

Teme 9

Binarna stabla

Binarna stabla

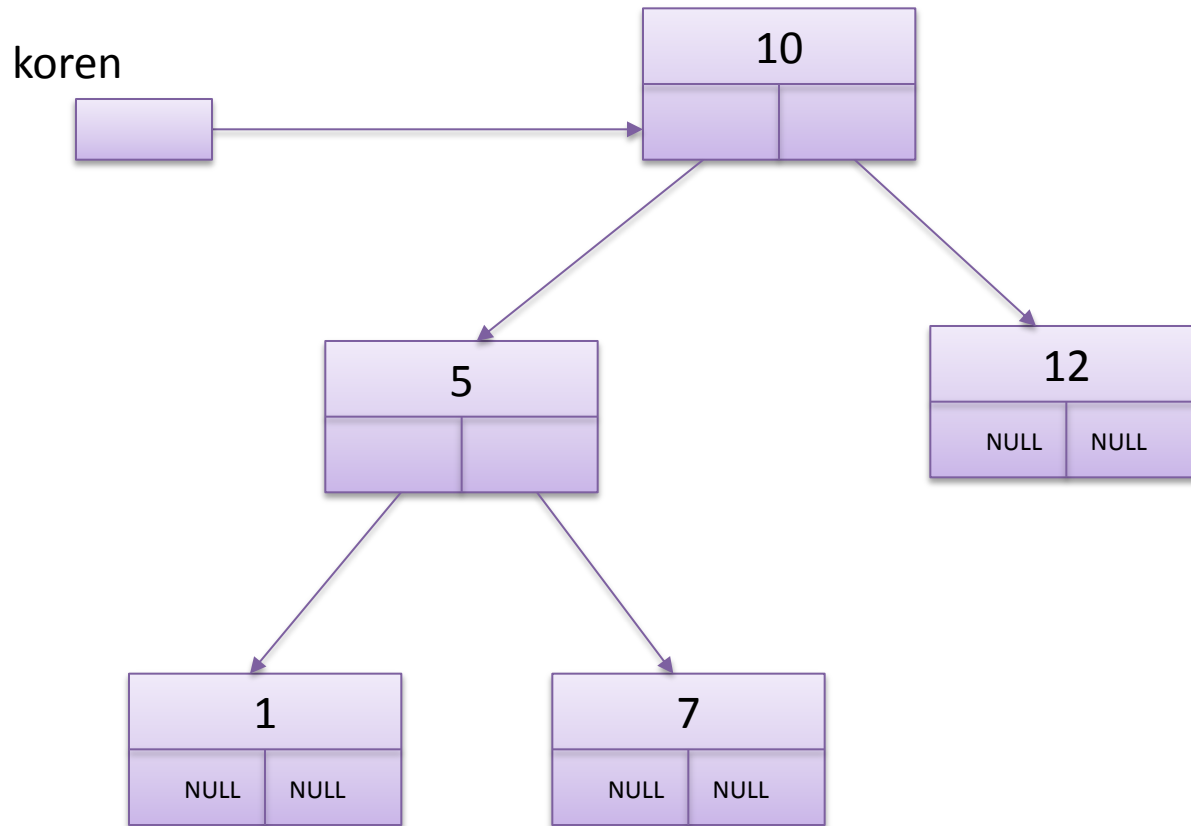
- Svaki čvor sadrži informaciju i dva pokazivača:
 - levi ukazuje na podstablo koje se sastoji iz čvorova manjih od tekućeg
 - desni ukazuje na podstablo koje se sastoji iz čvorova većih od tekućeg
- Postoji i osnovni čvor – koren
- Čvor koji nema ni jedan podčvor se zove list
- Dobre osobine
 - binarno stablo očuvava elemente sortirane
 - brza pretraga

Binarna stabla

- Primer:

```
typedef int TIP;  
typedef struct cvor_st  
{  
    TIP inf;  
    struct cvor_st *desni;  
    struct cvor_st *levi;  
} BCVOR;  
  
...  
BCVOR *koren;
```

Primer



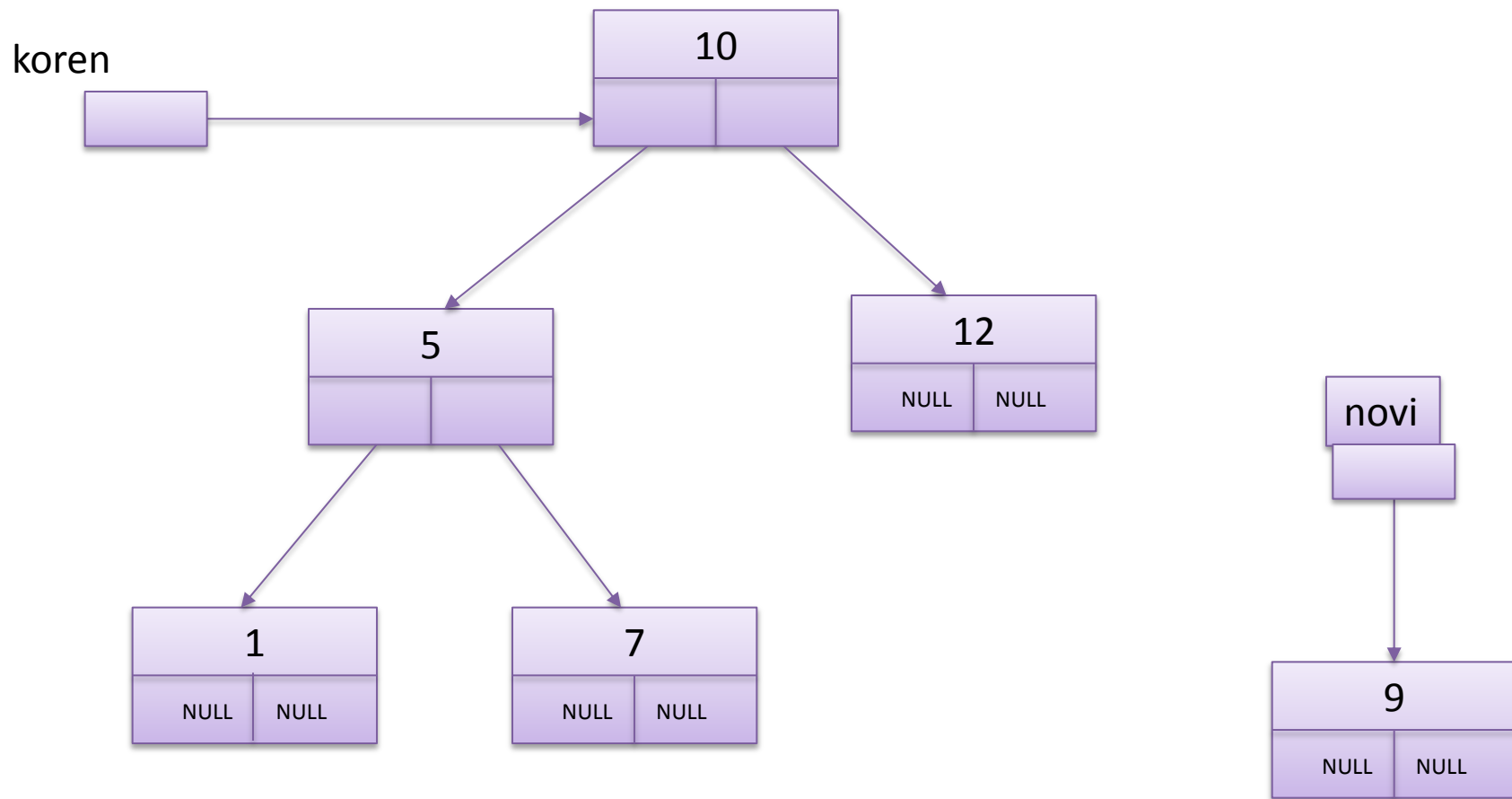
Operacije sa stablom

1. Ubacivanje u stablo
2. Pronalaženje čvora
3. Brisanje čvora
4. Brisanje stabla

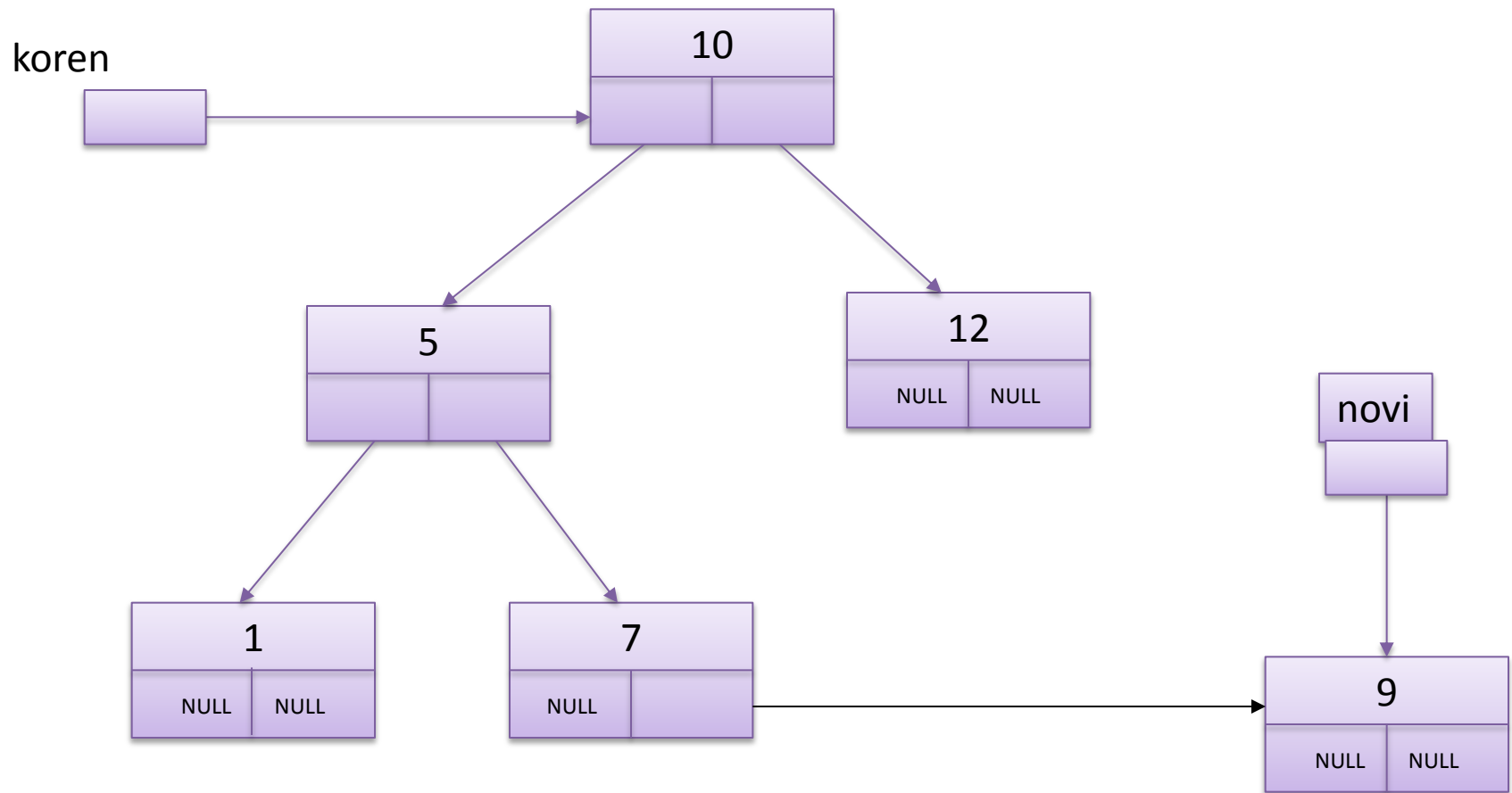
1. Ubacivanje u stablo

- Kreira se novi čvor
 - `novi = (BCVOR *)malloc(sizeof(BCVOR));`
- prođe se kroz stablo i kada se stigne do kraja stabla (naide na list), umetne se ispod njega (levo ili desno zavisi od sadržaja)

1. Ubacivanje u stablo



1. Ubacivanje u stablo



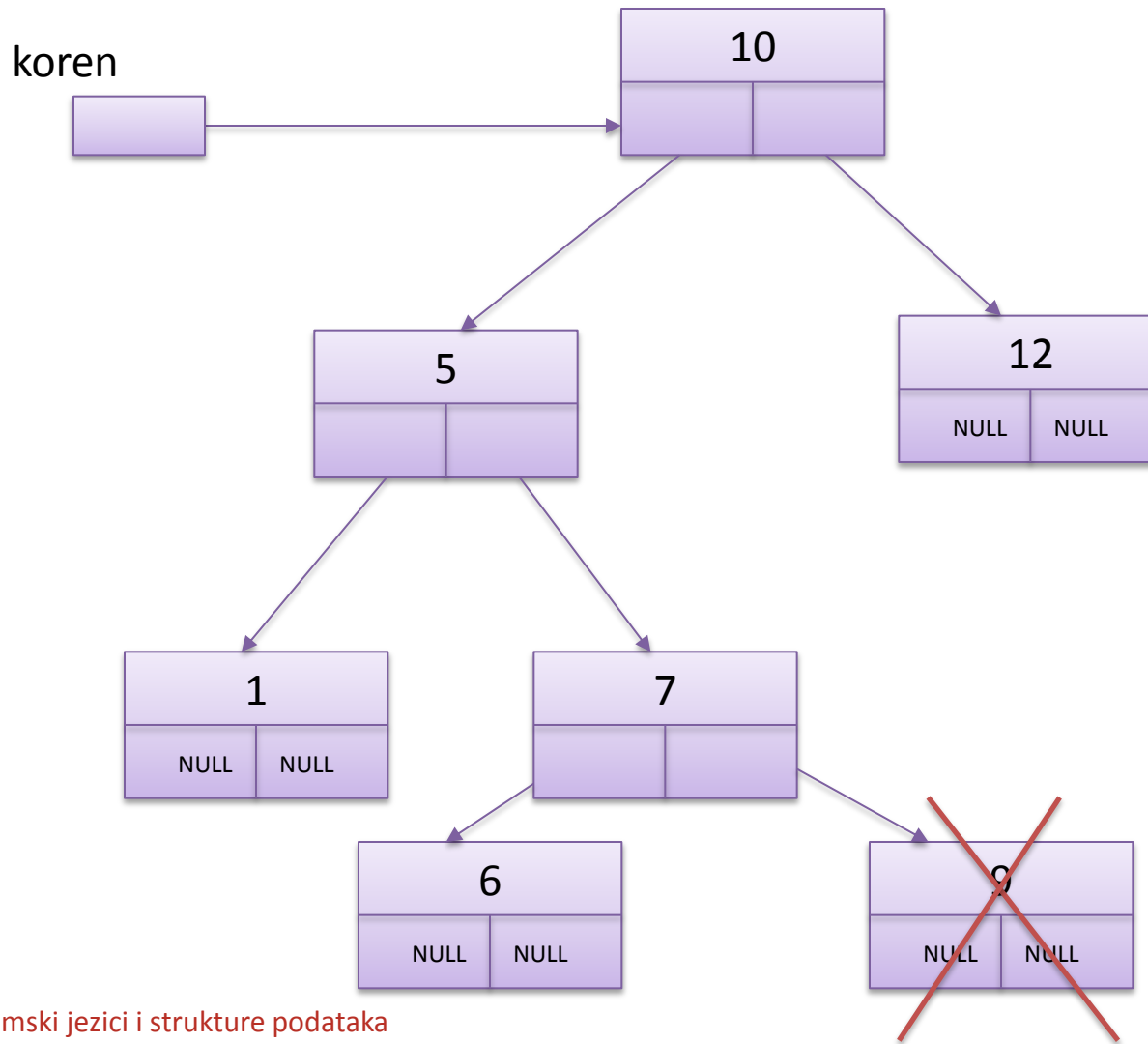
2. Pronalaženje čvora

- Krene se od korena i gleda da li je sadržaj traženog čvora manji ili veći od tekućeg
- Ako je manji, ide se u levo podstablo
- Ako je veći, ide se u desno podstablo
- Ako je jednak, vrati se pokazivač na tekući

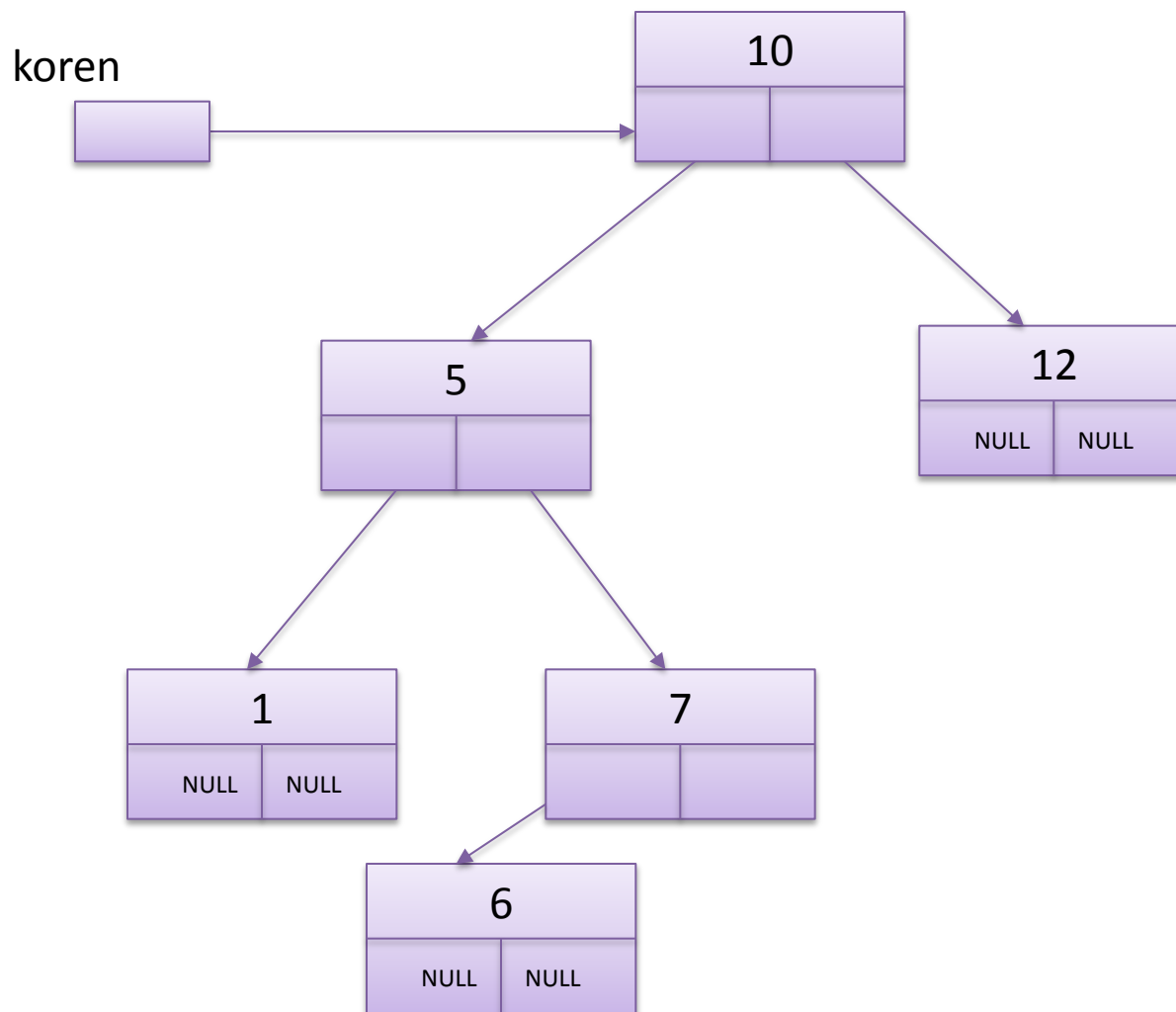
3. Brisanje čvora

- Krene se od korena i gleda se da li je sadržaj traženog čvora manji ili veći od tekućeg
- Ako je manji, ide se u levo podstablo
- Ako je veći, ide se u desno podstablo
- Ako je jednak, obriše se čvor i obavi se prevezivanje
 - a) ako je čvor koji se briše list, samo se obriše, a onaj čvor koji je ukazivao na njega se ažurira (NULL)
 - b) ako čvor koji se briše ima samo jedan podčvor, samo se obriše, a onaj koji je ukazivao na obrisani se preveže na podčvor
 - c) ako čvor koji se briše ima oba podčvora, on se obriše, a na njegovo mesto se uveže najmanji element u njegovom desnom podstablu
 - a) najmanji element koji je uvezan umesto izbrisanog se uklanja sa originalne lokacije i radi se isto prevezivanje

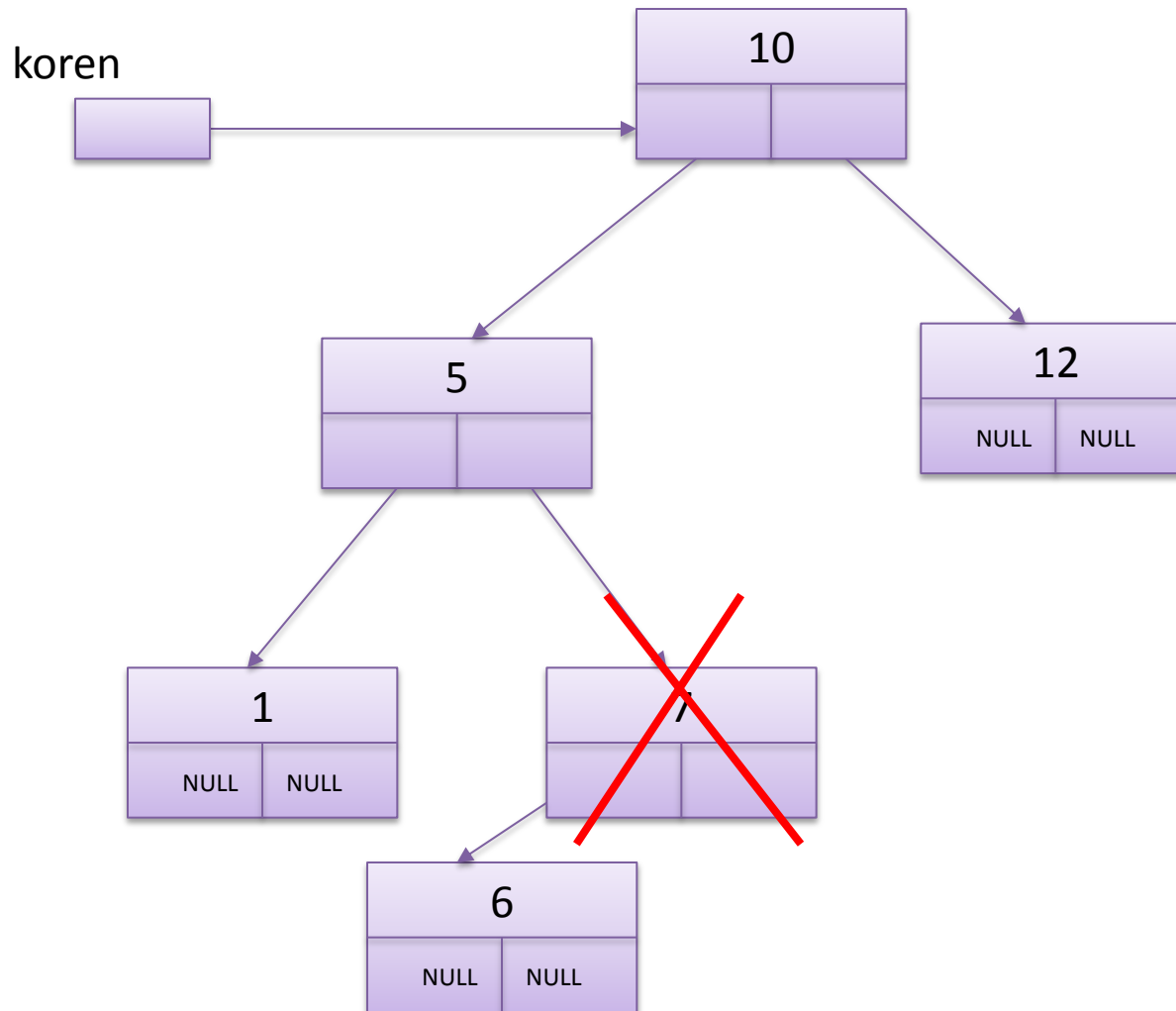
a) brisanje lista



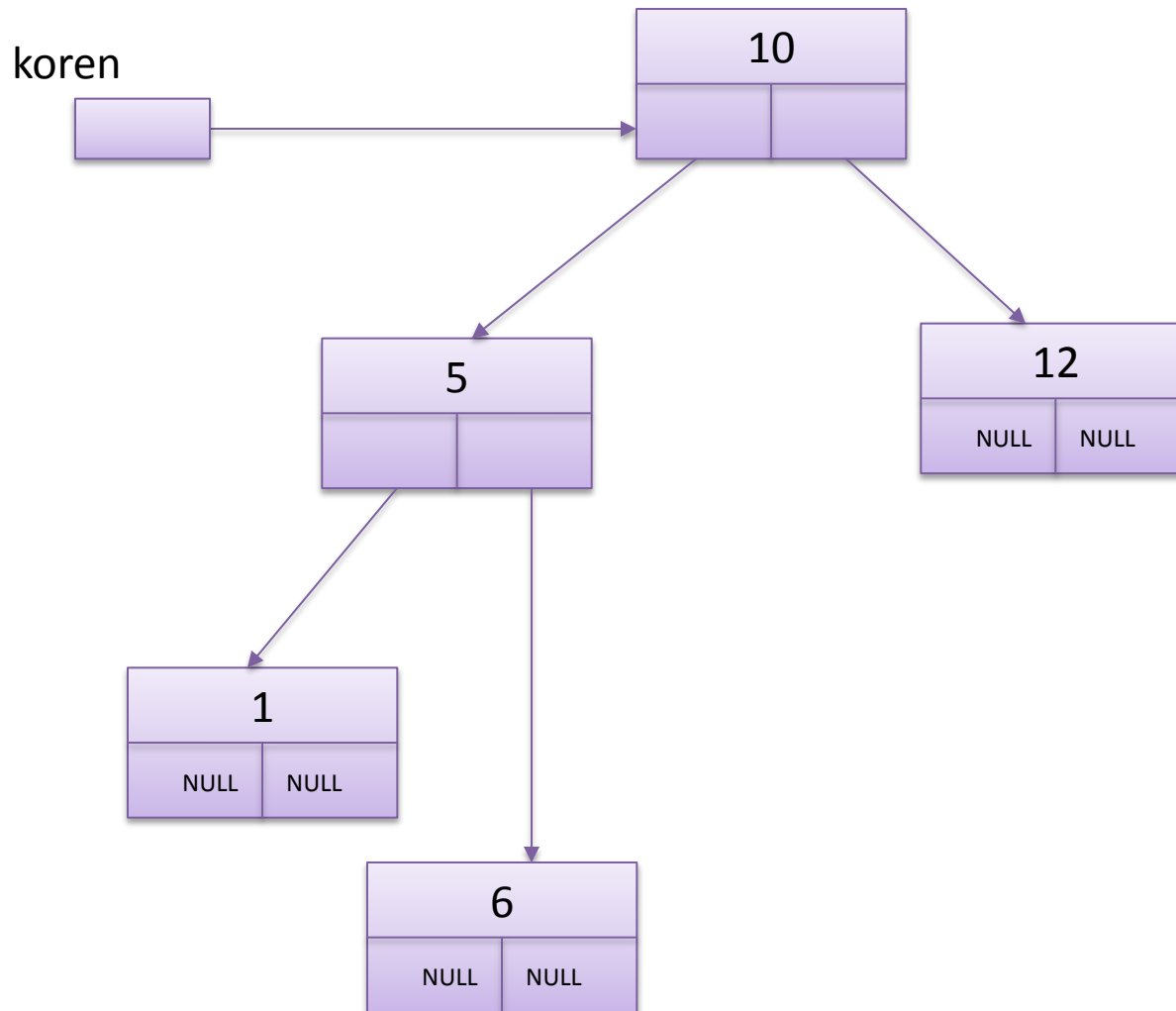
a) brisanje lista



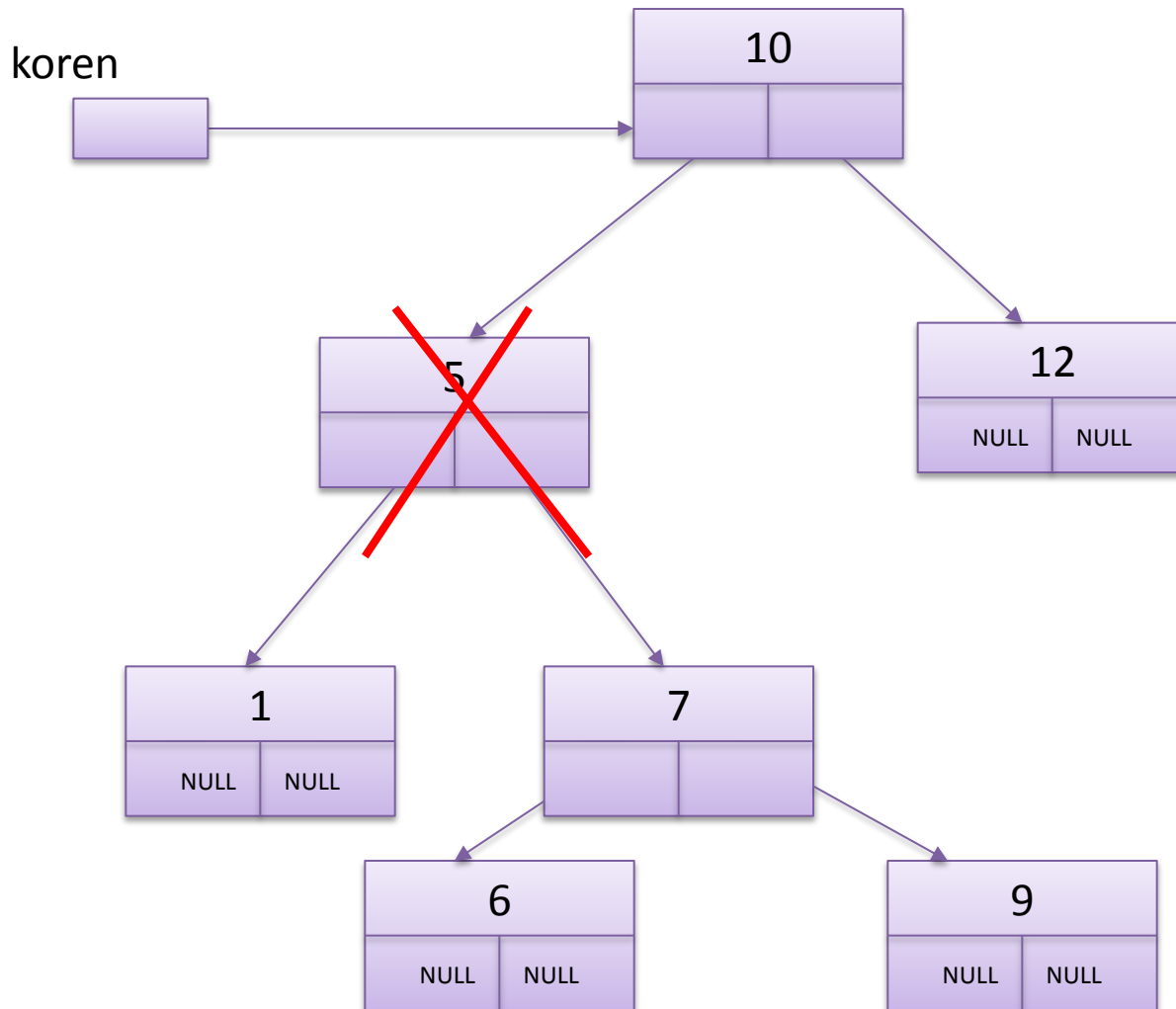
b) brisanje čvora sa jednim podstablom



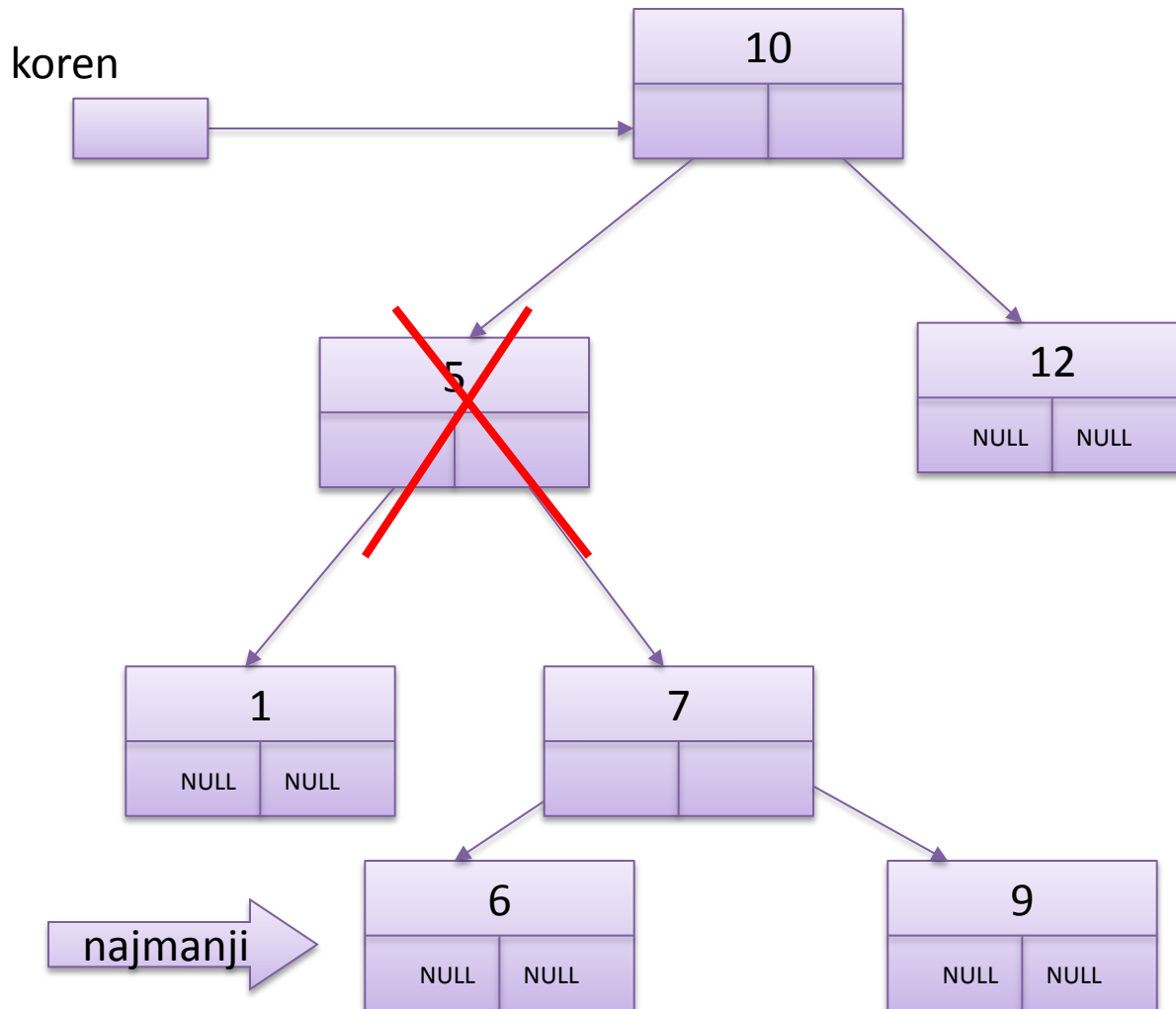
b) brisanje čvora sa jednim podstablom



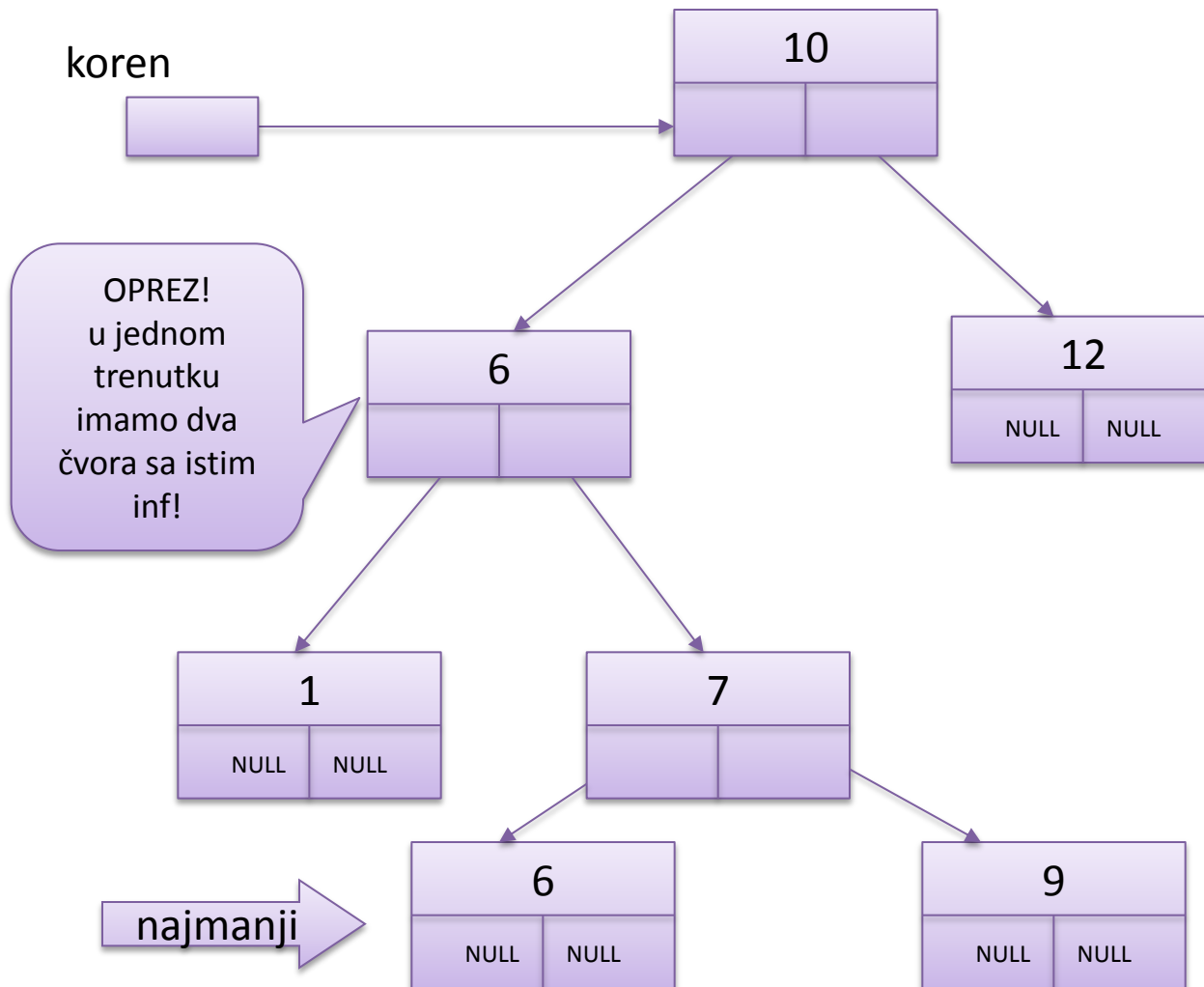
c) brisanje čvora sa dva podstabla



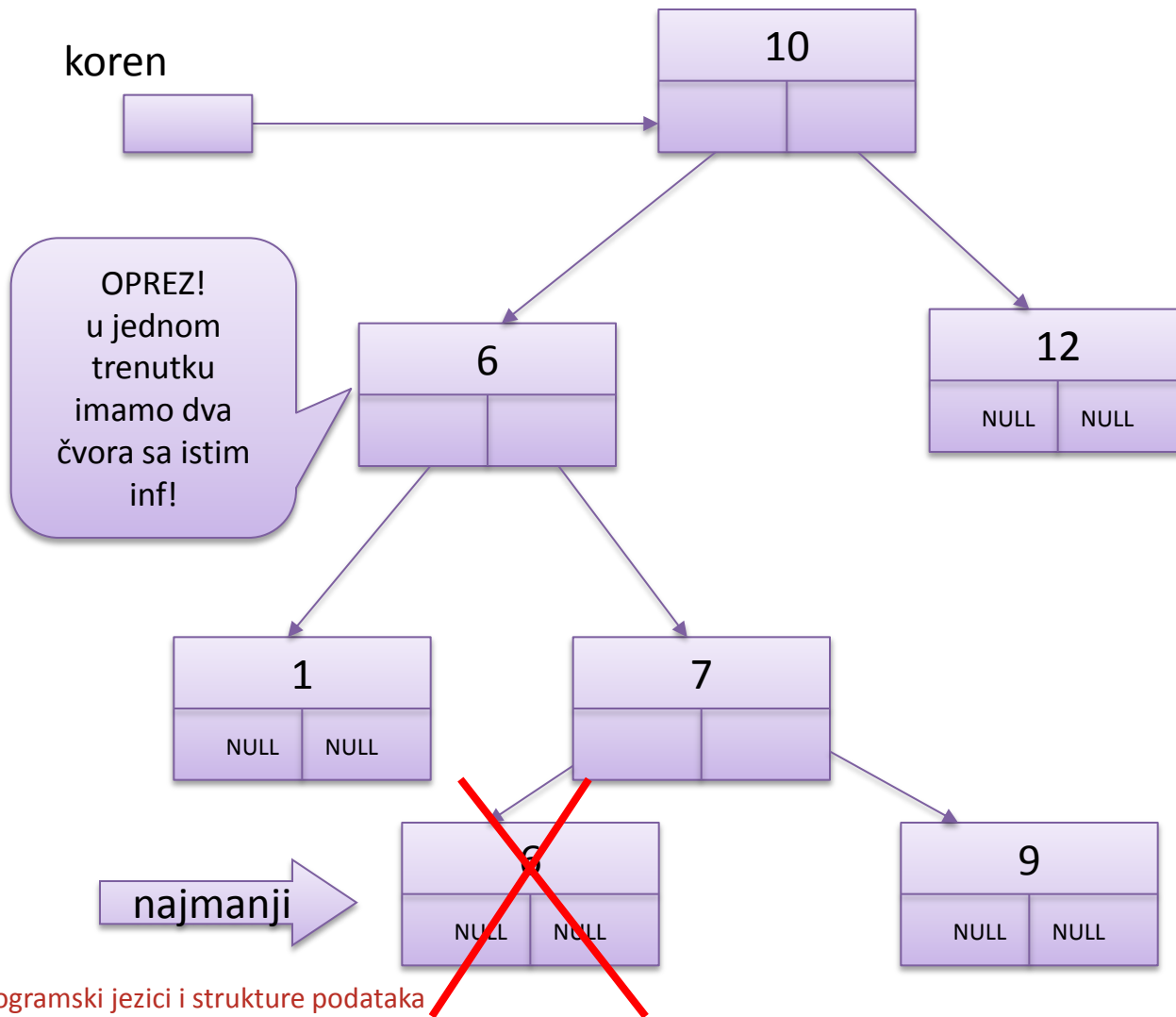
c) brisanje čvora sa dva podstabla



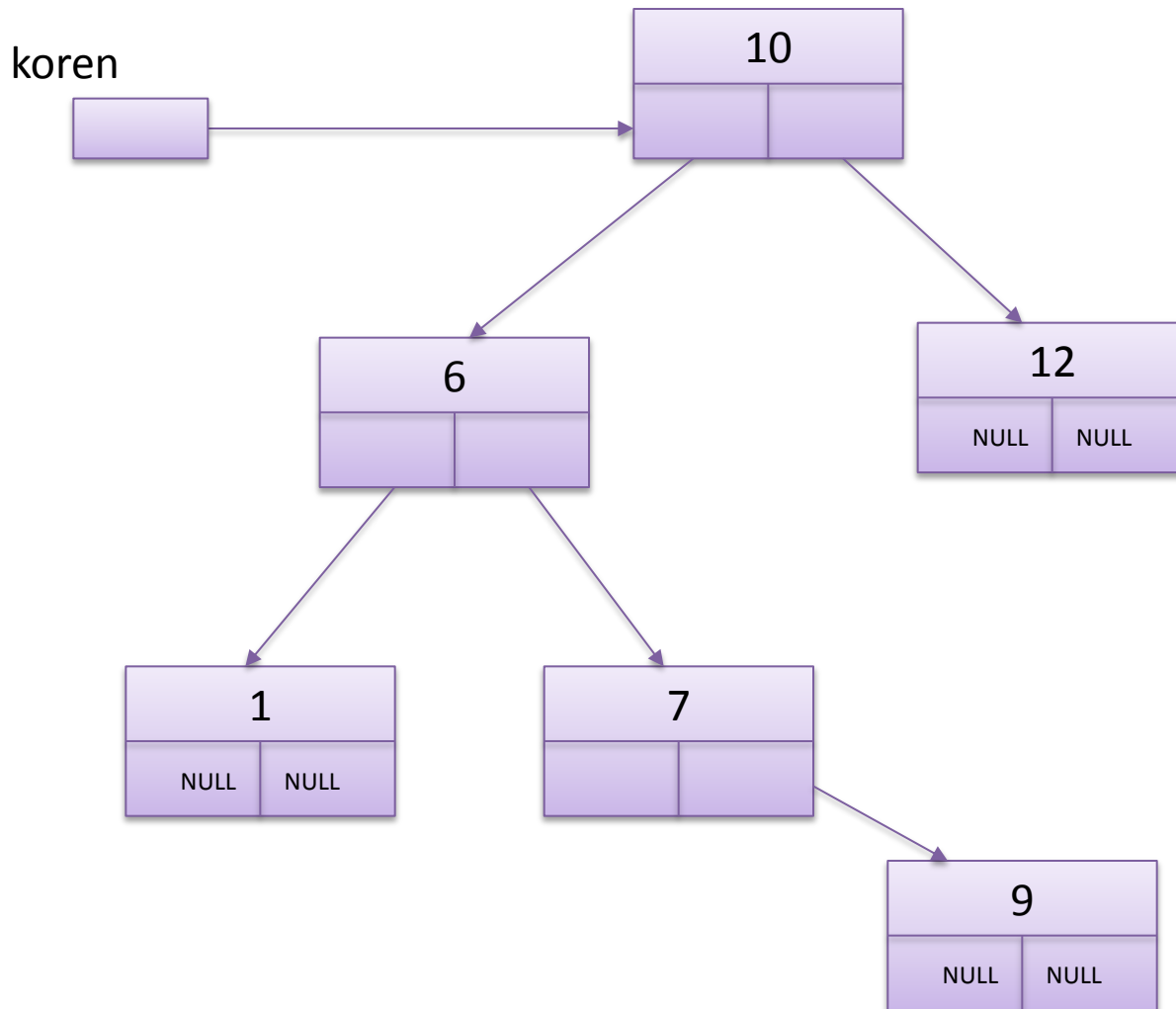
c) brisanje čvora sa dva podstabla



c) brisanje čvora sa dva podstabla



c) brisanje čvora sa dva podstabla



4. Brisanje stabla

- Krene se od korena i briše se levo i desno podstablo, pa onda koren

Zadatak 1.

- Napisati funkciju void ispis(BCVOR *koren) koja rekurzivno ispisuje sadržaj stabla
 - a) prvo ispisati sadržaj čvora a zatim njegovih podčvorova sa leva na desno
 - b) prvo ispisujemo sadržaj levog podstabla, zatim samog čvora i na kraju desnog podstabla (obilazak sa leva na desno)
 - c) obilazak sa desna na levo

Zadatak 2

- Napisati program koji manipuliše strukturom auto (Tema 5 – Strukture - Zadatak 3) i ima sledeće stavke u glavnom meniju:
 - učitavanje podataka o automobilima iz datoteke u binarno stablo
 - dodavanje novog automobila u stablo, po marki atumobila
 - pronalaženje automobila po marki i ispis ostalih podataka

Zadatak 2

- brisanje zadatog automobila
- snimanje podataka o automobilima iz binarnog stabla u datoteku
- pronalaženje najnovijeg automobila sa kubikažom ne većom od zadate i prikaz njegovih podataka na standardni izlaz

Zadatak 3

- Napisati rekurzivnu funkciju koja:
 - izračunava sumu elemenata nepraznog stabla (TIP je float);
 - izračunava broj elemenata binarnog stabla
 - ispisuje elemente svih listova binarnog stabla
 - ispituje da li se element E nalazi u binarnom stablu
 - određuje maksimalnu vrednost u binarnom stablu
 - određuje pokazivač na maksimalnu vrednost u binarnom stablu

Zadatak 3

- Napisati rekurzivnu funkciju koja:
 - određuje maksimalnu dubinu nepraznog stabla, tj. broj elemenata u najdužem putu od korena do lista (koren tretirati kao 0-ti nivo)
 - određuje broj elemenata na N-tom nivou binarnog stabla (koren tretirati kao 0-ti nivo)