

Keresőfák III., Hash

- Adott egy $n = 2^k - 1$ pontú teljes bináris keresőfa. A fában tárolt elemek egészek az $I = [1, 2^k]$ intervallumból és egy szám legfeljebb egyszer fordul elő a fában. Utóbbi feltétel szerint pontosan egy olyan $i \in I$ egész van, amely nincs a fában. Adjunk egy hatékony módszert i meghatározására!
 - Adottak a sík egész koordinátájú $P_1 = (x_1, y_1), P_2 = (x_2, y_2), \dots, P_n = (x_n, y_n)$ pontjai. Javasoljunk $O(n \log n)$ költségű módszert annak eldöntésére, hogy vannak-e olyan P_i, P_j pontok ($i \neq j$), melyek távolsága nem több mint 2.
-
- Mi a baja az $f(K) = K^2 \pmod{7}$ hash-függvénynek, ahol 7 a táblaméret?
 - A hash-függvény legyen $f(K) = K$, a táblaméret $M = 7$, és $1 \leq K \leq 20$. Helyezzük el a táblában a 3, 4, 7, 11, 14, 17, 20 kulcsokat ebben a sorrendben
 - lineáris
 - kvadratikus maradékpróbálást használva az ütközések feloldására.
 - Egy M méretű hash-táblába $n < M$ elemet raktunk be nyitott címezéssel, kvadratikus próbával. Ennek során t_1 ütközés történt, azaz ennyiszor kellett tovább próbálkoznunk – egy elem beszúrása során több ütközés is lehetett. Ugyanezt az n elemet ugyanabban a sorrendben beszúrtuk egy M^2 méretű hash-táblába is, de most lineáris próbával, $M \cdot h(x) + 1$ hash-függvénnyel, ekkor t_2 ütközés történt. Igazoljuk, hogy $t_2 \leq t_1$! (ZH 2010. tavasz)
 - A $T[0 : M]$ táblában $2n$ elemet helyeztünk el az első $3n$ helyen ($3n < M$) egy ismeretlen hash-függvény segítségével. A táblában minden $3i$ indexű hely üresen maradt ($0 \leq i < n$). Legfeljebb hány ütközés lehetett, ha az ütközések feloldására
 - lineáris próbálást
 - kvadratikus maradék próbálást használtunk?
 - Nyitott címezéssel hasheltünk egy 11 elemű táblába a $h(k) = k \pmod{11}$ hash-függvény és kvadratikus maradék próba segítségével.

A következő kulcsok érkeztek (a megadott sorrendben): 6, 5, 7, 17, 16, 3, 2, 14. Adjuk meg a tábla végső állapotát!
 - A $b_0 \dots b_n$ alakú $n + 1$ hosszú bitsorozatokat akarjuk tárolni. Tudjuk, hogy a b_0 paritásbit, ami a sorozatban az egyesek számát párosra egészíti ki. Ha nyitott címezésű hash-elést használunk $h(x) \equiv x \pmod{M}$ hash-függvénnyel és lineáris próbával, akkor $M = 2^n$ vagy $M = 2^n + 1$ méretű hash-tábla esetén lesz kevesebb ütközés?
 - A kezdetben üres M méretű hash-táblába sorban beraktuk a $k_1, k_2, \dots, k_n$ kulcsokat a $h(x) \equiv x \pmod{M}$ hash-függvénnyel, lineáris próbával. Jelölje t_1 a keletkezett táblában az egymás melletti foglalt mezők maximális számát. (Ciklikusan értve, azaz t_1 a következő beszúráskori leghosszabb próbasorozat hossza.) Amikor ugyanezt a $k_1, k_2, \dots, k_n$ sorozatot ugyanabban a sorrendben egy üres $2M$ méretű táblába rakjuk be a $h(x) \equiv x \pmod{2M}$ hash-függvénnyel, lineáris próbával, akkor a kapott táblában legyen t_2 az egymás melletti foglalt mezők maximális száma.
 - Igazolja, hogy $t_2 \leq t_1$
 - Igaz-e, hogy $t_1 \leq 2t_2$?
 - Egy 10 méretű hash-táblába kettős hash-elést használva helyezzük el a következő elemeket: 12, 5, 20, 11, 25, 2. A használt hash-függvény legyen $h(x) = 3x \pmod{10}$, a második hash-függvény pedig $h'(x) = (x \bmod 9) + 1$. (PPZH 2010. tavasz) Mi a baj a $h'(x)$ értékeivel?
 - A $T[0 : M - 1]$ táblában rekordokat tárolunk nyitott címezésű hashelt szervezéssel. Az ütközések feloldására lineáris próbálást alkalmazunk. Tehát ha a $h(K)$ sorszámú cella foglalt, akkor a K kulcsú rekordot a $h(K) - 1, h(K) - 2, \dots$ sorszámú cellák közül az első üresbe tesszük. Tegyük fel, hogy a tábla használata során egy hibás törlés történt: egy cellából kitöröltünk egy rekordot a törlés-bit beállítása nélkül.
 - Igaz-e, hogy a hibás törlés helye mindig megtalálható?
 - Adjunk hatékony (lineáris időigényű) algoritmust a tábla megjavítására. (Módosítsuk úgy a táblát, hogy megszűnjének a hibás törlés negatív következményei.)