

Rând 7

1. Analizați timpii mediu și defavorabil pentru următorul algoritm. Justificați.

function F(n)

Require: n: Întreg

$s \leftarrow 0$

$m \leftarrow n$

while $m \neq 0$ **do**

$s \leftarrow s + \lfloor m/10 \rfloor$

$m \leftarrow m/10$

end while

$f \leftarrow s$

end function

procedure A

$s \leftarrow 0$

for $i = 1, n$ **do**

$m \leftarrow 2i + 1$

$s \leftarrow s + f(m)$

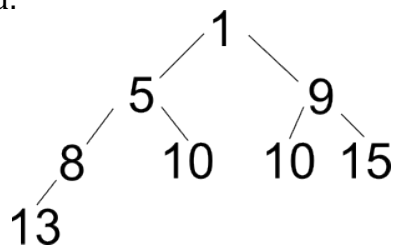
end for

@ scrie s

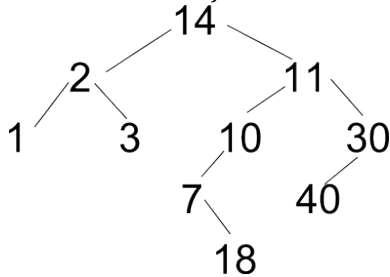
end procedure

2.1. Fie o tabelă de dispersie inițial vidă, cu 5 locații și funcția $d(c) = c \bmod 5$ în care coliziunile sunt rezolvate prin înlănțuire folosind arbore AVL pentru memorarea coliziunilor. Ce se întâmplă la inserarea 35, 2, 18, 6, 3, 10, 8, 5.

2.2. Scrieți ansamblul rezultat la inserarea valorii 3 în următorul ansamblu:



2.3. Care e înălțimea nodului 11 în arborele:



3. Răspundeți, justificând.

3.1. Inserarea unui element într-un vector ordonat x_1, \dots, x_n se poate face în: a) $O(\log_2 n)$ b) $O(n)$ c) $\theta(n)$ d) $\theta(\log_2 n)$

3.2 Considerați $4 + 3 * (6 * 3 - 12)$. Folosiți o stivă pentru a converti expresia în formă postfixată. Care este numărul maxim de simboluri care vor apărea la un moment dat în stivă?

a) 1 b) 2 c) 3 d) 4 e) 5

3.3. Care metodă de acces definește o Coadă?

a) FIFO b) LIFO c) FILO d) HPOF

3.4 Fie un arbore binar cu 18 noduri. Care e adâncimea minimă?

a) 1 b) 2 c) 3 d) 4 e) 5

3.5 Care afirmații sunt adevărate?

a) orice ansamblu este un ABC b) orice arbore binar de căutare are structura de ansamblu c) orice ABC verifică proprietatea de ansamblu d) niciuna

3.6. Ce tip de inițializare trebuie pentru o tabelă de dispersie cu adresare deschisă?

a) nici una b) cheile la fiecare poziție în vector trebuie inițializate c) primul element din fiecare listă înălțuită trebuie setat pe NIL d) legătura fiecărei locații din tabelă trebuie setată nulă

4. Alegeți o SD pentru implementarea unui Dicționar Ordonat astfel încât operația de căutare să aibă complexitate minimă. Pseudocod. Precizați complexitatea operației.

5. Operația de adăugare într-un ABC. Arborele se reprezintă înălțuit cu alocare dinamică. Fiecare nod va memora: adâncimea nodului, referința către cei 2 subarbori și către părinte, se va folosi o procedură nerecursivă. Se va preciza complexitatea operației.