

Natjecateljsko programiranje

3.predavanje

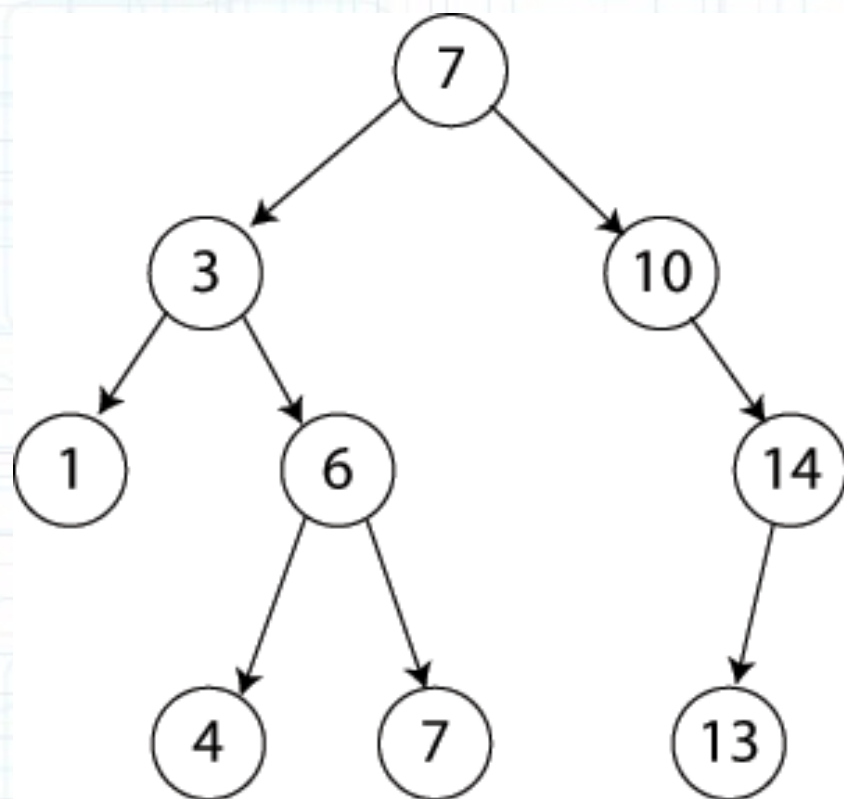
Napredne strukture podataka

Filip Pavetić
fpavetic@gmail.com

Osveta (zadaca)

- A = maksimalan broj ≤ 1000000
- Podjelom na \sqrt{A} blokova moguće je svaki upit riješiti u $O(\sqrt{A})$

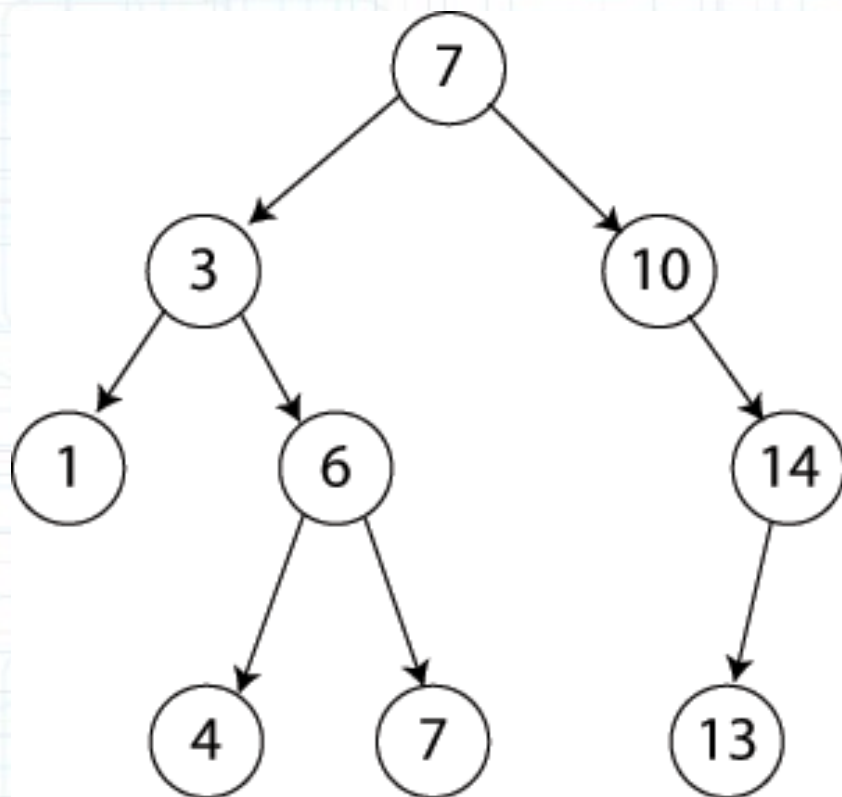
Binarno stablo pretraživanja



- Binarno stablo pretraživanja takva je struktura u kojoj svaki čvor ima dva djeteta
- Vrijednost lijevog djeteta manja je od vrijednosti roditelja, a vrijednost desnog veća

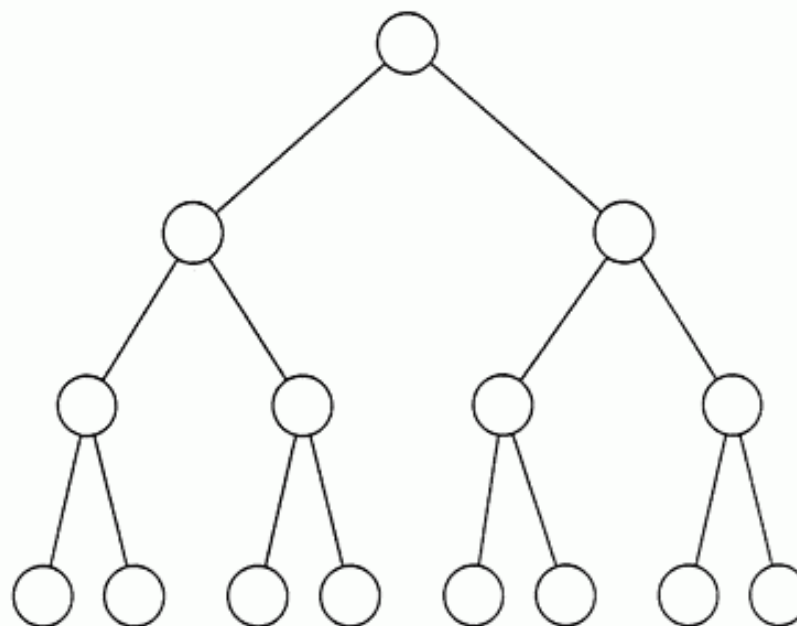
Binarno stablo pretraživanja

- Najgori slučaj ubacivanja elementa u stablo – $O(n)$ (ako redom ubacujemo 1,2,3...n)



Balansirano binarno stablo

- Balansirano binarno stablo je stablo kojem je dubina približno $\log_2(n)$
- Stabla koja održavaju svojstvo balansiranosti u svojim operacijama su složena



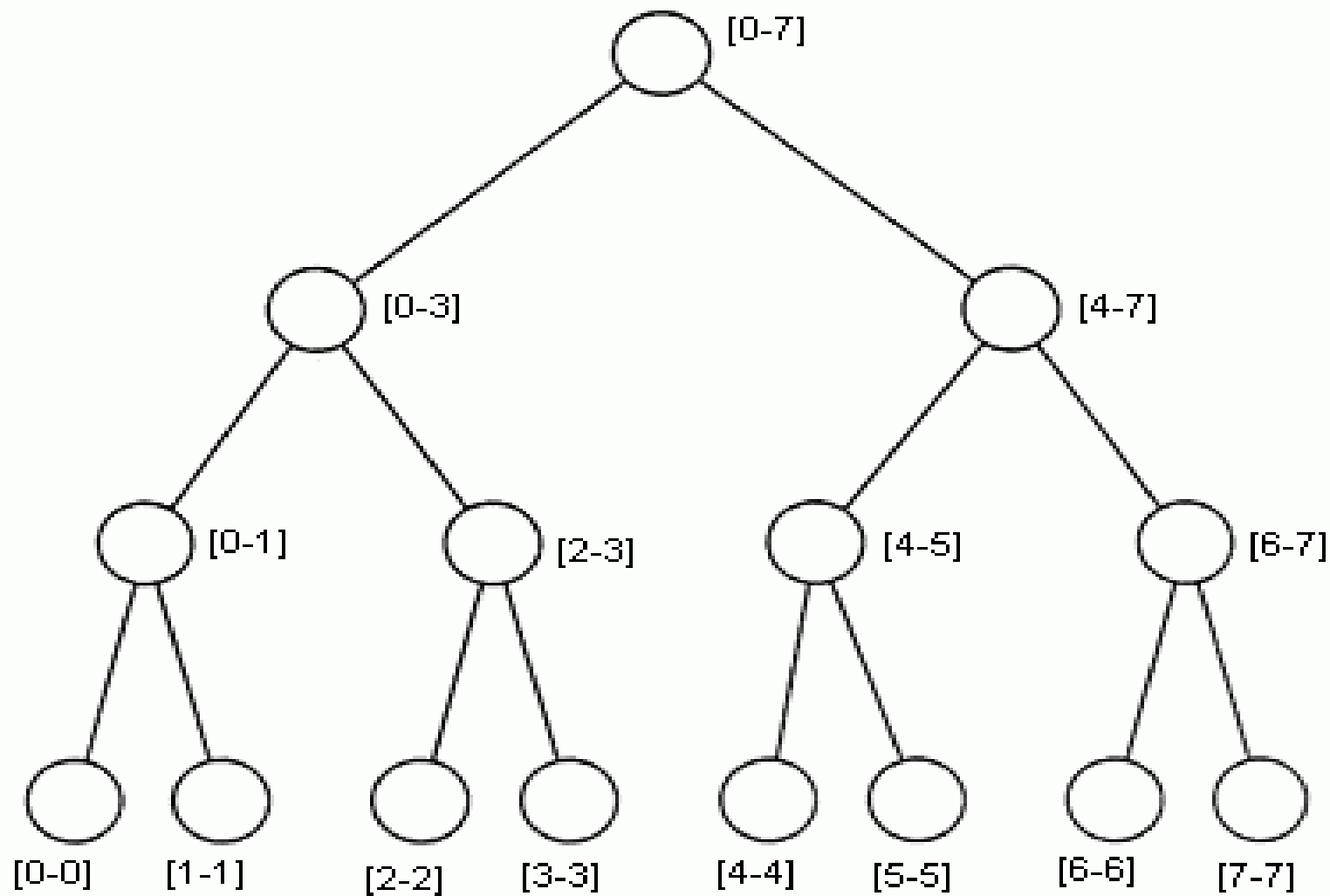
Balansirano binarno stablo

- Iz sortiranog niza možemo **statički** izgraditi balansirano binarno stablo
- Jednostavnom modifikacijom kod običnog binarnog stabla može se postići složenost $O(\sqrt{N})$ po operaciji – nakon što dubina stabla dostigne \sqrt{N} samo “razbijemo” stablo i izgradimo ga balansirano

Interval stablo

- Balansirano stablo u kojem je svaki čvor “zadužen” za podatke o nekom intervalu
- Poznato još pod nazivima:
 - Segmentno stablo
 - Tournament stablo

Interval stablo



Interval stablo - “Osveta”

- Svaki čvor je zadužen za praćenje sume u intervalu
- Dodavanje broja moguće je u $O(\log^2 A)$
- Traženje k-tog po redu – $O(\log^2 A)$
- Brisanje elementa – $O(\log^2 A)$
- Odgovor na upit iz zadatka – $O(\log^2 A)$

Fenwickovo stablo - “logaritamska”

- Uređenje niza tako da element na indeksu k čuva sumu prethodnih 2^i elemenata gdje je 2^i najveća potencija broja 2 s kojom je k djeljiv – $\text{lobit}(k)$
- $\text{lobit}(k) = k \& -k$
- Update elementa – $O(\log_2 n)$
- Upit: suma elemenata $1 \dots n$ – $O(\log_2 n)$

Fenwickovo stablo - “logaritamska”

```
void add( int i, int value ) {  
    for( ; i <= MAX; i += i&-i )  
        suma[i] += value;  
}  
  
int get( int i ) {  
    int ret = 0;  
    for( ; i > 0; i -= i&-i ) ret += suma[i];  
    return ret;  
}
```


Fenwickovo stablo - “logaritamska” - “Osveta”

- Dodavanje elementa – $O(\log^2 A)$
- Brisanje elementa – $O(\log^2 A)$
- Odgovor na upit – $O(\log^2 A)$
- Traženje k-tog po redu – $O(\log^2 A)$ –
binary search + upit sume

Zadaci

- <https://www.spoj.pl/problems/BRCKTS/>
- Najlijeviji element u intervalu s nekim svojstvom
- Brojanje inverzija
- Podniz s najvećom sumom kada su dozvoljene promjene elemenata
- Histogram

Literatura

- <http://www.topcoder.com/tc?module=Static&d1=tutorials>
- <http://www.topcoder.com/tc?module=Static&d1=tutorials&d2=binaryIndexedTrees>