(y), \neg P(a)\})[y/a]$

$C_2^{'} = \{Q(a), P(f(a))\}$

(3) C_3 C_6

$L = (\{\neg P(y), \neg P(a)\})[y/a]$

$C_3^{'} = \{Q(a), \neg S(f(a))\}$

(4) $C_1^{'}$ C_8

$C_1^{''} = \{T(f(a), a)\}$

(5)

$$\begin{tikzcd}[column sep=small] C_2' \arrow[dr, no head] & & C_8 \arrow[dl, no head] \\ & C_2'' = \{P(f(a))\} & \end{tikzcd}$$

(6)

$$\begin{tikzcd}[column sep=small] C_3' \arrow[dr, no head] & & C_8 \arrow[dl, no head] \\ & C_3'' = \{\neg S(f(a))\} & \end{tikzcd}$$

(7)

$$\begin{tikzcd}[column sep=small] C_1 \arrow[dr, no head] & & C_5 \arrow[dl, no head] \\ & C_9 = \{\neg P(z), Q(z), \neg P(f(z)), \neg T(f(z),z), \neg R(z), R(f(z))\} & \end{tikzcd}$$

(redenumire în C_1: [y/z])
L = ({T(f(z),z), T(x, y)})
[x/f(z)][y/z]

(8) C_9 C_1''

$\mathcal{L} = (\{T(f(z),z), T(f(a),a)\})$
$[z/a]$

$\{\rceil P(a), Q(a), \rceil P(f(a)), \rceil R(a), R(f(a))\}$ C_2''

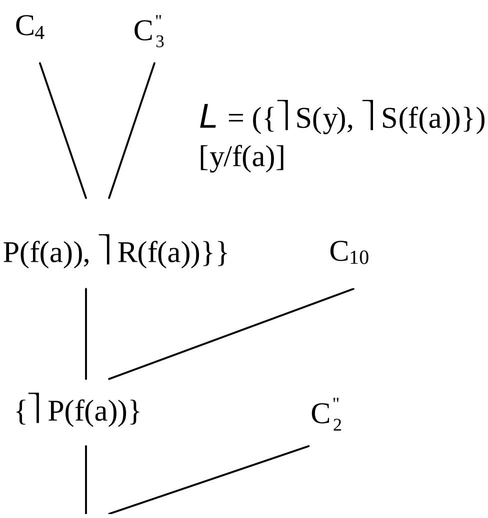
$\{\rceil P(a), Q(a), \rceil R(a), R(f(a))\}$ C_8

$\{\rceil P(a), \rceil R(a), R(f(a))\}$ C_7

$\{\rceil P(a), R(f(a))\}$ C_6

$C_{10} = \{R(f(a))\}$

(9)



Puteți folosi ca alternativă rezoluția de bază ? Raspundeți singuri la celelalte întrebări din enunț.

§6. Exerciții propuse

Exercițiile propuse în acest ultim modul sunt destinate unei ***autoevaluari*** pentru fiecare cititor şi *nu sunt şi rezolvate în lucrare (exceptând unele indicații care sunt furnizate „pe loc”)*, pentru a nu exista tentația de copiere a rezolvărilor (fie şi parțial). Se urmăreşte astfel atât o trecere finală în revistă a cantității/calității informațiilor asimilate, precum şi testarea capacității de a lucra la nivel global cu aceste cunoştințe. Dacă fiecare rezolvare ar fi apreciată cu un punctaj de la 1 la 10, *realizarea a 75 de puncte este o cerință minimală pentru a ști că pregătirea este cât de cât satisfăcătoare*.

V.F.1. Determinați toate funcțiile autoduale având (minim 1 şi) maxim 4 argumente.

V.F.2. Aduceți la **FNC(D)(P)** şi **FNC(D)M** următoarele funcții booleene (am renunțat la introducerea explicită a simbolului •, presupunând în acelaşi timp că acesta este prioritar față de +):

$f(x, y, z) = x + \bar{x}y + \bar{x}\,\bar{y}$

$f(x, y, z) = (x + y)(x + \bar{y})(\bar{x} + \bar{z})(\bar{x} + z)$

$f(u, x, y, z) = u\bar{x} + \bar{y}z$

$f(u, x, y, z) = ux + yz$.

V.F.3. Se ştie că intersecția a două mulțimi închise de funcții booleene este închisă. Ce se poate spune despre reuniune?

V.F.4 ([OHL]). Definiți structural (întâi pentru termi şi pentru formule atomice, apoi pentru formule) următoarele ***funcții recursive*** (*calculabile prin algoritmi*):

- ND : $\mathcal{T} \cup \mathcal{At} \cup$ **LP1** $\to$ **N**. Pentru fiecare F $\in$ **LP1** (t $\in \mathcal{T}$, A $\in \mathcal{At}$), considerăm toate variabilele şi constantele care apar. Pentru fiecare asemenea element, se poate evident determina „*gradul său de imbricare*”, adică numărul total de simboluri disticte (funcționale, predicative, cuantificatori) în domeniul (sintactic al) cărora se află acel element. Valoarea lui ND va fi dată de gradul maxim găsit. De exemplu, ND(x) = 0, ND(f(x, y)) = 1, ND(($\forall$x)(P(f(x) $\wedge$ Q(x))) = 4, etc.

- NS : $\mathcal{T} \cup \mathcal{At} \cup$ **LP1** $\rightarrow$ **N**, unde NS(e) = numărul de simboluri din $\mathcal{Alf}$, distincte de paranteze şi virgule, peste care este construit elementul respectiv (e $\in \mathcal{T} \cup \mathcal{At} \cup$ **LP1**). De exemplu, NS(($\forall$x)P(f(x, x))) = 6.

Demonstraţi apoi prin inducţie structurală că NS(F) $\geq$ ND(F), pentru fiecare F $\in$ **LP1**. *Ce* se modifică în cele de mai sus dacă termii, formulele atomice şi formulele se reprezintă ca arbori?

V.F.5. Ne plasăm în **LP**. Fie F $\rightarrow$ G o tautologie, astfel încât F şi G nu au formule atomice în comun. Arătaţi că fie F este nesatisfiabilă, fie G este tautologie (neexclusiv). Argumentaţi faptul că presupunerea „nu există variabile comune" este esenţială.

V.F.6. Exprimaţi următoarele afirmaţii din matematică sub forma unor formule din **LP1** sau **LP1**$_{=}$. Comentaţi (ne)satisfiabilitatea acestora în diverse structuri.

- x divide y.
- x este număr prim.
- x este cel mai mare divizor comun pentru y şi z.
- x este cel mai mic multiplu comun pentru y şi z.
- Şirul numerelor prime este infinit.
- Mulţimea X are 2 elemente.
- Mulţimea X are n $\geq$ 2, n $\in$ **N** elemente.
- Mulţimea X este parţial ordonată.
- Mulţimea X este bine ordonată.
- Mulţimea {x, y, z} formează un triunghi.

- Mulțimea {x, y, z} formează un triunghi isoscel.
- Dacă două puncte ale unei drepte aparțin unui plan, atunci dreapta aparține planului.

V.F.7. Ne plasăm în **LP1**. Construiți o formă normală Skolem, notată C, pentru formula $B = (\forall x)(\exists y)Q(x, y)$ şi arătați că formulele B şi C nu sunt tare echivalente.

V.F.8. Ne plasăm din nou în **LP1**. Arătați că formula

$F = (\forall x)(\forall y)(\forall z)((Q(x, x)) \wedge ((Q(x, y) \wedge Q(y, z)) \to Q(x, z)) \wedge ((Q(x, y) \vee Q(y, x)) \to (\exists y)(\forall x)Q(y, x))$

este satisfiabilă dar nevalidă. Mai mult, arătați că ea este adevărată în orice univers finit.

V.F.9. Demonstrați, folosind deducția naturală, că:

- $(\forall x)(A) \to (\exists x)A$.
- $A \to (\forall x)A$, dacă x nu apare (intră) liber în A.
- $(\exists x)(A) \to A$, dacă x nu apare liber în A.
- $(\forall x)(B \to A) \leftrightarrow ((\exists x)(B) \to A)$, dacă x nu apare liber în A.

V.F.10. Arătați că următoarea mulțime de literali L din **LP1** este unificabilă şi găsiți un cel mai general unificator:

$L = \{\rceil P(f(z, g(a, y)), h(z)), \rceil P(f(f(u, v), w), h(f(a, b)))\}$.

Ca o indicație, luând doar câte doi literali la fiecare pas de unificare, **Algoritmul lui J. Robinson** se termină după patru paşi, cu un posibil m.g.u. de forma $sub = [z/f(u, v)]\bullet[w/g(a,y)]\bullet[u/a]\bullet[v/b]$.

V.F.11 ([OHL])**. *Problema reacției în lanț a lui Schubert.*** Lupii, vulpile, păsările, omizile şi melcii sunt *animale*, şi există măcar câte un asemenea animal. Există, de asemenea şi cereale, cerealele fiind *plante*. Despre fiecare animal ştim că **fie** îi place să mănânce toate plantele, **fie** îi place să mănânce toate animalele mult mai mici ca el însuşi, animale cărora, în plus, le place să mănânce anumite plante. Omizile şi melcii sunt mult mai mici decât păsările, care sunt mult mai mici decât vulpile, care la rândul lor sunt mult mai mici decât lupii. Lupilor nu le place să mănânce vulpi sau cereale, în timp ce păsărilor le place să mănânce omizi dar nu şi melci. Omizilor şi melcilor le place să mănânce anumite plante. Putem trage concluzia că există (cel puțin) un animal căruia îi place să mănânce un (alt) animal, acestuia din urmă plăcându-i să mănânce cereale?

V.F.12 ([OHL]). Să ne imaginăm o logică propozițională *multivaluată* (de fapt, *trivalentă*), notată, să spunem, cu **LP3**. Mai precis, luăm **LP3** = **LP**, *sintaxa nemodificându-se*. Prin urmare, mulțimea de formule rămâne aceeaşi, cu observația că admitem şi prezența explicită a tuturor conectorilor logici mai des întâlniți, adică a lui non, şi, sau, implică şi echivalent (aceştia vor căpăta însă, pentru evitarea unor confuzii, indicele inferior 3). Notând **B'** = **B** ∪ {u}, o *asignare (structură)* va fi orice funcție $S : A \rightarrow$ **B'**, iar conectorii amintiți vor fi interpretați semantic ca funcții de arități corespunzătoare peste **B'**, funcții definite prin următoarele *tabele de adevăr generalizate*:

x	ù $_3$
0	1

1	0
u	u

x	**y**	**x Ú$_3$ y**	**x Ù$_3$ y**	**x ®$_3$ y**	**x «$_3$ y**
0	0	0	0	1	1
0	1	1	0	1	0
1	0	1	0	0	0
1	1	1	1	1	1
u	0	u	0	0	0
u	1	1	u	1	0
0	u	u	0	1	0
1	u	1	u	u	0
u	u	u	u	1	1

Se observă imediat că dacă restricționăm pe $\rceil_3$, $\vee_3$, $\wedge_3$, $\rightarrow_3$ şi $\leftrightarrow_3$ la **B** (le notăm la fel), atunci acestea coincid cu funcțiile corespunzătoare din semantica logicii propoziționale clasice ($\rceil_3(x) = \overline{x}$, $x \vee_3 y = x + y$, $x \wedge_3 y = x \bullet y$, $x \rightarrow_3 y = \overline{x} + y$ respectiv $x \leftrightarrow_3 y = (\overline{x} + y) \bullet (\overline{y} + x)$). Valoarea u (*nedefinit*, undefined) nu trebuie înțeleasă ca fiind fie adevărat, fie fals, dar nu ştim exact care dintre ele, ci mai degrabă în sensul: „*cineva care are 2.00 metri este cu siguranță înalt* - *1; cineva care are 1.50 metri este cu siguranță mic de statură – 0; cineva care are 1.75 este*...u”. Se observă uşor şi alte lucruri, cum ar fi faptul că există o unică extensie la **LP3** (notată ulterior la fel), care păstrează operațiile, pentru fiecare asignare S, adică **LP3** este extensional. Dar, oare, **câte/care** dintre lucrurile definite/demonstrate pentru **LP** se păstrează în **LP3**? Verificați astfel dacă afirmația ***pentru fiecare structură*** S, ***fiecare*** F,G $\in$ **LP3**, ***avem*** $S(F \rightarrow_3 G) = S((\rceil_3 F) \vee_3 G)$, este adevărată. În caz contrar, găsiți un contraexemplu, sau, modificați corespunzător tabela lui $\rightarrow_3$.

V.F.13 ([OHL])**. *Alice în țara minunilor***. Când Alice a intrat în pădurea uitării, ea nu a uitat chiar totul, ci doar anumite lucruri. De exemplu, își uita destul de des propriul său nume. Totuşi, în cea mai mare parte a timpului, ea uita în care anume zi din săptămână se găseşte în momentul respectiv. Leul şi inorogul erau şi ei vizitatori destul de frecvenți ai pădurii uitării. În plus, ei erau ființe destul de ciudate. Astfel, leul mințea întotdeauna lunea, marțea şi miercurea, în timp ce în restul zilelor spunea adevărul. Inorogul mințea joia, vinerea şi sâmbăta şi spunea adevărul în celelalte zile. Odată, Alice a întâlnit în pădure leul şi inorogul, care se odihneau sub un copac. Ei au făcut următoarele afirmații:

- *Leul*: Ieri a fost una dintre acele zile în care eu spun minciuni.
- *Inorogul*: Şi pentru mine ieri a fost una dintre zilele în care eu spun minciuni.

Folosind aceste informații, Alice, care era o fată deşteaptă, a putut să determine în ce zi din săptămână se afla atunci. Puteți afla şi dumneavoastră?

V.F.14 ([OHL]). Fie formulele din $\mathbf{LP1}_=$:

F = $(\forall y)(\exists x)P(x, f(x, y))$

G = $(\forall x)((x = b) \rightarrow (\exists\, y)(g(y, y) = x))$

H = $(\forall x)(P(x, b) \rightarrow (\exists y)(g(y, y) = x))$.

Arătați că aceste formule sunt satisfiabile dar nevalide, folosind structurile Herbrand.

V.F.15 ([OHL]). Similar cu cazul prezentat în **V.F.12**, să considerăm o *logică propozițională 4-valuată*, construită cu ajutorul conectorilor $\vee_4$, $\wedge_4$, $\rceil_4$, definiți semantic respectiv prin tabelele:

$\vee_4$	1	b	n	0
1	1	1	1	1
b	1	b	b	b
n	1	b	n	0
0	1	b	0	0

$\wedge_4$	1	b	n	0
1	1	b	n	0
b	b	b	n	0
n	n	n	n	n
0	0	0	n	0

şi

$\rceil_4$	
1	0
b	b
n	n
0	1

După cum se poate observa, pentru simplitate, am schimbat uneori modalitatea de reprezentare a unui tabel în cele de mai sus (sperând că nu există dificultăţi de înţelegere). **Intuitiv**, această logică se poate interpreta după cum urmează: *Să presupunem că cineva trimite nişte chestionare unui anumit număr de persoane, cerându-le sa răspundă prin „DA” sau „NU” la o întrebare Q. Rezultatul acestei activităţi este notat cu 1 dacă la toate chestionarele s-a răspuns prin „DA” şi cu 0 dacă la toate s-a răspuns „NU”. Dacă s-au returnat toate chestionarele dar există atât răspunsuri „DA” cât şi răspunsuri „NU”, sau dacă nu s-au returnat toate chestionarele (ci doar o parte, indiferent de răspunsuri), atunci activitatea va fi apreciată cu* b*. Dacă nici un chestionar nu este returnat, rezultatul activităţii va fi notat cu* n*. În situaţia descrisă, ù $_4$ ar putea fi interpretat ca un rezultat al răspunsurilor la întrebarea* non(Q)*, iar Ú$_4$ respectiv Ù$_4$ ar putea fi interpretaţi ca fiind rezultatul răspunsurilor la nişte întrebări de genul Q = Q_1* sau *Q_2 respectiv Q = Q_1* şi *Q_2 (de exemplu, dacă rezultatul* *pentru Q1 este b şi rezultatul pentru Q2 este n, atunci rezultatul pentru Q_1 Ù$_4$ Q_2 este, natural, tot n).*

Demonstraţi, de exemplu, că atât $\wedge_4$ cât şi $\vee_4$ sunt operaţii comutative pe **B''** = **B** ∪ {b, n}. Ce alte lucruri *interesante* mai puteţi deduce pentru această logică?