PROGRAMARE LOGICĂ
SEMINAR 4
- REZOLUŢIA SLD -

Teorie:

- O *clauză definită* este o formulă de forma:
  - $P(t_1, \ldots, t_n)$ (formulă atomică), unde P este un simbol de predicat, iar $t_1, \ldots, t_n$ termeni
  - $P_1 \wedge \ldots \wedge P_n \to Q$, unde toate P_i, Q sunt formule atomice.
- O regulă din Prolog $\mathtt{Q :- P_1, \ldots, P_n}$ este o clauză $P_1 \wedge \ldots \wedge P_n \to Q$, iar un fapt din Prolog $\mathtt{P(t_1, \ldots, t_n)}$ este o formulă atomică $P(t_1, \ldots, t_n)$.
- O clauză definită $P_1 \wedge \ldots \wedge P_n \to Q$ poate fi gândită ca formula $Q \vee \neg P_1 \vee \ldots \vee \neg P_n$.
- Pentru o mulţime de clauze definite T, *regula rezolutiei SLD* este

$$\text{SLD} \quad \boxed{\frac{\neg P_1 \vee \cdots \vee \neg P_i \vee \cdots \vee \neg P_n}{(\neg P_1 \vee \cdots \vee \neg Q_1 \vee \cdots \vee \neg Q_m \vee \cdots \vee \neg P_n)\theta}}$$

  unde $Q \vee \neg Q_1 \vee \cdots \vee \neg Q_m$ este o clauză definită din T (în care toate variabilele au fost redenumite) şi θ este c.g.u pentru P_i şi Q.
- Fie T o mulţime de clauze definite şi $P_1 \wedge \ldots \wedge P_m$ o ţintă, unde P_i sunt formule atomice. O derivare din T prin rezoluţie SLD este o secvenţă $G_0 := \neg P_1 \vee \ldots \vee \neg P_m$, G_1, ..., G_k, ... în care G_{i+1} se obţine din G_i prin regula SLD. Dacă există un k cu $G_k = \square$ (clauza vidă), atunci derivarea se numeşte *SLD-respingere*.

Teoremă 1 (Completitudinea SLD-rezoluţiei)**.** *Sunt echivalente:*

(1) există o SLD-respingere a lui $P_1 \wedge \ldots \wedge P_m$ din T,
(2) $T \vdash_b P_1 \wedge \ldots \wedge P_m$,
(3) $T \models P_1 \wedge \cdots \wedge P_m$.

- Fie T o mulţime de clauze definite şi o ţintă $G_0 = \neg P_1 \vee \ldots \vee \neg P_m$. Un *arbore SLD* este definit astfel:
  - Fiecare nod al arborelui este o ţintă (posibil vidă)
  - Rădăcina este G_0
  - Dacă arborele are un nod G_i, iar G_{i+1} se obţine din G_i folosind regula SLD folosind o clauză $C_i \in T$, atunci nodul G_i are copilul G_{i+1}. Muchia dintre G_i şi G_{i+1} este etichetată cu C_i.
- Dacă un arbore SLD cu rădăcina G_0 are o frunză $\square$ (clauza vidă), atunci există o SLD-respingere a lui G_0 din T.

Exerciţiul 1: *Găsiţi o SLD-respingere pentru următoarele programe Prolog şi ţinte:*

(a)

```
1.  r :- p,q.      5.  t.      ?- w.
2.  s :- p,q.      6.  q.
3.  v :- t,u.      7.  u.
4.  w :- v,s.      8.  p.
```

(b)

```
1.  q(X,Y) :- q(Y,X), q(Y,f(f(Y))).      ?- q(f(Z),a).
2.  q(a,f(f(X))).
```

(c)

```
1.  p(X) :- q(X,f(Y)), r(a).      4.  r(X) :- q(X,Y).      ?- p(X), q(Y,Z).
2.  p(X) :- r(X).                 5.  r(f(b)).
3.  q(X,Y) :- p(Y).
```

Rezolvare:

(a)

$G_0 = \neg w$
$G_1 = \neg v \vee \neg s$ *(4)*
$G_2 = \neg t \vee \neg u \vee \neg s$ *(3)*
$G_3 = \neg u \vee \neg s$ *(5)*
$G_4 = \neg s$ *(7)*
$G_5 = \neg p \vee \neg q$ *(2)*
$G_6 = \neg q$ *(8)*
$G_7 = \Box$ *(6)*

(b)

$G_0 = \neg q(f(Z), a)$
$G_1 = \neg q(a, f(Z)) \vee \neg q(a, f(f(a)))$ *(1 cu $\theta(X) = f(Z)$ şi $\theta(Y) = a$)*
$G_2 = \neg q(a, f(Z))$ *(2 cu $\theta(X) = a$)*
$G_3 = \Box$ *(2 cu $\theta(Z) = f(X)$)*

(c)

$G_0 = \neg p(X) \vee \neg q(Y, Z)$
$G_1 = \neg r(X_1) \vee \neg q(Y, Z)$ *(2 cu $\theta(X) = X_1$)*
$G_2 = \neg q(Y, Z)$ *(5 cu $\theta(X_1) = f(b)$)*
$G_3 = \neg p(Z_1)$ *(3 cu $\theta(X) = Y_1$ şi $\theta(Y) = Z_1$)*
$G_4 = \neg r(X)$ *(2 cu $\theta(Z_1) = X$)*
$G_5 = \Box$ *(5 cu $\theta(X) = f(b)$)*

Exerciţiul 2: Desenaţi arborele SLD pentru programul Prolog de mai jos şi ţinta `?- p(X,X).`

```
1.  p(X,Y) :- q(X,Z), r(Z,Y).
2.  p(X,X) :- s(X).
3.  q(X,b).
4.  q(b,a).
5.  q(X,a) :- r(a,X).
6.  r(b,a).
7.  s(X) :- t(X,a).
8.  s(X) :- t(X,b).
9.  s(X) :- t(X,X).
10. t(a,b).
11. t(b,a).
```

Rezolvare: