

R14.

1. Subalgoritm dat de recurență:

$$T(n) = \begin{cases} 0, & n=1 \\ 2 \cdot T\left(\frac{n}{2}\right) + O(n^3), & \text{altfel} \end{cases}$$

Calculați complexitatea

2.1. chi: 10, 22, 31, 4, 15, 28, 17, 88, 59

$T\Delta$ de lungime $m=11$

adresare de chi

$$d'(c) = c \% m$$

verificare pătratică $c_1 = 1$, $c_2 = 2$

Ilustrați inserarea.

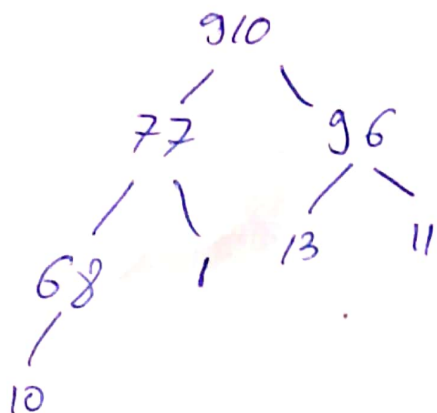
2.2. Ansamblul rezultat prin ștergerea unei valori din a

310
/ \
22 0 6

verificare pătratică $c_1 = 1$, $c_2 = 2$

Ilustrați inserarea.

2.2. Ansamblul rezultat prin ștergerea unei valori din ansamblul.



2.3. Exemplu rotație de simplă opre dreapta AVL.
Justificarea necesității aplicării rotației.

3.1. Complexitate $f(n) = \sum_{i=1}^n 3^i$ + justificare

a) $\Theta(2^n)$ b) $\Theta(3^n)$ c) $O(3^n)$

3.2. ~~Clasa~~ Tipul de listă cel mai potrivit pentru
demonstrarea extragerii elementului de pe poziția n , + justificare

a) listă pe vector b) ~~LDI~~ c) ~~SI~~ d) ~~LDI~~ și ~~LSI~~

3.3. Coadă cu Priorități ce are 2 elemente cu aceeași
prioritate. Ce este rezultatul?

68 13 11
10

2.3. Exemplu rotație de simplă opoziție dreapta AVL.
Justificarea necesității aplicării rotației.

3.1. Complexitate $f(n) = \sum_{i=1}^n 3^i$ + justificare

a) $\Theta(2^n)$ b) $\Theta(3^n)$ c) $O(3^n)$

3.2. ~~Clasa~~ Tipul de listă cel mai potrivit pentru
extragerea elementului de pe poziția n . + justificare

a) listă pe vector b) ~~LDI~~ c) ~~LSI~~ d) LDI și LSI

3.3. Coadă cu Priorități ce are 2 elemente de aceeași
prioritate. Ce rezultă operația element?

a) una din 2 aleator b) prima inserată c) cel mai recent inserat
d) nu e posibil (violarea precondiției)

3.4. Nr. minim noduri arbore binar complet cu adâncime 3.
a) 3 b) 4 c) 8 d) 11 e) 7

3.5. Stergerea în ABC este "comutativă".

Adică: $\text{sterge}(x); \text{sterge}(y); \Leftrightarrow \text{sterge}(y); \text{sterge}(x);$
a) True b) False

3.6. În AVL este adăuginat:

- a) nr noduri subarbore stâng diferă prin cel mult 1 de nr noduri subarbore drept
- b) diferența adâncime a celor doi subarbori e -1, 0 sau 1
- c) diferența noduri subarbori: -1, 0 sau 1
- d) înălțimea lui n este egală cu adâncimea sa

4. Multime pe TD cu adresare deschisă, verif banală
Stabilități

- b) diferența adâncime a celor doi subarbori $\in -1, 0$ sau 1
- c) diferența noduri subarbori: $-1, 0$ sau 1
- d) mulțimea lui n este egală cu adâncimea sa

4. Mulțime pe TD cu adresare deschisă, verificăm lista.
Specificăm, indicăm reprezentarea, descriem în pseudocod
subalgoritmul de adăugare. Precizăm complexitatea.

5. Arbore binar cu elem. distincte. Descrie în pseudocod
operația de del. a mulținii lui x . Nerecursiv.
Complexitate.