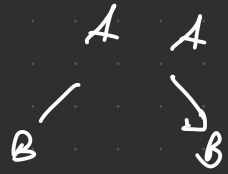


Arborii Binari

→ maxim 2 copii

→ subarbori stang / drept



Transf. A Gen → A Ben
prin fiu, frate dreapta

Implementare

Index	
data	
Stanga	
Dreapta	

Nr. max de noduri de pe nivelul $i \rightarrow 2^{i-1}$

Nr. max noduri : $\sum_{i=1}^h 2^{i-1} - 1 \quad h > 0$

→ arbore plin de înălțime h

Arbore binar complet

Părintele nodului i are indicele $i/2$

Fiul stâng al lui $i \rightarrow 2 * i$
drept $2 * i + 1$

$2 * i + 1 > n$
- nu are fiu

Implementare cu pointeri

```
struct t {  
    val  
    * st  
    * dr  
};
```

Parcurgeri \rightarrow RSD
SRD
SDR

Threaded Trees

```
void pre (AB * r) {
    if (r != NULL) {
        print r -> chere
        pre (r -> st)
        pre (r -> dr)
    }
}
```

-> contine pointer la parinte

Arbori Binari Ordonați

$S \leq R \leq D$ -> trebuie balansat
pt căutări în $\log_2 n$

• Insertia -> $< \text{root}$ -> stanga
 $> \text{root}$ -> dreapta

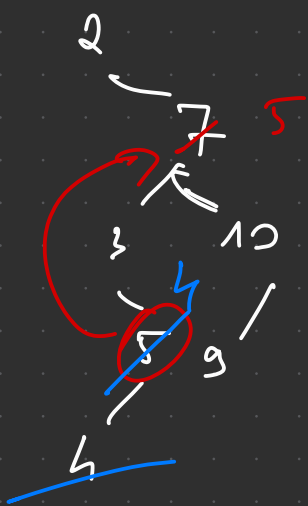
AB * insert (AB * r, int c)

```
if (r != NULL) {
    if (c < r -> chere)
        insert (r -> st, c)
    else insert (r -> dr, c)
}
else {
    r = malloc
    r -> chere = c
    r -> st = NULL
    r -> dr = NULL
    return r
}
```

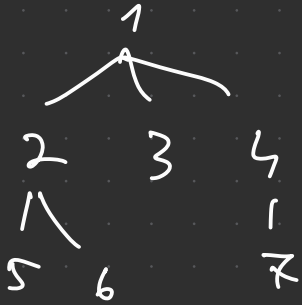
Suprimare

ultimul mod de intrare la
parcurgerea în ordine

SRA



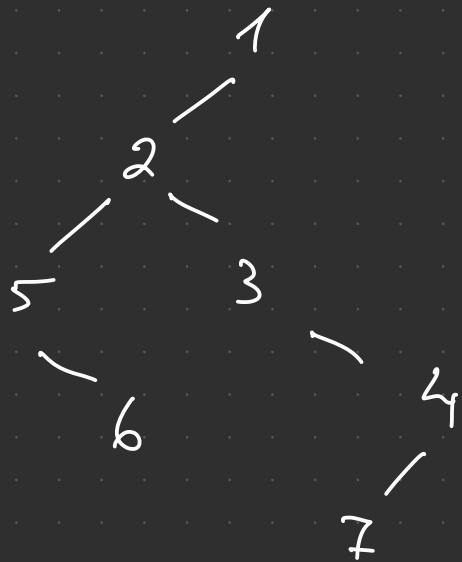
FCRS



SRA

- 5
- 2
- 6
- 1
- 3
- 7
- 4

B



- 5
- 6
- 2
- 3
- 7
- 4
- 1