

Keresés  
ALGORITMUSOK ÉS GRÁFOK  
3. gyakorlat  
2025.

**Lineáris keresés.**

Egy  $n$ -méretű tömbben a lineáris keresés lépésszáma  $O(n)$ .

**Bináris keresés.**

Egy  $n$ -méretű rendezett tömbben a bináris keresés lépésszáma  $O(\log_2 n)$ .

1. A tanult bináris keresést (lásd az alábbi kódot) futtatjuk az 1, 3, 6, 8, 10, 13 tömbön, amiben az

(a)  $s = 8$ ;                      (b)  $s = 2$ ;                      (c)  $s = 0$

elemet keressük. Határozzuk meg, hogy hány összehasonlítás történik, és hogy mi lesz az **eleje** és **vege** változók értéke, amikor a futás véget ér.

**Input:** egy  $n \geq 1$  darab különböző számot tartalmazó, növekvően rendezett  $T$  tömb, valamint egy  $s$  szám

**Output:** ha  $T$  tartalmazza az  $s$  számot, akkor az  $s$  szám  $T$ -beli indexe, ellenkező esetben **nincs**

**eleje** := 0

**vege** :=  $n - 1$

**ciklus:** amíg **eleje** ≤ **vege**

**kozepso** :=  $\left\lfloor \frac{\text{eleje} + \text{vege}}{2} \right\rfloor$

**ha**  $T[\text{kozepso}] == s$ :

**return** **kozepso**

**különb**en **ha**  $T[\text{kozepso}] > s$ :

**vege** := **kozepso** - 1

**különb**en:

**eleje** := **kozepso** + 1

**ciklus vége**

**return** "nincs"

2. Egy  $n$ -elemű tömb első néhány eleme 1-es, a többi pedig 2-es. Adjunk  $O(\log_2 n)$  lépésszámú algoritmust, ami megkeresi a tömbben az első előforduló 2-es indexét, ha van ilyen. (Például az 1, 1, 1, 2, 2 tömb esetén a kimenet legyen 3, viszont az 1, 1, 1 tömb esetén a kimenet legyen **nincs**.)

3. Egy  $n$ -elemű, rendezett tömbben pontosan három különböző érték szerepel. Adjunk  $O(\log_2 n)$  lépésszámú algoritmust ennek a három értéknek a megkeresésére. (Például a 0, 0, 1, 1, 1, 8 tömb esetén a kimenet legyen 0, 1, 8.)
4. Adott egy  $n$  darab különböző egész számot tartalmazó, rendezett  $T$  tömb. Adjunk  $O(\log_2 n)$  lépésszámú algoritmus egy olyan  $i$  index meghatározására, melyre  $T[i] = i$ , feltéve, hogy van ilyen  $i$ .
5. Adjunk  $O(n \log_2 n)$  lépésszámú algoritmust annak eldöntésére, hogy egy  $n$  darab különböző elemet tartalmazó, rendezett  $T$  tömbben van-e két olyan elem, aminek az összege 2025.
6. Egy láncolt listában növekvően rendezetten tároltunk  $n$  darab pozitív számot, de valaki a láncolt lista néhány elemét (nem tudjuk, hogy melyikeket) megszorozta  $(-1)$ -gyel. Adjunk  $O(n)$  lépésszámú algoritmust, ami az így kapott láncolt listát növekvően rendezi.

Keresés  
ALGORITMUSOK ÉS GRÁFOK  
3. gyakorlat  
2025.

**Lineáris keresés.**

Egy  $n$ -méretű tömbben a lineáris keresés lépésszáma  $O(n)$ .

**Bináris keresés.**

Egy  $n$ -méretű rendezett tömbben a bináris keresés lépésszáma  $O(\log_2 n)$ .

1. A tanult bináris keresést (lásd az alábbi kódot) futtatjuk az 1, 3, 6, 8, 10, 13 tömbön, amiben az

(a)  $s = 8$ ;                                      (b)  $s = 2$ ;                                      (c)  $s = 0$

elemet keressük. Határozzuk meg, hogy hány összehasonlítás történik, és hogy mi lesz az **eleje** és **vege** változók értéke, amikor a futás véget ér.

**Input:** egy  $n \geq 1$  darab különböző számot tartalmazó, növekvően rendezett  $T$  tömb, valamint egy  $s$  szám

**Output:** ha  $T$  tartalmazza az  $s$  számot, akkor az  $s$  szám  $T$ -beli indexe, ellenkező esetben **nincs**

**eleje** := 0

**vege** :=  $n - 1$

**ciklus:** amíg **eleje**  $\leq$  **vege**

**kozepso** :=  $\left\lfloor \frac{\text{eleje} + \text{vege}}{2} \right\rfloor$

**ha**  $T[\text{kozepso}] == s$ :

**return** **kozepso**

**különben ha**  $T[\text{kozepso}] > s$ :

**vege** := **kozepso** - 1

**különben:**

**eleje** := **kozepso** + 1

**ciklus vége**

**return** "nincs"

2. Egy  $n$ -elemű tömb első néhány eleme 1-es, a többi pedig 2-es. Adjunk  $O(\log_2 n)$  lépésszámú algoritmust, ami megkeresi a tömbben az első előforduló 2-es indexét, ha van ilyen. (Például az 1, 1, 1, 2, 2 tömb esetén a kimenet legyen 3, viszont az 1, 1, 1 tömb esetén a kimenet legyen **nincs**.)

3. Egy  $n$ -elemű, rendezett tömbben pontosan három különböző érték szerepel. Adjunk  $O(\log_2 n)$  lépésszámú algoritmust ennek a három értéknek a megkeresésére. (Például a 0, 0, 1, 1, 1, 8 tömb esetén a kimenet legyen 0, 1, 8.)
4. Adott egy  $n$  darab különböző egész számot tartalmazó, rendezett  $T$  tömb. Adjunk  $O(\log_2 n)$  lépésszámú algoritmus egy olyan  $i$  index meghatározására, melyre  $T[i] = i$ , feltéve, hogy van ilyen  $i$ .
5. Adjunk  $O(n \log_2 n)$  lépésszámú algoritmust annak eldöntésére, hogy egy  $n$  darab különböző elemet tartalmazó, rendezett  $T$  tömbben van-e két olyan elem, aminek az összege 2025.
6. Egy láncolt listában növekvően rendezetten tároltunk  $n$  darab pozitív számot, de valaki a láncolt lista néhány elemét (nem tudjuk, hogy melyikeket) megszorozta  $(-1)$ -gyel. Adjunk  $O(n)$  lépésszámú algoritmust, ami az így kapott láncolt listát növekvően rendezi.