LISTE ÎN PROLOG

I/ Definirea listelor în Prolog

- lista nevidă cu elemente reale : $L = [H_1, H_2, \dots, H_n]$, $H_i \in \mathbb{R}$
- listă vidă : $L = [\,]$.
- se utilizează reprezentarea recursivă a listelor: $\boxed{L = [H|T]}$ cu H = primul element din listă (Head), iar T = restul listei (Tail).
- exemple :
  - $L_1 = [7, 9, 11, 13]$ → $H_1 = 7$; $T_1 = [9, 11, 13]$ → $HT_1 = 9$; $TT_1 = [11, 13]$
  - $L_2 = [\overset{p_1}{[1,-1]}, \overset{p_2}{[2,-2]}, \overset{p_3}{[3,-3]}]$ → $H_2 = [1,-1]$; $T_2 = [[2,-2], [3,-3]]$
- declararea domeniului listă se realizează prin:

```
domains
    lista = tip_date*
```

Exemple:

```
lista = real*
matrice = lista*
```

II/ Prelucrarea listelor în Prolog

1) Verificarea apartenenței unui element într-o listă de numere reale.

$X \in \mathbb{R}$, $L = [H_1, H_2, \dots, H_n]$, $H_i \in \mathbb{R}$, $i = \overline{1,n}$, $X \in L$? (true/false)

Program Prolog

```
domains
    lista = real*
predicates
    member(real, listă)
clauses
    member(X, [X|T]) :- !.
    member(X, [Y|T]) :- member(X, T).
```

Execuție: member(7, [2,-1,7,3,10]). member(10, [1,-1,7,2,8]).

2) Lungimea (dimensiunea) unei liste de numere reale.

$L = [H_1, H_2, \dots, H_n] \Rightarrow \bar{L} = n.$

$L = [H_1|T_1] \Rightarrow \boxed{\bar{L} = 1 + \bar{T_1}}.$

$\bar{L}$: N ; $\bar{T_1}$: N1

Program Prolog

```
domains
    lista = real*
predicates
    lungime(lista, integer)
clauses
    lungime([], 0).
    lungime([H|T], N) :- lungime(T, N1), N = 1 + N1.
```

Execuție: lungime([2,-1,0,1,3,7,8,2,1], Lungime).