Python学习笔记03 归并排序

核心思想

归并排序的核心思想有两点：
1） 是把给定的数列不停地展开为两个子数列
2） 再把子数列层层**有序**地合并回原有的数列长度

代码实现

代码主要有两个子过程，
1 是mergesort （arr）负责将数列arr不停地二分为left和right两个子数组然后分别对左右两个数组再次分别调用mergesort过程进行迭代 当数列长度小于2时停止（掉入上图堆栈中的最内/下层） 之后对左右两边数组调用merge（left，right）子过程，这一过程确保left和right两边被有序合并
2.merge（left，right）中新开一个数组队列 用i和j两个指针不停控制左右两边较小的元素入列 直至左右两边全部入队

具体代码如下：

```
import math


def mergesort(arr):

    if len(arr)<2:

         return arr

    mid=len(arr)//2

    lft=arr[:mid]

    rgt=arr[mid:]

    lft=mergesort(lft)

    rgt=mergesort(rgt)

    return merge(lft,rgt)



def merge(a_l,a_r):

     merged=[]
```

```
    i=0
    j=0
    while (i<len(a_l)) and (j<len(a_r)):
        if a_l[i]<a_r[j]:
            merged.append(a_l[i])
            i+=1
        else:
            merged.append(a_r[j])
            j+=1
    while i<len(a_l):
       merged.append(a_l[i])
       i+=1
    while j<len(a_r):
       merged.append(a_r[j])
       j+=1
    return merged


import random

n=int(input())
arr=[]
scale=int(input())
for k in range(n):
    arr.append(random.randint(0, scale))
print(arr)
print(mergesort(arr))
```

eg. 归并排序的复杂度计算

```
$Cn=C_{\lceil n/2 \rceil}+C_{\lfloor n/2 \rfloor}+N \text{ For N>1 with }C_1=1$
```

每一次排序都等于左边和右边的复杂度 加上合并左边右边的Merge操作

我们先考虑2的指数幂 设$N = 2^n$

$$a_n = 2a_{n-1} + 2^n \text{ with } a_0 = 0$$

两边同时除以 2^n

$$\frac{a_n}{2^n} = \frac{a_{n-1}}{2^{n-1}} + 1$$

展开后求和

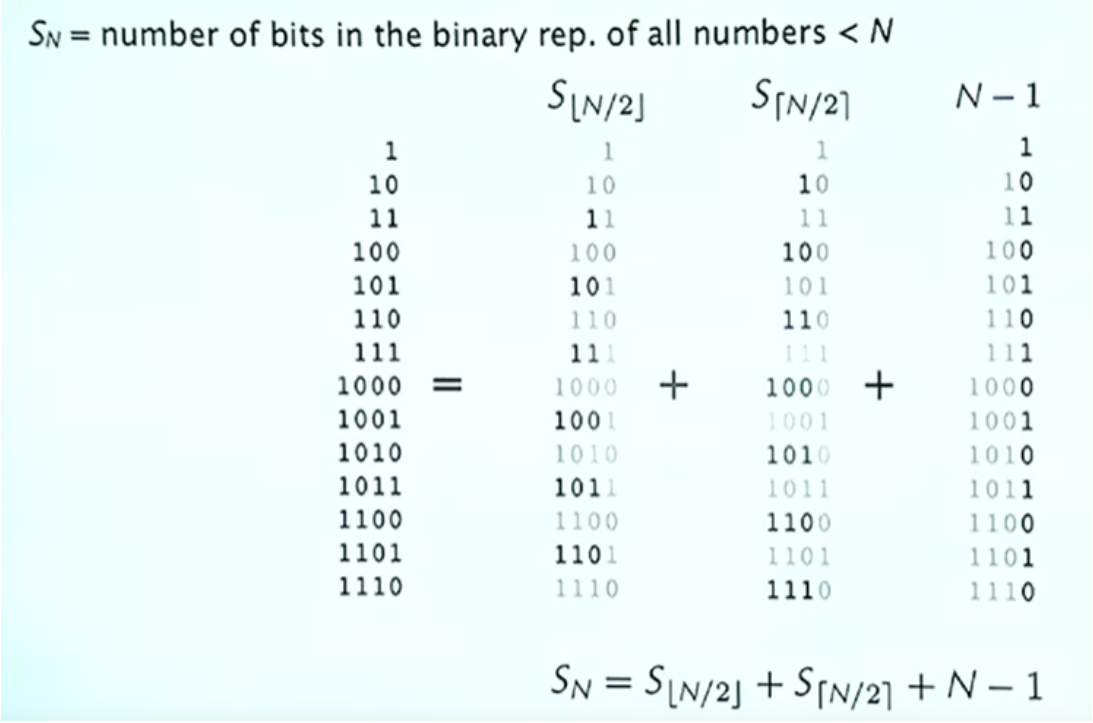
$$\frac{a_n}{2^n} = n$$

得到：

$$a_n = n2^n$$

那么$C_N = a_{lgN} = NlgN$

下面讨论更一般的情况
$CN = C_{\lceil N/2 \rceil} + C_{\lfloor N/2 \rfloor} + N \text{ For N>1 with } C_1 = 1$
$C_{N+1} = C_{\lfloor (N+1)/2 \rfloor} + C_{\lceil (N+1)/2 \rceil} + N + 1$
$= C_{\lceil N/2 \rceil} + C_{\lfloor N/2 \rfloor + 1} + N + 1$
$C_{n+1} - C_n = C_{\lfloor N/2 \rfloor + 1} - C_{\lfloor N/2 \rfloor} + 1$
定义$D_N = C_{N+1} - C_N$
得到：$D_N = D_{\lceil N/2 \rceil} + 1$
从之前的证明中我们知道：$D_N = \lfloor lgN \rfloor + 2$ (首项有差异)
于是$Cn = N - 1 + \sum_{1<=k<n} (\lfloor lgk \rfloor + 1)$
注意：上式后面这部分是小于N的所有数的二进制位数的合，若我们令其为Sn 则：
我们发现：



要求小于N的所有数字位数，不妨把从0到N-1的数字二进制都写在一个：$N \times (\lfloor lgN \rfloor + 1)$ 的表中，Sn等于这个表的长乘以宽再减去位于存在于前面所有的零（一个等比数列的求和）：

```
![[Pasted image 20240401192326.png]]
代回后得到：
$$C_N=N-1+S_N$$
$$=N-1+N\lfloor lgN\rfloor-2^{\lfloor lgN\rfloor+1}+N+1=N\lfloor lgN\rfloor-2^{\lfloor lgN\rfloor+1}+2N$$
```