Kereső algoritmusok összehasonlítása

Kémeri Martin SSR0TI

Lineáris keresés:

A lineáris keresés sorra egymás után megvizsgálja a tömb elemeit, így a végrehajtási idő a tömbelemek számával arányos. Erre utal az algoritmus elnevezése. Ha több elem is rendelkezik az adott tulajdonsággal, akkor a lineáris keresés a legkisebb indexűt határozza meg közülük. Természetesen a keresést a tömb végétől visszafelé is végezhetjük. Így a legnagyobb indexű elemre akadunk rá először.

```
1 usage
def linearisKereses(v, n, T):
    i = 0

    while i < n and v[i] != T:
        i += 1

    if i < n:

        return True

    else:

        return False
```

Strázsás keresés:

A strázsás keresés egy kicsit gyorsabb, mert tömbelemenként csak egy vizsgálatot kell végezni – arra nem kell figyelni, hogy elértük-e a ciklussal a tömb végét. Mivel a végére betettük a keresett elemet, ezért a ciklus feltétele előbb-utóbb hamis lesz, legkésőbb a strázsánál.

Csak akkor használható, ha

- tudjuk, hogy van még hely a tömbben (legalább eggyel nagyobb)

- szabad módosítani a tömböt.

```
1 usage
def strazsasKereses(v, T):
    i = 0
    v.append(T)

    while v[i] != T:
        i += 1

    if i < len(v) - 1:

        return True

    else:

        return False
```

Bináris keresés:

A bináris keresés egy erősen optimalizált keresési eljárás, amely csak rendezett adatsoron alkalmazható. Az algoritmus alapelve, hogy a rendezett tömb adott elemével összehasonlítva a keresett elemet, a keresés a megfelelő intervallumban (az adott elemnél kisebb vagy nagyobb elemek halmazában) folytatható: a keresési intervallum ilyen finomításával így végül megtaláljuk a keresett elemet, ha benne van a rendezett tömbünkben. Az algoritmus futási ideje logaritmikus.

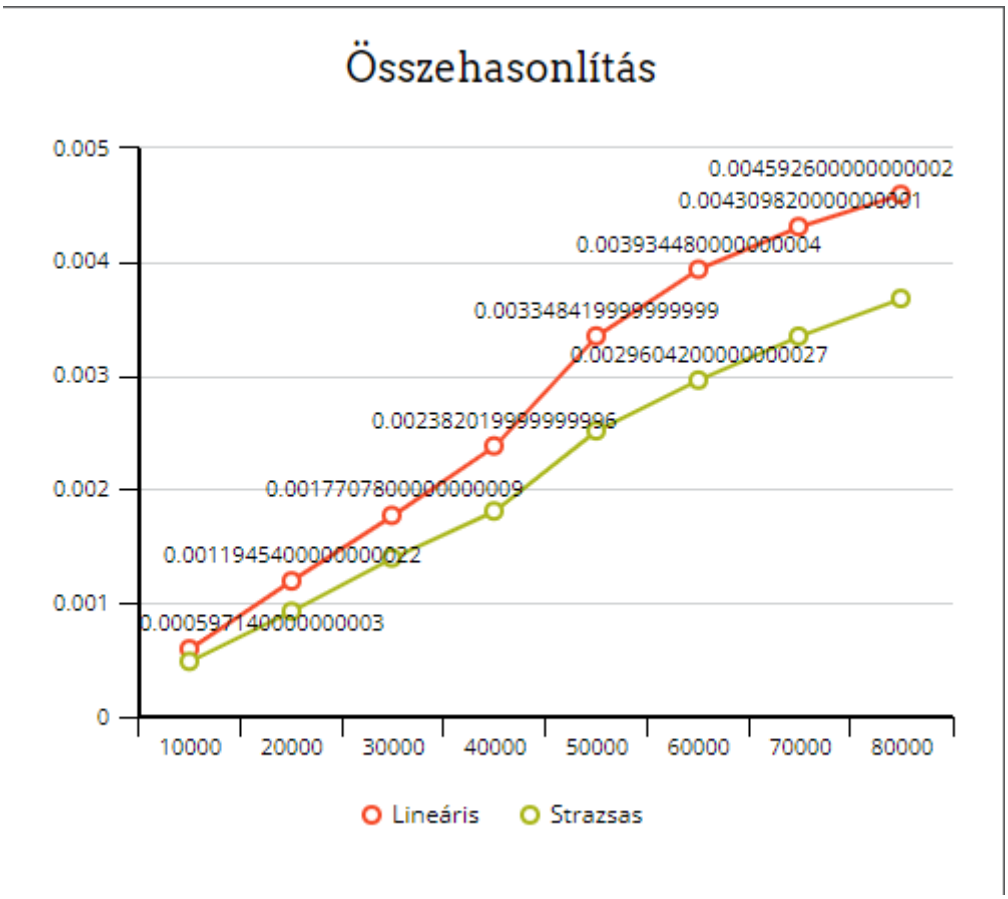
```
1 usage
def binarisKereses(v, T):
    e = 0
    u = len(v) - 1
    k = 0
    while e <= u:
        k = (u + e) // 2
        if v[k] < T:
            e = k + 1
        elif v[k] > T:
            u = k - 1
        else:
            return k
    return False
```

Ugró keresés:

A bináris kereséshez hasonlóan ennek az algoritmusnak is az a célja, hogy minél kevesebb lépésből megtalálja a keresett T tulajdonsású elemet. Az algoritmus nem egyesével, hanem egy előre megadott ugrás értékkel vizsgálja meg a sorozat elemeit. Amennyiben túlugrott a keresett T tulajdonságú elem értékén, akkor az utolsó ugrás intervallumán belül lineáris keresést valósít meg.

```
1 usage
def ugraloKereses(v, n, T ):

    lepesKoz = math.sqrt(n)

    elozo = 0

    while v[int(min(lepesKoz, n)-1)] < T:

        elozo = lepesKoz

        lepesKoz += math.sqrt(n)

        if elozo >= n:

            return -1

    while v[int(elozo)] < T:

        elozo += 1

        if elozo == min(lepesKoz, n):

            return -1

    if v[int(elozo)] == T:

        return int(elozo)

    return False
```

Összehasonlítás rendezetlen sorozaton:



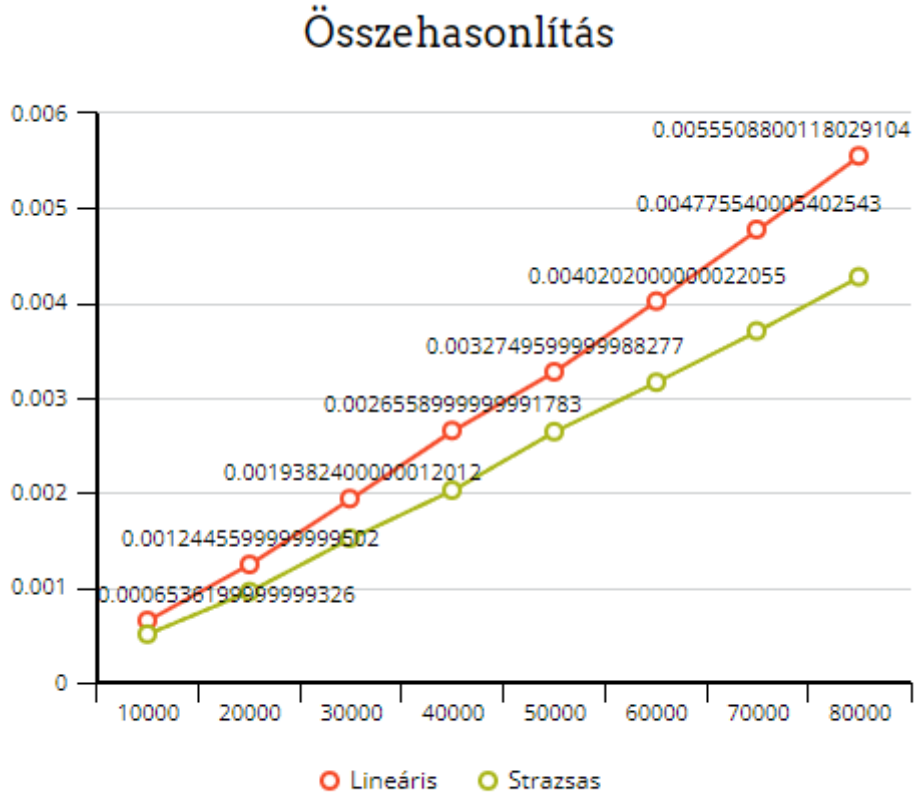
Lineáris keresés:

Elem	Átlag idő
10000	0.000597140000000003
20000	0.00119454000000022
30000	0.00177078000000009
40000	0.00238201999999996
50000	0.00334841999999999
60000	0.00393448000000004
70000	0.00430982000000001
80000	0.00459260000000002

Strázsás keresés:

Elem	Átlag idő
10000	0.0004870199999999914
20000	0.000927100000000001
30000	0.0013962799999999984
40000	0.0018090800000000046
50000	0.0025148599999999994
60000	0.00296042000000027
70000	0.0033463400000000006
80000	0.0036782399999999993

Összehasonlítás rendezett sorozaton:



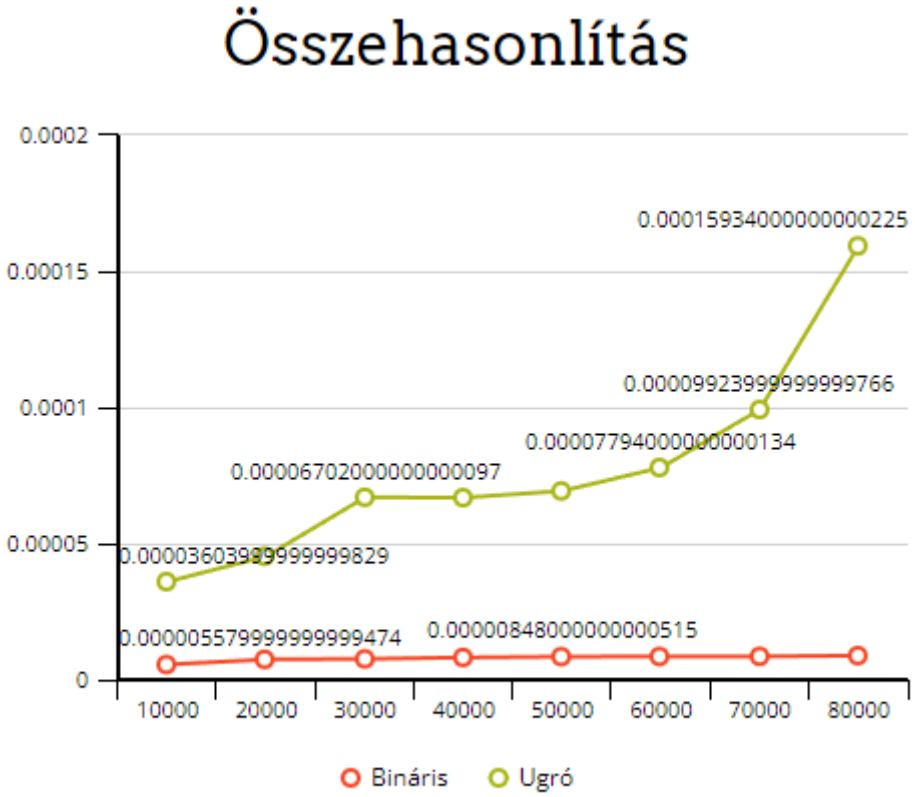
Lineáris keresés:

Elem	Átlag idő
10000	0.0006536199999999326
20000	0.0012445599999999502
30000	0.0019382400000012012
40000	0.0026558999999991783
50000	0.0032749599999988277
60000	0.0040202000000022055
70000	0.004775540005402543
80000	0.0055508800118029104

Strázsás keresés:

Elem	Átlag idő
10000	0.0005102600000002511
20000	0.0009579400000006899
30000	0.001520799999998701
40000	0.0020273399999979347
50000	0.0026441800000014835
60000	0.0031648399999937736
70000	0.003705499997886064
80000	0.004276155997878554

Bináris és Ugró keresés összehasonlítása:



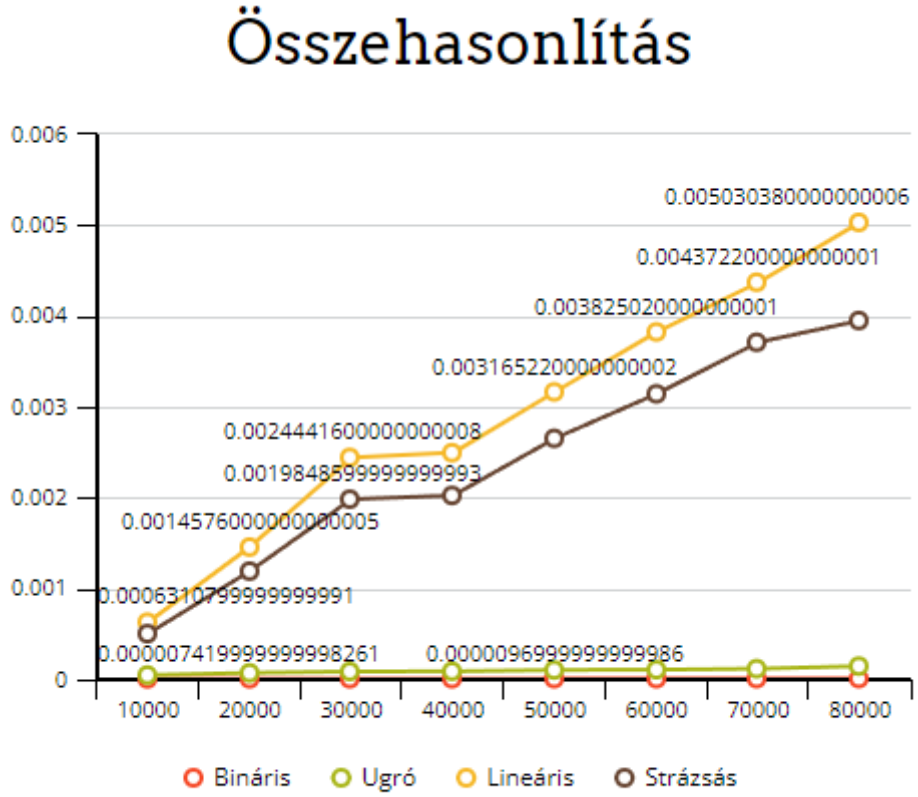
Bináris:

Elem	Átlag idő
10000	5.579999999999474e-06
20000	7.479999999997211e-06
30000	7.659999999996837e-06
40000	8.179999999996523e-06
50000	8.48000000000515e-06
60000	8.5400000000041e-06
70000	8.5799999999997e-06
80000	8.859999999996649e-06

Ugró:

Elem	Átlag idő
10000	3.603999999999829e-05
20000	4.534000000000204e-05
30000	6.702000000000097e-05
40000	6.687999999999971e-05
50000	6.934000000000107e-05
60000	7.794000000000134e-05
70000	9.923999999999766e-05
80000	0.0001593400000000225

Lineáris, Strázsás, Bináris és Ugró keresés összehasonlítása:



Bináris:

Elem	Átlag idő
10000	7.419999999998261e-06
20000	8.400000000001461e-06
30000	8.880000000001387e-06
40000	9.059999999999624e-06
50000	9.6999999999986e-06
60000	9.899999999998799e-06
70000	1.0280000000001399e-05
80000	1.1000000000002675e-05

Ugró:

Elem	Átlag idő
10000	4.906000000000077e-05
20000	7.339999999999986e-05
30000	8.827999999999892e-05
40000	9.074000000000026e-05
50000	0.00010507999999999907
60000	0.00010836000000000457
70000	0.00011952000000000074
80000	0.00014768000000000281

Lineáris:

Elem	Átlag idő
10000	0.0006310799999999991
20000	0.0014576000000000005
30000	0.0024441600000000008
40000	0.0024965000000000013
50000	0.003165220000000002
60000	0.003825020000000001
70000	0.004372200000000001
80000	0.005030380000000006

Strázsás:

Elem	Átlag idő
10000	0.0005052999999999974
20000	0.001192899999999998
30000	0.0019848599999999993
40000	0.0020257200000000004
50000	0.0026543200000000043
60000	0.0031441999999999998
70000	0.0037114999999999982
80000	0.003950840000000002