Bináris keresőfák, piros-fekete fák

Algoritmuselmélet

8. gyakorlat

2023.

1. (a) Építsünk beszúrásokkal bináris keresőfát az alábbi sorrendben érkező számokból: 10, 3, 8, 12, 1, 5, 15, 4, 6, 13.
   (b) Hajtsuk végre rendre a TÖRÖL(4), TÖRÖL(12) és TÖRÖL(10) műveleteket.
   (c) Milyen sorrendben írja ki a preorder, inorder és posztorder bejárás a fában tárolt értékeket?
2. 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?
3. Egy bináris fa inorder bejárása $j, b, k, g, i, a, c, d, f, e, h$, preorder bejárása pedig $a, b, j, g, k, i, d, c, e, f, h$. Rekonstruáljuk a fát.
4. Egy bináris keresőfában n egész számot, egy másikban pedig m egész számot tárolunk. Rendezzük az $n + m$ elemet $O(n + m)$ lépésben.
5. Egy bináris keresőfában n különböző egész számot tárolunk. Adjunk algoritmust, ami $O(n)$ lépésben eldönti, hogy van-e a tárolt számok között két olyan, melyek különbsége 2013.
6. Adott két bináris keresőfa, mindegyikben n különböző elemet tárolunk. Adjunk $O(n)$ lépésszámú eljárást, ami eldönti, hogy igaz-e, hogy a két fában ugyanazok a számok szerepelnek.
7. Adott két teljes bináris keresőfa, mindegyikben n elemet tárolunk. Adjunk $O(\log n)$ lépésszámú eljárást, ami eldönti, hogy az első fa összes eleme nagyobb-e a második fa összes eleménél.
8. Adott egy n-csúcsú bináris keresőfa, ami csupa különböző elemet tárol. Ennek minden v csúcsára meg akarjuk határozni, hogy a v gyökerű részfában hány darab v-nél kisebb elem van tárolva. Adjunk algoritmust, ami ezt a feladatot $O(n)$ lépésben megoldja.
9. Lehetséges-e hogy az alábbi ábrákon egy piros-fekete fa csúcsait ábrázoltuk? (Az üres leveleket nem rajzoltuk fel, a fekete körök fekete csúcsokat, fehér négyzetek piros csúcsokat jelölnek.)

10. Mennyi a tárolható elemek számának minimuma, illetve maximuma egy olyan piros-fekete fában, aminek a fekete magassága 3?
11. Egy piros-fekete fában a $2010, 42, 100\pi, 1848, 3$ elemeket tároljuk úgy, hogy a gyökérben levő elem a 42. Hogyan nézhet ki a fa? (Adja meg az összeset és indokolja meg, hogy más nincs.)
12. Előfordulhat-e, hogy egy piros-fekete fában a KERES művelet végrehajtása során bejárt nemlevél csúcsokban sorban a 2, 20, 12, 5, 8, 15, 10 elemeket találjuk?

13. Lehetséges-e, hogy egy piros-fekete fából a preorder bejárás 6, 1, 5, 3, 2, 4 sorrendben olvassa ki az elemeket?

14. Egy piros-fekete fában valamelyik, a gyökértől egy levélig vezető úton sorban az alábbi színű pontok vannak: fekete, piros, fekete, fekete. Mennyi a fában tárolt elemek számának minimuma?

15. Határozzuk meg egy piros-fekete fa azon v csúcsait, melyek esetén a v csúcsú részfa is egy piros-fekete fa.

16. Tervezzünk adatstruktúrát a következő feltételekkel. Természetes számokat kell tárolni, egy szám többször is szerepelhet, és a szükséges műveletek a következők. BESZÚR(i): az i szám egy újabb példányát tároljuk; TÖRÖL(i): az i szám egy példányát töröljük; MINDTÖRÖL(i): az i szám összes példányát töröljük; DARAB(i): visszaadja, hogy hány példány van az i számból. Az adatstruktúra legyen olyan, hogy ha m-féle elemet tárolunk, akkor mindegyik művelet lépésigénye $O(\log m)$.

17. Egy orvosi rendelőben a regisztrációnál kell bejelentkezni, ahol az ott dolgozók eldöntik, hogy a beteg az épp rendelő két orvos közül A-hoz vagy B-hez kell kerüljön, vagy bármelyikükhöz kerülhet. Ezen kívül, a beutaló ismeretében, a beteghez egy, a sürgősséget kifejező, számot is rendelnek. Amikor valamelyik orvos végzett egy beteggel, akkor azon betegek közül, akiket nem csak a másik orvos láthat el, behívja a legkisebb sürgősségi számút. Tegyük fel, hogy a kiosztott sürgősségi számok egymástól különbözőek. Írjunk le egy olyan adatszerkezetet, ami abban az esetben, ha n beteg várakozik, akkor a regisztráción az új beteg beillesztését, illetve az orvosoknak a következő beteg kiválasztását $O(\log n)$ lépésben lehetővé teszi.

18. Tervezzünk olyan adatszerkezetet, ami egy rendezett halmaz elemeinek tárolására szolgál és a megvalósítandó műveletek a következők. FELÉPÍT(A): az A halmaz elemeiből felépíti a struktúrát; MINTÖR, MAXTÖR: a minimális, illetve maximális elem törlése; BESZÚR(x): az x elemet a struktúrába illeszti. A FELÉPÍT művelet költsége legyen O(n), a MINTÖR, MAXTÖR, BESZÚR műveleteké pedig $O(\log n)$, ahol n a tárolt elemek száma.

19. Egy gyökeres szintezett fán A és B a következő játékot játssza: felváltva mozgatnak egy bábut, ami kezdetben az első szinten, a gyökérben van. Minden lépésben a soron következő játékos az aktuális v csúcsból v valamelyik fiába mozgatja a bábut. A játéknak akkor van vége, ha a bábu a fa egyik levelébe kerül. A levelek egy része zöldre van festve. A kezdő A játékos akkor nyer, ha a játék egy zöld levélben ér véget. Adott a fa éllistája, amiből minden levélről kiderül, hogy az zöld-e. Mutassunk egy $O(n)$ lépésszámú algoritmust, amely meghatározza, hogy az A játékos hogyan játsszon, hogy biztosan nyerjen (feltéve, hogy van ilyen nyerő stratégiája).