

quicksort.py

```
1 import time
2
3 def isSorted(a):
4     b = a[0]
5     for x in a:
6         if b > x:
7             return False
8         b = x
9     return True
10
11 def qsort(a, l, r):
12     if l >= r:
13         return
14
15 #pivot = a[l + int((r-l)/2)]
16 pivot = a[r]
17
18 i, j = l, r #リストの1つ目の値をi, 後ろから一つ目をjとする
19
20 while i < j: #i(最初は0が入る)がjより小さいと入る
21     while a[i] < pivot: #a[i]がより小さかったらi+1
22         i += 1
23     while a[j] > pivot: #j番目の値がpivotより小さければpivotより、小さくなるまで探す。
24         j -= 1
25     i < jの場合:
26         a[i] == a[j]の場合:
27             j -= 1
28     それ以外の場合:
29         a[i], a[j] = a[j], a[i] #iとjが違う値だった場合、iとjの値を変える。(マイナスの値がない場合は、i=0だから最初に来る。)
30     qsort(a, l, i- 1 ) #i-1番目の値がjになり、もう一度読まれる。
31     qsort(a, j+ 1 , r) #j+1番目の値がiになり、繰り返し読まれます。
32
33
34 a = [ 7 , 2 , 1 , 4 , 6 , 0 , 8 , 5 , 9 , 3 ]
35 長さ =長さ(a)
36 印刷(a)
37 start = time.time()
38 qsort(a, 0, lenght- 1 )
39 end = time.time()
40 print(end-start)
41 print(isSorted(a))
42 print(a)
43
44 #実行結果
45 # [7, 2, 1, 4, 6, 0, 8, 5, 9, 3]
46 # 2.5033950805664062e-05
47 # True
48 # [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
49
```