

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 tabeli atunci cand se insereaza urmatoarele chei: {32, 11, 23, 44, 24, 65, 78, 17, 37, 10} pentru fiecare metoda de rezolvarea a coliziunilor, anume:

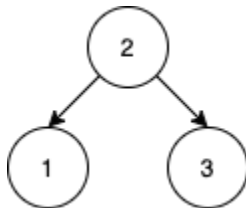
- Chaining (Separate Chaining).
- Open Addressing (Linear Probing).

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 QuickSort. In comparatie trebuie sa discutati diferente de abordare (implementare) dar si eficienta spatiu-timp.
- (2.5 puncte) Se da un arbore binar de cautare (Binary Search Tree – BST) ca intrare la o functie/procedura. Arborele este dat ca un vector de numere ce sunt intregi de la 1 la n. Presupunem ca o sa construim arborele BST din acest vector. Gasiti numarul de moduri diferite sa reordonam sirul astfel incat, atunci cand construim arborele BST, acesta sa fie identic cu cel initial.

Exemplul 1:

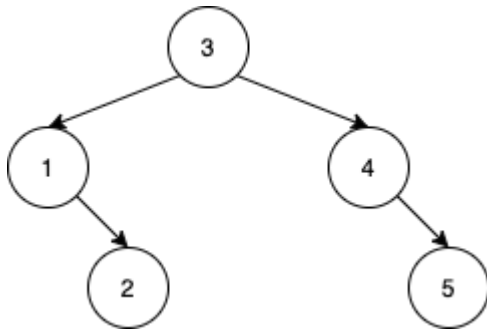


Input: [2,1,3]

Output: 1

Explicatie: Nu putem reordona numerele astfel incat din [2,1,3] sa rezulte acelasi BST.

Exemplul 2:



Input: [3,4,5,1,2]

Output: 5

Explicatie:

[3,1,2,4,5]

[3,1,4,2,5]

[3,1,4,5,2]

[3,4,1,2,5]

[3,4,1,5,2]

5. (3.5 puncte) Consideram un vector de numere intregi nesortate primite ca parametru la o functie si un intreg reprezentand o suma. Returnati daca exista un triplet in vector care sa aiba suma data.

Input:

vector = [2, 7, 4, 0, 9, 5, 1, 3]

suma = 6

Output: True

Explicatie: Triplete sunt {0, 1, 5}, {0, 2, 4}, {1, 2, 3}

Se va folosi obligatoriu o tabela de dispersie.

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