

Mätpunkter

1000, 2000, 4000, 8000, 10000, 20000, 40000, 60000, 70000, 80000, 100000
För att få en bra kurva

Gör ett program som räknar medel själv så går det undan

Kanske till och med itererar genom alla mätdata.

Bubblesort

Vår bubbelsort är implementerad så att den jämför om den under senaste varvet har behövt byta något element. Har den inte behövt göra det avbryts sorteringen.

Dubbla loopar.

Antal körningar på varje: 10 st

Worst case

När listan är sorterad i omvänd ordning.

$O(n^2)$

Tider: 1000 st
0,00686097
0,00679898
0,00676894
0,00676012
0,00676703
0,00676012
0,00676799
0,00679111
0,00679994
0,00683808

10000 st
0,684377
0,685198
0,685202
0,687142
0,684120
0,686088
0,682605
0,684475
0,685051
0,682982

20000 st

2,7306
2,73148
2,73528
2,73044
2,73171
2,73051
2,73559
2,73036
2,73141
2,73126

50000 st
17,2220
17,2309
17,2267
17,2257
17,2203
17,2216
17,2206
17,2234
17,2935
17,2219

100000 st
69,1949
69,1884
69,1732
69,1823
69,1692
69,1786
69,1687
69,1627
69,1653
69,1726

Avrage Case

Sumpade värden i en lång lista, inte unika värden. Dessa verkar dock ta längre tid? Men der verkar okej. Det kan vara en konstant skillnad vilket är okej. Dock borde det vara för något?
 $O(n^2)$

Tider: 1000 st
0,00681281
0,00674820
0,00679088
0,00681591
0,00678110
0,00671506
0,00660586
0,00659108

0,00666404
0,00662994

10000 st
0,702117
0,703693
0,707952
0,703003
0,702871
0,703883
0,705235
0,702676
0,701026
0,701739

20000 st
2,81161
2,81607
2,80367
2,82638
2,81508
2,82284
2,80390
2,81501
2,81719
2,82419

50000 st
17,6489
17,6064
17,6653
17,6448
17,6738
17,6237
17,6821
17,6623
17,6556
17,6681

100000 st
70,9432
71,0154
71,4977
71,1066
71,0427
71,1776
71,075
71,0232
71,1147
71,0699

Best case

Redan sorterad borde vara det iaf $O(n)$

Tider: 1000 st

1,1*10⁻⁵
9,06*10⁻⁶
1,0*10⁻⁵
9,06*10⁻⁶
9,06*10⁻⁶
1,0*10⁻⁵
1,0*10⁻⁵
9,06*10⁻⁶
9,06*10⁻⁶
9,06*10⁻⁶

10000 st

4,79*10⁻⁵
4,79*10⁻⁵
4,81*10⁻⁵
4,79*10⁻⁵
4,69*10⁻⁵
4,69*10⁻⁵
4,79*10⁻⁵
4,79*10⁻⁵
4,79*10⁻⁵
4,79*10⁻⁵

20000 st

9,60827*10⁻⁵
8,79765*10⁻⁵
9,0837*10⁻⁵
8,89301*10⁻⁵
8,89301*10⁻⁵
9,01222*10⁻⁵
8,89301*10⁻⁵
8,91685*10⁻⁵
9,01222*10⁻⁵
8,91285*10⁻⁵

50000 st

0,0002170
0,0002160
0,0002139
0,0002182
0,0002151
0,0002141

0,0002148
0,0001160
0,0002151
0,0002170

100000 st
0,0005490
0,0004261
0,0004559
0,0004301
0,0004570
0,0004389
0,0004580
0,0004280
0,0004311
0,0004311

1000000 st
0,004812
0,004250
0,004164
0,004151
0,004246
0,004236
0,004144
0,004179
0,004254
0,004264

Insertionsort

Vi arbetar endast med originalarrayen och flyttar inte till någon annan array.

Dubbla loopar.

Antal körningar på varje: 10 st

Worst case

När listan är sorterad i omvänd ordning. $O(n^2)$

Se fil

Quick Sort

Vi väljer första element som pivot.

Worst Case

Sorterad