

5. tétel

Adatszerkezetek és algoritmusok vizsga

Frissült: 2013. január 28.

A hierarchikus adatszerkezet

A hierarchikus adatszerkezet olyan $\langle A, R \rangle$ rendezett pár, amelynél van egy kitüntetett $r \in A$ elem, ez a **gyökérem**, úgy, hogy:

1. r nem lehet végpont, azaz

$$\forall a \in A \quad \neg R(a, r)$$

2. $\forall a \in \{A \setminus \{r\}\}$ elem egyszer és csak egyszer 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 valamilyen értelemben a lista általánosításai.

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. (Pl. 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.

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.

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. Minden közbenső (és levél) 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*.

Csomópont foka: a csomópont-hoz 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.

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. (W)

Fa magassága: a fa szintjeinek a száma. (Mindig eggyel nagyobb, mint a legnagyobb szintszám.)

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 *kiegyensúlyozottnak* 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.

Egy *f* fokú *m* magasságú fa maximális csomópontjainak száma.

$$(f^{m+1} - 1)/(f - 1)$$

Fák műveletei

Lekérdező

- Üres e - logikai értéket ad vissza
- Gyökérelem - visszaadja a gyökér adatelemet
- Keres(e) - adott adatelemet keres, elem mutatóját adja vissza

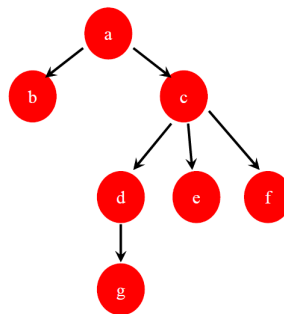
Módosító

- Üres - üres fát létrehoz
- Beszúr(e) - adott e adatelemet beszúr
- MódosítGyökér(e) - adott e adatelem lesz a gyökér
- Töröl(e) - törli az e adatelemet (egy előfordulást – összes előfordulást)
- TörölFa - törli az összes elemet

Fa bejárása

fa csomópontjaiban általában adatokat tárolunk. Ezeket valamilyen sorrendben szeretnénk egymás után elérni. Általános fa esetén a bejárési stratégiák:

- Gyökérkezdő (preorder): gyökér, majd a részfák bejárása sorban (pl. balról jobbra)



Pl. a fenti fa esetében: $a\ b\ c\ d\ g\ e\ f$

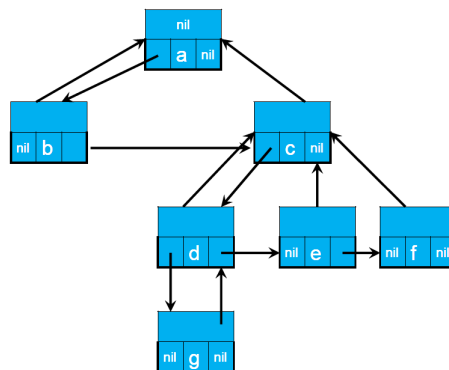
- Gyökérvégző (postorder): részfák bejárása sorban, majd a gyökér
Példában: $b\ g\ d\ e\ f\ c\ a$

Reprezentációja

Bal gyermek, jobb testvér.

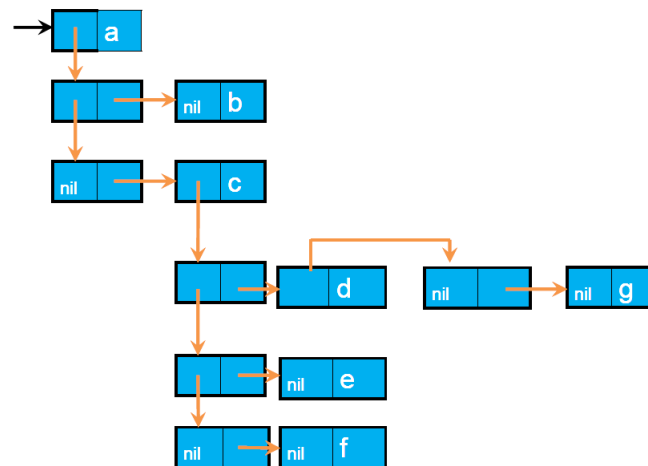
Minden csomóponthoz három mutató tartozik:

- 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



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.



Egyéb

- aritmetikai ábrázolás

- láncolt, ahol minden csomópontnak van pontosan k db mutatója a max. k gyerekre

1. Bináris fa

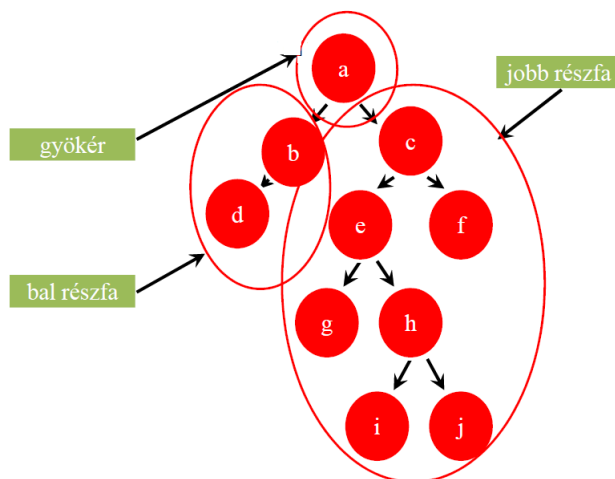
A bináris fa olyan fa, amelynek csúcspontjaiból maximum 2 részfa nyílik (azaz fokszáma 2).

Bejárása

A szülő mindig a gyerekek között helyezkedik el, itt gyökérközepű, inorder bejárás is van.

Itt a bejárások:

- Gyökérkezdő (preorder): gyökér, bal részfa, jobb részfa
- Gyökérközepű (inorder): bal részfa, gyökér, jobb részfa



A fenti fa bejárása: $d b a g e i h j c f$

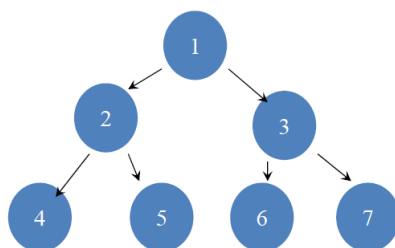
- Gyökérvégző (postorder): bal részfa, jobb részfa, gyökér

Reprezentáció

Aritmetikai ábrázolás

Szint-folytonosan ábrázolunk egy tömbben.

Például ezt a fát



így tároljuk el:

1	2	3	4	5	6	7
---	---	---	---	---	---	---

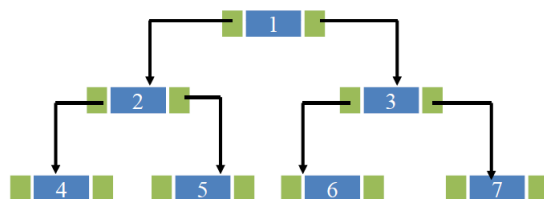
Az indexfüggvények:

$$ind(bal(c)) = 2 * ind(c)$$

$$ind(jobb(c)) = 2 * ind(c) + 1$$

Láncolt ábrázolás

Minden elem esetében mutató a bal és a jobb gyerekre (többnyire a szülőre is).



Bináris fák tulajdonságai

Egy bináris fa akkor **tökéletesen kiegyensúlyozott**, ha minden elem bal-, illetve jobboldali részfájában az elemek száma legfeljebb eggyel tér el. **Teljesnek** nevezünk egy bináris fát, ha minden közbenső elemének pontosan két leágazása van. **Majdnem teljes**, ha csak a levelek szintjén van esetleg hiány.

Műveletek

- üres fa inicializálása
- az üres fa gyökérelemének definiálása
- a gyökér és a két részfa csoportosítása (az egyik részfa lehet üres)
- egy elem hozzáadása egy olyan elem bal (jobb) oldalához, amelynek nincs bal (jobb) oldali leágazása
- jelezzük, ha a fa üres
- jelezzük, ha nincs bal (jobb) oldali leágazása az aktuális elemnek a gyökérelem elérése

- egy adott elem elérése, egy elem bal (jobb) oldali részfájának az elérése a gyökérből
- egy fa kettéválasztása egy elemre (régi gyökér) és egy vagy két részfára (attól függően, hogy a gyökérnek egy vagy két leágazása volt)
- egy (rész-)fa törlése (a fa lehet egyetlen elem?)
- egy részfa helyettesítése egy másik részfával
- stb.