

ADATSZERKEZETEK ÉS ALGORITMUSOK

Hierarchikus adatszerkezetek
„Hierarchikus adatszerkezetek, keresési fák”

Hierarchikus adatszerkezetek

- A hierarchikus adatszerkezet olyan $\langle A, R \rangle$ rendezett pár, amelynél van egy kitüntetett r elem, ez a **gyökérelem**, úgy, hogy:
 1. r nem lehet végpont
 $\forall a \in A$ esetén $\neg R(a, r)$
 2. $\forall a \in \{A \setminus \{r\}\}$ elem egyszer és csak egyszer lehet végpont, azaz $\forall a \in \{A \setminus \{r\}\}$ -hez $\exists! b \neq a, b \in A: R(b, a)$
 3. $\forall a \in \{A \setminus \{r\}\}$ elem r -ből elérhető, azaz
 $\exists a_1, a_2, \dots, a_n \in A, a_n = a:$
 $R(r, a_1), R(a_1, a_2), \dots, R(a_{n-1}, a_n))$
- A hierarchikus adatszerkezetek bizonyos értelemben a lista általánosításai

Hierarchikus adatszerkezetek

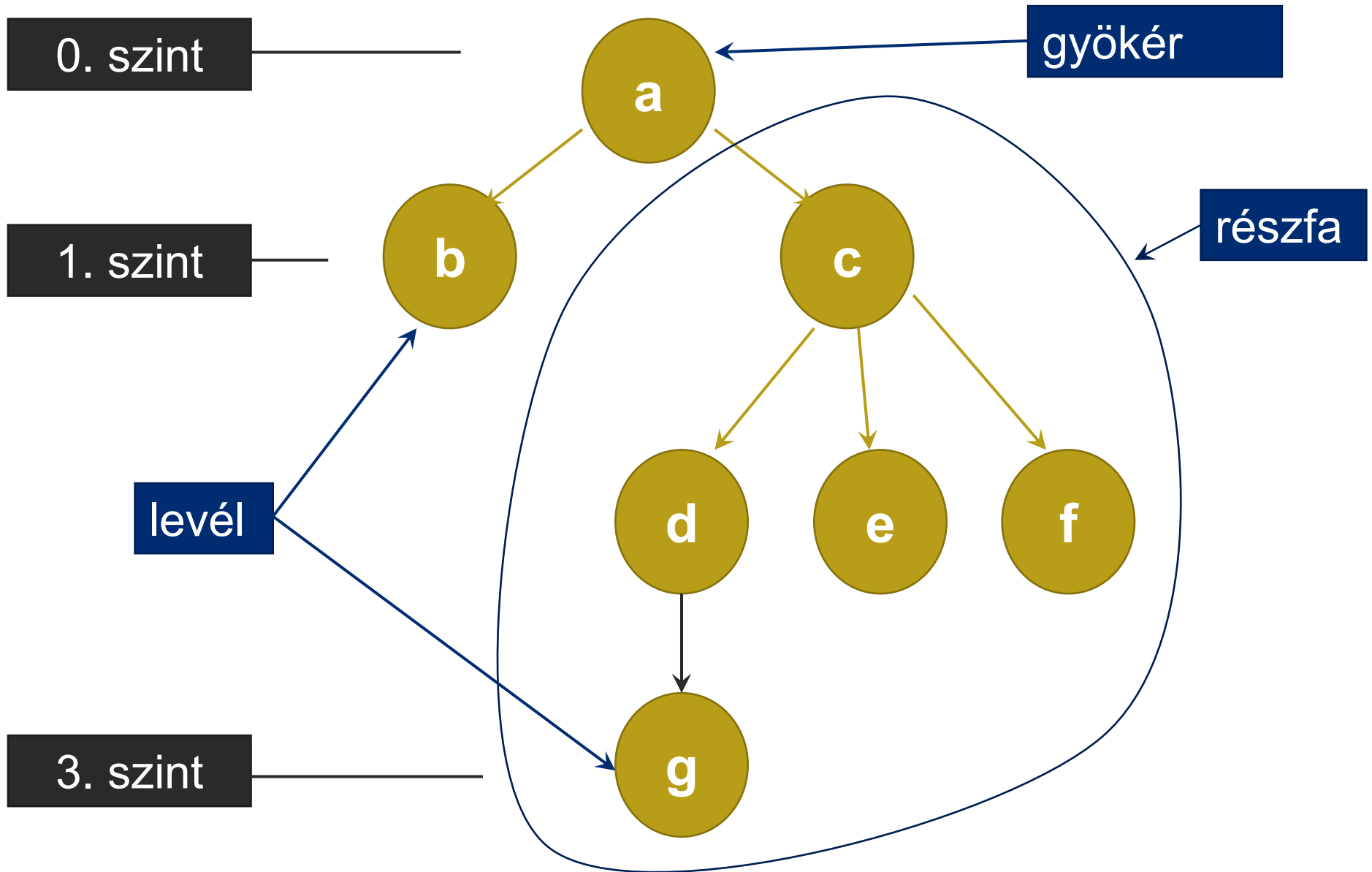
- Egy elemnek akárhány rákövetkezője lehet, de minden elemnek csak egyetlen megelőző eleme van, azaz az adatelemek között **egy-sok** jellegű kapcsolat áll fenn
- Minden adatelem csak egy helyről érhető el, de egy adott elemből tetszés szerinti számú adatelem látható
 - Például
 - Fa
 - összetett lista
 - B-fa

Fák

- A **fa** egy hierarchikus adatszerkezet, mely véges számú csomópontból áll, és igazak a következők:
 - Két csomópont között a kapcsolat egyirányú, az egyik a kezdőpont, a másik a végpont
 - Van a fának egy kitüntetett csomópontja, ami nem lehet végpont
 - Ez a fa gyökere
 - Az összes többi csomópont pontosan egyszer végpont

Fák

- A **fa** rekurzív definíciója:
 - A fa vagy üres, vagy
 - Van egy kitüntetett csomópontja, ez a gyökér.
 - A gyökérhez 0 vagy több diszjunkt fa kapcsolódik
 - Ezek a gyökérhez tartozó részfák
- A fával kapcsolatos algoritmusok gyakran rekurzívak



Fák

- Az adatszerkezetben
 - A fa **csúcsai** az adatelemeknek felelnek meg,
 - Az **élek** az adatelemek egymás utáni sorrendjét határozzák meg – egy csomópontból az azt követőbe húzott vonal egy él
 - A **gyökérelem** a fa első eleme, amelynek nincs megelőzője
 - **Levélelem** a fa azon eleme, amelynek nincs rákövetkezője
 - **Közbenső elem** az összes többi adatelem

Fák

- Az adatszerkezetben
 - Minden közbenső elem egy részfa gyökereként tekinthető, így a fa részfákra bontható:
 - részfa: „t” részfája "a" -nak, ha
 - "a" a gyökere, azaz közvetlen megelőző eleme „t”-nek, vagy
 - „t” részfája "a" valamely részfájának
 - **elágazásszám**: közvetlen részfák száma
 - a fa **szintje** a gyökértől való távolságot mutatja.
 - A gyökérelem a 0. szinten van.
 - A gyökérelem rákövetkezői az 1. szinten. stb.
 - a fa szintjeinek száma a fa **magassága**

Fák

- További definíciók:
 - **Csomópont foka**: a csomóponthoz kapcsolt részfák száma
 - **Fa foka**: a fában található legnagyobb fokszám
 - **Levél**: 0 fokú csomópont
 - **Elágazás** (közbenső v. átmenő csomópont): > 0 fokú csomópont
 - **Szülő** (ős): kapcsolat kezdőpontja
 - csak a levelek nem szülők
 - **Gyerek** (leszármazott): kapcsolat végpontja
 - csak a gyökér nem gyerek
 - ugyanazon csomópont leszármazottai egymásnak testvérei

Fák

- További definíciók

- **Szintszám**: gyökértől mért távolság.
 - A gyökér szintszáma 0.
 - Ha egy csomópont szintszáma n , akkor a hozzá kapcsolódó csomópontok szintszáma $n + 1$.
- **Útvonal**: az egymást követő élek sorozata
 - Minden levélelem a gyökértől pontosan egy úton érhető el.
- **Ág**: az az útvonal, amely levélben végződik
- **Üresfa** az a fa, amelyiknek egyetlen eleme sincs. (Ω)
- **Fa magassága**: a levelekhez vezető utak közül a leghosszabb
 - Mindig eggyel nagyobb, mint a legnagyobb szintszám

Fák

- További definíciók:
 - **Minimális magasságú** az a fa, amelynek a magassága az adott elemszám esetén a lehető legkisebb.
 - Valójában ilyenkor minden szintre a maximális elemszámú elemet építjük be.
 - Egy fát **kiegyensúlyozott**nak nevezünk, ha csomópontjai azonos fokúak, és minden szintjén az egyes részfák magassága nem ingadozik többet egy szintnél.
 - **Rendezett fa**: ha az egy szülőhöz tartozó részfák sorrendje lényeges, azok rendezettek.

Feladat

- Maximum hány csomópont helyezhető el egy f fokú, m szintet tartalmazó fában?

$$1 + f + f^2 + f^3 + \dots$$

$$\frac{(f^{m+1} - 1)}{(f - 1)}$$

Fák műveletei

- Lekérdező
 - Üres_e – logikai értéket ad vissza
 - Gyökérelem – visszaadja a gyökér adatelemet
 - Keres(e) – adott e adatelemet keres, egy ilyen elem mutatóját adja vissza

Fák műveletei

- Módosító

- Üres
- Beszúr(e)
- MódosítGyökér(e)
- Töröl(e)

- TörölFa

- létrehoz egy üres fát
- adott e adatelemet beszúr
- adott e adatelem lesz a gyökér
- törli az e adatelemet
 - egy előfordulást
 - összes előfordulást
- törli az összes elemet

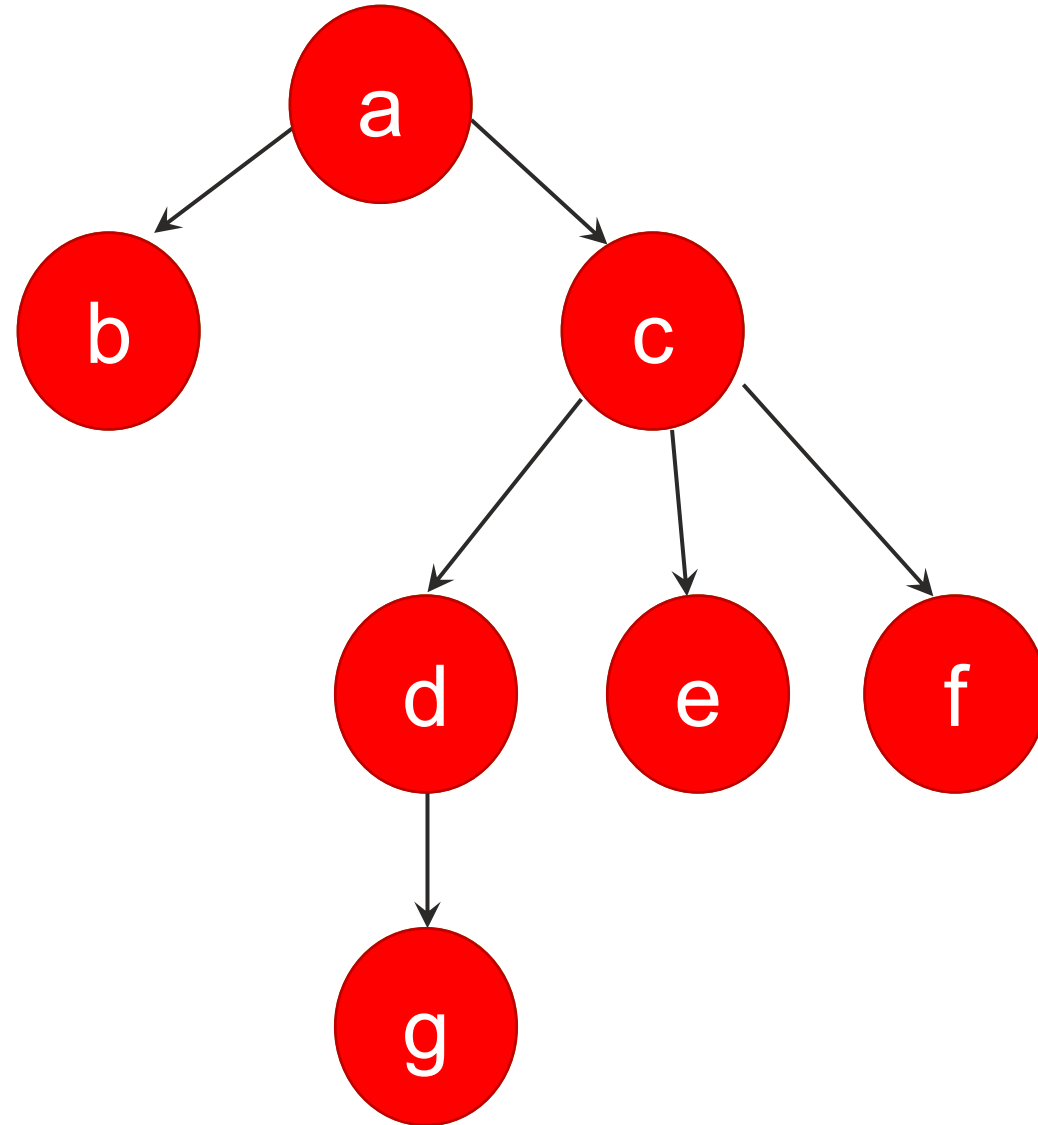
Fák műveletei

- Fák **bejárása**
 - A fa csomópontjaiban általában adatokat tárolunk. Ezeket valamilyen sorrendben szeretnénk egymás után elérni.
- Általános fa esetén a bejárási stratégiák
 - **Gyökérkezdő** (preorder)
 - gyökér, majd a részfák bejárása sorban
 - például balról jobbra
 - **Gyökérvégző** (postorder)
 - részfák bejárása sorban, majd a gyökér

Preorder bejárás

Gyökér, majd a részfák bejárása sorban (balról jobbra)

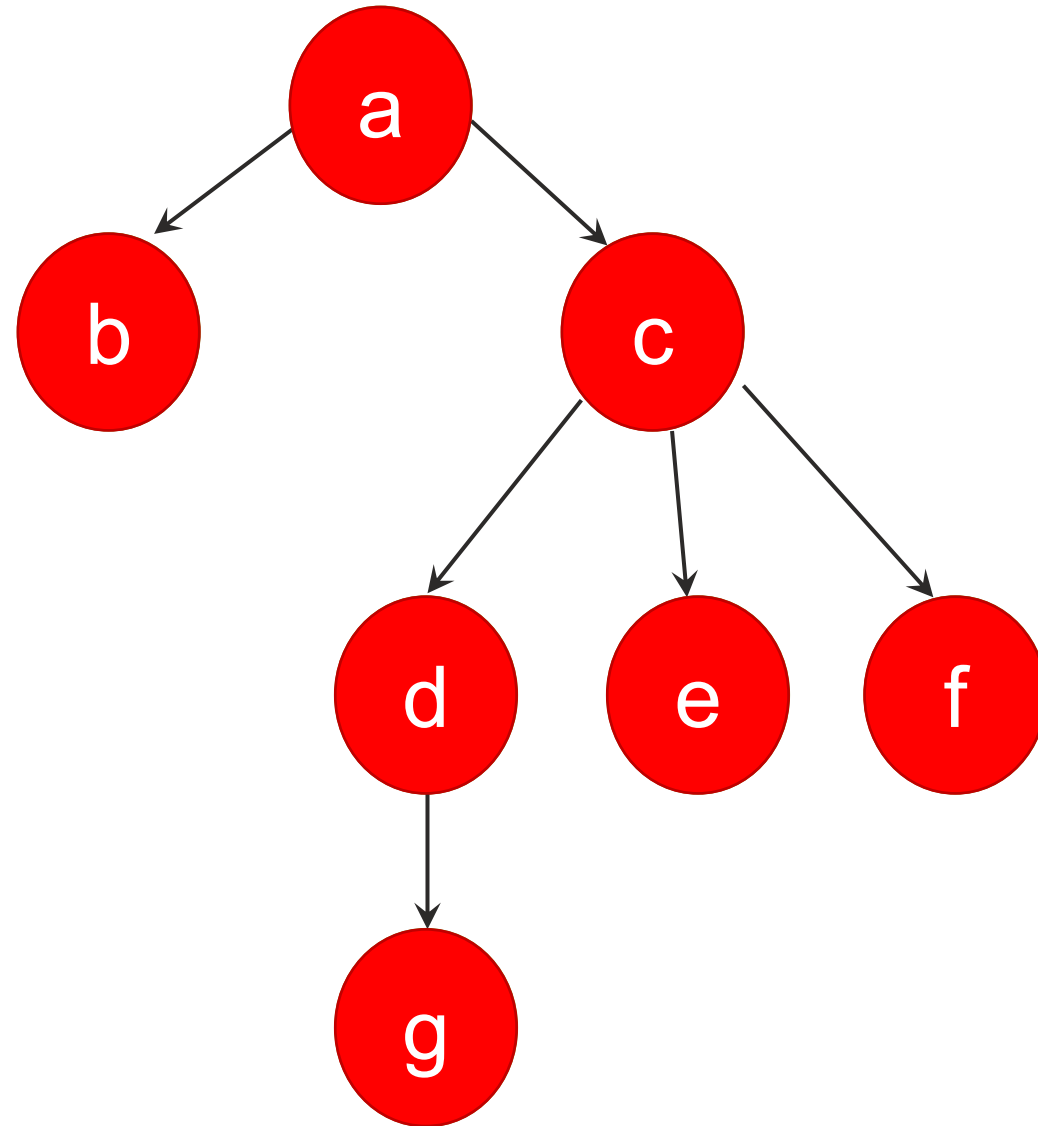
a b c d g e f



Postorder bejárás

Részfák bejárása sorban,
majd a gyökér

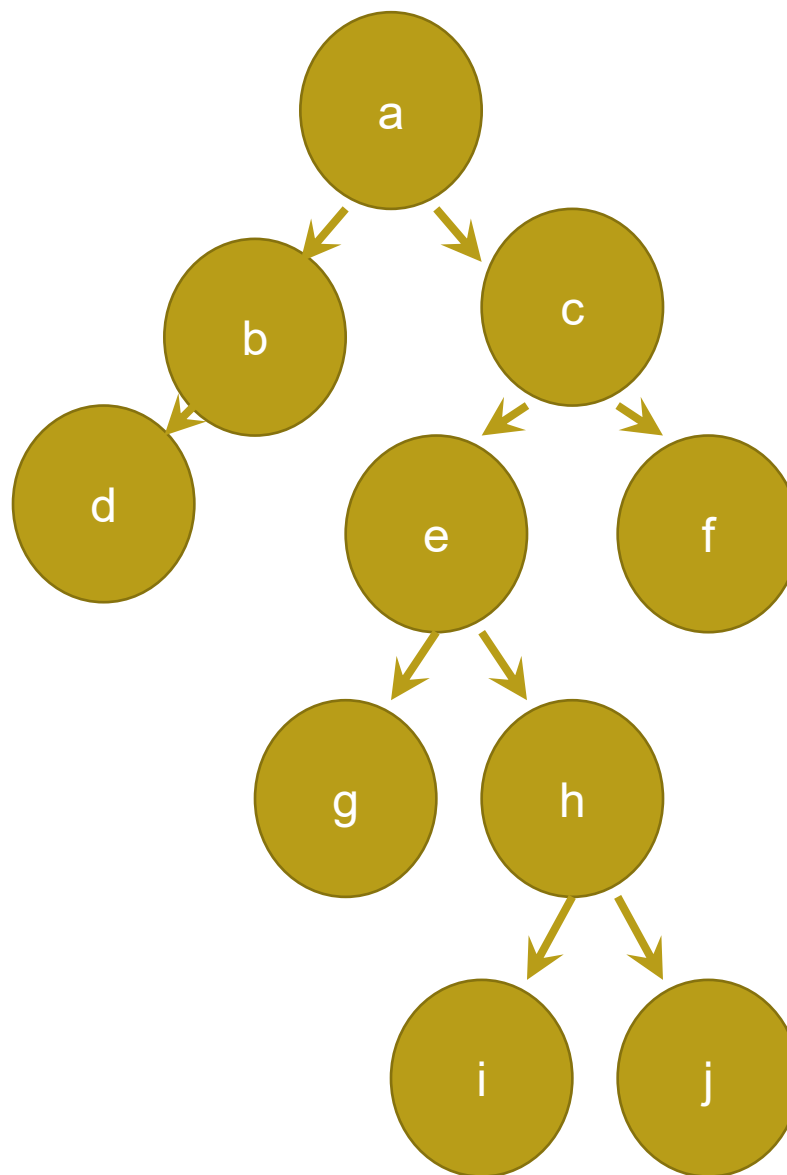
b g d e f c a



Bináris fák

- A bináris fa olyan fa, amelynek csúcspontjaiból ***maximum*** 2 részfa nyílik
 - Azaz fokszáma 2
- A szülő mindig a gyerekek között helyezkedik el
 - Van értelme a „gyökérközepű” (inorder) bejárásnak

Bináris fa



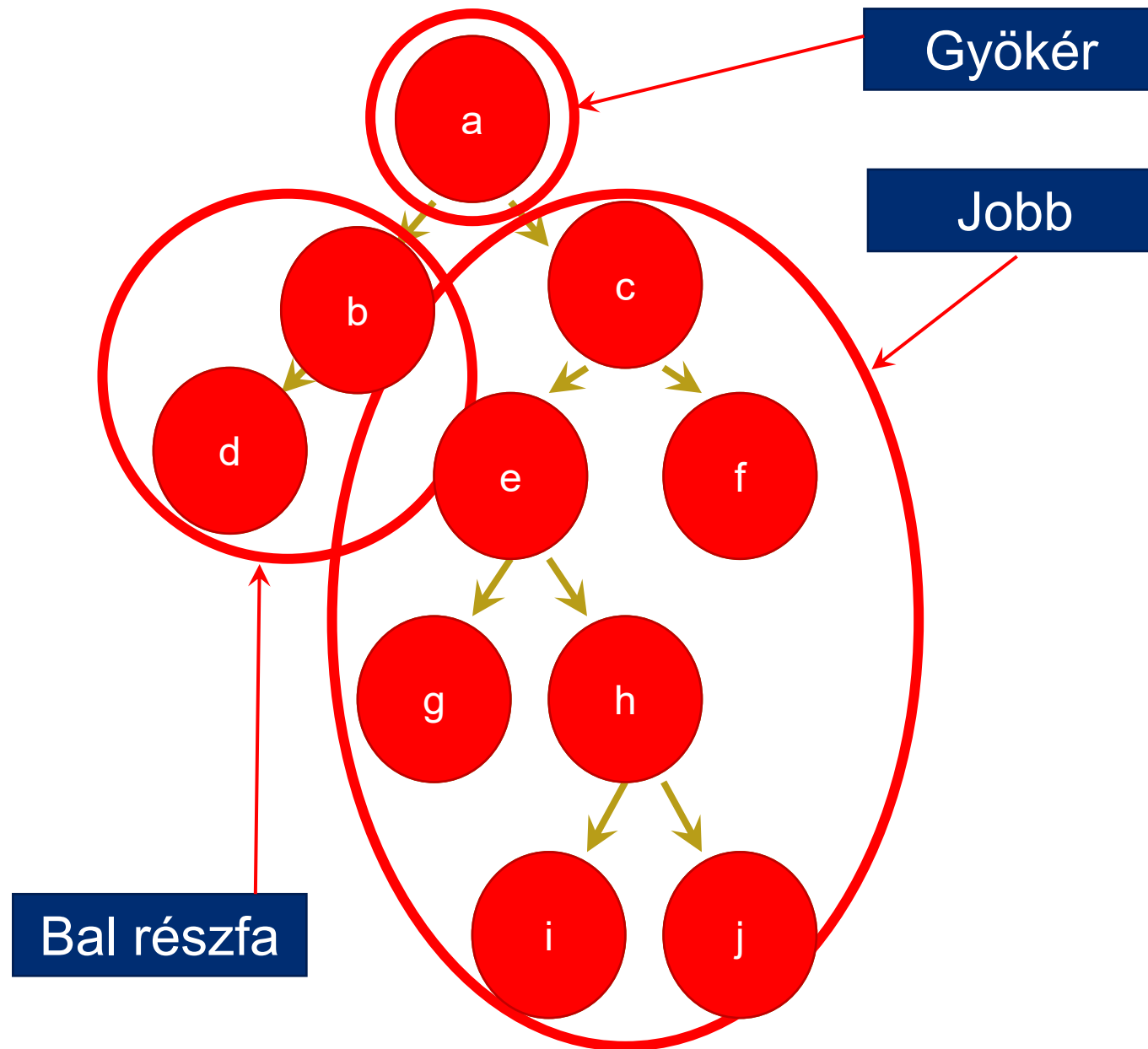
Bináris fák bejárása

- A bejárési stratégiák
 - **Gyökerkezdő** (preorder)
 - gyökér, bal részfa, jobb részfa
 - **Gyökerközepű** (inorder)
 - bal részfa, gyökér, jobb részfa
 - **Gyökérvégző** (postorder)
 - bal részfa, jobb részfa, gyökér

Preorder

Gyökér,
Bal részfa,
Jobb részfa

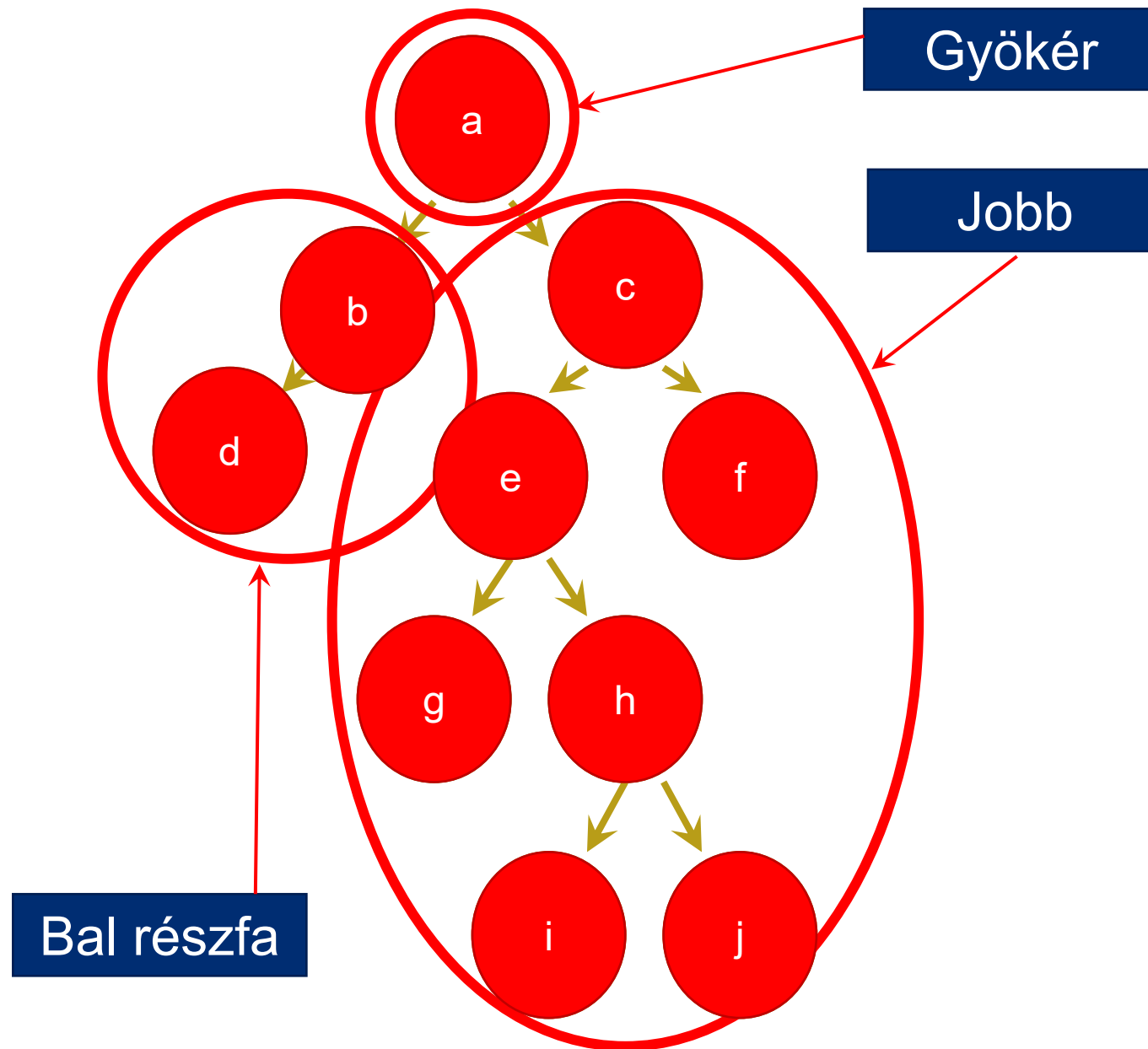
a b d c e g h i j f



Inorder

Bal részfa,
Gyökér,
Jobb részfa

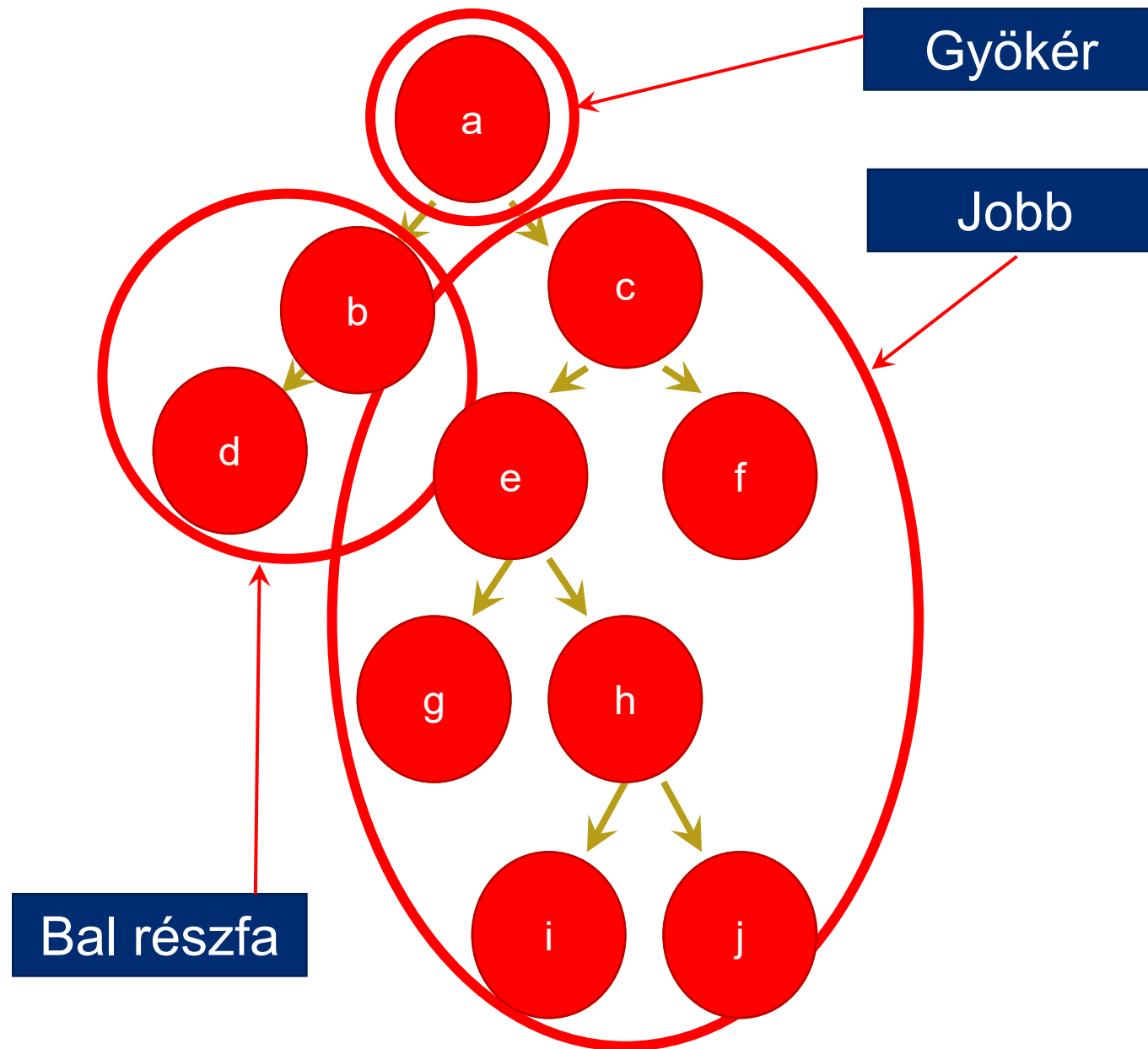
d b a g e i h j c f



Postorder

Bal részfa,
Jobb részfa,
Gyökér

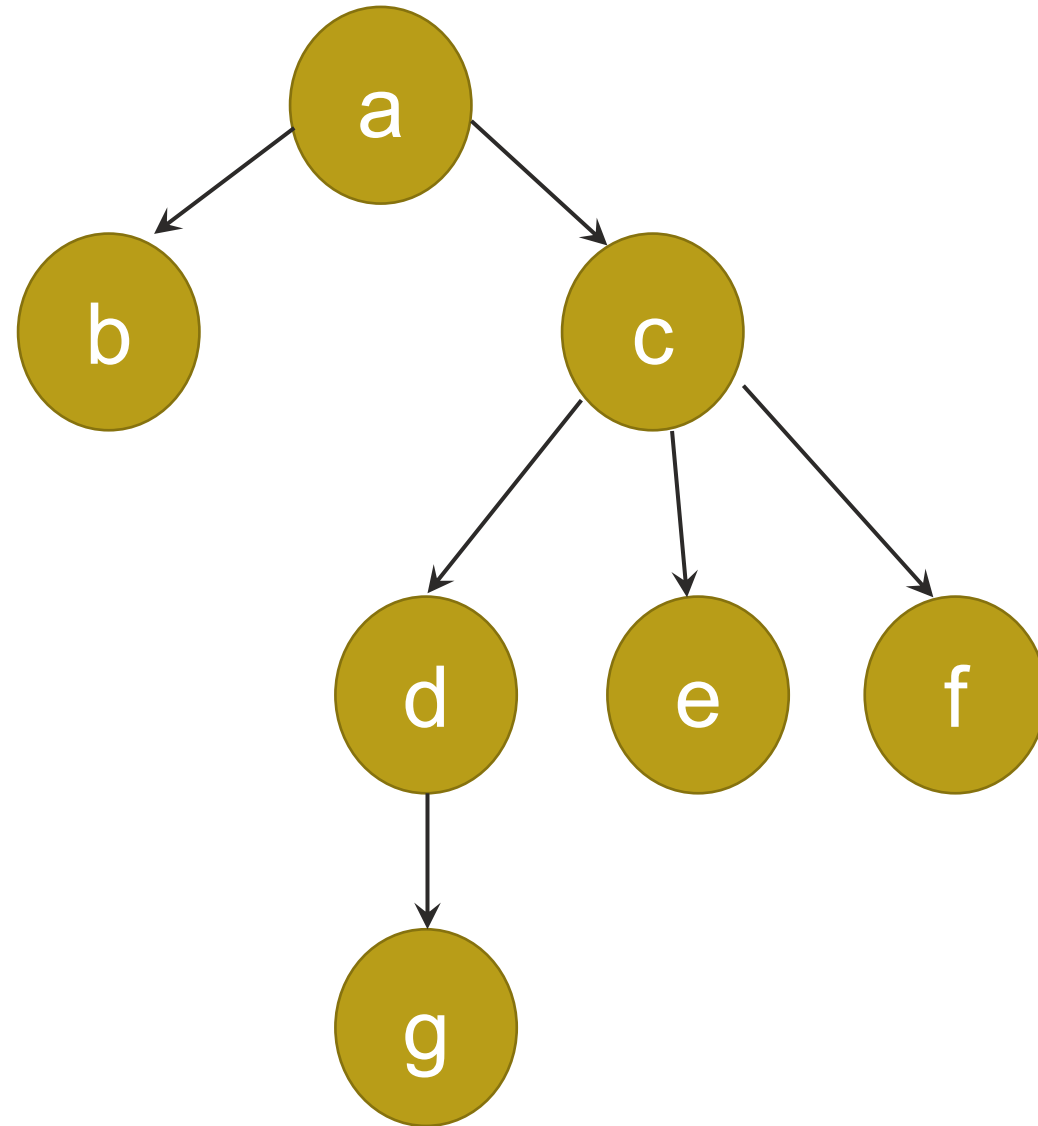
d b g i j h e f c a



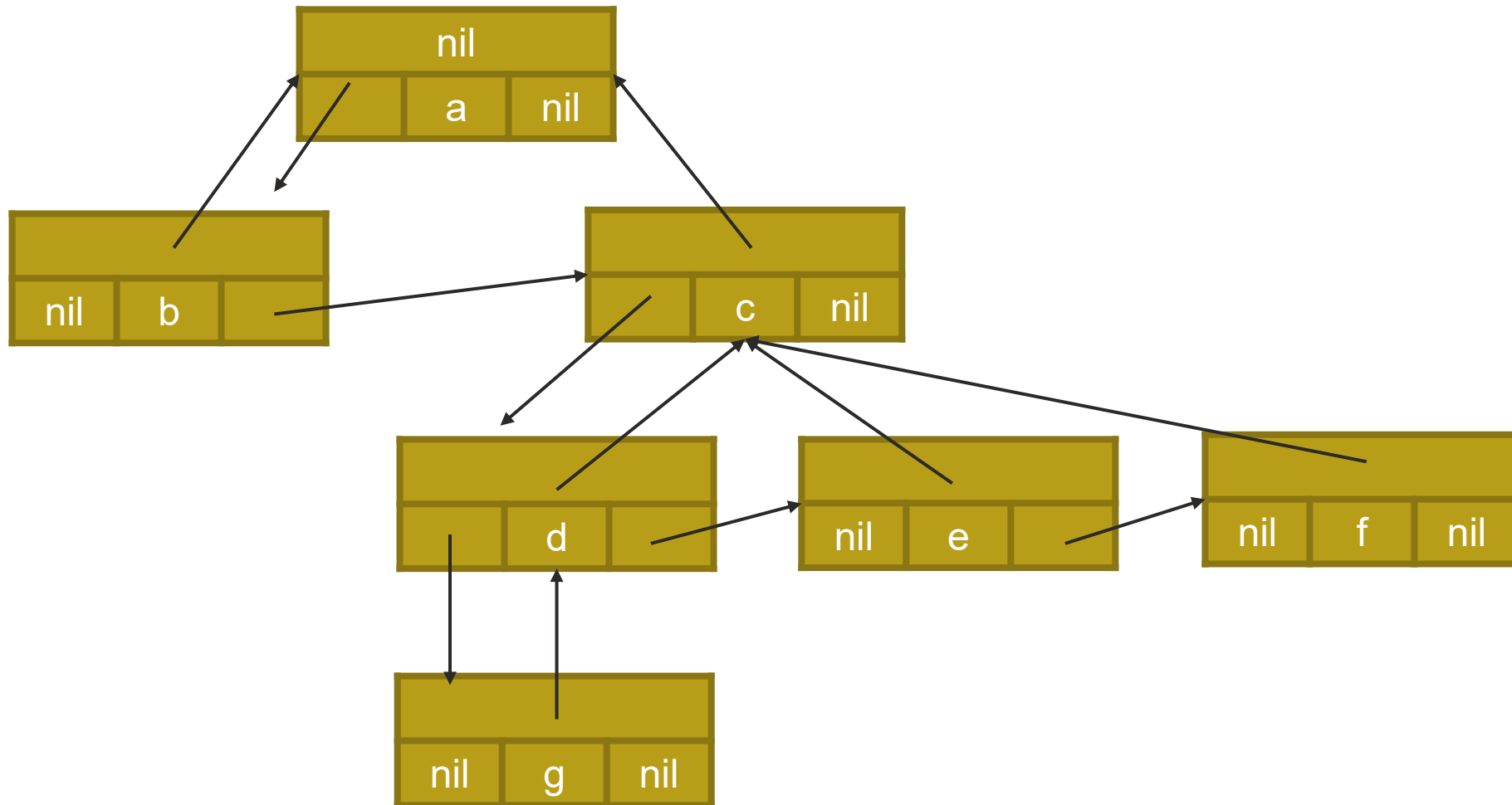
Reprezentáció

- Általános fa esetén
 - „**bal-gyermek, jobb-testvér**”
 - Minden csomóponthoz tartozik három mutató
 - **bal-gyermek** – a csúcs gyermekei közül a bal szélsőre mutat
 - **jobb-testvér** – a csúcsnak arra a testvérére mutat, amelyik közvetlenül jobbra mellette található (azonos szinten ugyanahhoz az őshöz tartozó következő szomszédos elemre)
 - **szülő**

Általános fa



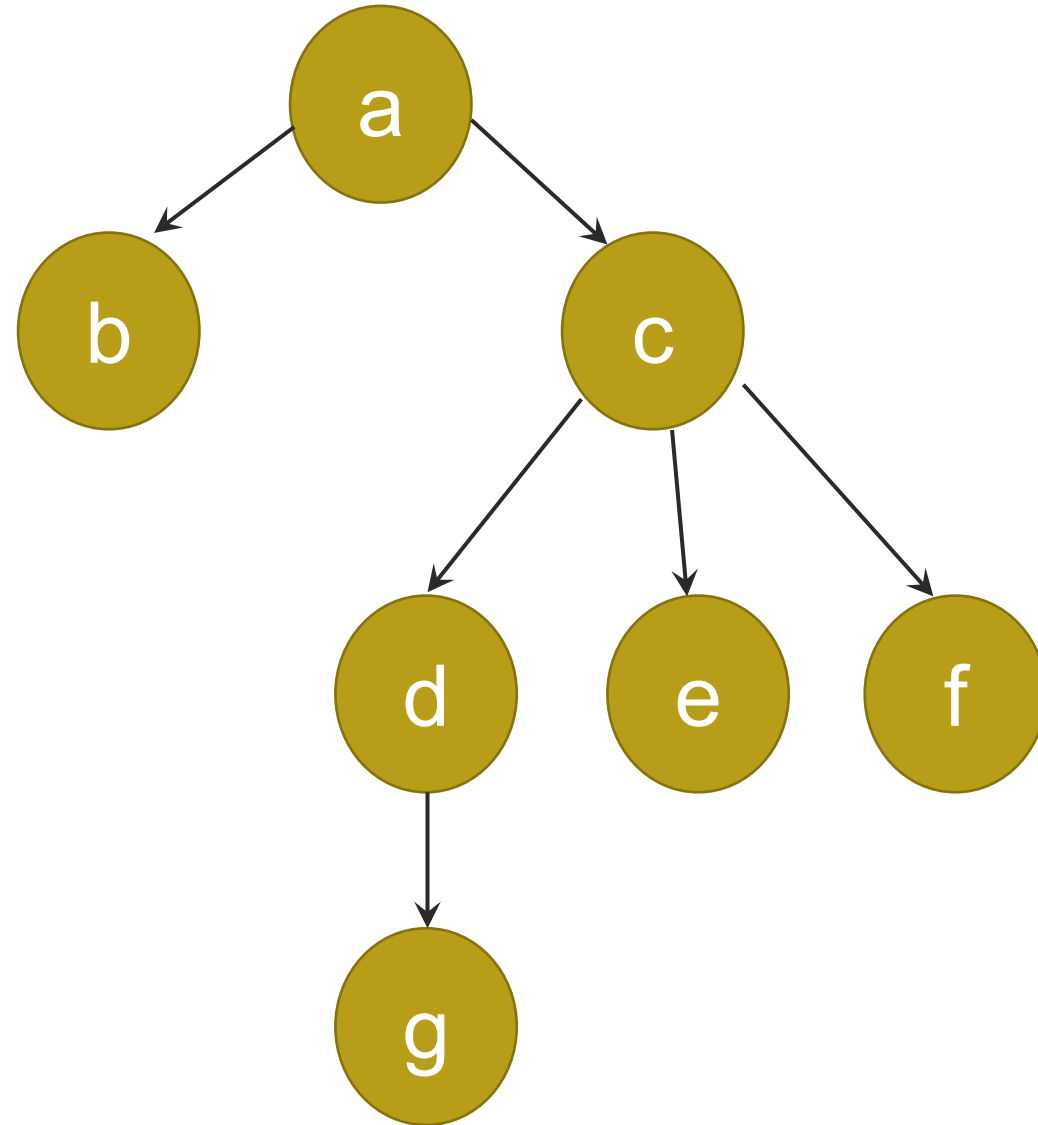
Bal-gyermmek, jobb-testvér



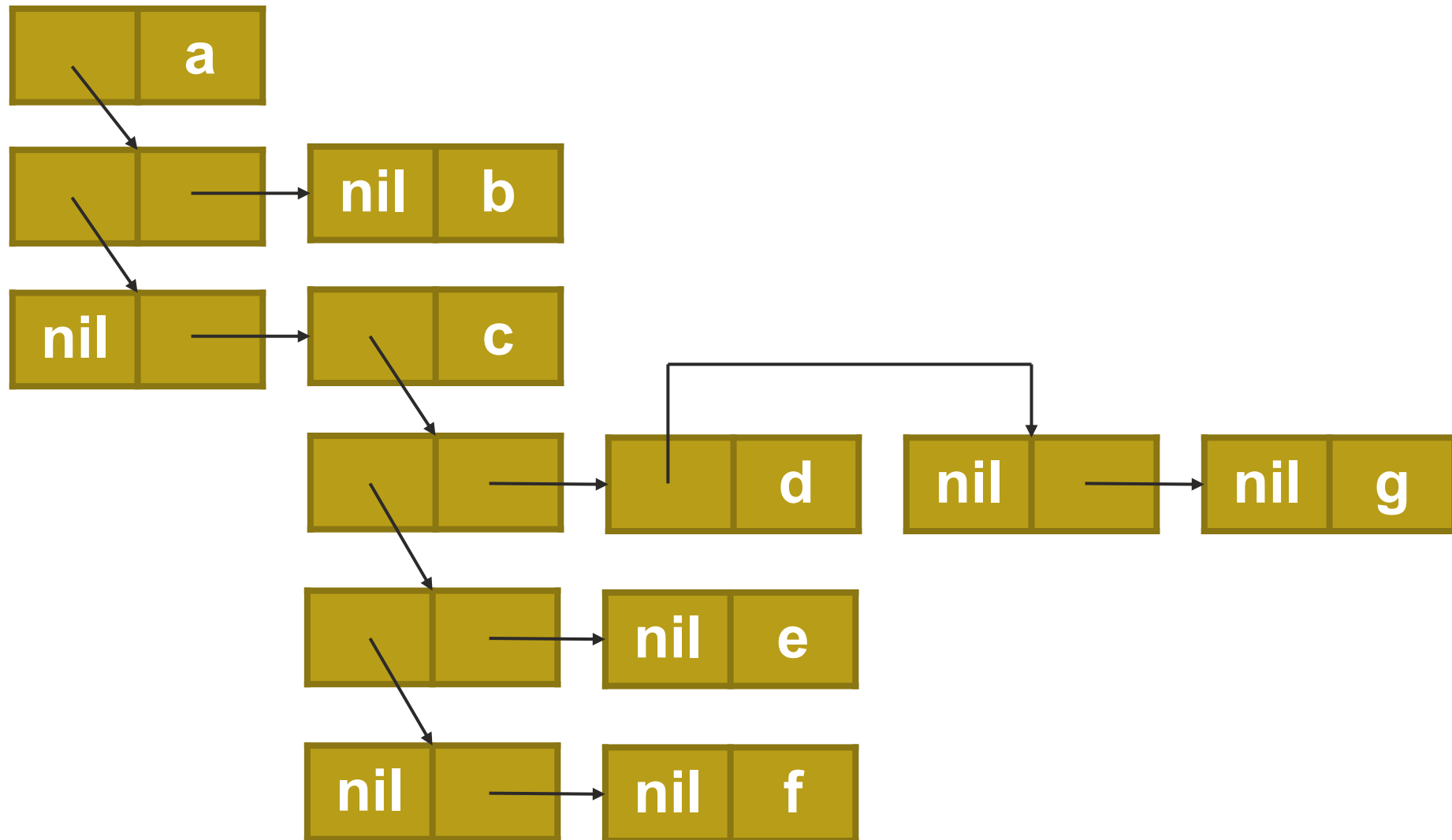
Reprezentáció

- Általános fa esetén például **multilista**:
 - Minden csomóponthoz tartozik egy lineáris lista, amelynek első eleme az adat, a többi a kapcsolatok listája
 - Annyi kapcsolati elem, ahány fokú a csomópont.
 - A kapcsolatok újabb csomópontokra, illetve lineáris listákra mutatnak.

Általános fa



Multilista



Reprezentáció

- Korlátos általános fa esetén további lehetőség
 - Aritmetikai ábrázolás
 - Láncolt, ahol minden csomópontnak van pontosan k db mutatója a maximum k gyerekre