

Adatszerkezetek/2

1. Egy 2-3 fa gyökerének három fia van, a benne szereplő két érték 40 és 50. Mennyi lehet a tárolt elemek minimális, illetve maximális száma, ha tudjuk, hogy csak pozitív egész számokat tárol a fa?
2. Az S_1 és S_2 kulcshalmazokat olyan kiegészített 2-3-fákban tároljuk, melyekben minden csúcsnál fel van jegyezve az onnan induló részfa magassága. Tegyük fel, hogy az S_1 -beli kulcsok mind kisebbek az S_2 -belieknél. Javasoljon hatékony algoritmust a két fa egyesítésére! A cél egyetlen olyan kiegészített 2-3-fa, amelyben a kulcsok $S_1 \cup S_2$ elemei.
3. Egy B_{20} -fának 10^9 levele van. Mekkora a fa szintjeinek minimális, illetve maximális száma?
4. Nyitott címrésszel hash-eltünk egy 11 elemű táblába a $h(k) = k \pmod{11}$ hash-függvény segítségével. A következő kulcsok érkeztek (a megadott sorrendben): 10, 22, 31, 4, 15, 28, 17, 88, 59. Adja meg a tábla végső állapotát a következő két próbamódszerre:
 - (a) lineáris próba;
 - (b) kvadratikus maradék próba!
5. Nyitott címrésszel hashelünk egy kezdetben üres $M = 11$ méretű táblába a $h(x) = x \pmod{M}$ hash-függvénnyel, kettős hasheléssel, ahol a második hash-függvény $h'(x) = 7x \pmod{(M-1)}$. Mi lesz a tábla állapota, ha a 4, 5, 14, 7, 8, 26, 18 kulcsokat szűrjük be ebben a sorrendben?
6. Előfordulhat-e nyitott címrészes hash-elés esetén, hogy az $n > 3$ méretű táblában csak 3 elem van, de a keresés lépésszáma n ?
7. Jó választás-e $M = 7$ méretű táblánál az $f(x) = x^2 \pmod{7}$ hash-függvény?
8. A $T[0 : M]$ táblában $2n$ elemet ($n < M/3$) helyeztünk el valamilyen hash-függvény segítségével, amikor azt tapasztaltuk, hogy az elemek mindegyike az első $3n$ hely egyikére került. Ha törlés közben nem volt és a végén a táblában minden $3i$ indexű hely üres ($0 \leq i < n$), akkor legfeljebb hány ütközés lehetett, ha az ütközések feloldására lineáris próbálást, illetve ha kvadratikus maradék próbálást használtunk?
9. Egy orvosi rendelőben 2 orvos rendel, A és B . Bizonyos betegeket csak az egyik orvos láthat el (minden ilyen Betegre adott, hogy ez az orvos A vagy B), más betegeket mindkettőjük elláthat. Emellett minden beteg érkezésekor kap egy prioritást, mely az eset súlyosságát jelzi. Adjunk olyan adatszerkezetet, amely a következő műveleteket támogatja:
 BEHÍV(X): meghatározza a legkisebb prioritású beteget az X orvos által elláthatóak közül ($X \in \{A, B\}$).
 TÖRÖL(X): törli a legkisebb prioritású beteget az X orvos által elláthatóak közül.
 BESZÚR(b, Y): egy új b beteget szűr be az adatszerkezetbe, Y a hozzárendelt orvos (lehet A , B vagy AB).
 A BEHÍV lépésszáma legyen konstans, a másik két művelet lépésszáma pedig $O(\log k)$, ha k beteg van!
10. Tervezzon olyan adatszerkezetet, ami egy rendezett halmaz elemeinek tárolására szolgál. A megvalósítandó műveletek:
 FELÉPÍT(a_1, \dots, a_n): n elemből felépíti a struktúrát;
 MINTÖR, MAXTÖR: a min. illetve max. elem törlése;
 BESZÚR(x): az x elemet a struktúrába illeszti.
 A FELÉPÍT lépésszáma legyen $O(n)$, a többi műveleté pedig $O(\log n)$, ha n a tárolt elemek száma.
11. Írjon le egy olyan adatszerkezetet, amivel egész számok véges sok részhalmazát tárolhatjuk, ha minden tárolandó T_i halmaznak véges sok eleme van. Három műveletet definiálunk,
 BESZÚR(i, x): a T_i halmazhoz hozzáveszi az x egész számot
 METSZETMÉRET(i, j): megadja a két halmaz metszetének $|T_i \cap T_j|$ elemszámát
 UNIÓMÉRET(i, j): megadja a két halmaz uniójának $|T_i \cup T_j|$ elemszámát.
 A BESZÚR lépésszáma legyen $O(|T_i|)$, a másik két műveleté pedig $O(|T_i| + |T_j|)$.