

Keresés, rendezés I.
Adatstruktúrák és algoritmusok
2. gyakorlat

1. Valaki gondolt egy számra a $0, 1, \dots, 31$ számok közül.
 - (a) Legfeljebb hány kérdéssel lehet kibarkochbázní?
 - (b) Adjuk meg a kérdéseket előre! (Tehát egyszerre kapjuk meg a választ az összes kérdésünkre.)
 - (c) * Hány kérdésre van szükségünk, ha előre kell megadnunk a kérdéseinket és az egyik kérdésünkre nem kapunk választ?
2. Tudjuk, hogy az $a_1, a_2, \dots, a_n$ láncolt lista egy csupa pozitív elemű rendezett láncolt listából úgy keletkezett, hogy annak minden elemét vagy 2-vel vagy (-2) -vel szoroztuk. Adjunk $O(n)$ lépést használó algoritmust, ami az $a_1, \dots, a_n$ láncolt listát rendezi.
3. Pontosán hány összehasonlítás kell ahhoz, hogy n darab szám közül a legkisebb 10 szám egyikét megtaláljuk? (Az eredmény bármelyik lehet a legkisebb tíz közül: tehát pl. az első éppúgy megfelel, mint a tizedik.)
4. Adott az $A[1 : n]$ egész számot növekvő sorrendben tartalmazó tömb. Adjunk $O(\log n)$ lépést használó algoritmust, ami meghatározza, hogy hány olyan elem van, aminek az értéke megegyezik az $A[1]$ értékkel.
5. Adott az $A[1 : n]$ csupa különböző egész számot növekvő sorrendben tartalmazó tömb. Adjunk hatékony algoritmust egy olyan i index meghatározására, melyre $A[i] = i$, feltéve, hogy van ilyen i .
6. Az $A[1 : n]$ tömbben levő elemekről tudjuk, hogy $A[1] \neq A[n]$. Adjunk $O(\log n)$ összehasonlítást használó algoritmust, amely talál egy olyan i indexet, hogy $A[i] \neq A[i + 1]$.
7. Az $A[1 : n]$ tömb különböző egészeket tartalmaz. Az $A[i]$ elem lokális minimum, ha $A[i - 1] > A[i]$ és $A[i] < A[i + 1]$ teljesül ($A[1]$ illetve $A[n]$ elég, ha az egy szomszédjánál kisebb). Adjunk $O(\log n)$ összehasonlítást használó algoritmust egy lokális minimum megkeresésére. (Ha több lokális minimum van, akkor elég egyet megtalálni közülük, mindegy melyiket.)