Gyakorlás

Algoritmuselmélet

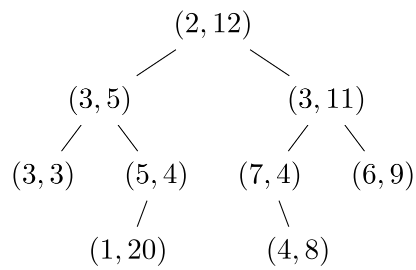
2. konzultáció

2024.

1. Egy kupac tömbös reprezentációja $3, 5, x, y, 12, 10, 100, 7, 16, 21$.
   (a) Rajzoljuk fel a kupacot bináris fa alakban.
   (b) Egy MINTÖR végrehajtása után a gyökérbe x kerül. Rajzoljuk fel ezt az új kupacot is.
   (c) Határozzuk meg x és y összes lehetséges értékét, ha tudjuk, hogy a tárolt elemek mind különböző számok.
2. Egy kupac elemeit preorder bejárás szerint kiolvasva a következő számsorozatot kapjuk: $1, 17, 19, 21, 22, 31, 37, 2, 8, 3$. Rekonstruálható-e ebből a kupac?
3. Definiáljunk egy rendezési relációt a pozitív egész számpárokból álló alaphalmazra a következőképpen: $(a, b) < (c, d)$ akkor és csak akkor áll fenn, ha

$$\text{vagy} \qquad a \cdot b < c \cdot d, \qquad \text{vagy} \qquad a \cdot b = c \cdot d \text{ és } \min(a, b) < \min(c, d).$$

   Az alábbi ábrán egy bináris keresőfa látható ezzel a rendezéssel.
   (a) Szúrjuk be a keresőfába a $(6, 4)$ párt.
   (b) Az így kapott keresőfából töröljük az $(2, 12)$ elemet.



4. Egy piros-fekete fában az x csúcs gyermekei y_1 és y_2, az y_2 gyermekei pedig z_1 és z_2. Tudjuk, hogy még, hogy y_1 levél. Mi mondható el x, y_1, y_2, z_1, z_2 színéről?
5. Egy 2-3 fában az $1, 5, 7, 8, 12, 13, 20, 21$ kulcsokat tároljuk, és a levelek feletti szinten a csúcsoknak (balról jobbra haladva) 3, 3, 2 levelük van.
   (a) Rajzoljuk fel a 2-3 fát.
   (b) Szúrjuk be a fába a 6-ot.
6. Kettős hash-elést használva akarunk beszúrni elemeket egy kezdetben üres, 11-elemű hash-táblába. Első hash-függvénynek a $h(K) = K \pmod{11}$ függvényt, második hash-függvénynek a $h'(K) = \big(K \pmod 6\big) + 1$ függvényt választjuk. Hogyan változik a tábla a $3, 6, 14, 9, 25$ kulcsok ezen sorrendben történő beszúrása során?
7. Éllistával adott egy irányítatlan, élsúlyozott G gráf és benne egy kijelölt $v \in V(G)$ csúcs. Javasoljunk $O(m \log n)$ lépésszámú algoritmust, ami eldönti, hogy létezik-e olyan minimális súlyú feszítőfa G-ben, amelyben a v csúcs elsőfokú, és ha létezik, akkor meg is ad egy ilyet.
8. Éllistával adott egy $G = (V, E)$ egyszerű, irányítatlan, összefüggő gráf, melynek az élei súlyozottak. Tegyük fel, hogy az élsúlyok mind különbözőek. A gráf egy körének értéke legyen a körben szereplő legnagyobb élsúly. Olyan kört szeretnénk találni a G gráfban, aminek az értéke minimális. Adjunk algoritmust, mely $O\big(|V|^2 \log |V|\big)$ lépésben talál egy ilyen kört vagy jelzi, hogy nincs kör G-ben.
9. Tegyük fel, hogy van egy algoritmusunk ami 1 lépésben tetszőleges $\varphi(x_1, \ldots, x_m)$ Boole-formuláról eldönti, hogy kielégíthető-e. Adjunk polinomiális algoritmust, ami egy tetszőleges kielégíthető formulára megad egy kielégítést, azaz a változók olyan értékadását, amire a formula értéke igaz.

10. Bizonyítsuk be, hogy az alábbi probléma coNP-beli.

    Bemenet: $s_1, \dots, s_n \in \mathbb{Z}^+$ számok.

    Kérdés: igaz-e, hogy $\{s_1, \dots, s_n\}$ bármely két különböző részhalmazára más a részhalmazban levő számok összege?

11. A 3KLASZTER eldöntési probléma bemenete egy K_n teljes gráf, egy $w\colon E(G) \to \mathbb{R}$ súlyfüggvény és egy $k \geq 0$ szám, és azt akarjuk eldönteni, hogy G csúcsai particionálhatók-e 3 osztályba úgy, hogy egy osztályon belül bármely él súlya legfeljebb k (de különböző osztálybeli csúcsok között bármilyen súlyú él futhat). Bizonyítsuk be, hogy a 3KLASZTER probléma NP-teljes.

12. Bizonyítsuk be, hogy az alábbi probléma NP-teljes.

    Bemenet: irányítatlan, összefüggő G gráf.

    Kérdés: ki lehet-e színezni G csúcsait 5 színnel úgy, hogy legfeljebb egyetlen olyan él legyen, amelynek a két végpontja azonos színű?

13. Tegyük fel, hogy $\mathrm{NP} \subseteq \mathrm{P}$. Következik-e ebből, hogy az alábbi eldöntési probléma coNP-beli?

    Bemenet: G irányítatlan, páros gráf.

    Kérdés: igaz-e, hogy G-ben létezik két éldiszjunkt maximális párosítás?

14. Igaz-e, hogy ha SAT $\prec X$ fennáll az alábbi X eldöntési problémára, akkor $\mathrm{P} \neq \mathrm{NP}$.

    Bemenet: irányítatlan G gráf.

    Kérdés: igaz-e, hogy G-ben nincsen 2012-elemű független ponthalmaz?

15. Adott az $a_1, a_2, \dots, a_n$ számsorozat. Ebből néhány tagot akarunk úgy kiválasztani, hogy ne legyen közöttük két szomszédos eleme a sorozatnak, és a kiválasztott elemek négyzetösszege maximális legyen. Írjuk fel egészértékű programozási feladatként ezt a problémát.

16. Adott egy $G = (V, E)$ egyszerű, irányítatlan gráf. A G gráf minden éléhez egy súlyt akarunk rendelni, a súlyok mindegyike a $0, 1, \dots, 10$ egész számok közül kerülhet ki. Célunk, hogy G-ben az élek súlyainak összege maximális legyen, de egyik csúcsnál se legyen a rá illeszkedő élek súlyainak összege 15-nél nagyobb. Írjuk fel egészértékű programozási feladatként ezt a problémát.

17. Egy $n \times n$-es táblázatban különböző egész számokat tárolunk úgy, hogy minden sorban balról jobbra és minden oszlopban fentről lefelé növekednek a számok. Adjunk $O(n)$ összehasonlítást használó algoritmust, ami eldönti, hogy egy adott k egész szerepel-e a táblázatban.

18. Tervezzünk adatstruktúrát különböző egész számok tárolására úgy, hogy az alábbi műveletek lépésszáma $O(\log n)$ legyen, ha n elemet tárolunk.

    BESZÚR(i): beszúrja az adatstruktúrába az i számot;

    MELYIK(K): megmondja, hogy a tárolt elemek között a nagyság szerinti növekvő rendezésben melyik elem a K-adik.

19. Diákok pontszámait szeretnénk nyilvántartani egy többfordulós verseny során, ahol a diákok kódokkal vannak azonosítva; minden kód egy tetszőleges egész szám. Tervezzünk adatszerkezetet, amiben a következő műveleteket kell tudni elvégezni.

    BESZÚR_DIÁK(x): beilleszt az adatszerkezetbe egy új, x kódú diákot 0 ponttal;

    PONT_NÖVELÉS(x, y): az x kódú diák pontjait y-nal megnöveli;

    HÁNY_PONT(x): megadja, hogy az x kódú diáknak hány pontja van;

    KI_A_LEGJOBB: visszaadja az összes olyan diák kódját, akinek a pontszáma a legmagasabb.

    A KI_A_LEGJOBB művelet lépésszáma $O(k)$ legyen, ahol k a legtöbb ponttal rendelkező diákok számát jelöli, a többi művelet lépésszáma pedig $O(\log n)$ legyen, ahol n a résztvevő diákok száma.