Demonstraţii semantice (i.e. prin tabel de adevăr) în Prolog

pentru proprietăţi ale operaţiilor şi relaţiilor între mulţimi

MATERIAL AJUTĂTOR PENTRU LABORATORUL DE LOGICĂ MATEMATICĂ ŞI COMPUTAŢIONALĂ

Claudia MUREŞAN, cmuresan@fmi.unibuc.ro, claudia.muresan@g.unibuc.ro

Universitatea din Bucureşti, Facultatea de Matematică şi Informatică

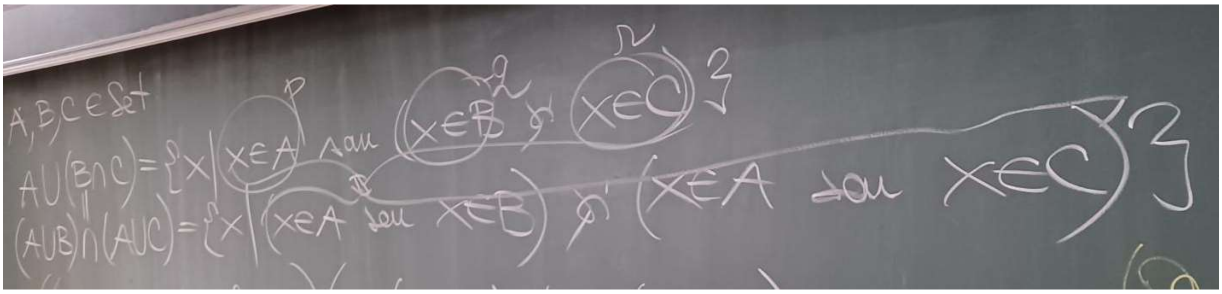
Semestrul II, 2024-2025

Exerciţiu: Să demonstrăm, printr-un program în Prolog, distributivitatea reuniunii faţă de intersecţie: pentru orice mulţimi A,B,C, A∪(B∩C)=(A∪B)∩(A∪C).

Soluţie: Demonstrăm semantic, i.e. printr-un tabel de adevăr, dar cu ajutorul Prologului, distributivitatea disjuncţiei faţă de conjuncţie: **p sau (q şi r) ⇔ (p sau q) şi (p sau r)**, oricare ar fi proprietăţile **p**,**q**,**r** având valorile de adevăr **fals** sau **adevărat**.

Apoi luăm un element x arbitrar şi aplicăm această proprietate enunţurilor: x∈A, x∈B, x∈C în locul lui **p**, **q**, respectiv **r**, şi obţinem:

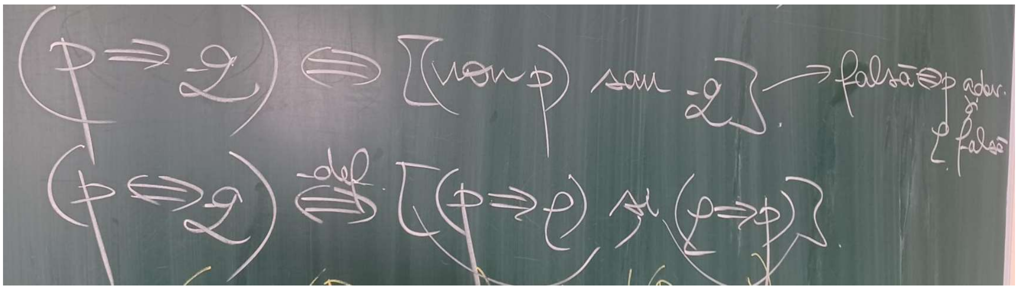
x∈A **sau (**x∈B **şi** x∈C**)** ⇔ **(**x∈A **sau** x∈B**) şi (**x∈A **sau** x∈C**)**,   adică: x∈A∪(B∩C) ⇔ x∈(A∪B)∩(A∪C).



Cum x este arbitrar şi, prin definiţie, două mulţimi sunt egale ddacă au aceleaşi elemente:

va rezulta că A∪(B∩C)=(A∪B)∩(A∪C).

Să ne amintim că două proprietăţi/enunţuri (care pot avea valorile de adevăr **fals** sau **adevărat**) sunt echivalente ddacă au aceeaşi valoare de adevăr:



Deci **p⇔q** este adevărată ddacă **p** şi **q** sunt *ambele* **false** sau *ambele* **adevărate**.

Aşadar avem de demonstrat că: oricare ar fi proprietăţile **p**,**q**,**r** având valorile de adevăr **fals** sau **adevărat**, enunţurile **[p sau (q şi r)]** şi **[(p sau q) şi (p sau r)]** au aceeaşi valoare de adevăr. Notăm aceste enunţuri cu **ms(p,q,r)**, respectiv **md(p,q,r)**. Deci avem de demonstrat că, dacă:

(∀P,Q,R∊{true,false})(ms(P,Q,R)=md(P,Q,R)), unde, pentru fiecare **P,Q,R∊{true,false}**, valorile de adevăr **ms(P,Q,R) = P sau (Q şi R)**, iar **md(P,Q,R) = (P sau Q) şi (P sau R)** (putem scrie chiar egal în loc de echivalenţă, cu semnificaţia: valoare de adevăr calculată din valorile **P,Q,R∊{true,false}** cu aceşti cuantificatori logici priviţi ca operaţii pe mulţimea valorilor de adevăr **{true,false}**).

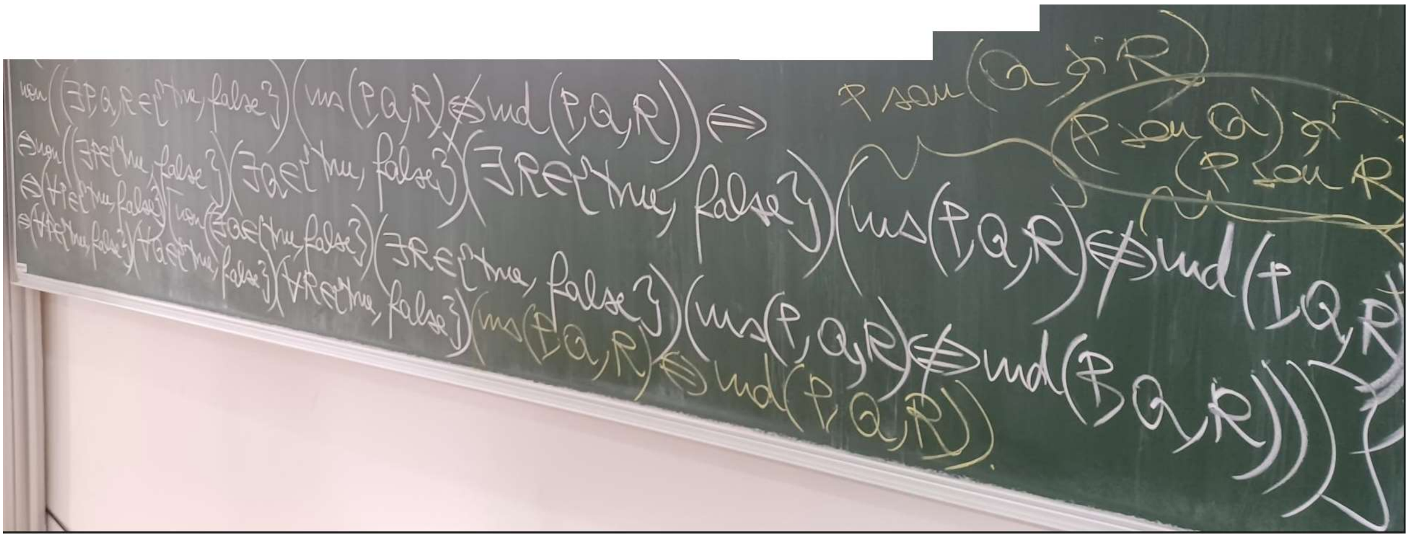
Enunţul acesta este doar o scriere prescurtată pentru enunţul cu trei cuantificatori universali:

(∀P∊{true,false})(∀Q∊{true,false})(∀R∊{true,false})(ms(P,Q,R)=md(P,Q,R)),

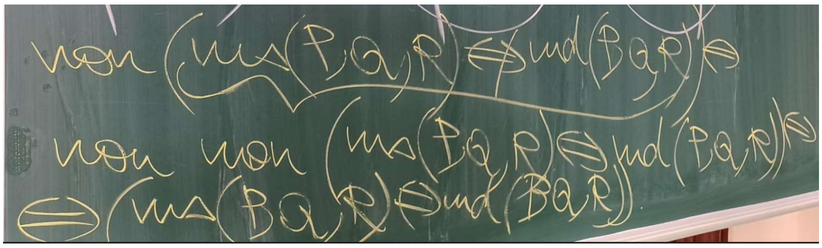
care este echivalent cu enunţul:

non[(∃P∊{true,false})(∃Q∊{true,false})(∃R∊{true,false})(ms(P,Q,R)≠md(P,Q,R))].

Într-adevăr, iterând procedeul de negare a enunţurilor cuantificate:



deoarece dubla negaţie este echivalentă cu identitatea:



Implementare, cu testarea faptului că Prologul trece prin toate cele 8 triplete de valori de adevăr, obţinând **false** sub acea negaţie de fiecare dată, apoi evaluează predicatul zeroar (i.e. de aritate 0, cu 0 argumente, deci propoziţia) *distrib* la **true**, prin afişarea tripletului de valori pentru *(P,Q,R)* la fiecare pas, punând câte un rând de paranteze în plus pentru că predicatele predefinite *not* şi *write* sunt unare (i.e. de aritate 1, cu câte un singur argument), aşa că trebuie să specificăm că virgulele de sub *not* sunt

conjuncţii, nu separatori de argumente, iar argumentul lui *write* este un **triplet**, nu sunt trei argumente separate prin virgulă, şi testând egalitatea valorilor de adevăr prin echivalenţă între acele expresii booleene, întrucât o simplă unificare nu le-ar evalua valorile booleene calculate, ci ar eşua, *ms* şi *md* fiind operatori diferiţi:

```
ms(P,Q,R) :- P;Q,R.

md(P,Q,R) :- (P;Q),(P;R).

implica(P,Q) :- not(P);Q.

echiv(P,Q) :- implica(P,Q),implica(Q,P).

distrib :- not((member(P,[true,false]), member(Q,[true,false]), member(R,[true,false]),
            write((P,Q,R)), nl, not(echiv(ms(P,Q,R),md(P,Q,R))))).
```