

# Adatszerkezetek

## 06. Fák

Vekov Géza

2023. március 29.



Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

# Hierarchikus adatszerkezet

# Hierarchikus adatszerkezet

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Definíció

- A szekvenciális adatszerkezet általánosítása
- Minden adatelemnek:
  - pontosan egy elődje és
  - tetszőleges számú (akár 0) utóda lehet
  - (kivéétel az első elem)

## Példa: Hierarchikus adatszerkezetek

- Fa
- Hierarchikus lista

## Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

### Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

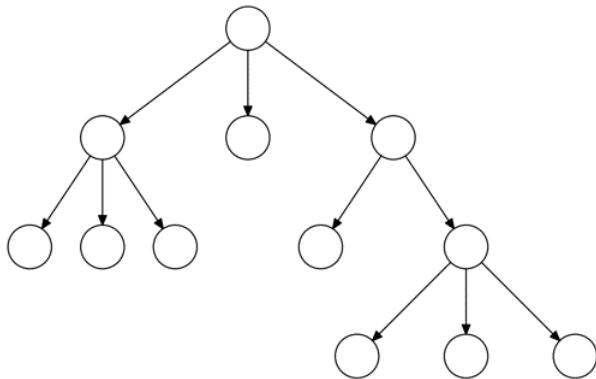
Tökéletesen  
egyensúlyozott

Bináris keresőfa

Fa

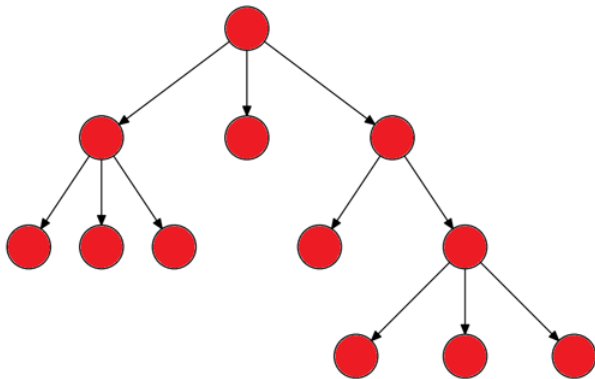
## Tulajdonságok

- homogén
- dinamikus
- hierarchikus adatszerkezet



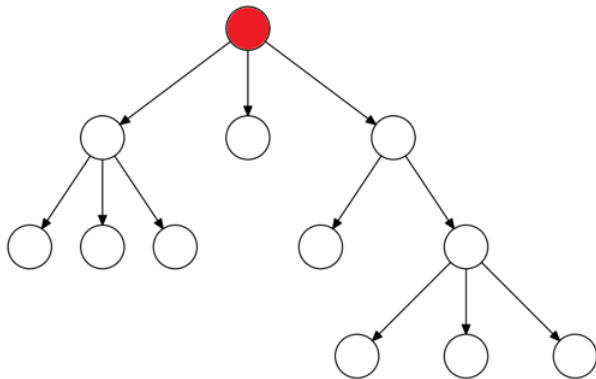
## Alapfogalmak

- **csúcs, csomópont**
- gyökér
- levél
- belső elem
- él
- út
- részfa
- szint
- magasság



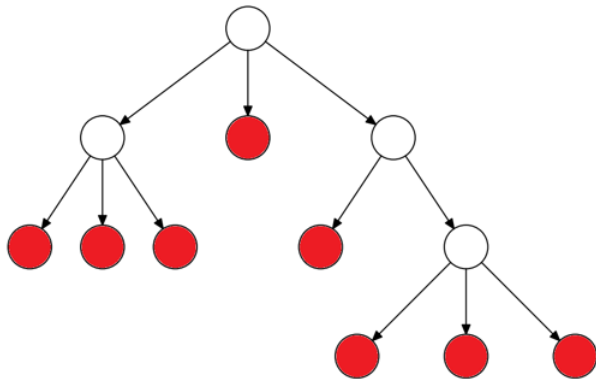
## Alapfogalmak

- csúcs, csomópont
- **gyökér**
- levél
- belső elem
- él
- út
- részfa
- szint
- magasság



## Alapfogalmak

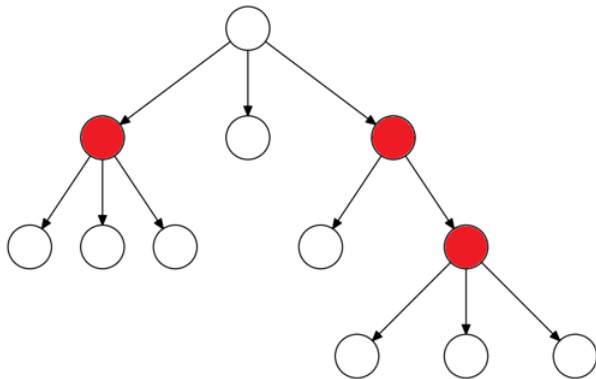
- csúcs, csomópont
- gyökér
- **levél**
- belső elem
- él
- út
- részfa
- szint
- magasság





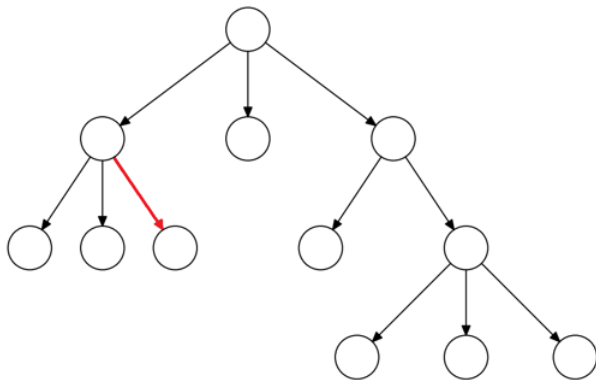
## Alapfogalmak

- csúcs, csomópont
- gyökér
- levél
- **belső elem**
- él
- út
- részfa
- szint
- magasság



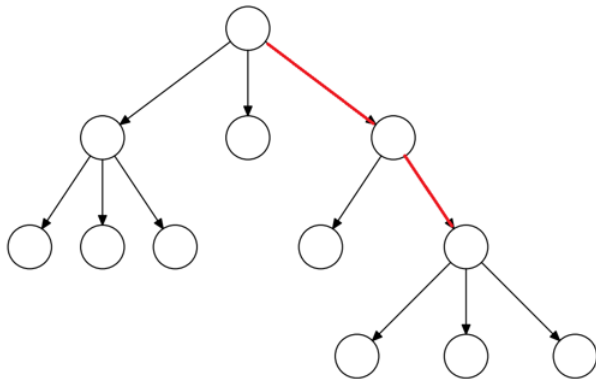
## Alapfogalmak

- csúcs, csomópont
- gyökér
- levél
- belső elem
- él
- út
- részfa
- szint
- magasság



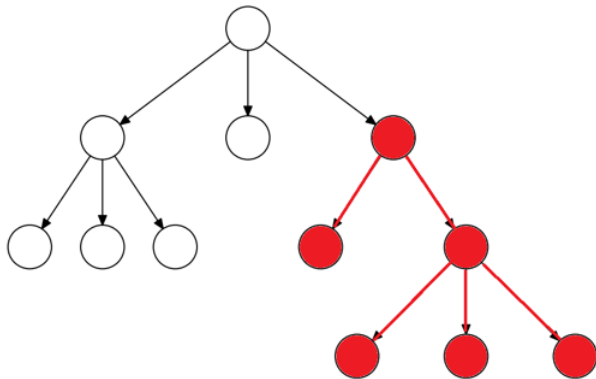
## Alapfogalmak

- csúcs, csomópont
- gyökér
- levél
- belső elem
- él
- **út**
- részfa
- szint
- magasság



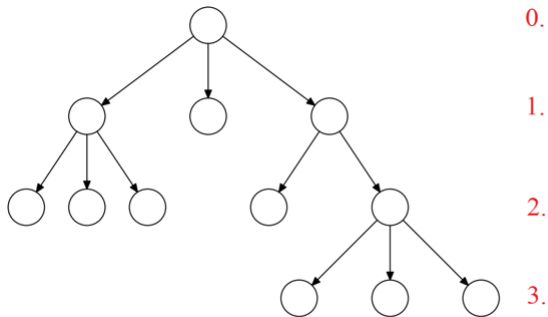
## Alapfogalmak

- csúcs, csomópont
- gyökér
- levél
- belső elem
- él
- út
- **részfa**
- szint
- magasság



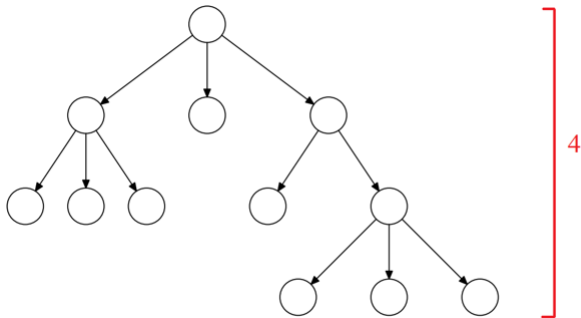
## Alapfogalmak

- csúcs, csomópont
- gyökér
- levél
- belső elem
- él
- út
- részfa
- **szint**
- magasság



## Alapfogalmak

- csúcs, csomópont
- gyökér
- levél
- belső elem
- él
- út
- részfa
- szint
- **magasság**



# Rendezett és rendezetlen fák

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Tulajdonság

- **Rendezetlen fa:** nem lényeges az ugyanazon csúcsból kiinduló élek sorrendje
- **Rendezett fa:** lényeges az ugyanazon csúcsból kiinduló élek sorrendje

# Rendezett és rendezetlen fák

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

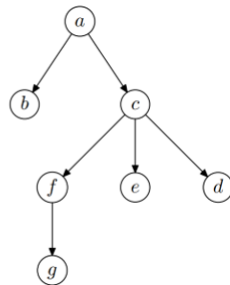
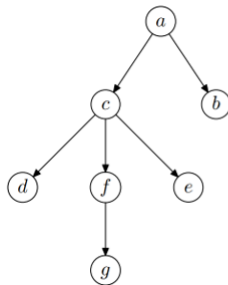
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Kérdés

Ekvivalens-e a két fa?





# Rendezett és rendezetlen fák

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

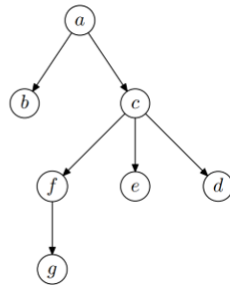
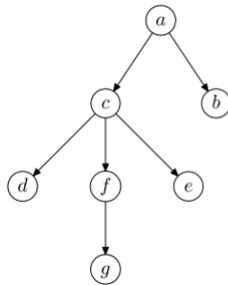
Bináris keresőfa

## Kérdés

Ekvivalens-e a két fa?

■ ha rendezetlen fák  $\rightarrow$  igen

■ ha rendezett fák  $\rightarrow$  nem



A tantárgy keretein belül

■ **fa = rendezett fa**

## Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

**Bináris fák**

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

# Bináris fák

## Definíció 1: Bináris fa

Knuth nyomán (rekurzívan is értelmezhető):

- Egy bináris fa vagy üres, vagy tartalmaz egy csomópontot, amelyhez bizonyos típusú információkat rendeltünk, valamint még két komponenst, amelyek szintén bináris fák (a bal és a jobb részfa).

## Definíció 2: Bináris fa

- olyan fa, melyben minden adatelemnek *legtöbb két* rákövetkezője (leszármazottja) van

## Definíció 3: Szigorú értelemben vett bináris fa

- a bináris fában minden adatelemnek *0 vagy 2* rákövetkezője (leszármazottja) van.

# Bináris fák

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa  
Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás  
Műveletek  
Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Definíció 4: Rendezett bináris fa

A fa elemeire értelmezhetők:

- bal/jobbs oldali rákövetkező (leszármazott)
- bal/jobbs oldali részfa

A tantárgy keretein belül

- **bináris fa = rendezett bináris fa**

# Nem bináris fa $\rightarrow$ bináris fa

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Módszer

Minden nem bináris fa reprezentálható bináris fával:

- A bináris fa gyökere legyen a nem bináris fa gyökere
- A bináris fa egy tetszőleges elemének a bal oldali rákövetkezője legyen a nem bináris fa megfelelő elemének a bal oldali (első) rákövetkezője
- - A bináris fa egy tetszőleges elemének a jobb oldali rákövetkezője legyen a nem bináris fa megfelelő elemének a következő (azonos szülőhöz tartozó) testvércsúcsa

# Nem bináris fa $\rightarrow$ bináris fa

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

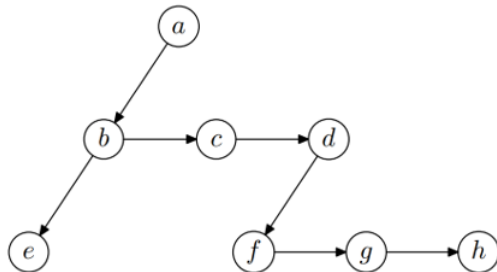
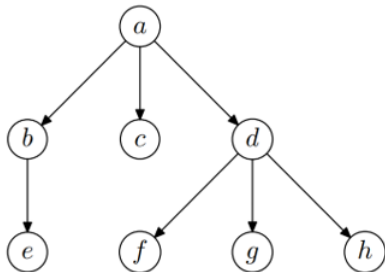
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Bináris fa: ábrázolás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Folytonos reprezentáció

- Három vektor segítségével, ahol a vektorok azonos indexű elemei a következő információkat tárolják:
  - **ADAT**: az adatelem értéke
  - **BAL**: a bal oldali rákövetkező adatelem vektorbeli indexét
  - **JOB**: a jobb oldali rákövetkező adatelem vektorbeli indexét
- Általában a fa gyökérelemét a vektor első eleme írja le

## Megjegyzés

- Amennyiben szükség van a *gyermek*→*szülő* kapcsolatra is, ki lehet bővíteni az ábrázolást még egy vektorral, ami minden elem esetében a szülő indexét tartalmazza.

# Bináris fa - folytonos ábrázolás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

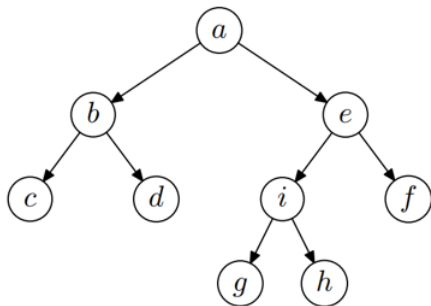
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



	ADAT	BAL	JOBB
1.	<i>a</i>	2	5
2.	<i>b</i>	3	4
3.	<i>c</i>	0	0
4.	<i>d</i>	0	0
5.	<i>e</i>	6	9
6.	<i>i</i>	7	8
7.	<i>g</i>	0	0
8.	<i>h</i>	0	0
9.	<i>f</i>	0	0



# Bináris fa: ábrázolás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Folytonos reprezentáció

### ■ Előnyök:

- Legtömörebb ábrázolásmód
- Gyors hozzáférés, jó teljesítmény

### ■ Hátrányok:

- Lassú beszúrás, törlés

## Megjegyzés

- Ha az adatszerkezetben nem sok változás történik (nem akarunk sok új elemet hozzáadni, gyakran törölni) érdemes ezt az ábrázolásmódot használni.

## Szétszórt (lista alapú) reprezentáció

### ■ Listaelem:

- **adat** - az adatelem értéke
  - **bal** - a bal oldali rákövetkezőt leíró listaelemet címzi
  - **jobb** - a jobb oldali rákövetkezőt leíró listaelemet címzi
- 
- A gyökérelemhez (és rajta keresztül az adatszerkezet többi eleméhez) a “gyökér” mutató segítségével tudunk hozzáférni.

# Bináris fa - szétszórt ábrázolás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

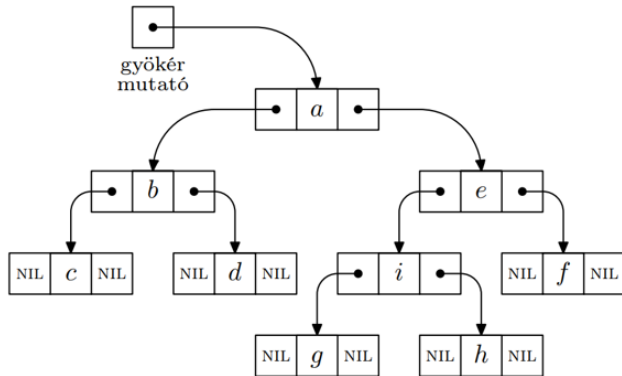
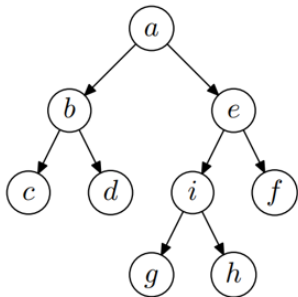
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Bináris fa - szétszórt ábrázolás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

```
typedef struct elem{  
    int adat;  
    elem * bal;  
    elem * jobb;  
    //elem * szülő  
} elem;  
  
elem *gyökér;
```

## Megjegyzések

- Legtöbb esetben nem számokat, hanem valamilyen típusú objektumot tároluk el a fában.
- Egy adott kulcs alapján (ID) szűrjük be, és férünk hozzá az elemekhez.
- Előfordulhat olyan feladat, amikor szükség lehet a felfele kapcsolatra is

# Bináris fa - ábrázolás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Szétszórt reprezentáció

### ■ Előnyök:

- Gyors műveletek (beszúrás, törlés)

### ■ Hátrányok:

- Rosszabb teljesítmény

## Megjegyzés

- Általános esetben ezt az ábrázolásmódot választjuk.

# Bináris fa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Műveletek

- Fa létrehozása \*
- Beszúrás: \*
  - Fa bővítése levél elemmel
  - Fa bővítése részfával
- Törlés: \*
  - Csomópont törlése a fából (lehet levél- vagy közbenső csomópont)
  - Részfa törlése a fából
- Bejárás:
  - Mélységi (preorder, inorder, postorder) és szélességi bejárás
- Keresés \*

## Megjegyzés

- \* műveletek függnek a bináris fa típusától

# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

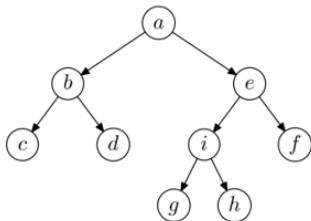
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemt
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon



# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

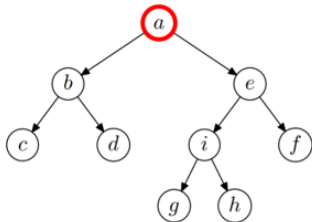
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon





# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

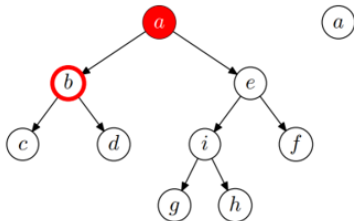
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemt
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon



# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

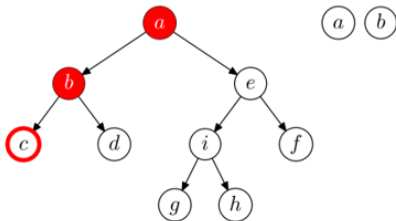
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon



# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

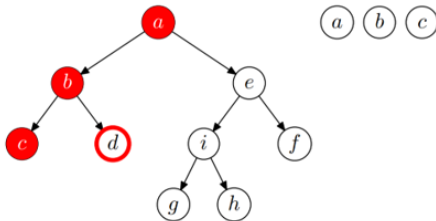
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemt
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon



# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

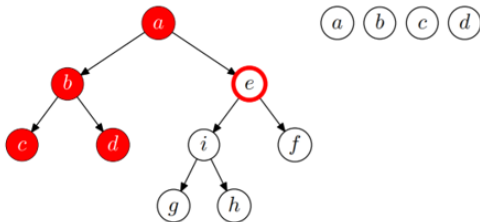
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon



# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

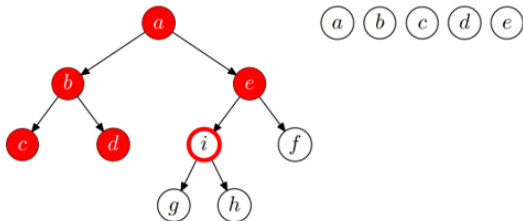
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon



# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

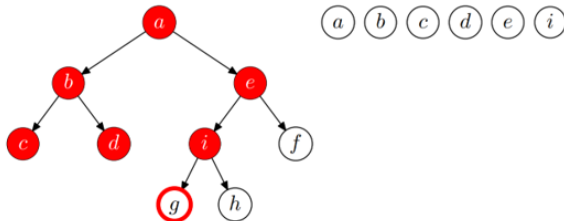
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon



# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

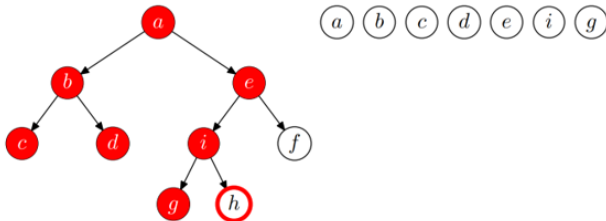
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemt
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon



# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

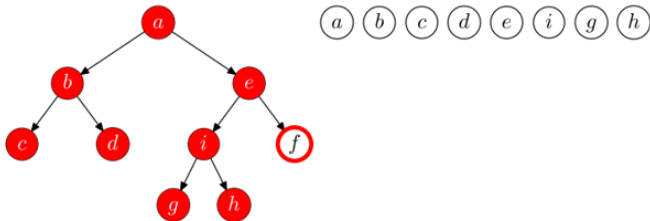
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon





# Fák bejárása: preorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

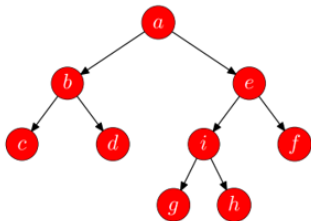
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Preorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem bal oldali részfáját *preorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *preorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

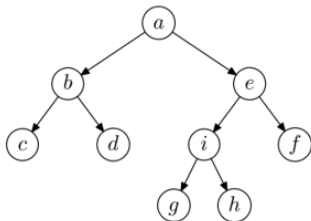
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

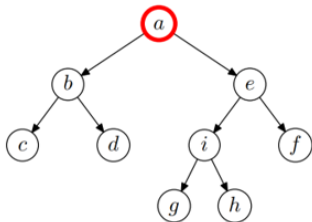
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

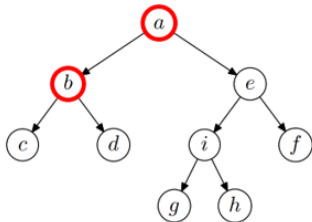
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa  
Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

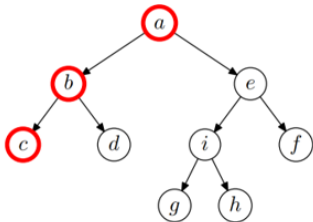
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

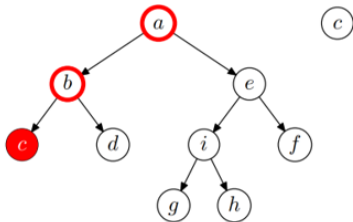
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

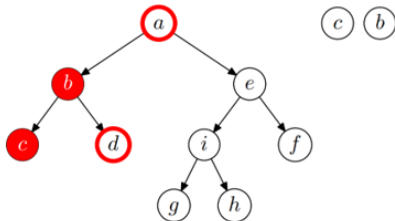
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

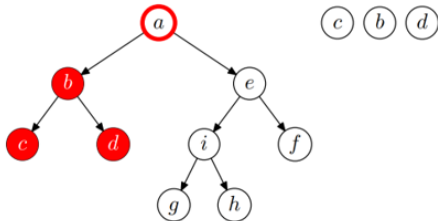
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon





# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

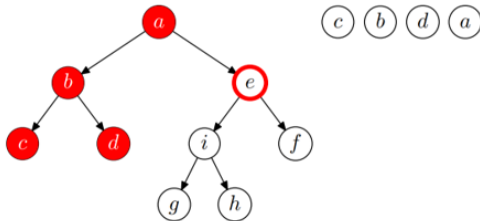
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

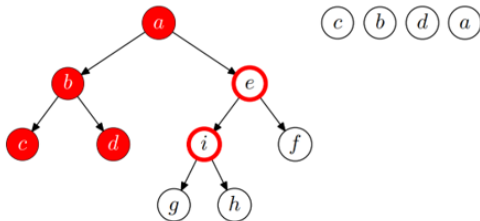
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

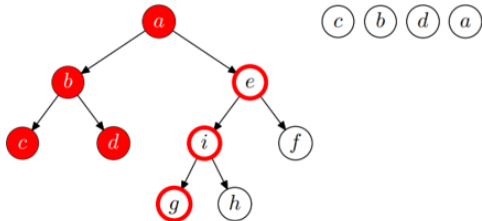
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

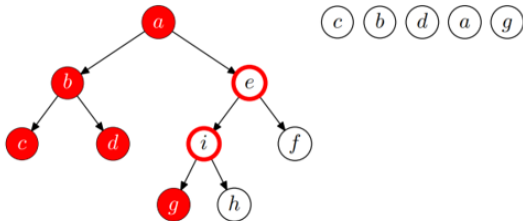
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

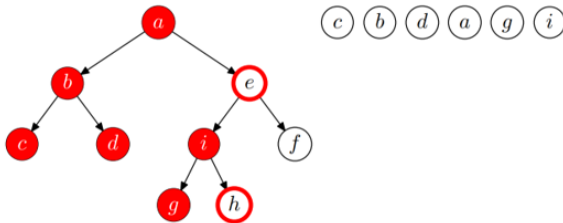
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

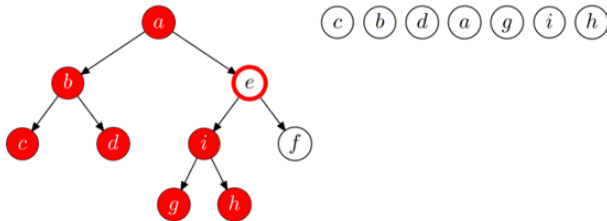
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

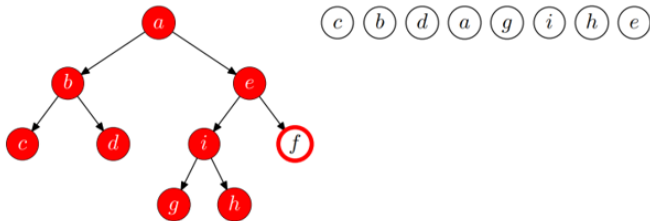
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon



# Fák bejárása: inorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

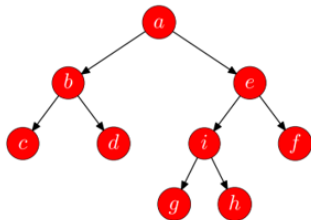
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## *Inorder* bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *inorder* módon
- Dolgozzuk fel a gyökérelemet
- Járjuk be a gyökérelem jobb oldali részfáját *inorder* módon





# Fák bejárása: postorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

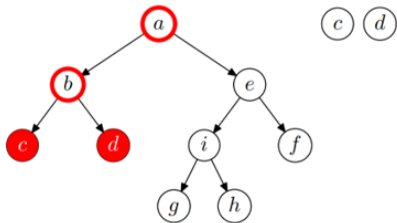
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Postorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *postorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *postorder* módon
- Dolgozzuk fel a gyökérelemet



# Fák bejárása: postorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

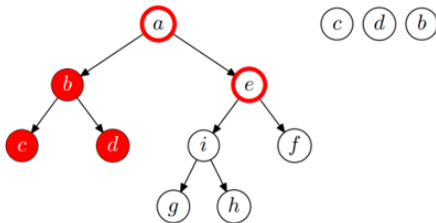
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Postorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *postorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *postorder* módon
- Dolgozzuk fel a gyökérelemet



# Fák bejárása: postorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

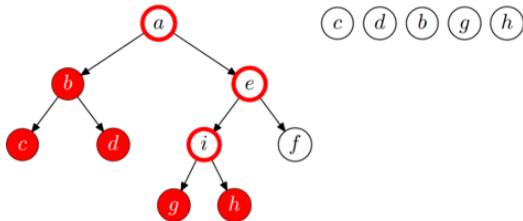
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Postorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *postorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *postorder* módon
- Dolgozzuk fel a gyökérelemet



# Fák bejárása: postorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

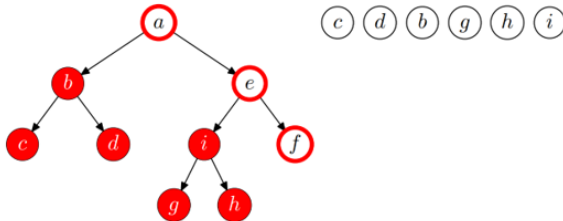
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Postorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *postorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *postorder* módon
- Dolgozzuk fel a gyökérelemet



# Fák bejárása: postorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

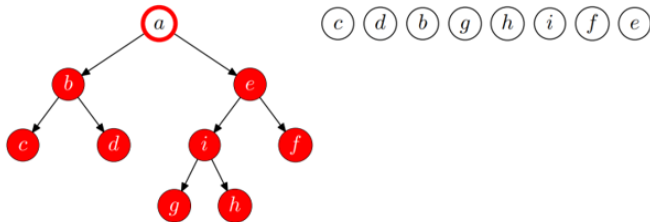
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Postorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *postorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *postorder* módon
- Dolgozzuk fel a gyökérelemet



# Fák bejárása: postorder

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

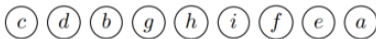
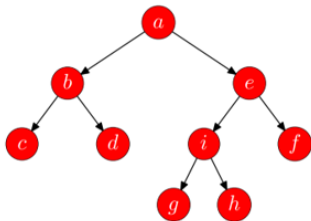
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Postorder bejárás

- Ha a fa üres  $\rightarrow$  algoritmus vége
- Járjuk be a gyökérelem bal oldali részfáját *postorder* módon
- Járjuk be a gyökérelem jobb oldali részfáját *postorder* módon
- Dolgozzuk fel a gyökérelemet



## Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

**Kifejezésfa**

Tökéletesen  
egyensúlyozott

Bináris keresőfa

# Kifejezésfa

## Bejárási sorrendek

A bejárások során a csomópontok listája egy bizonyos sorrendben jelenik meg, de ebből nem tudunk következtetni a fa szerkezetére.

Az alárendeltségi viszonyt zárójelekkel jelöljük:

- Prefix jelölés
  - **(gyökér, bal részfa, jobb részfa)** vagy **()** ha a fa üres
- Infix jelölés
  - **(bal részfa, gyökér, jobb részfa)** vagy **()** ha a fa üres
- Postfix jelölés
  - **(bal részfa, jobb részfa, gyökér)** vagy **()** ha a fa üres



# Zárójeles kifejezések

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Zárójelezés - példa

■ Preorder bejárás:  $a b c d e i g h f$

### ■ Prefix jelölés

■ (gyökér, bal részfa, jobb részfa) vagy () ha a fa üres:

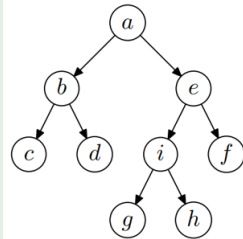
■  $( a ( b ( c ( ) ( ) ) ( d ( ) ( ) ) ) ( e ( i ( g ( ) ( ) ) ( h ( ) ( ) ) ) ( f ( ) ( ) ) ) )$

■ Egyszerűsítés: *nem ábrázoljuk az üres fákat*

■  $( a ( b ( c ) ( d ) ) ( e ( i ( g ) ( h ) ) ( f ) ) )$

■ Egyszerűsítés: *a leveleket nem tesszük zárójelbe*

■  $( a ( b c d ) ( e ( i g h ) f ) )$



# Kifejezésfa

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

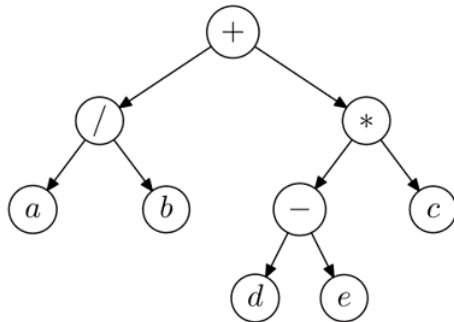
Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Kifejezésfa

A kifejezésfa egy olyan fa, amelyben

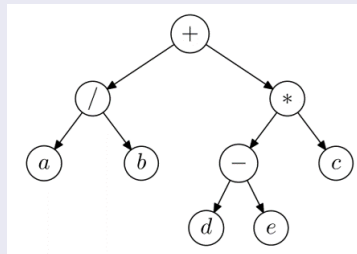
- a **levélelemek** a kifejezés **operandusait**,
- a **nem levélelemek** ugyanazon kifejezés **operátorait** tartalmazzák.



## Kifejezésfa bejárása

Aszerint, hogy a kifejezésfát melyik bejárási algoritmussal járjuk be, megkapjuk a kifejezés prefix, infix vagy postfix alakját.

- **Prefix:**  $+ / a b * - d e c$
- **Infix:**  $a / b + d - e * c$
- **Postfix:**  $a b / d e - c * +$
- A prefix és postfix alak egyértelmű.
- Az infix nem egyértelmű (de zárójelek használatával azzá tehető)



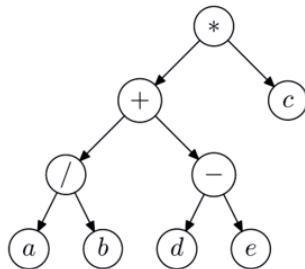
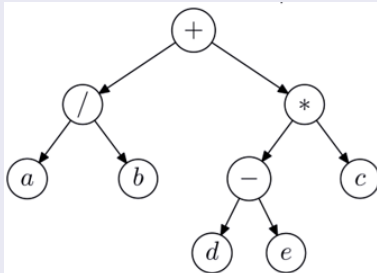
## Kifejezésfa bejárása

■ **Inorder** bejárás (mindkét fa esetén):  $a / b + d - e * c$

■ Zárójelezést alkalmazva:

■  $((a / b) + ((d - e) * c))$

$(( (a / b) + (d - e) ) * c)$



Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

Tökéletesen egyensúlyozott

# Tökéletesen kiegyensúlyozott bináris fa

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Minimális magasságú fa

- Egy fa minimális magasságú, ha adott számú elemet nem lehetne kisebb magasságú bináris fában elhelyezni.

## Tökéletesen kiegyensúlyozott bináris fa

- Egy bináris fa tökéletesen kiegyensúlyozott, ha bármely elemének bal és jobb oldali részfájában az elemek darabszáma legfeljebb 1-gyel tér el

## Megjegyzés

- Megjegyzés: minden tökéletesen kiegyensúlyozott fa minimális magasságú

# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Létrehozás

■ Bemenet:  $n$  elem

■ **Algoritmus:**

- Ha  $n = 0 \rightarrow$  eredmény: üres fa, algoritmus vége
- Az első elem legyen a tökéletesen kiegyensúlyozott bináris fa gyökere
- Osszuk két részre a megmaradt  $n-1$  elemet
  - Az első  $nb = \lfloor n/2 \rfloor$  elemből építsük fel a gyökérelem bal oldali tökéletesen kiegyensúlyozott részfáját
  - A megmaradt  $nj = n - 1 - nb$  elemből építsük fel a gyökérelem jobb oldali tökéletesen kiegyensúlyozott részfáját

# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

**Tökéletesen  
egyensúlyozott**

Bináris keresőfa





# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

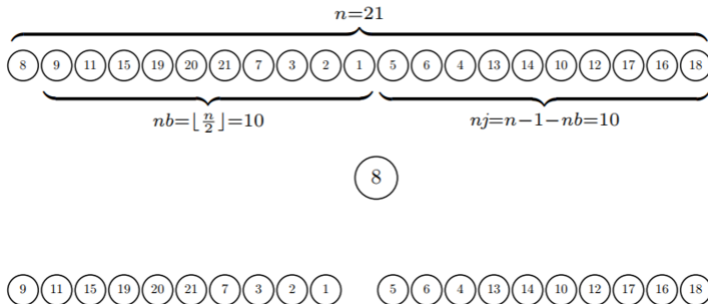
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

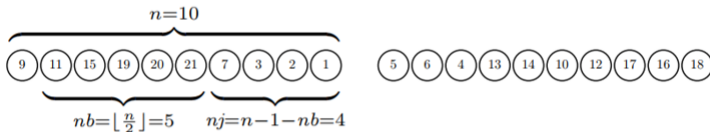
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

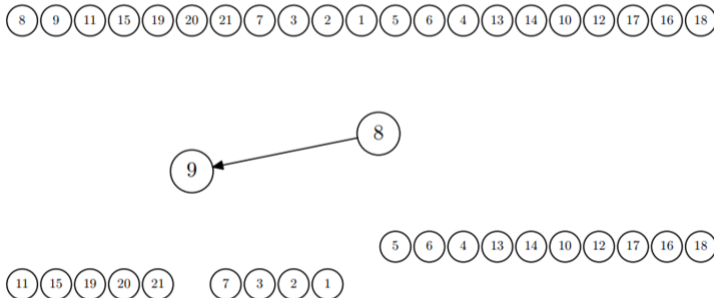
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

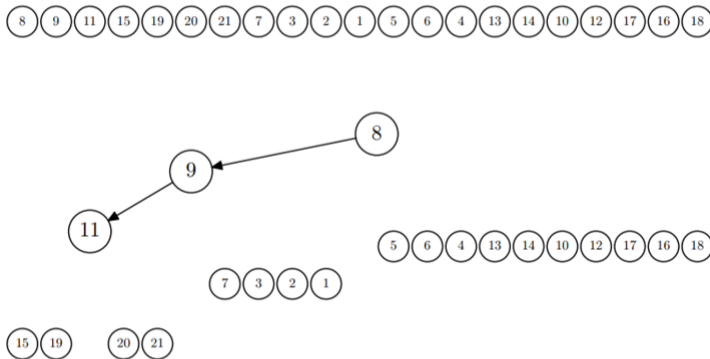
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

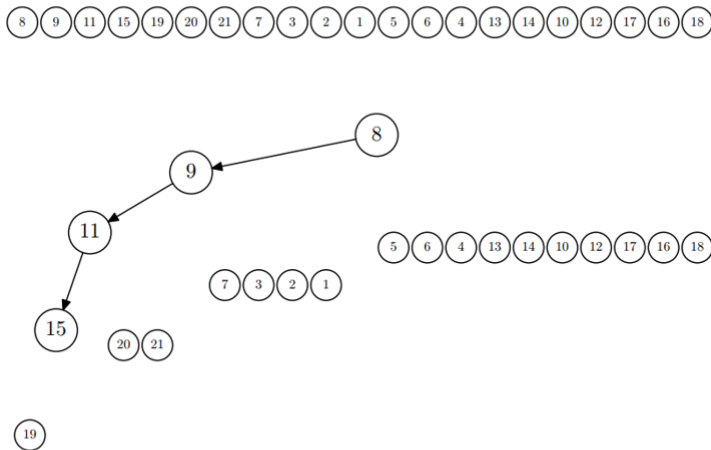
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

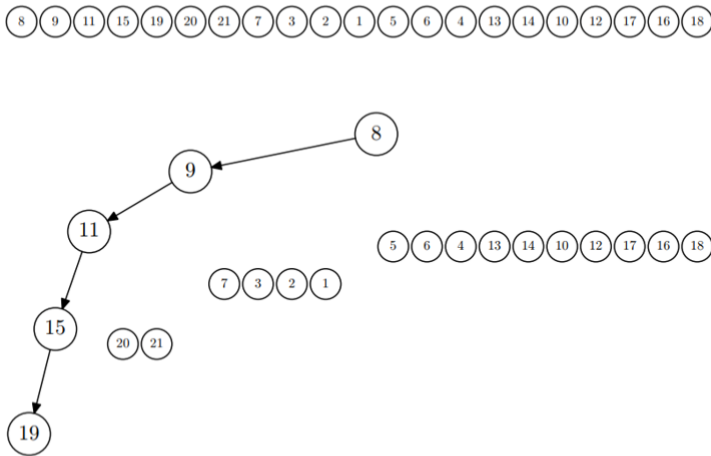
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

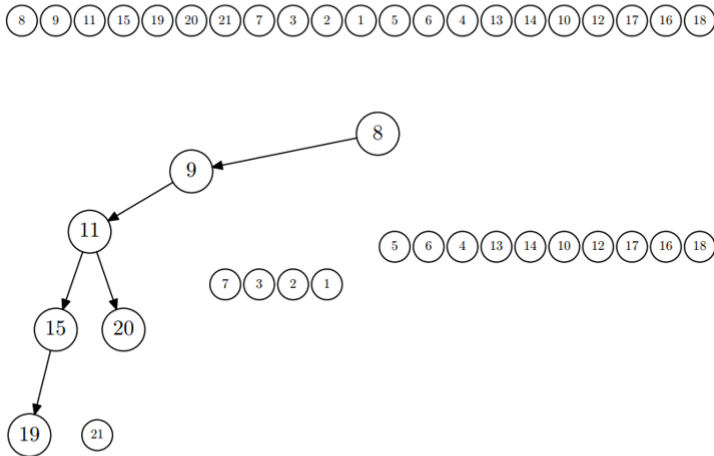
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

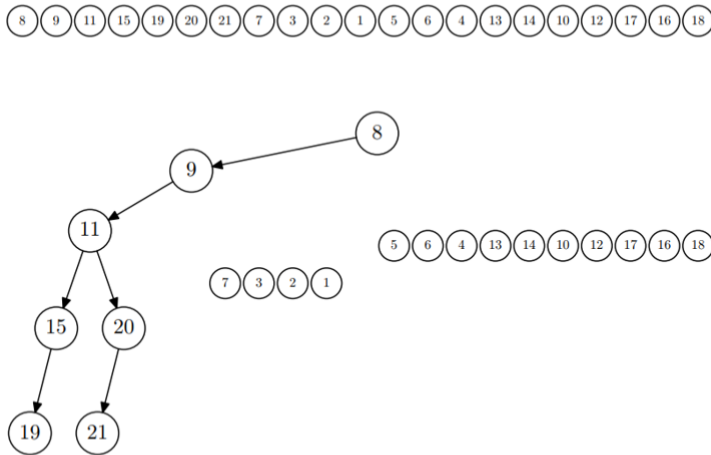
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa





# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

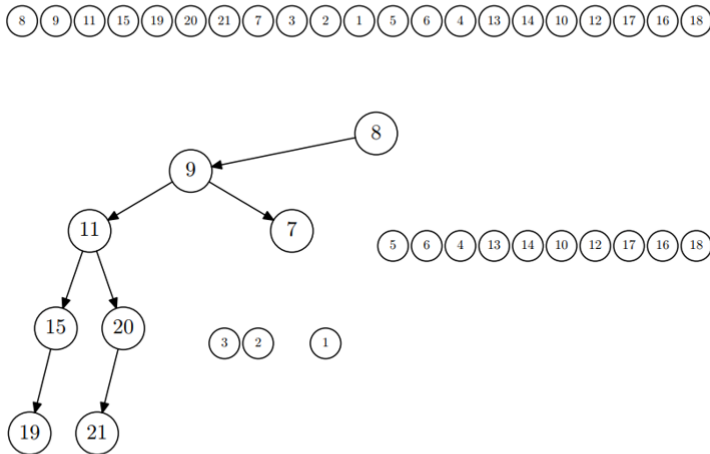
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

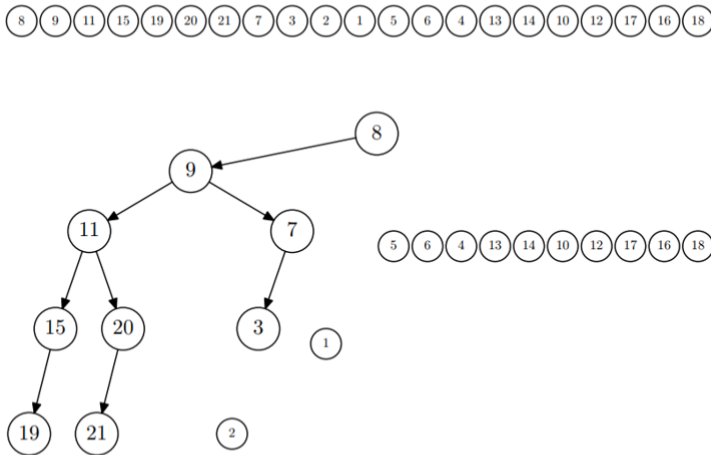
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

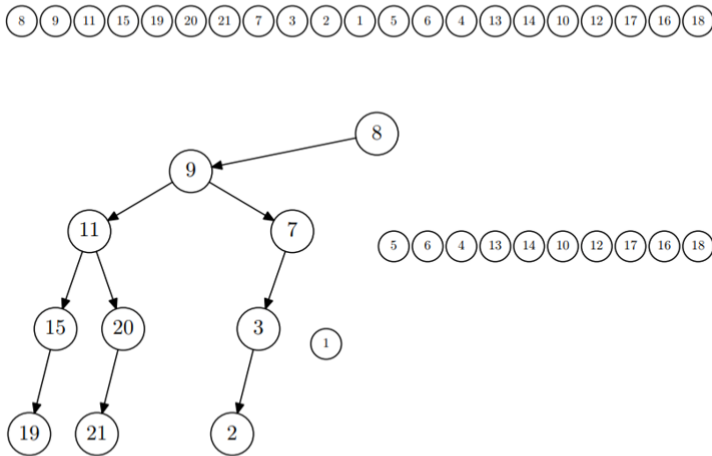
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

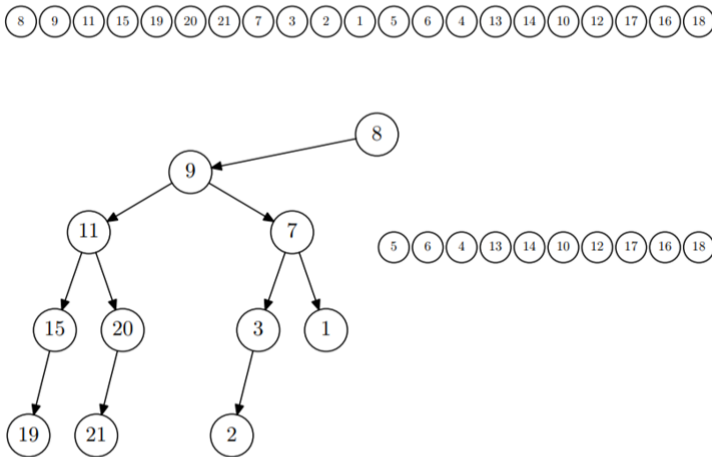
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



# Tökéletesen kiegyensúlyozott bináris fa: létrehozás

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

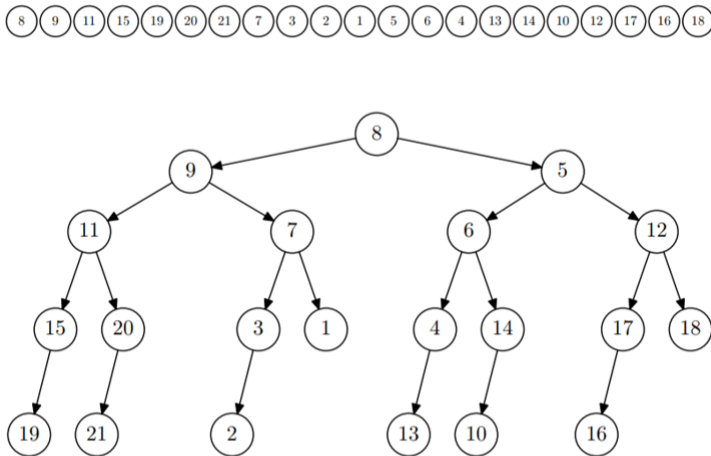
Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa



Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

# Bináris keresőfa

# Bináris keresőfa

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

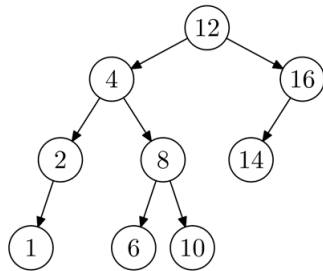
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Definíció

- Rendezett bináris fa
- Az adatelemek mindegyike rendelkezik egy *kulccsal*
- Minden adatelemre igaz, hogy:
  - Az adatelem *bal* oldali részfájában levő elemek kulcsai *kisebbek* az elem kulcsánál
  - Az adatelem *jobb* oldali részfájában levő elemek kulcsai *nagyobbak* az elem kulcsánál
- Minden kulcs egyedi. Amennyiben nem, a műveleteket megfelelően át kell értelmezni.



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Keresés

**x** - keresett elem

- **előfeltétel:** -
- **utófeltétel:** *igazat* térít ha megtalálta az elemet, *hamisat* ha nem (alternatíva: egy *mutatót* térít, ha megtalálta, *NULL*-t ha nem)\*

### Algoritmus:

- Ha a fa üres  $\rightarrow$  a keresett elem nem található a fában
- Összehasonlítjuk a gyökérben található kulcsot ( $c$ ) a keresett elemmel
  - Ha  $x = c$ : keresés vége
  - Ha  $x < c$ : megkeressük  $x$ -et a gyökérelem bal alfájában
  - Különben: megkeressük  $x$ -et a gyökérelem jobb alfájában

\* - függ a feladattól



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

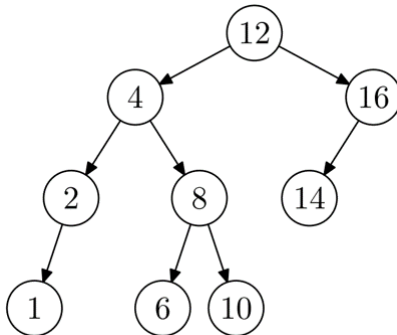
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Keressük meg a 2-est az ábrán látható bináris fában!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

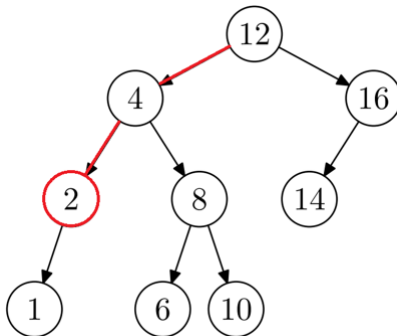
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Keressük meg a 2-est az ábrán látható bináris fában!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás  
Műveletek  
Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Beszúrás

- **x** - beszúrandó elem
- **előfeltétel:** -
- **utófeltétel:**  $x$  szabályos helyet foglal el a bináris fában (a bináris fának megmaradnak a jellemző tulajdonságai). Jelezzük, hogyha nem sikerült beszúrni az elemet\*.

## Algoritmus:

- Ha a fa üres  $\rightarrow$  a beszúrandó elem lesz a fa egyetlen eleme (levélelem és gyökér), algoritmus vége
- Összehasonlítjuk a gyökérben található kulcsot ( $c$ ) a keresett elemmel
  - Ha  $x < c$ : beszúrjuk az  $x$ -et a gyökérelem bal oldali részfájába
  - Ha  $x > c$ : beszúrjuk az  $x$ -et a gyökérelem jobb oldali részfájába
  - Ha  $x = c$ :  $x$  nem szúrható be a fába, mert nem szerepelhet két azonos kulcsú elem a keresőfában\*

\* - függ a feladattól

# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

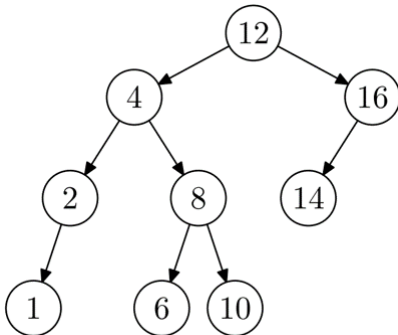
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Szúrjuk be a 3-ast az ábrán látható bináris fába!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

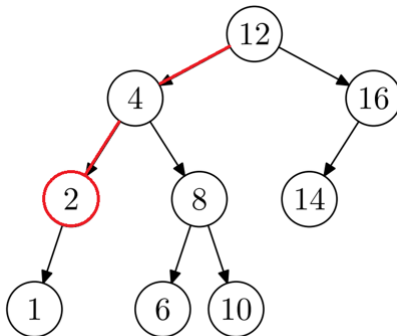
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Szúrjuk be a 3-ast az ábrán látható bináris fába!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

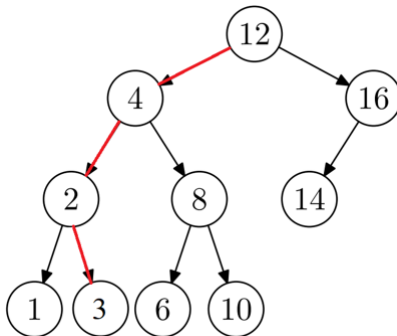
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Szúrjuk be a 3-ast az ábrán látható bináris fába!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Törlés

- **x** - törlendő elem
- **előfeltétel:** -
- **utófeltétel:** **x** törlődik a bináris keresőfa elemei közül\*. A fa tulajdonságai megmaradnak.

\* - függ a feladattól

# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Törlés

### Algoritmus:

- Ha a fa üres  $\rightarrow$  nem tudunk törölni, algoritmus vége
- Összehasonlítjuk a gyökérben található kulcsot ( $c$ ) a keresett elemmel
  - Ha  $x < c$ : töröljük az  $x$ -et a gyökérelem bal oldali részfájából
  - Ha  $x > c$ : töröljük az  $x$ -et a gyökérelem jobb oldali részfájából
  - Ha  $x = c$ : megnézzük, hogy a gyökérelemnek hány rákövetkezője van ( $k$ ):
    - $k = 0$ : töröljük a csomópontot
    - $k = 1$ : felülírjuk a gyökérelemet a rákövetkező elemmel (egy szinttel fennebb csúsztatjuk a gyökérelem nem üres részfáját)
    - $k = 2$ : a gyökérelemet felülírjuk a bal oldali részfája legjobboldalibb elemének az értékével, majd a bal oldali részfából töröljük a legjobboldalibb elemet  
*vagy*
    - $k = 2$ : a gyökérelemet felülírjuk a jobb oldali részfája legbaloldalibb elemének az értékével, majd a jobb oldali részfából töröljük a legbaloldalibb elemet



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

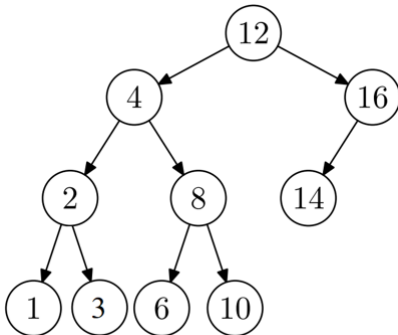
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Töröljük a 4-est az ábrán látható bináris fából!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

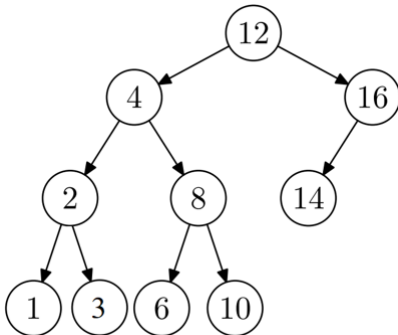
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Töröljük a 4-est az ábrán látható bináris fából!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

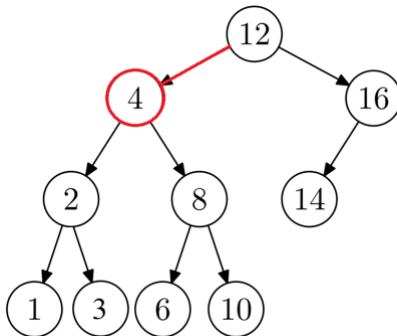
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Töröljük a 4-est az ábrán látható bináris fából!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

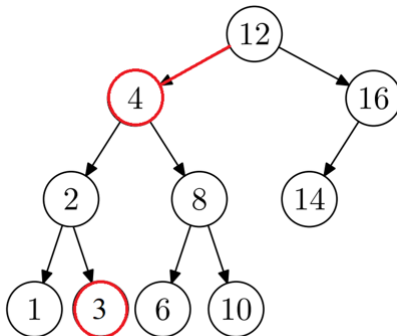
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Töröljük a 4-est az ábrán látható bináris fából!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

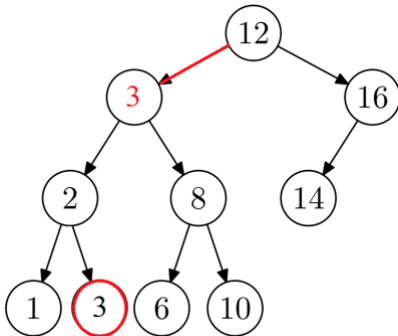
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Töröljük a 4-est az ábrán látható bináris fából!



# Bináris keresőfa: műveletek

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

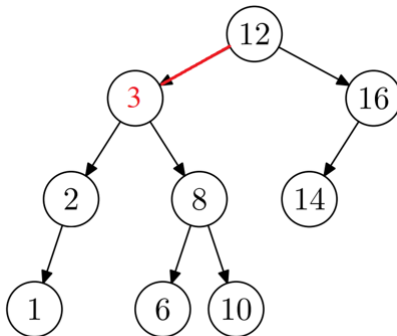
Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Feladat

- Töröljük a 4-est az ábrán látható bináris fából!



# Bináris keresőfa

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Műveletek

- keresés
- beszúrás
- törlés

## Kérdés

**Mennyi a műveletek futási ideje?**

# Bináris keresőfa

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Műveletek

- keresés
- beszúrás
- törlés

## Kérdés

**Mennyi a műveletek futási ideje?**

- $O(h)$ , ahol  $h$  a fa magassága



# Bináris keresőfa

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Műveletek

- keresés
- beszúrás
- törlés

## Kérdés

**Mennyi a műveletek futási ideje?**

- $O(h)$ , ahol  $h$  a fa magassága
- Legrosszabb eset?

# Bináris keresőfa

Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Műveletek

- keresés
- beszúrás
- törlés

## Kérdés

**Mennyi a műveletek futási ideje?**

- $O(h)$ , ahol  $h$  a fa magassága
- Legrosszabb eset?  $O(n)$  - hogyan lehet elkerülni?

## Adatszerkezetek

Vekov Géza

Hierarchikus  
adatszerkezet

Fa

Rendezett és  
rendezetlen fák

Bináris fák

Ábrázolás

Műveletek

Bejárások

Kifejezésfa

Tökéletesen  
egyensúlyozott

Bináris keresőfa

## Kérdések

Kérdések? 😊