

Pentru tipul LIST definit prin constructorii:

$[] : \rightarrow \text{LIST}$

$[a] : T \rightarrow \text{LIST}$

$a::x$ (sau $\text{cons}(a, x)$) : $T \times \text{LIST} \rightarrow \text{LIST}$ (adica a adauga la inceputul listei)

Folosind recursivitatea si inductia structurala, rezolvati urmatoarele exercitii:

1. Definiti functiile:

- $\text{append}(l1, l2)$ – concatenarea a doua liste
- $\text{member}(a, l)$ – verifica apartenenta elementului a in lista l
- $\text{size}(l)$ – dimensiunea listei
- $\text{maxelem}(l)$, $\text{minelem}(l)$ – intorc elementul maxim, respectiv minim din lista
- $\text{set}(l)$ – intoarce lista ce contine elemente distincte din l
- $\text{nbrelems}(l)$ – numarul de elemente distincte din lista

2. Aratati ca orice l de tip LIST:

- $\text{member}(a, l) \rightarrow \text{minelem}(l) \leq a \leq \text{maxelem}(l)$

3.

- $\text{size}(\text{set}(l)) \leq \text{size}(l)$
- $\text{nbrelems}(l) \leq \text{size}(l)$

4. $\text{size}(\text{set}(l)) = \text{size}(l) \rightarrow \text{set}(l) = l$

5. definiti functiile:

$\text{remove}(l, a) : \text{LIST} \times T \rightarrow \text{LIST}$ - intoarce o lista rezultata din eliminarea elementelor a din l

$\text{double}(l, a) : \text{LIST} \times T \rightarrow \text{LIST}$ - intoarce o lista rezultat din dublarea elementelor a din l (fiecare a devine aa)

Sa se demonstreze ca $\text{member}(a, l) \rightarrow \text{size}(\text{remove}(l, a)) \neq \text{size}(\text{double}(l, a))$

6. $\text{remove}(a, \text{append}(l1, l2)) = \text{append}(\text{remove}(a, l1), \text{remove}(a, l2))$

7. Se considera o stiva implementata folosind un vector. Operatiile definite pentru stiva sunt push si pop. Totusi, cand se adauga un element in stiva (push) si vectorul este plin, trebuie sa se aloce un nou vector si sa se copieze elementele din vectorul vechi in cel nou, iar apoi se va folosi vectorul nou pe post de stiva.

a. Care este complexitatea operatiilor push si pop in cazul cel mai defavorabil ?

Justificati.

b. Daca dimensiunea vectorului se incrementeaza cu 1 atunci cand este necesar un vector mai mare, care este costul total pentru n operatii si care este costul mediu (amortizat) al unei operatii pe stiva, folosind metoda agregarii ?

c. Daca dimensiunea vectorului se dubleaza atunci cand este necesar un vector mai mare, care este costul total pentru n operatii si care este costul mediu (amortizat) al unei operatii pe stiva, folosind metoda agregarii ?

8. Considerand un min-heap binar, cu operatiile insert si delete_min, se cere:

a. Care este complexitatea celor doua operatii in cazul cel mai defavorabil ? Justificati.

- b. Definiti o functie potential pentru min-heap astfel incat costurile amortizate ale celor doua operatii sa fie cat mai mici (delete_min sa fie $O(1)$).
- c. Demonstrati prin metoda potentialului care sunt costurile amortizate ale celor doua operatii.

9. Se considera o secventa de n operatii ce se executa pe o structura de date. Operatia i are costul i daca i este o putere a lui 2, respectiv, are costul egal cu 1 altfel. Aplicati metoda agregarii si metoda potentialului pentru a determina costul amortizat.

Notare: Fiecare exercitiu valoreaza 1p => 9 * 1p + 1p (oficiu) = 10p