

算法

枚举

又叫穷举，将问题的可能解一个个有序的列出来

举例子：？ $3 \times 6528 = 3? \times 8256$ ？

```
for i in range(10):
    if (i*10+3)*6528 == (30+i)*8256:
        print(i)
```

练习1：破解密码

请用python帮我找回密码，零星记得支付密码信息：

6位，前两位是85

最后两位数字相同

能被13和33整除

练习2：

有一盒乒乓球，9个9个拿，余7个，5个5个拿，余2个，4个4个拿，余1个。问有多少乒乓球

练习3：

把 1296 分拆成a、b、c、d四个正整数，如果a加上2，b减去2，c乘以2，d除以2，则这四个结果相等。现在请你编写程序求出这四个数。

补全下面的代码：

```
for a in range(1, ① ):
    b = ②
    for c in range(1,1296-a-b):
        d = ③
        if (b-2c*2) and (a+b+c+d ④ ):
            print(a,b,c,d)
```

排序

想想自己如何去排序

考虑所消耗的时间、空间

邻居好说话----冒泡排序

基本思想：每次比较两个相邻的元素，如果他们的顺序错误，就把他们交换过来。

例如: 12 35 99 18 76 这五个数**从大到小**

第一轮比较开始

先比较 第一位，第二位 12 35 交换

结果：35 12 99 18 76

继续比较第二位，第三位， 12 99 交换

结果: 35 99 12 18 76

继续比较第三位，第四位， 12 18 交换

结果：35 99 18 12 76

继续比较第四位，第五位， 12 76 交换

结果： 35 99 18 76 **12**

第一轮比较结束，成功的把最小的元素放在最后一位。

开始第二轮比较

第一轮已经把最小的放在了最后，第二轮只需要比较前4个就行

先比较 第一位，第二位 35 99交换

结果：99 35 18 76 12

继续比较第二位，第三位， 35 18 不交换

结果：99 35 18 76 12

继续比较第三位，第四位， 18 76 交换

结果： 99 35 76 **18 12**

第二轮比较结束，成功的把第二小的元素放在最后一位。

开始第三轮比较

。 。 。

思考：

1. 像不像是在冒泡泡，把最小的放在最后

2. n个数排序，要运算几轮？

n-1轮，每一轮都需要从第 1 位开始进行相邻两个数的比较，将较小的一个数放在后面，比较完毕后向后挪一位继续比较下面两个相邻数的大小，重复此步骤，直到最后一个尚未归位的数（这个数无需比较，因为也没有其他数和他比较了）

3. 试试算一算一共比较几次

像不像平时照相排队的时候被别人换来换去

code

```
a = [12, 35, 99, 18, 76]
count = len(a)
for i in range(0, count - 1):
    for j in range(0, count - 1 - i):
        if a[j] > a[j+1]:
            a[j], a[j+1] = a[j+1], a[j]
print(a)
```

题目：

80, 125, 64, 68, 46, 升序，第二遍的结果是多少？

23, 25, 18, 63, 84, 77, 65, 9, 33 升序，冒泡，需要交换几次？