

EXAMPLE

1. Se dă o listă liniară...

```
defun lista (l)
  (cond
    ((null l) nil)
    ((numberp (car l)) (cons (car l) (lista (cdr l))))
    (t (lista (cdr l))))
  )
)
```

(lista '(1 a 2 b 3 c)) →

```
(defun lista_aux (l col)
  (cond
    ((null l) col)
    ((numberp (car l)) (lista_aux (cdr l) (append col (list (car l)))))
    (t (lista_aux (cdr l) col))
  )
)

(defun lista (l)
  (lista_aux l nil)
)
```

2. Se dă o listă liniară...

```
(defun per (e l)
  (cond
    ((null l) nil)
    (T (cond
      ((< e (car l)) (cons (list e (car l)) (per e (cdr l))))
      (T (per e (cdr l)))
    ))
  )
)

(defun perechi (l)
  (cond
    ((null l) nil)
    (t (append (per (car l) (cdr l)) (perechi (cdr l))))
  )
)
```

(perechi '(1 2 3)) →

3. Se dă o listă liniară l, $0 < n \leq (\text{lungimea lui } l) + 1$...

```
(defun ins (el n l)
  (cond
    ((= n 1) (cons el l))
    (t (cons (car l) (ins el (- n 1) (cdr l)))))
  )
)
```

```
(defun ins_toate(el n l)
  (cond
    ((= n 0) nil)
    (t (cons (ins el n l) (ins_toate el (- n 1) l))))
)
```

```
(defun inserare (el l)
  (ins_toate el (+ (length l) 1) l)
)
```

(inserare '1 '(2 3)) →