Rendezések

1. Rendezze a $7, 3, 15, 1, 5, 4, 8, 2$ sorozatot gyorsrendezéssel úgy, hogy mindig a tömb első elemét választja particionáló elemnek!

2. Egy tömbön gyorsrendezést futtatva az első particionálás után az eredmény: $4, 2, 3, 1, 6, 8, 11$. Mi lehetett a particionáló elem?

3. Igazolja, hogy nincs olyan összehasonlításokkal rendező algoritmus, amelynél akármi is a bemenet, minden elem legfeljebb 2020 összehasonlításban szerepel!

4. Adjon egy $O(n)$ időigényű algoritmust n olyan egész számból álló sorozat rendezésére, melynek elemei az
   (a) $\{1, \dots, 3n\}$ halmazból vannak!
   (b) $\{1, \dots, n^3 - 1\}$ halmazból vannak!

5. A $G = (V,E)$ többszörös élet nem tartalmazó irányított gráf csúcshalmaza legyen $V = \{1, 2, \dots, n\}$. Tegyük fel, hogy a gráf olyan éllistával adott, amelyben minden csúcsnál a szomszédok tetszőleges sorrendben vannak felsorolva. Adjon algoritmust, ami $O(|V|+|E|)$ lépésben olyan éllistát hoz létre, amiben a szomszédok minden csúcsnál növekvő sorrendben vannak!

6. Egy bináris keresőfában csupa különböző egész számot tárolunk. Mely x egész számokra lehetséges, hogy egy KERES(x) hívás során a keresési út mentén a 20, 18, 3, 15, 5, 8, 9 kulcsokat látjuk ebben a sorrendben?

7. (a) Építsen beszúrásokkal bináris keresőfát az alábbi sorrendben érkező számokból: 7, 3, 2, 9, 8, 12, 6, 4.
   (b) Járja be pre-. in-, és posztorder bejárással a kapott fát!
   (c) Az (a) rész keresőfáján hajtsa végre az alábbi műveletsort: BESZÚR(5), TÖRÖL(2), TÖRÖL(7), TÖRÖL(6).

8. Egy bináris fa csúcsai 0 és 9 közötti egész számokkal vannak megcímkézve. Az inorder bejárás során a címkék sorrendje: 9, 3, 1, 0, 4, 2, 7, 6, 8, 5, a posztorder bejárásnál pedig 9, 1, 4, 0, 3, x, 7, 5, y, 2. Mi lehet az x és mi az y?

9. Határozza meg azokat a bináris fákat, amelyekben a preorder bejárás szerinti sorrend éppen a posztorder bejárás által adott sorrend fordítottja!

10. Az A keresőfában n egész számot, a B-ben pedig m-et tárolunk. Rendezzük az $n + m$ elemet $O(n + m)$ lépésben!

11. Igazolja, hogy minden olyan algoritmus, ami csak összehasonlításokkal fel tud építeni egy bináris keresőfát n elem esetén $\Omega(n \log n)$ összehasonlítást használ!

12. Adott egy n csúcsú bináris keresőfa, melyben csupa különböző elemeket tárolunk. Ennek minden v csúcsára meg akarjuk határozni, hogy a v gyökerű részfában hány darab v-nél kisebb elem van tárolva. Adjon algoritmust, ami ezt a feladatot $O(n)$ lépésben megoldja!