quicksort

Execute | Share main.py STDIN

```python
import time
def partition(arr,low,high):
    i = ( low-1 )         # index of smaller element
    pivot = arr[high]     # pivot

    for j in range(low , high):
        if arr[j] <= pivot:
            i = i+1
            arr[i],arr[j] = arr[j],arr[i]

    arr[i+1],arr[high] = arr[high],arr[i+1]
    return ( i+1 )
start = time.time()
print ("time:",(time.time()-start)*1000)

def quickSort(arr,low,high):
    if low < high:

        pi = partition(arr,low,high)

        quickSort(arr, low, pi-1)
        quickSort(arr, pi+1, high)

arr = [10000,15409,14569,25374,10867,14356]
n = len(arr)
quickSort(arr,0,n-1)

print ("Sorted array is:")
for i in range(n):
    print ("%d" %arr[i]),

```

Result

$python main.py

('time:', 0.00095367431640625)
Sorted array is:
10000 10867 14356 14569 15409 25374

Shell Sort

```python

import time
def shellSort(arr):
    n = len(arr)
    gap = n/2

    while gap > 0:

        for i in range(gap,n):
            temp = arr[i]
            j = i
            while j >= gap and arr[j-gap] >temp:
                arr[j] = arr[j-gap]
                j -= gap
            arr[j] = temp
        gap /= 2


arr = [10834,23123,54578,12346,78763,35312]
start = time.time()
print ("time:",(time.time()-start)*1000)
n = len(arr)
print ("Array before sorting:")
for i in range(n):
    print(arr[i]),

shellSort(arr)

print ("\nArray after sorting:")
for i in range(n):
    print(arr[i]),

```

$python main.py

('time:', 0.00095367431640625)
Array before sorting:
10834 23123 54578 12346 78763 35312
Array after sorting:
10834 12346 23123 35312 54578 78763