

Algoritmuskészítés gyakorlat Keresőfák

Jelölés Piros-fekete fában fekete csúcs karikával, piros négyzettel jelölve.

1. Milyen egy bináris keresőfa struktúrája? Miben különbözik a bináris kupactól? Hogyan járja be a fát pre-, az in- és a postorder bejárás?
2. Mely bejárásoknál lehetséges az, hogy a tárolt elemek legnagyobbika megelőzi a legkisebbet?
3. Egy bináris fa inorder bejárása: $j, b, k, g, i, a, c, d, f, e, h$ preorder bejárása: $a, b, j, g, k, i, d, c, e, f, h$. Rekonstruáljuk a fát!
4. *Gondolkodtató HF*: Egy bináris fa preorder bejárása $a, b, c, d, e, f, g, h, i, j, k, l, m, n, o$, postorder bejárása $e, d, c, h, i, g, f, b, l, n, o, m, k, j, a$. Egyértelmű-e ebből az inorder bejárás? Ha igen, akkor mi a csúcslista, ha nem, akkor miért nem?
5. Egy bináris keresőfában csupa különböző egész számot tárolunk. Lehetséges-e, hogy egy $KERES(x)$ hívás során a keresési út mentén a 20, 18, 3, 15, 5, 8, 9 kulcsokat látjuk ebben a sorrendben? Ha nem lehetséges, indokoljuk meg miért nem, ha pedig lehetséges, határozzuk meg az összes olyan x egész számot, amire ez megtörténhet.
6. Mik a piros-fekete fák? Hogyan definiáljuk? Mit biztosítanak a piros-fekete fák az elemek kereshetőségére vonatkozóan a bináris keresőfákhoz képest?
7. Hány csúcsa van egy piros-fekete fának, ha k elemet tárolunk benne?
8. Melyik az a 10 pontú piros-fekete fa, aminek a legnagyobb a magassága?
9. Milyen az piros-fekete fa, aminek minden csúcsa fekete?
10. Mennyi a tárolható elemek számának minimuma, illetve maximuma egy olyan piros-fekete fában, aminek a fekete magassága 3, illetve ha 4?
11. Adott egy n csúcsú és egy k csúcsú piros-fekete fa. A két fában tárolt összes elemből $O(n+k)$ lépésben szeretnénk készíteni egy rendezett tömböt, tudjuk, hogy meg lehet csinálni. Adjunk megfelelő algoritmust!
12. *Gondolkodtató HF*: Illesszük be egy piros-fekete fába sorban a 8, 2, 4, 7, 5, 3, 1, 6 elemeket. Ezután töröljük az 5, majd a 7 elemet. Okoz-e a naív beszúrás valamilyen problémát?
13. Szúrjuk be egy bináris keresőfába a 6, 2, 1, 4, 5, 8, 9, 7, 10 elemeket, majd töröljük a 7, 8, 2 elemeket!
14. Lehetséges-e, hogy egy piros-fekete fából a piros-fekete tulajdonság megsértése nélkül pirosról feketére illetve feketéről pirosra átszínezzünk
 - (a) néhány csúcsot?
 - (b) pontosan egy csúcsot?
 - (c) pontosan k csúcsot?
15. Hogyan definiáljuk a jobbra forgatást piros-fekete fákban?
16. Milyen eseteket különböztetünk meg piros-fekete fába való beszúráskor? Mi az egyes esetekben a teendőnk, hogy fenntarthassuk a piros-fekete fa által előírt tulajdonságokat a művelet után is?
17. *Ezek alapján*: Illesszük be egy piros-fekete fába sorban a 8, 2, 4, 7, 5, 3, 1, 6, 9 elemeket. Ezután töröljük az 5, majd a 7 elemet.
18. Milyen költséggel tudunk bináris keresőfával illetve piros-fekete fával rendezni egy n elemű listát?
19. Egy sportklub teniszezői kialakítottak egy erőssorrendet. Ezt a következő szabályok szerint tartják karban: (1) új játékos mindig a sorrend végére kerül, (2) a rangsor szerinti i -edik játékos kihívhatja az $i + 1$ -ediket, és ha legyőzi, helyet cserélnek. Tervezzük olyan hatékony adatszerkezetet, mely lehetővé teszi a rangsor számítógépes kezelését! A szükséges funkciók a következők:
BESZÚR(név): az új játékos "név" azonosítóval a rangsor végére teszi
KIHÍV(név): az $i + 1$. helyen álló játékos nevét adja, ha a "név" azonosítójú játékos az i . helyen áll és $i \geq 1$.
KICSERÉL(i): kicseréli a rangsor i . és $i + 1$. helyén lévő játékosokat, ha $i \geq 1$.