

Test de seminar – 3

Timp de lucru: 50 minute

Punctaj Maxim: 10 puncte

Puncte din oficiu: 0 puncte

Numar exercitii: 5

1. (2 puncte) Se da o tabela hash cu urmatoarea functie hash: $h(x) = x \bmod m$, unde $m = 10$. Sa se realizeze cate o schema a tabelii atunci cand se insereaza urmatoarele chei: {62, 91, 13, 14, 34, 75, 28, 7, 17, 100} pentru fiecare metoda de rezolvarea a coliziunilor, anume:
- Chaining (Separate Chaining).
 - Open Addressing (Linear Probing).

ATENTIE, TREBUIE 2 TABELE, CATE UNA PENTRU FIECARE METODA DE REZOLVARE A COLIZIUNILOR.

Fiecare tabela (pentru fiecare metoda de coliziune) va fi punctata cu 1 punct (in total 2 puncte).

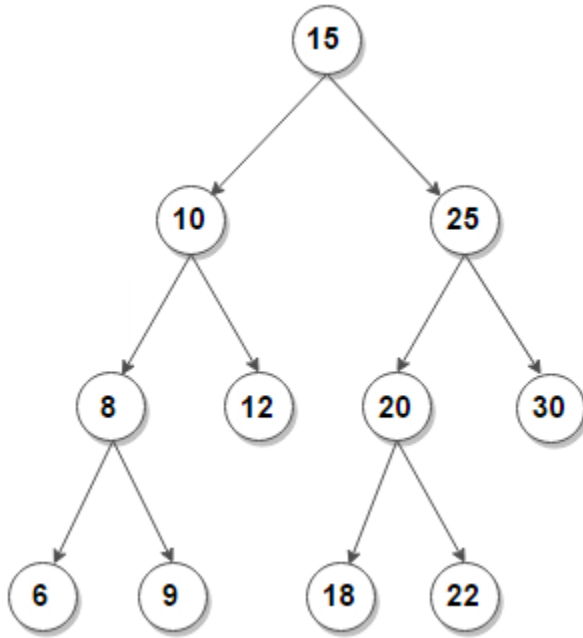
- (1 punct) Dati un exemplu de secventa de intrare (de minim 6 elemente, numere intregi) pentru care algoritmul HeapSort are complexitatea timp $\Theta(n \log n)$ (0.5p). Motivati alegerea facuta intr-un paragraf (0.5p).
- (1 punct) Realizati o comparatie in 2-3 paragrafe intre algorimul de sortare HeapSort si algorimul de sortare BubbleSort. In comparatie trebuie sa discutati diferente de abordare (implementare) dar si eficienta spatiu-timp.
- (2.5 puncte) Se dau doi vectori de numere intregi X si Y, reprezentand chei intr-un binary search tree (BST). **Fara a construi arborii**, verificati daca cei 2 vectori reprezinta acelasi arbore BST. Vom presupune ca insertia cheilor in arbore se face in aceeasi ordine cu aparitia in array.

Exemplu:

$X = \{ 15, 25, 20, 22, 30, 18, 10, 8, 9, 12, 6 \}$

$Y = \{ 15, 10, 12, 8, 25, 30, 6, 20, 18, 9, 22 \}$

Ambii vectori sunt defapt urmatorul arbore: (pag 2)



Detaliati in 2-3 paragrafe tehnica utilizata si aproximati complexitatea spatiu-timp.

5. (3.5 puncte) Consideram un vector de numere intregi, gasiti indexul minim al elementului care se repeta printr-o singura parcurgere a vectorului (timp liniar).

Exemplu:

Input: { 5, 6, 3, 4, 3, 6, 4 }

Output: Indexul minim este 1

Input: { 1, 2, 3, 4, 5, 6 }

Output: Invalid

Se va folosi obligatoriu o tabela de dispersie.

Detaliati in 2-3 paragrafe tehnica utilizata si aproximati complexitatea spatiu-timp.