

R1. Subalgoritmi recursivi

Sa se scrie modelul matematic recursiv si subalgoritmii corespunzatori pentru urmatoarele cerinte:

- Să se numere de câte ori apare un element în listă.
- Să se șteargă aparițiile unui element din listă.
- Să se determine produsul elementelor de pe poziții impare din listă.

Pentru început, vom descrie formulele matematice recursive de rezolvare a subproblemelor. Precizăm că formulele recursive le vom descrie matematic, independent de vreun limbaj de programare, urmând ca implementarea mecanismului recursiv să se facă în funcție de particularitățile limbajului de programare folosit.

Vom presupune în cele ce urmează că lista pe care o prelucrăm L e formată din elementele l_1, l_2, \dots, l_n :

$L = (l_1, l_2, \dots, l_n)$. Pentru descrierea în Pseudocod, vom considera lista reprezentată secvențial pe tablou.

1. Să se adauge un element la sfârșitul unei liste.

a. Model matematic recursiv

$$adaug(e, l_1 l_2 \dots l_n) = \begin{cases} (e) & \text{daca } n = 0 \\ l_1 \cup adaug(e, l_2 \dots l_n) & \text{altfel} \end{cases}$$

b. Descriere Pseudocod

Funcția $adaug(e, L)$ este:

Dacă $L.n=0$ atunci

$adaug \leftarrow creeaza(e)$ {"creeaza(e)" returnează o listă care conține doar elementul "e"}

altfel

$L1 \leftarrow sublista(L, 2)$ {"sublista(L,i)" returnează sublista $li \dots ln$ }

$adaug \leftarrow adaugaInceput(adaug(e, L1), L.[1])$ {"returnează lista obținută adăugându-l pe $l1$ la începutul listei $L1$ }

SfDacă

SfFuncție

Funcția $creeaza(e)$ este:

$L.n \leftarrow 1$

$L.[1] \leftarrow e$

$creeaza \leftarrow L$

SfFuncție

Funcția $sublista(L, i)$ este:

Pentru $j \leftarrow 1, L.n-i+1$ execută

$L1.[j] \leftarrow L.[i+j-1]$

```

SfPentru
L1.n ← L.n-i+1
sublista ← L1
SfFuncție

```

Funcția *adaugaInceput*(L,e) este:

```

Pentru i ← L.n+1,2,-1 executa
    L1.[i] ← L.[i-1]
SfPentru
L1.[1] ← e
L1.n ← L.n+1
adaugaInceput ← L1
SfFuncție

```

2. Să se numere de câte ori apare un element în listă.

Model matematic recursiv

$$nrap(E, l_1 l_2 \dots l_n) = \begin{cases} 0 & \text{daca } l \text{ e vida} \\ 1 + nrap(E, l_2 \dots l_n) & \text{daca } l_1 = E \\ nrap(E, l_2 \dots l_n) & \text{altfel} \end{cases}$$

.....

3. Să se șteargă aparițiile unui element din listă.

Model matematic recursiv

$$stergere(E, l_1 l_2 \dots l_n) = \begin{cases} () & \text{daca } l \text{ e vida} \\ l_1 \cup stergere(E, l_2 \dots l_n) & \text{daca } l_1 \neq E \\ stergere(E, l_2 \dots l_n) & \text{altfel} \end{cases}$$

.....