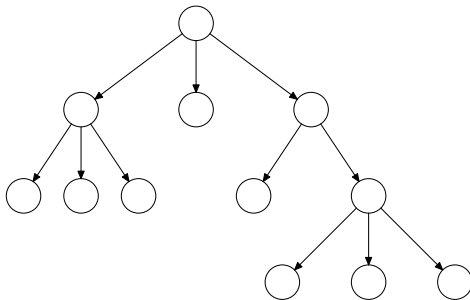


## A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

Fa adatszerkezetekkel kapcsolatos fogalmak:

- csúcs, csomópont
- gyökérelem
- levélelem
- közbenső elem
- él
- út
- részfa
- szint
- magasság

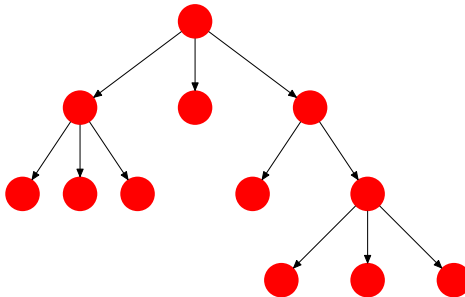


# A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

Fa adatszerkezetekkel kapcsolatos fogalmak:

- csúcs, csomópont
- gyökérelem
- levélelem
- közbenső elem
- él
- út
- részfa
- szint
- magasság

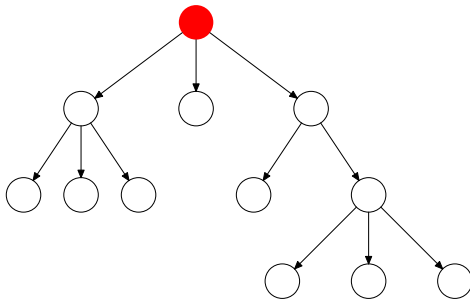


## A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

Fa adatszerkezetekkel kapcsolatos fogalmak:

- csúcs, csomópont
- gyökérelem
- levélelem
- közbenső elem
- él
- út
- részfa
- szint
- magasság

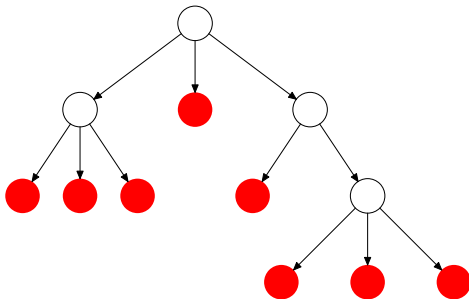


## A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

Fa adatszerkezetekkel kapcsolatos fogalmak:

- csúcs, csomópont
- gyökérelem
- **levélelem**
- közbenső elem
- él
- út
- részfa
- szint
- magasság

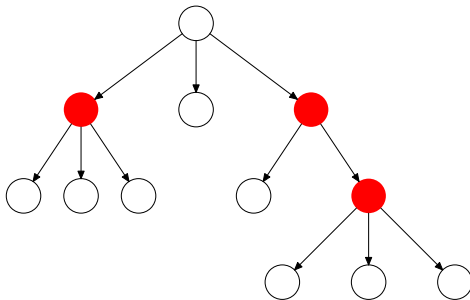


## A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

Fa adatszerkezetekkel kapcsolatos fogalmak:

- csúcs, csomópont
- gyökérelem
- levélelem
- **közbenső elem**
- él
- út
- részfa
- szint
- magasság

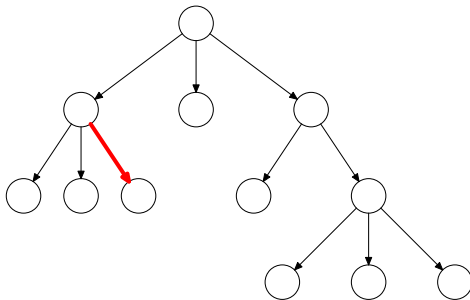


## A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

Fa adatszerkezetekkel kapcsolatos fogalmak:

- csúcs, csomópont
- gyökérelem
- levélelem
- közbenső elem
- él
- út
- részfa
- szint
- magasság

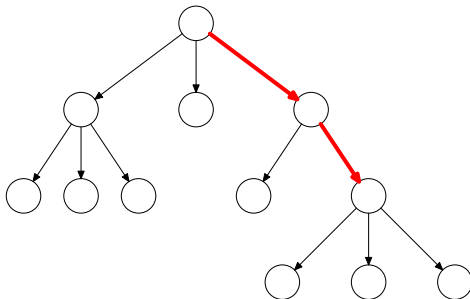


## A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

Fa adatszerkezetekkel kapcsolatos fogalmak:

- csúcs, csomópont
- gyökérelem
- levélelem
- közbenső elem
- él
- út
- részfa
- szint
- magasság

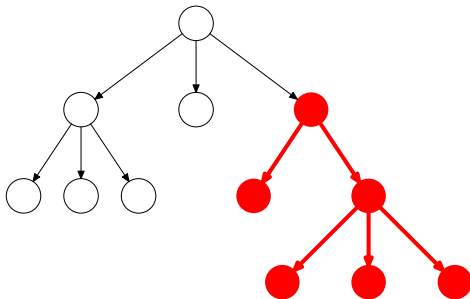


## A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

Fa adatszerkezetekkel kapcsolatos fogalmak:

- csúcs, csomópont
- gyökérelem
- levélelem
- közbenső elem
- él
- út
- **részfa**
- szint
- magasság





## A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

Fa adatszerkezetekkel kapcsolatos fogalmak:

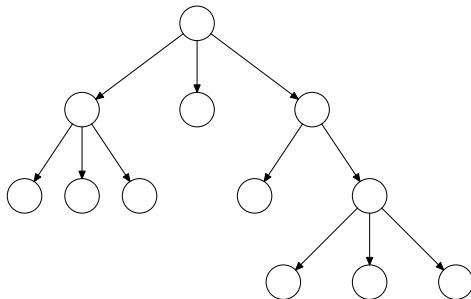
- csúcs, csomópont
- gyökérelem
- levélelem
- közbenső elem
- él
- út
- részfa
- **szint**
- magasság

0.

1.

2.

3.

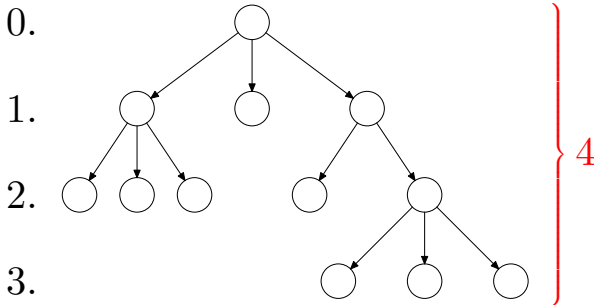


## A fa adatszerkezet

Homogén, dinamikus, hierarchikus adatszerkezet.

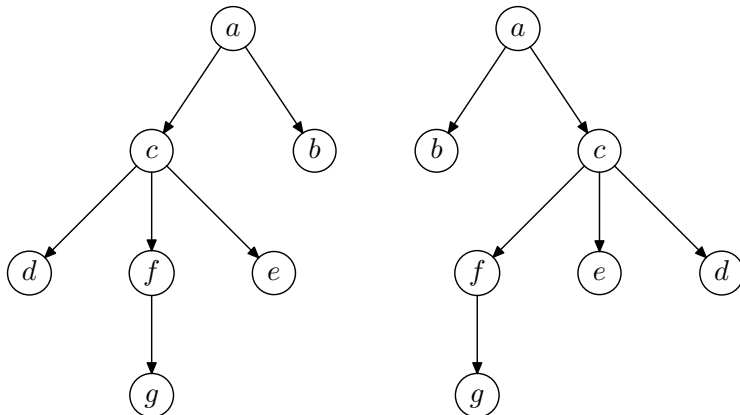
Fa adatszerkezetekkel kapcsolatos fogalmak:

- csúcs, csomópont
- gyökérelem
- levélelem
- közbenső elem
- él
- út
- részfa
- szint
- **magasság**



## Rendezetlen és rendezett fák

**Rendezetlen** fáknál nem lényeges az ugyanazon csúcsból kiinduló élek sorrendje, **rendezett** fáknál viszont igen.



Az ábrán látható két fa ekvivalens egymással, ha eltekintünk az ugyanazon csúcsokból kiinduló élek sorrendjétől (azaz ha rendezetlen fáknak tekintjük őket).

Mivel az informatikában a rendezett adatszerkezetek játszanak fontos szerepet, a továbbiakban rendezett fákkal foglalkozunk.



## Bináris fa

Olyan fa, melyben minden adatelemnek legfeljebb két rákövetkezője van.

## Szigorú értelemben vett bináris fa

Szigorú értelemben vett bináris fáról beszélünk, ha a bináris fában minden adatelemnek 0 vagy 2 rákövetkezője van.

## Rendezett bináris fa

Rendezett bináris fa elemeire értelmezhetők a következő fogalmak:

- bal/jobbs oldali rákövetkező
- bal/jobbs oldali részfa

A továbbiakban bináris fa alatt – hacsak mást nem mondunk – rendezett bináris fát értünk.

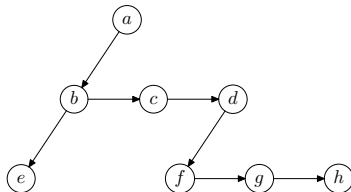
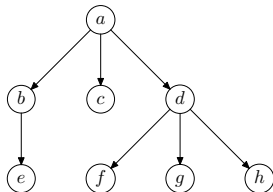


# Nem bináris fa rendezett bináris fává alakítása

Minden nem bináris fa reprezentálható rendezett bináris fával.

## Tetszőleges fa **binarizálásának** algoritmus

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



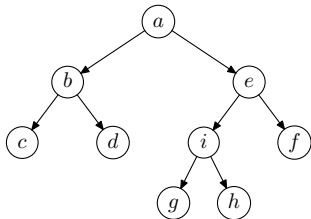


## Bináris fával végezhető műveletek

- **Létrehozás:** üres fa.
- **Bővítés:** egy elemmel vagy egy részfával, általában levélelemnél.
- **Törlés:** részfát vagy egy elemet, utóbbi esetben a fát a legtöbb esetben újra kell szervezni (hogyan továbbra is fa maradjon).
- **Csere:** megengedett.
- **Rendezés:** nincs.
- **Keresés, elérés és feldolgozás:** a bejárás algoritmusai alapján.
- **Bejárás:** szokás szerint olyan algoritmus, amelynek segítségével a bináris fa elemeit leképezzük egy sorra (preorder, inorder vagy postorder módon).

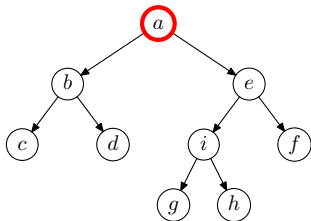
## Preorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.



## Preorder bejárás algoritmusai

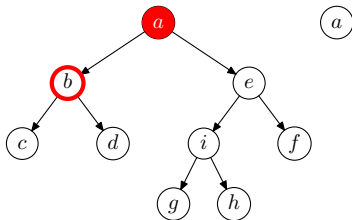
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.





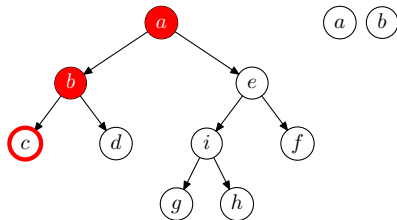
## Preorder bejárás algoritmusai

- 1 Ha a bejárandó fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.



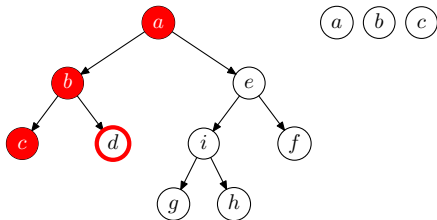
## Preorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.



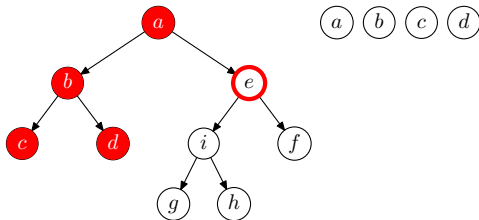
## Preorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.



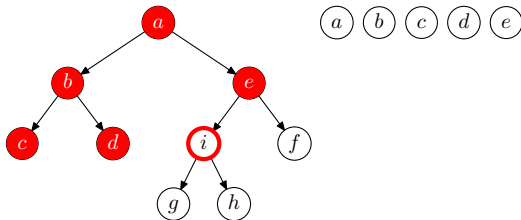
## Preorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.



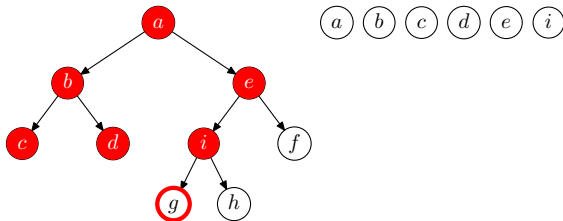
## Preorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.



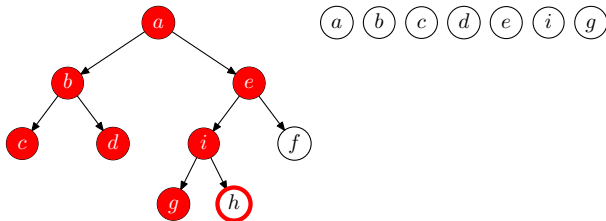
## Preorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.



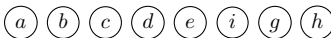
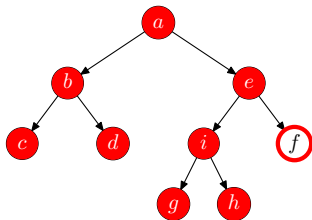
## Preorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.



## Preorder bejárás algoritmusai

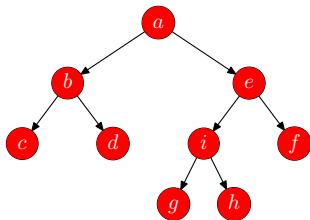
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.





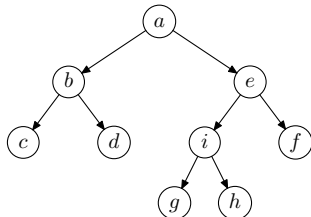
## Preorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 3 Járjuk be a gyökérelem bal oldali részfáját preorder módon.
- 4 Járjuk be a gyökérelem jobb oldali részfáját preorder módon.



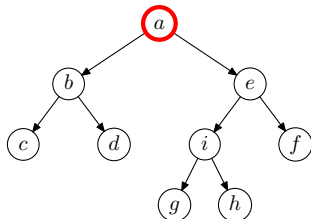
## Inorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.



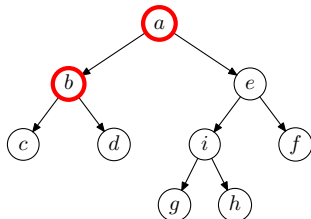
## Inorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.



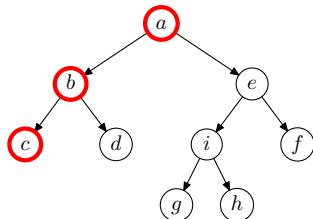
## Inorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.



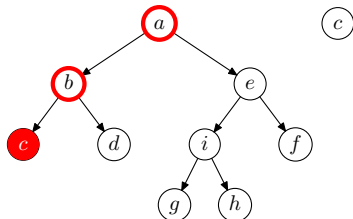
## Inorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.



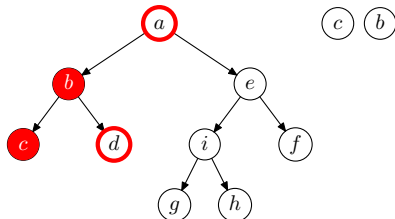
## Inorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.



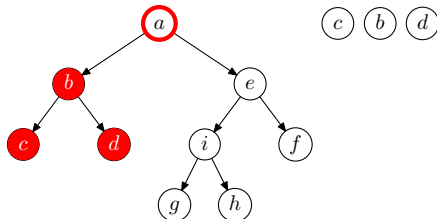
## Inorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.



## Inorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.

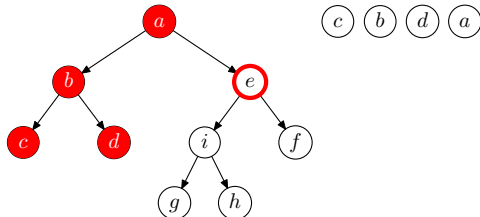






## Inorder bejárás algoritmus

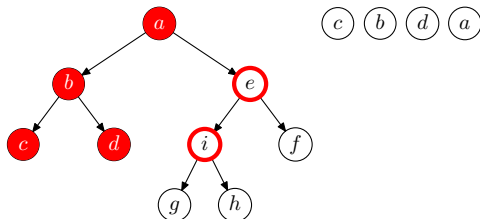
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.





## Inorder bejárás algoritmus

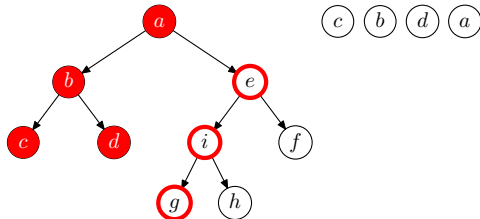
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.





## Inorder bejárás algoritmus

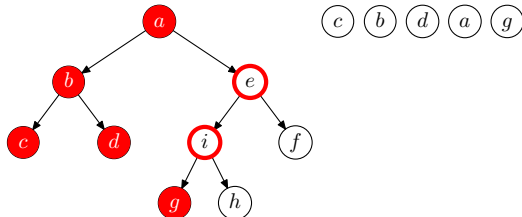
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.





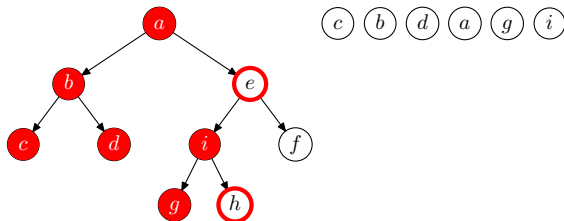
## Inorder bejárás algoritmus

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.



## Inorder bejárás algoritmusai

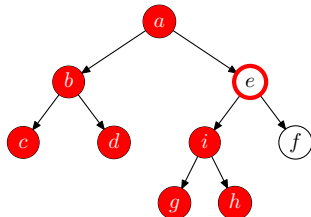
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.





## Inorder bejárás algoritmus

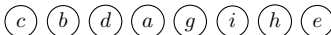
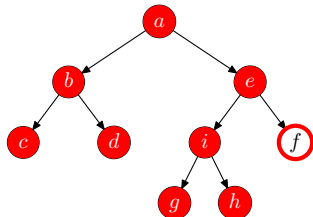
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.





## Inorder bejárás algoritmus

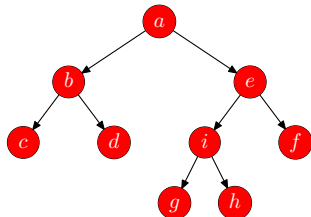
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.





## Inorder bejárás algoritmus

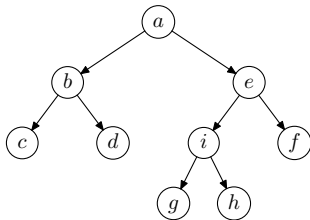
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját inorder módon.
- 3 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).
- 4 Járjuk be a gyökérelem jobb oldali részfáját inorder módon.





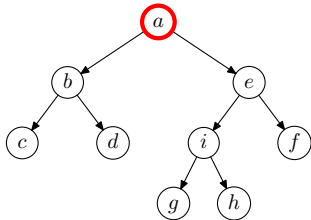
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



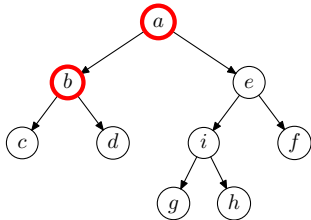
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



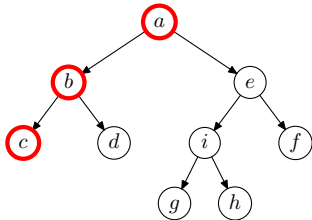
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



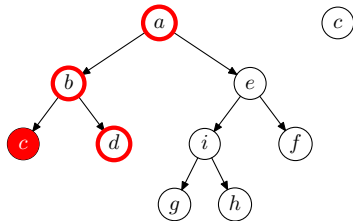
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



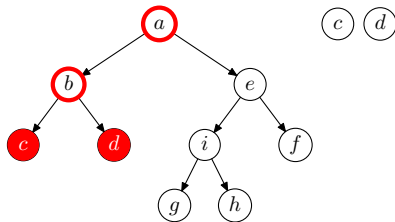
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



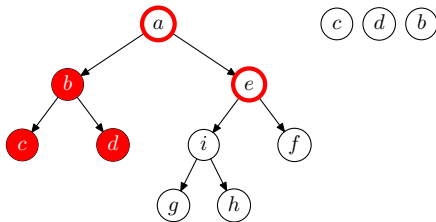
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



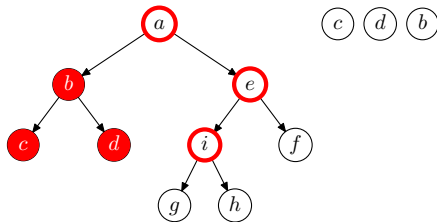
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



## Postorder bejárás algoritmusai

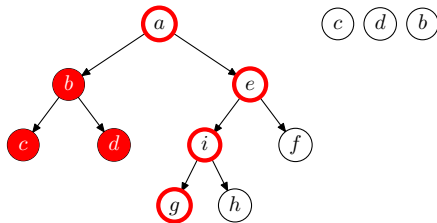
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).





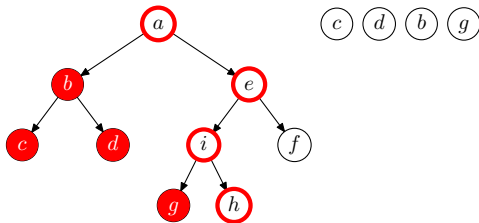
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



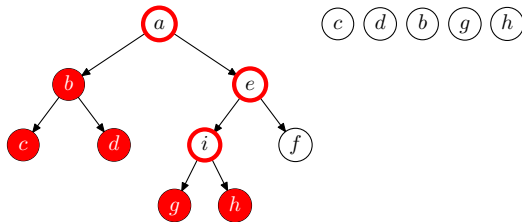
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



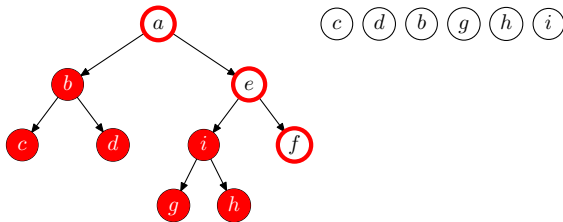
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



## Postorder bejárás algoritmusai

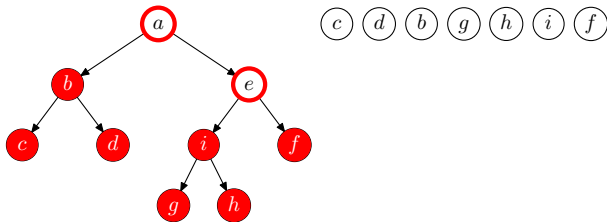
- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).





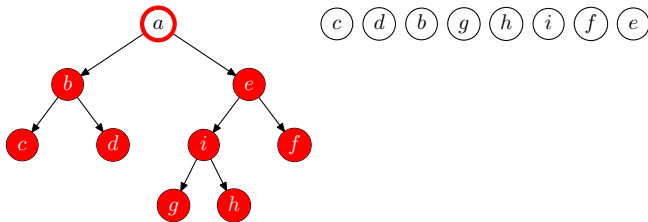
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



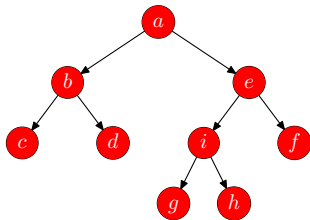
## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).



## Postorder bejárás algoritmusai

- 1 Ha a bejárando fa üres, az algoritmus véget ér.
- 2 Járjuk be a gyökérelem bal oldali részfáját postorder módon.
- 3 Járjuk be a gyökérelem jobb oldali részfáját postorder módon.
- 4 Dolgozzuk fel a gyökérelemet (más szavakkal: helyezzük a gyökérelemet a sor végére).

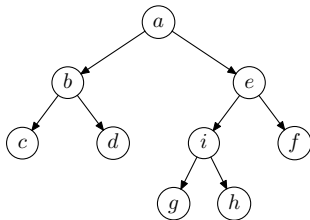


## Bináris fa folytonos reprezentációja

Három vektor segítségével, ahol a vektorok azonos indexű elemei a következő információkat tárolják:

- az ADAT vektorban az adatelem értékét,
- a BAL vektorban a bal oldali rákövetkező vektorbeli indexét,
- a JOBB vektorban a jobb oldali rákövetkező vektorbeli indexét.

Általában a fa gyökérelemét e vektorok első eleme írja le.



	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

