实验四习题解答

#四种排序算法时间性能比较

​

​

#四种排序算法

#冒泡排序

def bubble\_sort(nums):

lens = len(nums)

for i in range(lens-1): #i表示冒泡次数

for j in range(lens-1-i): #j表示当前元素

if nums[j] > nums[j+1]:

nums[j], nums[j+1] = nums[j+1], nums[j]

return nums

​

def choose\_sort(nums):

lens = len(nums)

for i in range(lens-1):

min = i

for j in range(i+1, lens):

if nums[j] < nums[i]:

min = j

nums[i], nums[min] = nums[min], nums[i]

return nums

​

def insert\_sort(nums):

lens = len(nums)

for i in range(1, lens):

for j in range(i-1, -1, -1):

if nums[j] > nums[j+1]:

nums[j], nums[j+1] = nums[j+1], nums[j]

else:

break

return nums

​

def merge\_sort(nums):

lens = len(nums)

count = lens

n = 2

def sort\_two\_list(list1,list2):

len1 = len(list1)

len2 = len(list2)

list3 = []

i,j=0,0

while i<len1 and j<len2:

if list1[i] < list2[j]:

list3.append(list1[i])

i += 1

else:

list3.append(list2[j])

j += 1

list3 += list1[i:]

list3 += list2[j:]

return list3

for i in range(0,lens-1,2):

if nums[i] > nums[i+1]:

nums[i],nums[i+1] = nums[i+1],nums[i]

while n<=lens:

list3 = []

for j in range(0,lens,2\*n):

list3 += sort\_two\_list(nums[j:j+n],nums[j+n:j+2\*n])

nums = list3

n += n

return nums

​

#产生大小为n的随机数组

def create\_random(n):

import random

random\_list = []

for i in range(n):

random\_list.append(random.randint(0, n))

return random\_list

​

​

#手动设置断点分析函数执行时间

def \_time\_analyze\_(func, nums):

from time import perf\_counter

start = perf\_counter()

func(nums)

finish = perf\_counter()

print(func.\_\_name\_\_, ":", finish-start, "s")

​

def simple\_profile(n): #n为用于测试的数组大小

print("\*" \* 40, "\nSimple time analze")

nums\_ = create\_random(n)

for fun in [bubble\_sort, choose\_sort, insert\_sort, merge\_sort]:

nums = nums\_

\_time\_analyze\_(fun, nums)

​

#用python的timeit模块分析函数执行时间

def \_timeit\_analyze\_(func, nums):

from timeit import Timer

t1 = Timer("%s(%s)" % (func.\_\_name\_\_,nums), "from \_\_main\_\_ import %s" % func.\_\_name\_\_)

​

print(func.\_\_name\_\_, ":", t1.timeit(1))

​

def timeit\_profile(n):

print("\*" \* 40, "\nModule timeit analyze")

nums\_ = create\_random(n)

for fun in [bubble\_sort, choose\_sort, insert\_sort, merge\_sort]:

nums = nums\_

\_timeit\_analyze\_(fun, nums)

​

if \_\_name\_\_ == "\_\_main\_\_":

len\_ = [100, 1000, 10000]

for i in len\_:

print('When the length of list is', i, ':\n')

simple\_profile(i)

timeit\_profile(i)

​