# 3) Metoda Quicksort

- împărțirea listei curente în funcție de primul element în două liste: o listă cu valori mai mici decât primul element și, respectiv, o listă cu elemente mai mari decât primul element
- se sortează cele două liste obținute la pasul anterior
- concatenarea rezultatelor obținute (a listelor sortate)

Program Prolog

```
domains
    lista = real *
predicates
    qsort(lista, lista)
    split(real, lista, lista, lista)
    concatenare(lista, lista, lista)
clauses
    qsort([], []).
    qsort([H|T], LS) :- split(H, T, Lmici, Lmari), qsort(Lmici, LS1), qsort(Lmari, LS2),
                        concatenare(LS1, [H|LS2], LS).
    split(K, [], [], []).
    split(K, [H|T], [H|T1], L2) :- H < K, split(K, T, T1, L2), !.
    split(K, [H|T], L1, [H|T2]) :- split(K, T, L1, T2).
    concatenare([], L2, L2).
    concatenare([H|T], L2, [H|TR]) :- concatenare(T, L2, TR).
```

Exercițiu:  qsort([20, -1, 12, 8], Lsort).

LSort = [-1, 8, 12, 20]

# Aplicație privind utilizarea metodelor de sortare

Să se determine lista sumelor elementelor de pe fiecare linie dintr-o matrice de valori reale și să se sorteze lista rezultat.

Fie matricea:

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix} \begin{array}{l} \rightarrow SL1 \\ \rightarrow SL2 \\ \cdots \\ \rightarrow SLn \end{array} \Rightarrow$$

$LR = [SL1, SL2, \ldots, SLn]$

$[HLR | TLR]$

sortare $(LR) \rightsquigarrow LSSort$ (lista sumelor pe linii sortată)

## Program ProLog

**domains**

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

**predicates**

```
sumaL (lista, real)
sumaM (matrice, lista)
sort_line_matrix (matrice, lista)
insertsort (lista, lista)
insert (real, lista, lista)
```

**clauses**

```
[sumaL ([], 0).
[sumaL ([H|T], S) :- sumaL(T, S1), S is S1+H.

[sumaM ([], []).
[sumaM ([HM|TM], [SHM|STM]) :- sumaL(HM, SHM), sumaM(TM, STM).
```

```
sort_line_matrix (M, LSSort) :- sumaM(M, LR),
                                 insertsort(LR, LSSort).
```

\* sortarea prin inserție (insertsort și insert)

Exemplu:

```
sort_line_matrix ([[1,2,-1], [2,3,0], [2,0,1]], LS).
```