

# Intrebari si Rezolvari Arbori (BST, AVL, Red-Black, Splay, B-Trees)

## 1. BST (Binary Search Tree)

1. Construiești un BST inserând pe rand: 15, 6, 18, 3, 7, 17, 20.

Rezolvare: Arbore rezultat

15

/ \

6 18

/ \ / \

3 7 17 20

2. Complexitatea worst-case pentru Search într-un BST?

Rezolvare:  $O(n)$ , când arborele e degenerat (elemente sortate).

3. Pași pentru stergerea unui nod cu doi copii?

Rezolvare: găsește succesorul/predecesorul în order, copiază cheia în nod, șterge succesorul (caz simplu).

## 2. AVL Tree

1. Inserări 10, 20, 30 într-un AVL → ce rotații?

Rezolvare: Caz RR → rotire stanga la 10 → arbore echilibrat cu 20 la radacina.

2. Inserări 30, 10, 20 într-un AVL → ce rotații?

Rezolvare: Caz LR → rotire stanga la 10, apoi rotire dreapta la 30 → radacina = 20.

3. De ce AVL e mai strict decât Red-Black?

Rezolvare: menține diferența de înălțime max 1 → mai echilibrat, căutare mai rapidă, dar necesită mai multe rotații.

## 3. Red-Black Tree

1. Cele 4 proprietăți fundamentale?

Rezolvare: radacina neagră; frunze NIL negre; nod roșu nu are copil roșu; toate drumurile au același număr de noduri negre.

2. Inseri nod roșu sub părinte roșu?

Rezolvare: daca unchi rosu  $\rightarrow$  recolorare; daca unchi negru  $\rightarrow$  rotatii + recolorari.

3. Inaltime maxima?

Rezolvare:  $h \leq 2 \cdot \log_2(n+1)$ .

## 4. Splay Tree

1. Ce este splaying?

Rezolvare: aducerea nodului accesat la radacina prin rotatii (zig, zig-zig, zig-zag). Cost amortizat  $O(\log n)$ .

2. Accesare nod 20 intr-un arbore  $10 \rightarrow 15 \rightarrow 20$ ?

Rezolvare: zig-zig dreapta (rotire stanga la 10, apoi la 15).

3. Avantaj fata de AVL/Red-Black?

Rezolvare: elementele accesate frecvent sunt aduse la radacina, deci acces foarte rapid pentru pattern-uri repetate.

## 5. B-Trees

1. Defineste un B-Tree si utilizari?

Rezolvare: arbore echilibrat cu multi copii, frunzele la acelasi nivel, operatii  $O(\log n)$ . Folosit in baze de date si sisteme de fisiere.

2. Intr-un B-Tree de ordin 3, cate chei/copii?

Rezolvare (daca ordin =  $t=3$ ): min chei=2, max chei=5, min copii=3, max copii=6.

3. De ce e preferat la DB?

Rezolvare: nodurile se aliniaza pe blocuri de disc, inaltime foarte mica, operatii cu putine accesari la disc.