

# Programski jezici i strukture podataka

14

STABLO

**NELINEARNE STRUKTURE**

# Stablo

- Stablo je nelinearna dinamička struktura podataka gde svaki čvor ima jednog i samo jednog prethodnika (u svetu stabla se naziva nadređeni), sem najvišeg čvora koji nema nadređenog i naziva se koren stabla, odnosno nema ili ima jednog ili više sledbenika (podređenih čvorova).
- Čvorovi stabla koji nemaju podređenih čvorova nazivaju se listovi stabla.
- Naziv ove strukture ilustruje njen oblik koji je pogodan za čuvanje pojava entiteta koji su po svojoj prirodi hijerarhijski, dekompozicioni ili analitičko-sintetički.

- Stablo je nelinearna struktura podataka koja predstavlja hijerarhijski odnos između elemenata.
- Primeri strukture stabla:
  - izgled sadržaja neke knjige,
  - rodoslovi porodice,
  - organizacione šeme.
- Stabla se u raznim pojavnim oblicima koriste u računarstvu:
  - operativni sistemi organizuju sistem datoteka i foldera na način koji odgovara strukturi stabla
  - predstavljanje izraza u programima prilikom njihovog prevodenja od strane prevodilaca
  - sintaksne kategorije se kod prevodilaca interno predstavljaju u obliku stabla izraza
- Indirektna primena stabla je za predstavljanje drugih struktura podataka:
  - disjunktne skupove
  - redove sa prioritetima

# Formalna definicija korenskog stabla

Stablo  $T$  je skup čvorova i skup grana između njih (tj. uređenih parova čvorova) sa sledećim osobinama:

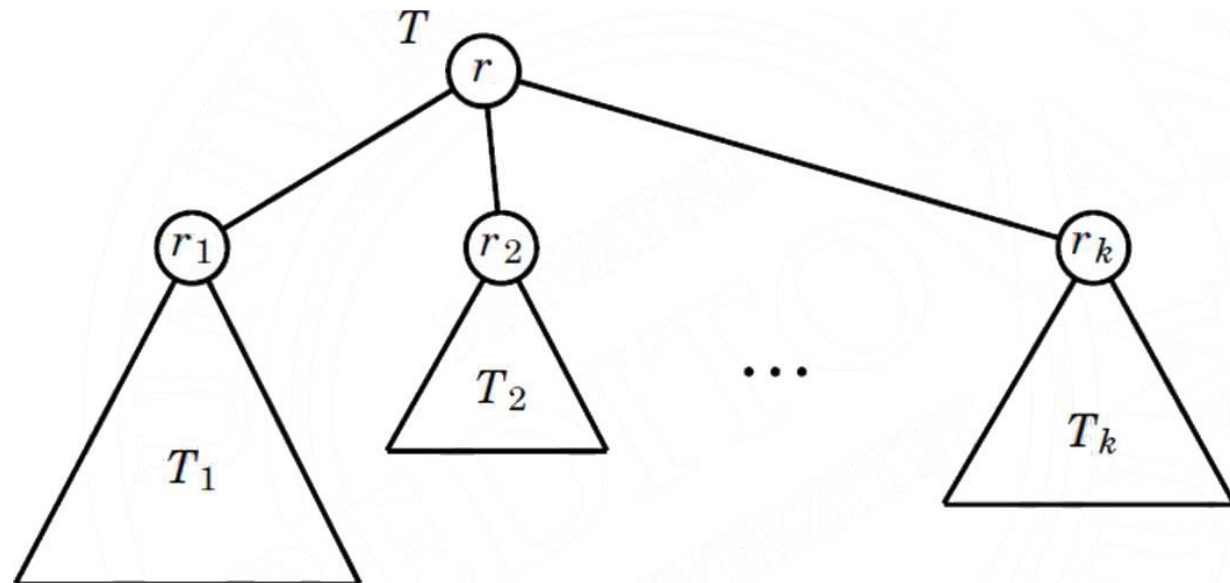
1. Ako je  $T$  neprazno stablo, ono ima jedinstven poseban čvor (koren) koji nema roditeljski čvor.
2. Svaki čvor  $u \in T$  osim korena ima jedinstven roditeljski čvor  $v$ .

Na osnovu ove formalne definicije stablo može biti prazno, što znači da takvo stablo nema nijedan čvor (i nijednu granu).

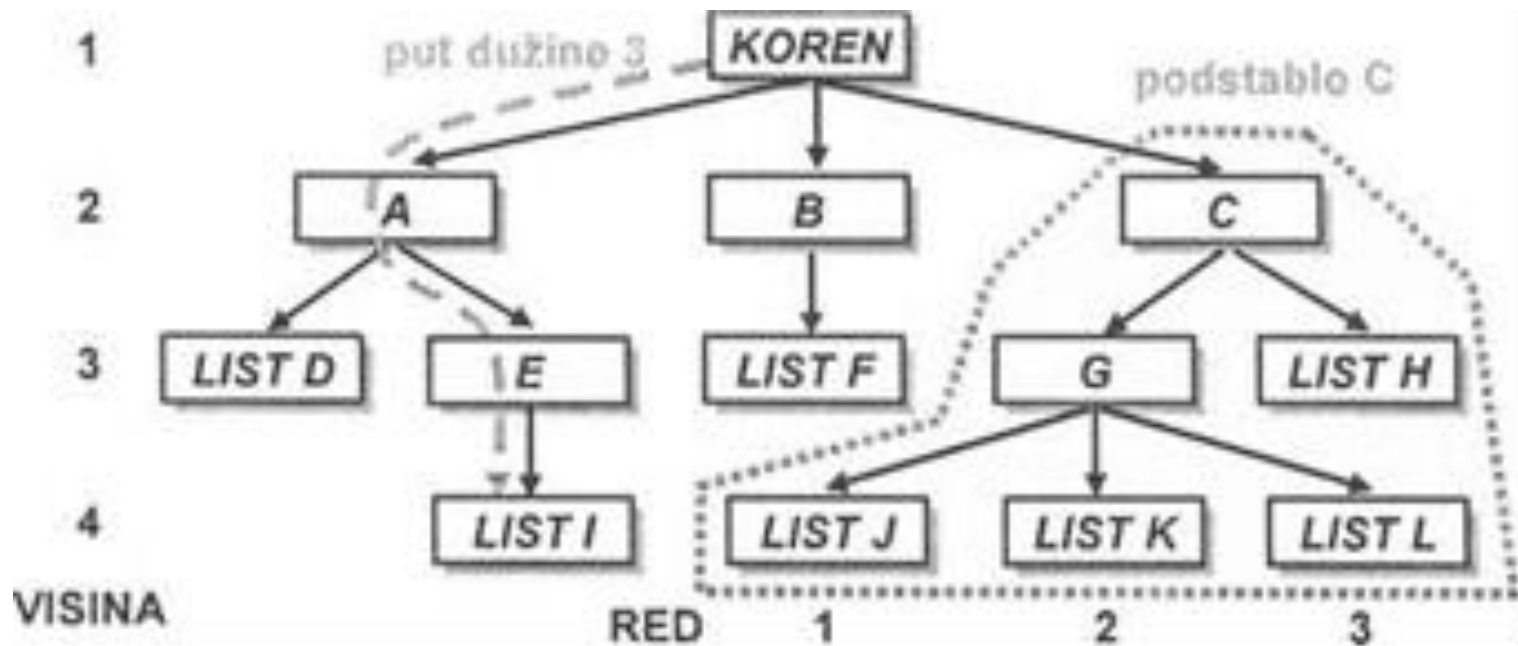
# Rekurzivna definicija korenskog stabla

Na osnovu prethodnog stabla se definiše i rekurzivno na način kojim se veća stabla konstruišu od manjih:

Stablo  $T$  je prazno ili se sastoji od čvora  $r$ , koji se naziva koren stabla  $T$ , i (moguće praznog) skupa stabala  $T_1, T_2, \dots, T_k$  čiji su koreni deca čvora  $r$ .



# Uopšteno stablo



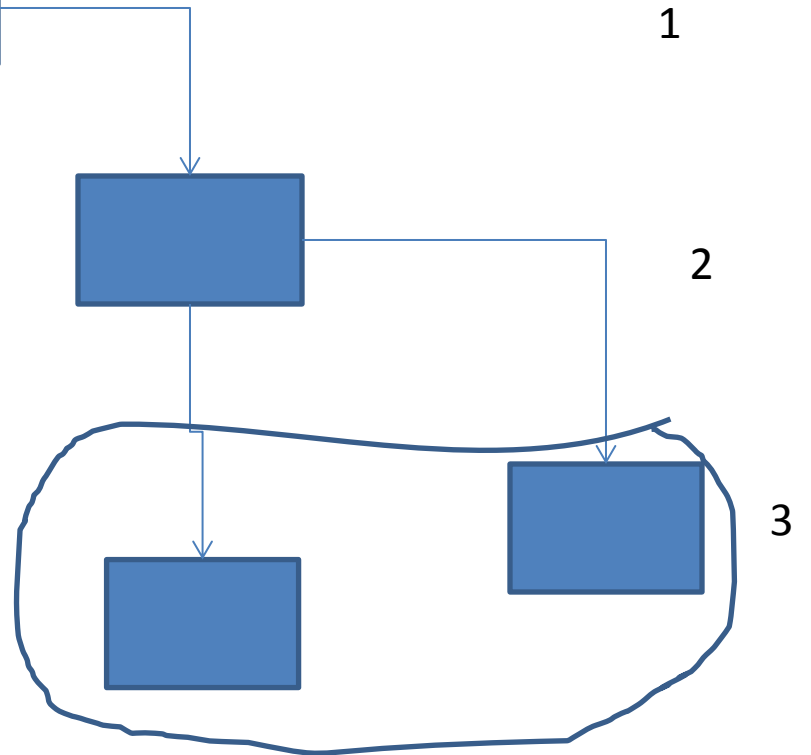
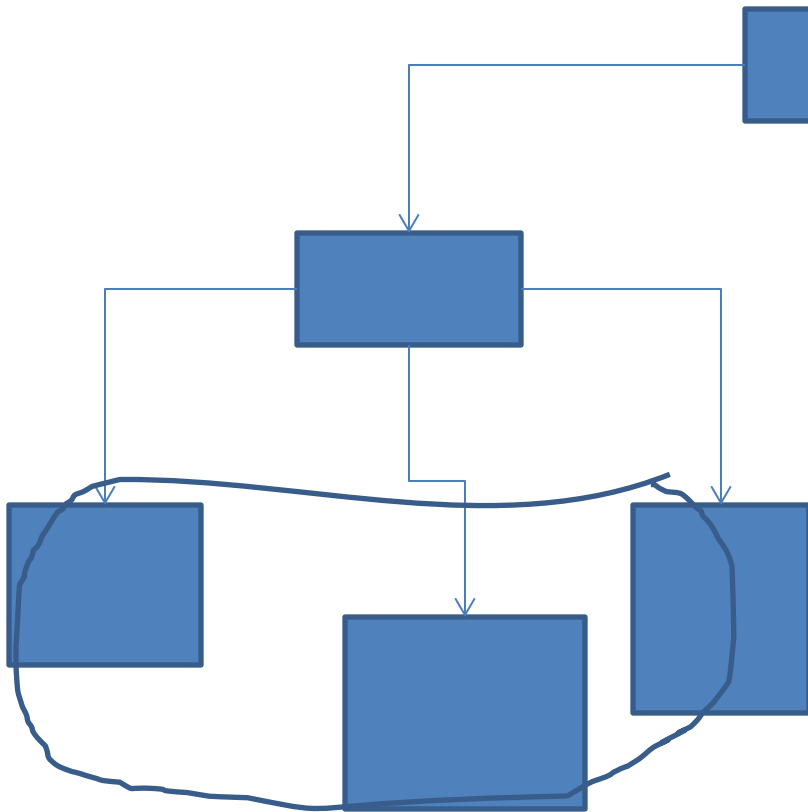
# Stablo

- Stablo obuhvata sve strukture podataka kojima pripada digraf tipa stablo.
- Ključna reč koja opisuje stablo je hijerarhija.
- Elementi stabla su u odnosu:
  - podređeni
  - nadređeni
- Skup čvorova koji su povezani granama se naziva **lanac**.
- Orjentisani lanac se naziva put.
- Ako su svaka dva čvora povezana jednim lancem kažemo da je digraf **slabo povezan**.

# Orjentisano stablo

- Postoji tačno jedan čvor u koji ne ulazi ni jedna grana (ulazni stepen je nula).
- U sve ostale čvorove ulazi tačno po jedna grana (ulazni stepen je jedan).
- Digraf je povezan, odnosno svaka dva čvora mogu se spojiti nizom uzastopnih grana ne uzimajući u obzir njihovu orijentaciju (digraf je slabo povezan).
- Visina stabla je jednaka je broju čvorova na najdužem putu u stablu.

- **Kompletno stablo** (reda  $n$ ): svi elementi osim listova imaju izlazni stepen  $n$ .
- **Puno stablo**: svi putevi od korena do listova su iste dužine.
- Svaki element strukture je dostupan preko korena.
- **Balansirano stablo**: ono stablo u kojem se broj elemenata podstabla na istom hijerarhijskom nivou razlikuje najviše za 1.



Balansirano stablo

# Struktura tipa stablo

- Digraf je tipa stablo
- Dozvoljen pristup svakom elementu
- Dodavanje i uklanjanje u domenu dozvoljenog (n-arno stablo, uopšteno-generalizovano stablo)

# Struktura tipa stablo

- Slično kao u slučaju osnovnih struktura podataka (nizova, listi, stekova, redova), da bismo stabla koristili u programima za organizaciju kolekcije podataka, moramo imati način za predstavljanje stabala u računaru i implementirati skup korisnih operacija nad stablima.

- U radu sa stablima se može pojaviti potreba za vrlo raznolikim operacijama:
  - dodavanje,
  - uklanjanje
  - određivanje deteta datog čvora,
  - operacije koje kao rezultat daju roditelja,
  - sasvim levog deteta
  - desnog brata datog čvora...

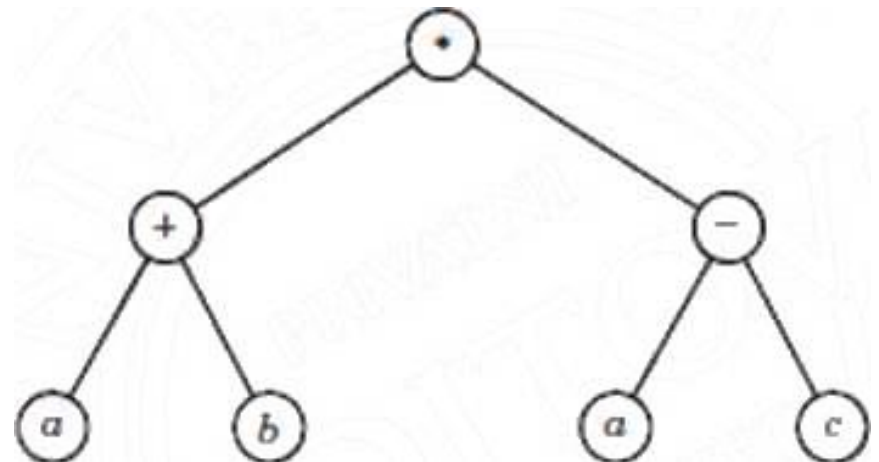
- Radi lakšeg programiranja korisno bi bilo realizovati i upitne operacije koje određuju da li je dato stablo prazno, ili da li je dati čvor koren, interni, eksterni i slično.
- Takođe i pomoćne operacije koje obuhvataju:
  - postupke konstruisanja novog stabla,
  - određivanja broja čvorova (veličine) stabla,
  - određivanje visine stabla i tako dalje.
- Međutim, ukoliko se koristi odgovarajuća reprezentacija stabla, sve ove operacije se mogu relativno lako implementirati

- Stablo predstavlja kolekciju podataka tako što se pojedinačni podaci u kolekciji nalaze u čvorovima stabla.
- stablo na slici predstavlja aritmetički izraz:

$$(a + b) * (a - c)$$

- koren tog stabla sadrži operator \* za množenje, a njegova deca su koreni dva podstabla koja predstavljaju podizraze:

$$a + b \text{ i } a - c$$

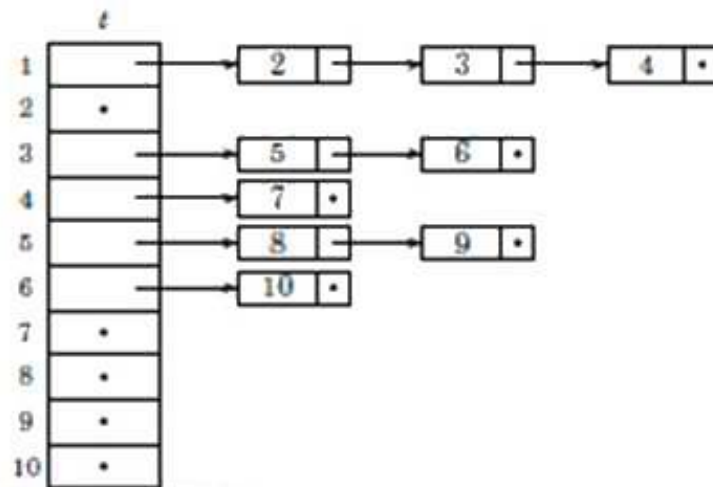
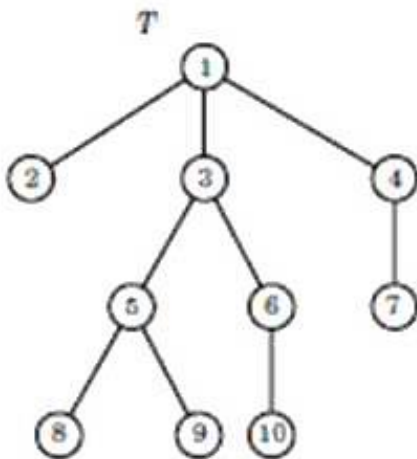


- Slično kao kod osnovnih struktura podataka, pojedinačni podaci u čvorovima stabla se na opštem nivou poistovećuju sa jednoznačnom vrednošću koja se naziva njihov ključ.
- Prirodna reprezentacija stabla bila bi da se svaki čvor stabla predstavlja:
  - objektom koji obuhvata polje ključa.
  - više polja u kojima se nalaze pokazivači na svako dete tog čvora i
  - polje za pokazivač na roditelja čvora.

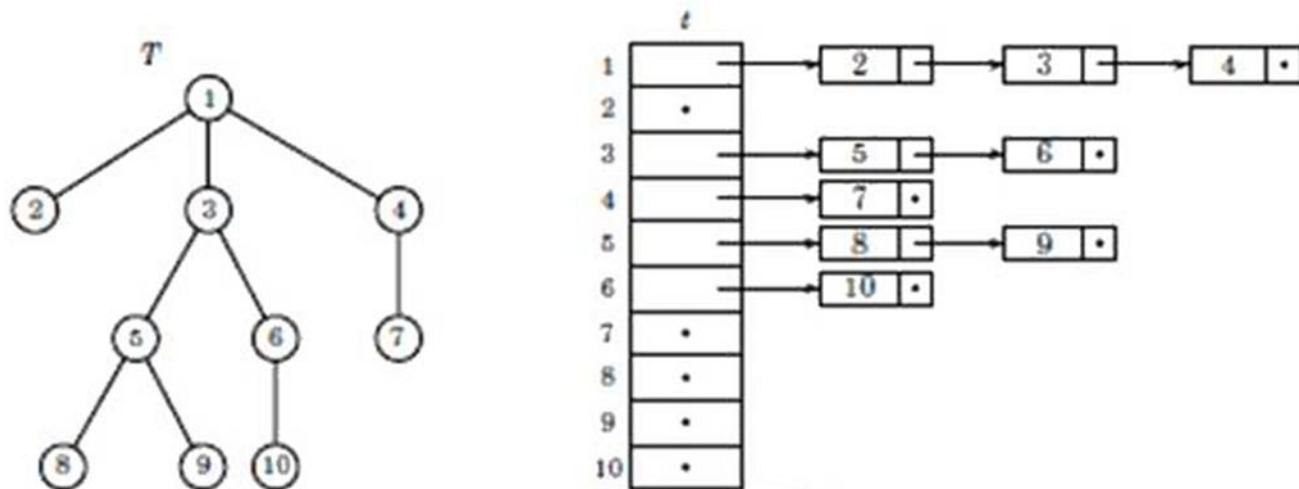
- Reč je o opštim stablima, odnosno broj dece po čvoru može biti vrlo različit i nije unapred ograničen, ovaj pristup nije najbolji jer proizvodi loše iskorišćenje memorijskog prostora.
- Zato se obično koriste dva druga načina za predstavijanje opštih stabala:
  - predstavljanje stabla listama dece čvorova
  - predstavljanje stabla pomoću sasvim levog deteta i desnog brata čvorova

# Predstavljanje stabla listama dece čvorova

- Jedan način za prevazilaženje problema nepoznatog broja dece nekog čvora u opštem stablu je formiranje povezane liste dece svakog čvora.
- Može se lako generalizovati za predstavljanje složenijih struktura kao što su grafovi.



- Imamo jedan niz pokazivača  $t$  čiji elementi predstavljaju čvorove i zato je njegova veličina jednaka ukupnom broju čvorova u stablu.
- Pored toga, svaki element niza  $t$  pokazuje na početak povezane liste čvorova stabla.
- Pri tome, elementi liste na koju pokazuje element niza  $t_i$  su čvorovi-deca čvora  $i$  tako da poredak elemenata u listi odgovara poretku dece sleva na desno u stablu.
- Na primer, element  $t_5$  pokazuje na listu 8->9 pošto su čvorovi 8 i 9 redom deca čvora 5.

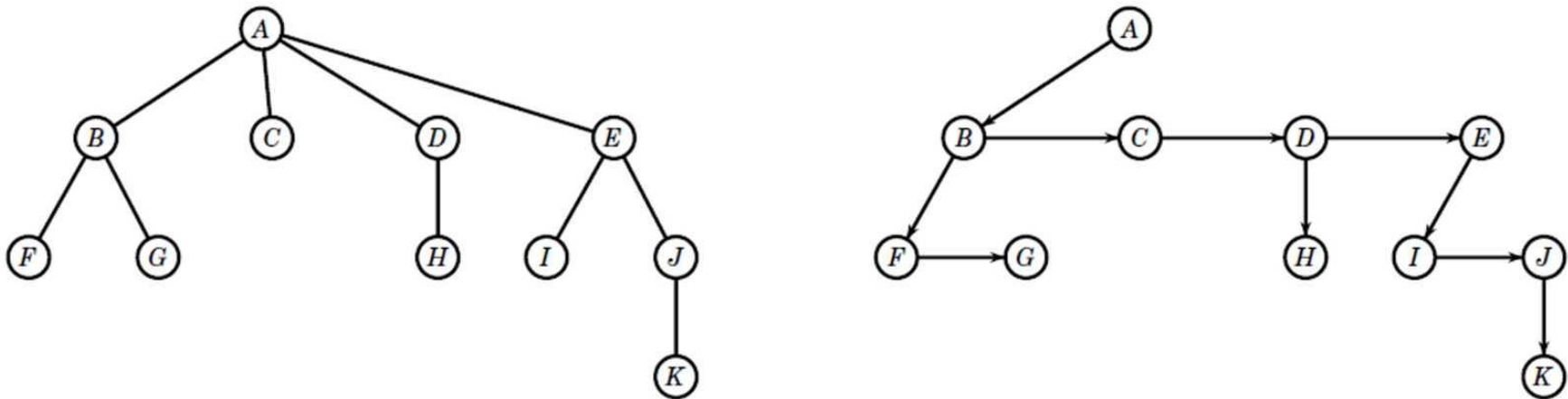


- Jedan nedostatak ovog načina predstavljanja stabla je to što ne možemo lako da konstruišemo veća stabla od manjih.
- To je posledica toga što svako stablo ima svoj niz pokazivača za svoje čvorove.
- Da bismo napravili novo stablo, recimo, od datog čvora kao korena i dva data podstabla  $T_1$  i  $T_2$ , morali bismo da kopiramo  $T_1$  i  $T_2$  u treće stablo i da dodamo listu za novi koren čija su deca koreni za  $T_1$  i  $T_2$ .

# Predstavljanje stabla pomoću sasvim levog deteta i desnog brata čvorova

- Drugo rešenje za problem predstavljanja opštih stabala je *predstavljanje sasvim levog deteta i desnog brata* svakog čvora stabla.
- U ovoj reprezentaciji svaki čvor pored ključa sadrži samo još dva pokazivača:
  - jedan pokazuje na njegovog sasvim levog deteta
  - drugi na njegovog desnog brata
- Ovi pokazivači imaju specijalnu vrednost **null** ukoliko čvor nema dece ili nema desnog brata.

- Na slici je za svaki čvor stabla strelicom nadole označen pokazivač na sasvim levog deteta, a strelicom nadesno je označen pokazivač na desnog brata.



- Sve operacije nad stablima, osim određivanja roditelja datog čvora, mogu se efikasno realizovati ako je stablo predstavljeno pomoću sasvim levog deteta i desnog brata svih čvorova.
- Ako je potrebno i da operacija određivanja roditelja bude efikasna, to se može lako rešiti dodavanjem još jednog pokazivača svakom čvoru koji direktno pokazuje na njegovog roditelja.

# n-arno stablo

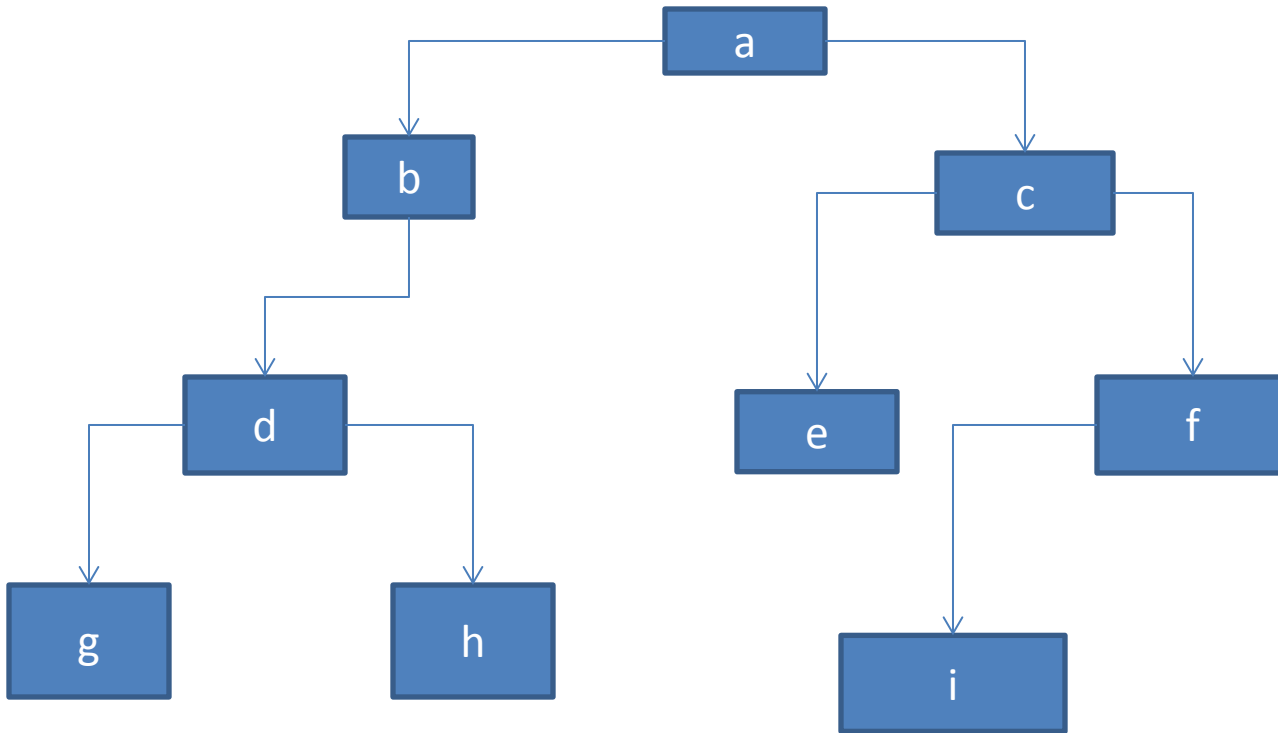
- je stablo reda  $n$ ,  $n$  je propisan i ograničen
- u skupu podređenih svakog elementa postoji eksplicitno uređenje
- najčešće korištena su binarna stabla,  $n=2$

# N-arno (binarno) stablo



# Obilazak binarnog stabla

- Obilazak s vrha ka dnu
  - pristupi nadređenom
  - pristupi levom podstablu
  - pristupi desnom podstablu
- Obilazak s leva u desno
  - pristupi levom podstablu
  - pristupi nadređenom
  - pristupi desnom podstablu
- Obilazak sa dna ka vrhu
  - pristupi levom podstablu
  - pristupi desnom podstablu
  - pristupi nadređenom
- Sve tri metode počinju od korena pošto se samo za njega zna gde je.



Obilazak s vrha ka dnu: a,b,d,g,h,c,e,f,i

Obilazak s leva u desno: g,d,h,b,a,e,c,i,f

Obilazak sa dna ka vrhu: g,h,d,b,e,i,f,c,a

Obilazak s vrha ka dnu  
 pristupi nadređenom  
 pristupi levom podstablu  
 pristupi desnom podstablu

Obilazak s leva u desno  
 pristupi levom podstablu  
 pristupi nadređenom  
 pristupi desnom podstablu

Obilazak sa dna ka vrhu  
 pristupi levom podstablu  
 pristupi desnom podstablu  
 pristupi nadređenom

# Fizička realizacija n-arnog stabla

- Fizička realizacija n-arnog stabla je u opštem slučaju spregnuta.
- Za mala i nepromenljiva stabla može biti sekvencijalna.
- Spregnuta fizička realizacija značajno uproštena u odnosu na uopšteno stablo (broj podređenih maksimalno  $n$ )
- Dovoljno je dopuniti informacioni sadržaj sa tačno  $n$  polja za pokazivače.

# Spregnut fizička realizacija stabla reda $n$

- Deskriptor sadrži:
  - Indikator tipa strukture
  - Ime stabla
  - Adresu korena
  - Opis elementa
  - Red stabla
- Element sadrži:
  - informacioni sadržaj
  - $n$  polja za pokazivače

# Binarna stabla pristupa

- Binarno stablo kod kojeg se u informacionom sadržaju nalazi i identifikator iz skupa sa strogim uređenjem, a raspored u njemu je takav da:
  - Svi elementi u levom podstablu elementa imaju manje vrednosti identifikatora od elementa samog.
  - Svi elementi u desnom podstablu elementa imaju veće vrednosti identifikatora od elementa samog.
- Namena binarnog stabla pristupa je skraćenje postupka traženja, a da dodavanje i uklanjanje bude moguće.

# Razlike između binarnog stabla i binarnog stabla pristupa

- Binarno stablo: Binarno stablo je stablo kod koga svaki čvor ima dva najviše dva deteta. U binarnom stablu, levo i desno dete mogu imati vrednosti nezavisne od vrednosti roditelja.
- Binarno stablo pristupa: Kod binarnog stabla pristupa, levo dete sadrži vrednost manju od roditelja, a desno dete sadrži vrednost koja je veća od vrednosti roditelja.

# Prednost binarnog stabla pristupa

- Algoritmi sortiranja i algoritmi pristupa kao npr. pretraga u dubinu (in-order) mogu biti veoma efikasni.
- Ostale prednosti su:
  - Binarno stablo pristupa ima brzo ubacivanje i brisanje elemenata kada je balansirano.
  - Kod je jednostavniji nego kod ostalih struktura podataka.
  - Čuva ključeve u čvorovima na takav način da se pretraga, ubacivanje i brisanje elemenata može uraditi efikasno.
  - Veoma jednostavna implementacija.
  - Čvorovi u stablu su dinamički po prirodi.

# Nedostaci

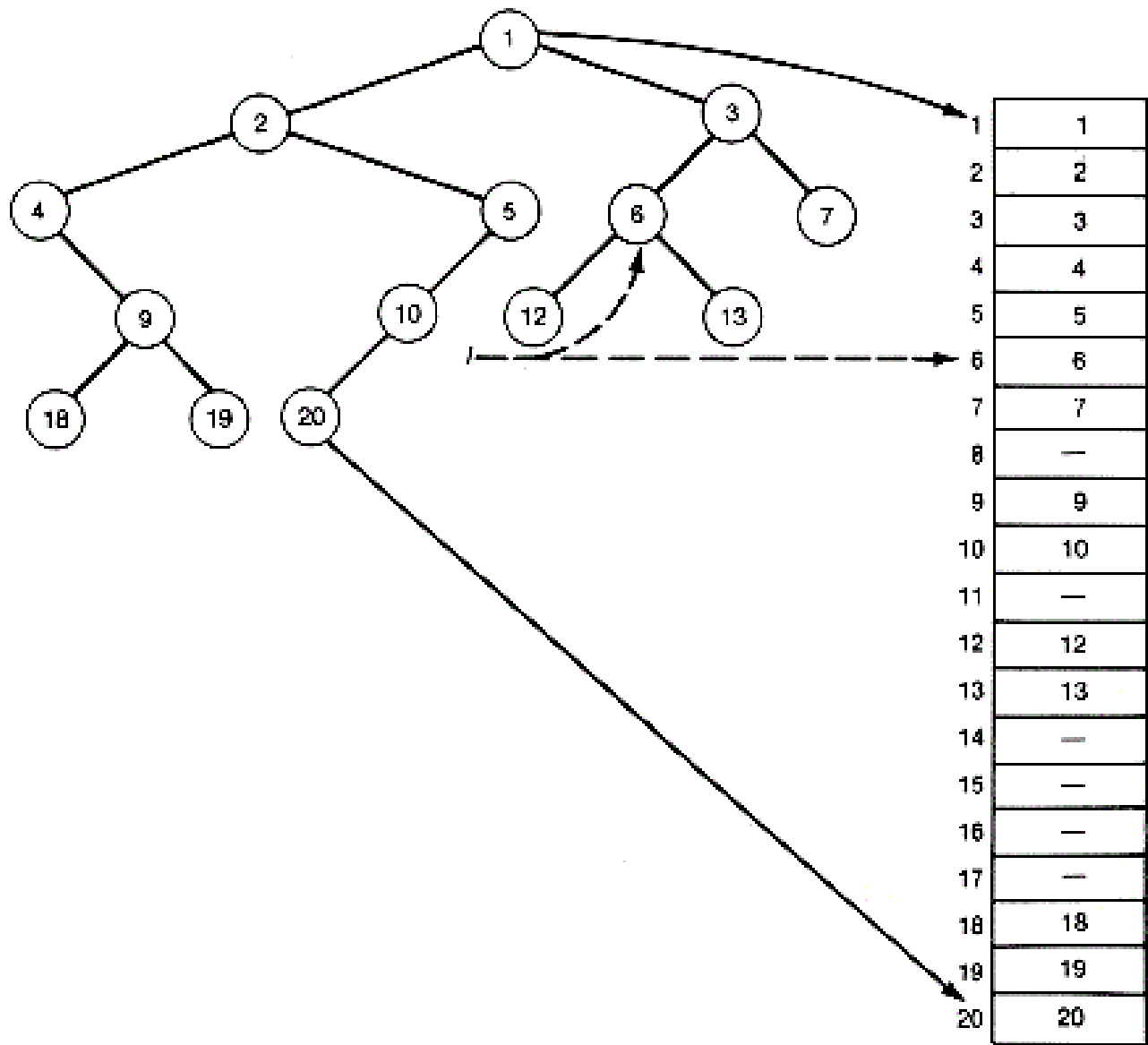
- Oblik binarnog stabla pristupa zavisi samo od reda ubacivanja elemenata, i može biti degenerisano.
- Kada ubacujemo ili tražimo element u binarnom stablu pristupa, ključ svakog posećenog čvora mora da se uporedi sa ključem elementa kog ubacujemo ili tražimo.
- Ključevi u binarnom stablu pristupa mogu biti dugački, što može uticati na vremensku složenost.

# Degenerisano stablo pristupa

- Degenerisano stablo je stablo gde za svaki roditeljski čvor, postoji samo jedno dete čvor.
- Ono je nebalansirano i, u najgorem slučaju, performanse se spuštaju na nivo povezane liste.
- Ako funkcija za dodavanje čvora ne podržava rebalansiranje, onda se lako može stvoriti degenerisano stablo ako mu dodajemo podatke koji su već sortirani.
- Ovo znači da u merenju performansi, stablo će se na kraju ponašati kao povezana lista.

# Sekvencijalna fizička realizacija

- Odvaja se memorijski prostor sa relativnim adresama.
- Koren se smešta u relativnu adresu jedan.
- Za ostale elemente važi: ako je element na relativnoj adresi  $i$ , levi podređeni je na adresi  $2*i$ , a desni na adresi  $2*i+1$ .
- Pristup je direktan pošto se adresa može izračunati unapred.



1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	—
9	9
10	10
11	—
12	12
13	13
14	—
15	—
16	—
17	—
18	18
19	19
20	20

prethodnik čvora i

čvor i

levi naslednik čvora i

desni naslednik čvora i

# Binarno stablo pristupa spregnuta realizacija

- Binarno stablo pristupa je binarno stablo u kome važi: ključ svakog čvora veći je od ključeva svih čvorova levog podstabla, a manji od ključeva svih čvorova desnog podstabla.
- Pretpostavimo zbog jednostavnosti da su ključevi svih čvorova različiti.
- Spregnuta realizacija **Z43.c**