Programare Logică

Exerciţii recapitulative din prima parte a materiei

Claudia MUREŞAN

Universitatea din Bucureşti, Facultatea de Matematică şi Informatică

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

2019–2020, Semestrul II

Exerciţiul 1. Fie p, q, r variabile propoziţionale. Aplicând tehnica rezoluţiei, să se determine dacă, în logica propoziţională clasică, următoarele enunţuri sunt satisfiabile:

(1) $[(\neg p \rightarrow q) \vee (p \wedge q \wedge \neg r)] \wedge \neg q$;

(2) $[(\neg p \rightarrow q) \vee (p \wedge q \wedge \neg r) \vee r] \wedge (\neg q \vee \neg r)$;

(3) $(p \vee q) \wedge (\neg q \vee r) \wedge (\neg q \vee \neg r) \wedge \neg p$.

Rezolvare: Punem enunţurile în FNC, apoi efectuăm derivări prin rezoluţie.
(1) $[(\neg p \rightarrow q) \vee (p \wedge q \wedge \neg r)] \wedge \neg q \sim [p \vee q \vee (p \wedge q \wedge \neg r)] \wedge \neg q \sim (p \vee q) \wedge \neg q$, întrucât $q \vee (p \wedge q \wedge \neg r) \sim q$ conform absorbţiei. Mulţimea de clauze corespunzătoare enunţului în FNC $(p \vee q) \wedge \neg q$ este $\{\{p, q\}, \{\neg q\}\}$ şi are unica derivare prin rezoluţie: $\dfrac{\{p, \cancel{q}\}, \{\cancel{\neg q}\}}{\{p\}}$, care nu conduce la clauza vidă $\square$, aşadar este satisfiabilă, deci enunţul $[(\neg p \rightarrow q) \vee (p \wedge q \wedge \neg r)] \wedge \neg q$ este satisfiabil.
(2) Ca mai sus, $[(\neg p \rightarrow q) \vee (p \wedge q \wedge \neg r) \vee r] \wedge (\neg q \vee \neg r) \sim [p \vee q \vee (p \wedge q \wedge \neg r) \vee r] \wedge (\neg q \vee \neg r) \sim (p \vee q \vee r) \wedge (\neg q \vee \neg r)$, iar acestei FNC îi corespunde mulţimea de clauze $\{\{p, q, r\}, \{\neg q, \neg r\}\}$, care are două derivări posibile prin rezoluţie: $\dfrac{\dfrac{\{p, \cancel{q}, r\}, \{\cancel{\neg q}, \neg r\}}{\{p, r, \neg r\}}}{\emptyset}$ şi $\dfrac{\dfrac{\{p, q, \cancel{r}\}, \{\neg q, \cancel{\neg r}\}}{\{p, q, \neg q\}}}{\emptyset}$: am eliminat clauzele triviale $\{p, r, \neg r\}$ şi $\{p, q, \neg q\}$ şi am obţinut, în fiecare caz, mulţimea vidă de clauze $\emptyset$ (nu clauza vidă $\square$), deci mulţimea de clauze de mai sus e satisfiabilă, aşadar enunţul $[(\neg p \rightarrow q) \vee (p \wedge q \wedge \neg r) \vee r] \wedge (\neg q \vee \neg r)$ e satisfiabil.
(3) Enunţul $(p \vee q) \wedge (\neg q \vee r) \wedge (\neg q \vee \neg r) \wedge \neg p$ este în FNC şi îi corespunde mulţimea de clauze $\{\{p, q\}, \{\neg q, r\}, \{\neg q, \neg r\}, \{\neg p\}\}$. De exemplu, o derivare prin rezoluţie a acestei mulţimi de clauze este: $\dfrac{\dfrac{\dfrac{\{p, \cancel{q}\}, \{\cancel{\neg q}, r\}, \{\neg q, \neg r\}, \{\neg p\}}{\{p, \cancel{r}\}, \{\neg q, \cancel{\neg r}\}, \{\neg p\}}}{\{\cancel{p}, \neg q\}, \{\cancel{\neg p}\}}}{\{\neg q\}}$, dar o altă derivare conduce la clauza vidă:

$$\begin{array}{c}\{p,q\},\{\neg\, q,\cancel{r}\},\{\neg\, q,\neg\cancel{r}\},\{\neg\, p\}\\ \hline \{p,\cancel{q}\},\{\neg\cancel{q}\},\{\neg\, p\}\\ \hline \{\cancel{p}\},\{\neg\cancel{p}\}\\ \hline \square\end{array}$$

, aşadar mulţimea de clauze de mai sus e nesatisfiabilă, prin urmare enunţul $(p \vee q) \wedge (\neg\, q \vee r) \wedge (\neg\, q \vee \neg\, r) \wedge \neg\, p$ e nesatisfiabil.

Exerciţiul 2. Considerăm un limbaj de ordinul I conţinând două simboluri distincte de operaţii binare f şi g şi două constante diferite a şi b. Fie X, Y şi Z variabile distincte. Să se determine dacă următorii termeni au unificator:

(1) $f(g(X, f(X, Y)), f(X, a))$ şi $f(g(f(a, b), f(f(Y, Y), Y)), f(X, Y))$;

(2) $f(g(X, f(X, Y)), f(X, a))$ şi $f(g(f(a, Z), f(f(Y, Y), Y)), f(X, Y))$.

Rezolvare: A se vedea fişierul REZ_EXERC_UNIF.PDF. Pentru mai multe cazuri ale ALGORITMULUI DE UNIFICARE, a se vedea seminarul cu exerciţii de unificare.

Exerciţiul 3. Să se scrie:

(1) un predicat binar *alternsort*, astfel încât *alternsort*(L, M) să fie satisfăcut ddacă L şi M sunt liste de liste de numere, iar M se obţine din lista L prin înlocuirea fiecărui element al său cu o listă sortată, astfel:

primul element, $L1$, al lui L, se înlocuieşte cu lista $S1$ obţinută prin sortarea crescătoare a lui $L1$;

al doilea element, $L2$, al lui L, se înlocuieşte cu lista $S2$ obţinută prin sortarea descrescătoare a lui L;

ş.a.m.d.: elementele lui L de pe poziţii impare se sortează crescător, iar cele de pe poziţii pare se sortează descrescător şi se depun, astfel sortate, în M;

să se scrie şi toate predicatele auxiliare necesare pentru implementarea lui *alternsort*, inclusiv un predicat pentru sortare de liste, cu o metodă de sortare la alegere;

(2) un predicat binar *catectnr* care să primească drept prim argument un termen arbitrar şi să calculeze, în al doilea argument, numărul de constante numerice care apar în scrierea termenului dat în primul argument; de exemplu, la o interogare de forma:

?- $catectnr(f(g(f(a, 10), f(f(5.5, Y), -2)), f(X, Y)), Cate)$.

răspunsul dat de Prolog trebuie să fie: Cate=3.

Rezolvare: A se vedea fişierul EXERCTIPEXAM.PL.