

贴瓷砖

抛硬币

抛 400 次，有 220 次是正面的概率

区间合并

[[2,4],[1,3],[6,9],[8,13],[21,33]]

斐波那契数列

总共 100 级台阶，每次可选择走 1 步、2 步，问走完这 100 级台阶有多少种走法

荷兰国旗

[2,0,1,2,1,0,1,2,2,1,0,2]

[0,0,0,1,1,1,2,2,2,2,2,2]

要求

时间复杂度 $O(n)$

空间复杂度 $O(1)$

单词逆序调整

abcde

$K=3$ 将前 k 个元素放到后面

cbaed
最终: deabc

K=2
baedc
最终: cdeab
时间复杂度 $O(n)$
空间复杂度 $O(1)$

对一个几乎有序的数组排序

可以用插入: 时间复杂度 $N*K$
改进的小根堆: $N*\log K$

有内存限制的输出 top100

需要排序的最短子数组长度

1 5 4 3 2 6 9
5 4 3 2 需要被排序 输出长度=4
时间复杂度 $O(N)$, 空间复杂度 $O(1)$

直接插入排序

```
lst = [6,3,5,7,0,4,1,2]
for i in range(1,len(lst)):
    key = lst[i]
    for j in range(i::-1):
        if lst[j]>key:
            lst[j+1]=lst[j]
            lst[j]=key
```

冒泡排序

```
lst = [6,3,5,7,0,4,1,2]
for i in range(len(lst)):
    for j in range(len(lst)-i-1):
        if lst[j+1]<lst[j]:
            lst[j+1],lst[j] = lst[j],lst[j+1]
print(lst)
```

选择排序

```
lst = [6,3,5,7,0,4,1,2]
for i in range(len(lst)):
    min_index = i
    for j in range(i,len(lst)):
        if lst[j]<lst[min_index]:
            min_index = j
    lst[i],lst[min_index] = lst[min_index],lst[i]

lst
```