**2024**转计算机学院机考试题（ 回忆版） **——Python**

2024年转专业机考，我是生死未卜。今年与以往两年有所不同，题目类型变了，是蓝桥杯里的题目了 qaq，我想的太多太乱，做的又不够，让正确的思路也没有很好的贯彻。



暑期增加：第五题，等差素数数列原先答案有问题，已经做出更改，感谢24级的刘家蔚同学，指正了我 原先的代码错误，并给出了他的使用递归方法而写出的题解。也非常感谢创建项目的哥，以及诸多计算 机转专业群内的大佬们，热心答疑，为我指明方向，虽然没有成功，但是我也收获颇多，非常感动，谢 谢各位！

**1.** 回文数字



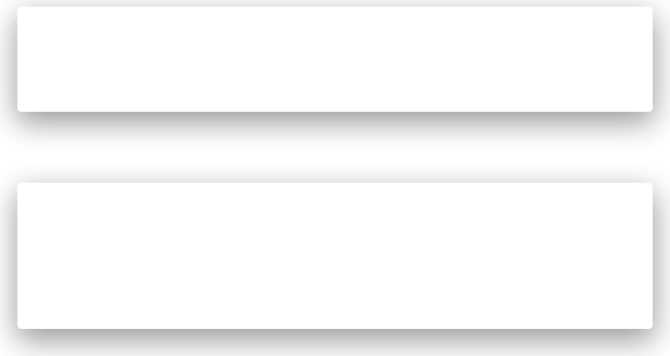
求10~1000内满足条件的回文整数。

要求如下

（1）这个整数本身是回文数，假设该整数为i，则i的平方，i的立方也均为回文数。 （2）逐行输出这些符合条件的数字，并把其对应的平方，立方在同一行输出。

输出结果展示：



11 121 1331

101 10201 1030301

111 12321 1367631

参考题解



def is\_palindrome(num\_str):

return num\_str == num\_str[::-1] for i in range(10, 1000):

if all(is\_palindrome(str(x)) for x in (i, i\*\*2, i\*\*3)): print(f"{i}, {i\*\*2}, {i\*\*3}")

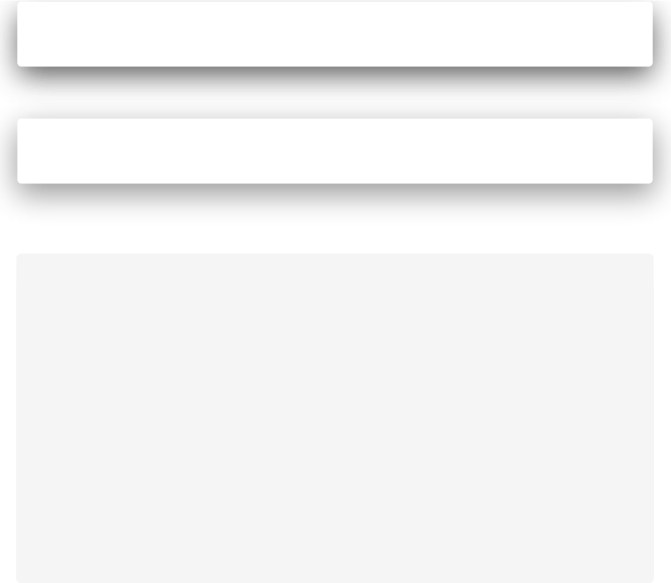
**2.**购买方案**(**组合**)**



买礼物，输入礼物的价格(升序空格隔开)，挑选其中三件，输出有多少种购买方式(组合) 例如:

输入



1 2 3 4 5

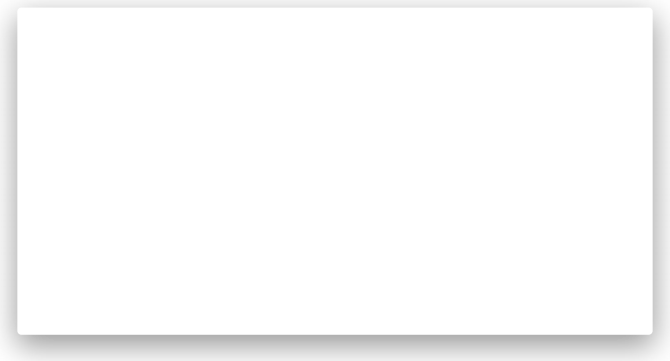
输出



10

参考题解



def combinations\_count(n): #计算组合数C(n,3)

if n < 3:

return 0 if n == 3:

return 1 result = 1

for i in range(1, 4):

result \*= (n - i + 1) result //= i

return result

gift\_prices = list(map(int,input().split())) num = len(gift\_prices)

print(combinations\_count(num))

**3.** 冒泡排序找字符



小蓝最近学习了一些排序算法，其中冒泡排序让他印象深刻。在冒泡排序中，每次只能交换相邻的两个 元素。小蓝发现，如果对一个字符串中的字符排序，只允许交换相邻的两个字符，则在所有可能的排序 方案中，冒泡排序的总交换次数是最少的。

例如，对于字符串 lan 排序，只需要 1次交换。对于字符串qiao 排序，总共需要 4 次交换。小蓝的幸运 数字是 V，他想找到一个只包含小写英文字母的字符串，对这个串中的字符进行冒泡排序，正好需要 V 次交换。

请帮助小蓝找一个这样的字符串。如果可能找到多个，请告诉小蓝最短的那个。如果最短的仍然有多 个，请告诉小蓝字典序最小的那个。请注意字符串中可以包含相同的字符。

输入格式

输入一行包含一个整数"V" ，为小蓝的幸运数字。 输出格式

输出一个字符串，为所求的答案。

样例输入



4

样例输出



bbaa

样例输入



100

样例输出



Jihgfeeddccbbaa

题解



def length(v): i = 1

while i \* (i - 1) // 2 < v: i += 1

return i

def create\_string(length):

goal\_string = "" alpha = 'a '

for \_ in range(length): goal\_string += alpha

alpha = chr((ord(alpha) - ord('a ') + 1) % 26 + ord('a ')) return goal\_string

def add(add\_number, demo\_string, length): list\_demo\_string = list(demo\_string)

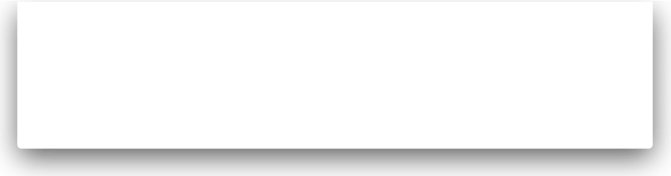
alpha = 'a ' index = 0

while add\_number > 0:

# 由于index从0开始 ，所以插入位置应该对应demo\_string的索引+1 list\_demo\_string.insert(index + 1, alpha)

index += 2

alpha = chr((ord(alpha) - ord('a ') + 1) % 26 + ord('a ')) add\_number -= 1



return ' '.join(sorted(list\_demo\_string,reverse=True)) V = int(input())

len\_needed = length(V)

add\_chars = len\_needed \* (len\_needed - 1) // 2 - V

demo\_string = create\_string(len\_needed - add\_chars)

final\_string = add(add\_chars, demo\_string, len\_needed) print(final\_string)

鼠鼠真的是太菜了！当我看到这道题目时候，知道冒泡排序,但就是没有任何思路了5555....考完试重新 做，加上参考大佬的做法，发现好像...主要还是自己题目理解有问题。没搞清楚就一头雾水的扎进去 了，当然...这也与备考题目类型差异过大有关（不是对自己菜的辩白），我想，机考作为一次应试考 试，准备上来说... 的确不是可以只看某一方面的内容的，盲求抱佛脚。

这道题，感觉题目的确写的比较抽象，字典序最小，那里是关键所在。比起正常的来说困难的地方我觉 得在补这个过程中，容易出问题。

**4.**分巧克力



儿童节那天有K位小朋友到小明家做客。小明拿出了珍藏的巧克力招待小朋友们。小明一共有N块巧克 力，其中第i块是Hi x Wi的方格组成的长方形。为了公平起见，小明需要从这 N 块巧克力中切出K块巧克 力分给小朋友们。切出的巧克力需要满足：

（1）形状是正方形，边长是整数（2）大小相同

例如，一块6x5的巧克力可以切出6块2x2的巧克力，或者2块3x3的巧克力。

当然小朋友们都希望得到的巧克力尽可能大，你能帮小Hi计算出，最大的边长是多少么？ 输入

第一行包含两个整数N和K 。(1 <= N, K <= 100000)

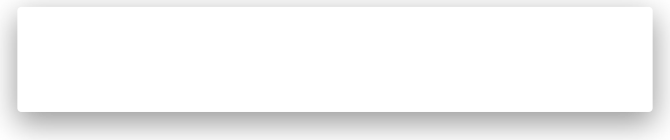
以下N行每行包含两个整数Hi和Wi 。(1 <= Hi, Wi <= 100000)

输入保证每位小朋友至少能获得一块1x1的巧克力。 输出

输出切出的正方形巧克力最大可能的边长。 运行示例

输入



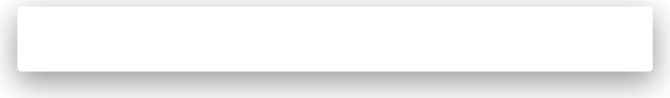
2 10

6 5

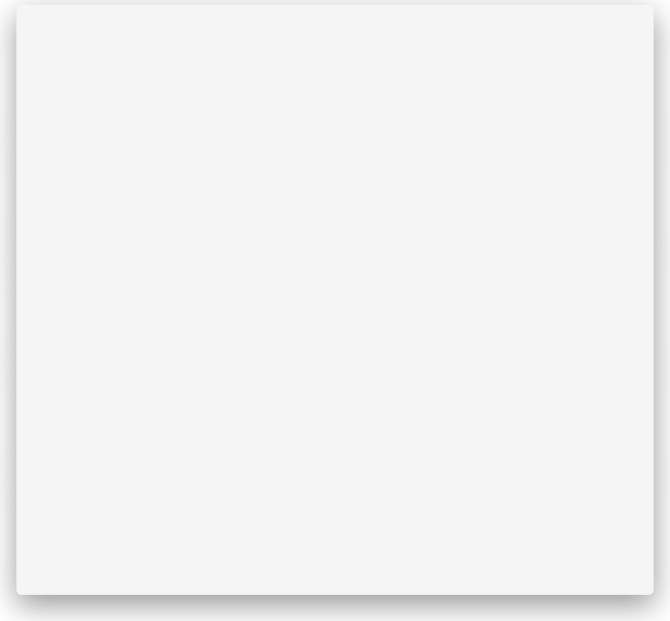
5 6

输出



2

题解





def cho\_minx(cho0, n, k): x\_min = 0

for i in range(n):

if min(cho0[i]) > x\_min:

x\_min = min(cho0[i]) x\_min = x\_min // k

return x\_min

def cho\_att(x\_min, n, k):

count = k + 1

x = x\_min - 1

while count >= k:

x += 1

if x == 0: x = 1

count = 0

for i in range(n):

a = cho0[i][0] // x b = cho0[i][1] // x count += a \* b

return x - 1

n, k = map(int, input().split()) cho0 = []

for i in range(n):

cho0.append(list(map(int, input().split()))) x\_min = cho\_minx(cho0, n, k)

x = cho\_att(x\_min, n, k) print(x)

**5.**等差素数数列



2,3,5,7,11,13, 是素数序列。

类似我1们有：7 ，37 ，67 ，97 ，127 ，157这样完全由素数组成的等差数列，叫等差素数数列，而这个 等差素数数列的公差为30，长度为 6。

2004 年，格林与华人陶哲轩合作证明了：存在任意长度的素数等差数列。 这是数论领域一项惊人的成 果！

有这一理论为基础，请你借助手中的计算机，满怀信心地搜索： 长度为10的等差素数列，其公差最小值是多少？

机考时候我极其昏头 想不出来解决办法，只知道暴力算（鼠鼠脑子太笨了）还是只写了一半（事后发 现没保存...太抽象了），这绝对是我的大错特错qaq。

在暑期时 ， **24**级的刘家蔚同学 ， 指正了我原先的代码错误 ， 并给出了他的使用递归方法而写出的 题解**:**

题解**(**刘同学提供**)**



# 由计算机转专业群 ，24数学刘家蔚同学提供

def is\_prime(n): #素数筛

for i in range(2, int(n\*\*0.5)+1): if n % i == 0:

return False return True

def loop\_check(num,step,time): #本函数用于检测是否符合等差素数数列条件 if time == 0: #base case

return True

if is\_prime(num+step): #递归

return loop\_check(num+step,step,time-1) return False

stop = 0 #状态值 ，0为继续 ，1为停止

for n in range(2,10000): if not is\_prime(n):

continue

elif stop == 0:

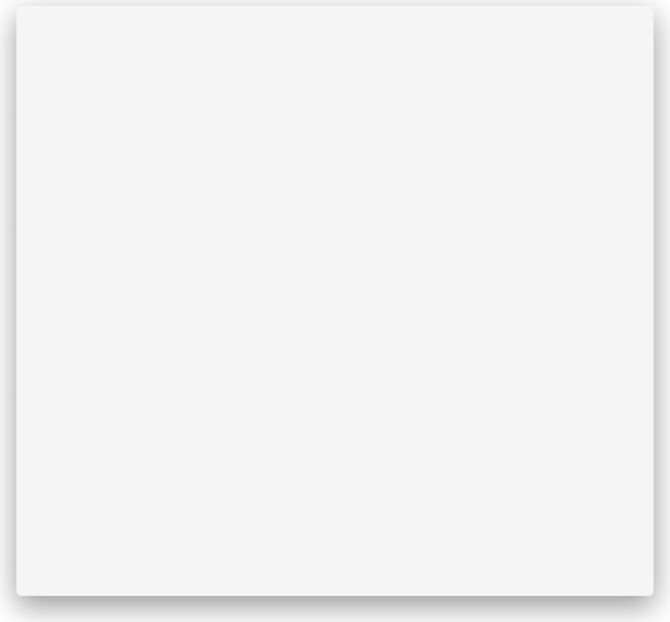
for step in range(2,10000): #枚举等差 ，逐个尝试 if loop\_check(n,step,8):

print(step)

stop = 1 #设置为停止

break elif stop == 1:

break

非常感谢刘同学的提醒，也让我能回头审视当时的一直未被完全解决的问题，但是这个暑期很忙碌，后 续如果有时间，还会再好好修改的，感谢这个项目的作者，能为我们这些想要转入计算机的同学增添力

量。

个人感想



本人码力较差，写的内容也比较平庸，此回忆版题目是在出成绩的前一天凌晨所写，花了挺长时间 的。只是在转机考上，我最后差了一点分数（这纯属是我个人原因了，计算机学院的老师是会捞人

的）。这里大家不要学我，在应试心态和精神状态上出了大问题，我没有把握好这个机会。所以，我也 希望想继续转入计算机的同学，一定要做足机试的心理建设和应试准备，加油，相信你一定可以的！

想起翁恺老师的话：学计算机，一定要有一个非常强大的心理状态，计算机的所有东西都是人做出 来的，别人想得出来，我也一定想得出来。在计算机里头没有任何黑魔法。所有东西只不过是我现在不 知道而已。总有一天，我会把所有的细节，所有的内部的东西全都搞明白的。

我觉得，不管你是有没有转入成功，只要你是热爱计算机科学的，只要你愿意在这条路上坚持下 去，开拓眼界，与时俱进，你一定能够在计算机领域，创造属于你的价值的！！！