

E6-2021级-航类第6次练习

A 第一次 最后一次

题目描述

输入

输出

输入样例

输出样例

样例解释

HINT

B 到底有多少次

题目描述

输入

输出

输入样例

输出样例

输入样例2

输出样例2

HINT

C e的呱呱泡蛙次方减一等价于

题目描述

输入

输出

输入样例

输出样例

其他

D zhnの柱状图

题目描述

输入

输出

输入样例

输出样例

E 简单的数列维护

题目介绍

输入格式

输出格式

输入样例1

输出样例1

数据范围

F LJF帮舍友找CP

题目描述

输入

输出

输入样例

输出样例

样例解释

G 测测你的专属超级英雄

题目描述

输入格式

输出格式

输入样例

输出样例

HINT

NOT HINT

H 图图的红包烦恼

题目描述

输入格式

输出格式

输入样例1

输出样例1

样例解释1

输入样例2

输出样例2

样例解释2

数据范围

HINT

出题参考资料

I 施密特正交化

题目描述

输入

输出

数据范围

输入样例

输出样例

样例解释

Hint

J 20长字符串

题目描述

输入格式
输出格式
输入样例
输出样例
样例解释
HINT

A 星空暗流

题目描述
输入
输出
输入样例
输出样例
数据限制
HINT

E6-2021级-航类第6次练习

- 练习时间2022/04/30 08:00 - 2022-05-07 23:50
- 总共11道题，编号为A~K，题目不一定按难度排序，同学们可以按照通过人数答题；或者提前读完所有题目，按照自己擅长题目的情况答题。
- 各题分值分布为如下，总分103分：

序号	A	B	C	D	E	F	G	H	I	J	K
分值	20	20	20	10	10	10	3	3	3	2	2

- 请严肃练习，严禁抄袭他人代码，课程组会在练习结束后进行代码查重并给予警告。

组题：爱吃猪脚的猪脚

更多题目作者：cbj、呱呱泡蛙、czy、魔法少女zhn、JJJ、LJF、wqh、LNB、助教头子cbd、Arthas

A 第一次 最后一次

题目描述

查找1个字符在字符串中出现的位置,求第一次出现和最后一次出现的距离.

输入

第一行为待查找的字符串,可能出现空格,长度保证小于等于250

第二行为1个字符

输出

输出一行

如果字符出现多次,输出第一次出现和最后一次出现的距离

如果字符出现一次,输出 `once`

如果字符没有出现,输出 `not exist`

输入样例

```
cbjlaji  
j
```

输出样例

```
3
```

样例解释

`cbjlaji` 中 `j` 字符出现2次,第2次在第1次后面3个字符的位置.

HINT

第2行待查找的字符可能是空格.为什么不用下 `strchr` 函数和 `strrchr` 函数?

author:cbj

B 到底有多少次

题目描述

简单的练习下 `strstr` 函数的用法:查找一个子字符串在字符串中总共出现了多少次

输入

两行,第一行一个长度小于等于256的字符串a,第二行一个长度小于等于32的字符串b

保证a比b长,且每行行尾不会出现 `\r`, 字符串a, b可能含有空格

输出

一个整数,字符串a中子字符串b出现的次数(没有出现就输出0)

输入样例

```
caijicbjzhichujiandanti  
ji
```

输出样例

```
2
```

输入样例2

```
aaa  
aa
```

输出样例2

```
2
```

HINT

样例2给了个特殊情况 字符串 `aaa` 从第1个字符开始和从第2个字符开始都能找到 `aa`

从第一个字符开始找,找到了以后从下一个字符的位置开始继续找,直到找不到更多目标字符串为止.

大家练习下指针的用法.题目描述里面说了 试试 `strstr` 函数吧

author:cbj

c e的呱呱泡蛙次方减一等价于

题目描述

考研名师呱呱泡蛙讲解等价无穷小的语录：

“e的呱呱泡蛙次方减一等价于？呱呱泡蛙，呱呱泡蛙趋于零。”

输入

一行字符串，格式为`e^{呱呱泡蛙}-1`。总长度不超过20。

输出

一行字符串，格式为`呱呱泡蛙, 呱呱泡蛙\to 0`

输入样例

`e^{x}-1`

输出样例

`x, x\to 0`

其他

语录参见张宇《考研数学30讲》。

当 🐸 $\rightarrow 0$ 时

$$\sin \text{🐸} \sim \text{🐸}$$

$$\tan \text{🐸} \sim \text{🐸}$$

$$\ln(1 + \text{🐸}) \sim \text{🐸}$$

$$e^{\text{🐸}} - 1 \sim \text{🐸}$$

$$\arcsin \text{🐸} \sim \text{🐸}$$

$$\arctan \text{🐸} \sim \text{🐸}$$

$$\log_a(1 + \text{🐸}) \sim \frac{\text{🐸}}{\ln a}$$

$$\text{🐸} - \ln(1 + \text{🐸}) \sim \frac{1}{2} \text{🐸}^2$$

$$1 - \cos \text{🐸} \sim \frac{1}{2} \text{🐸}^2$$

$$\ln(\text{🐸} + \sqrt{1 + \text{🐸}^2}) \sim \text{🐸}$$

$$\text{🐸} - \sin \text{🐸} \sim \frac{1}{6} \text{🐸}^3$$

$$\tan \text{🐸} - \text{🐸} \sim \frac{1}{3} \text{🐸}^3$$

$$(1 + \text{🐸})^\alpha - 1 \sim \alpha \text{🐸}$$

$$\arcsin \text{🐸} - \text{🐸} \sim \frac{1}{6} \text{🐸}^3$$

$$\text{🐸} - \arctan \text{🐸} \sim \frac{1}{3} \text{🐸}^3$$

$$\tan \text{🐸} - \sin \text{🐸} \sim \frac{1}{2} \text{🐸}^3$$

Author: 呱呱泡蛙

D zhnの柱状图

题目描述

魔法少女zhn给了你四行字符串，请你统计出每个字母出现了多少次，以星号柱状图的形式输出。

保证所有字母都是大写，空格和标点符号不计入最后统计。如果字母没有出现过，横坐标也要有这个字母。

具体形式见样例。

注意：每行最后不要有多余的空格，最后一行不要有多余的换行。

输入

四行字符，由大写字母和标点符号组成，每行不超过100个字符

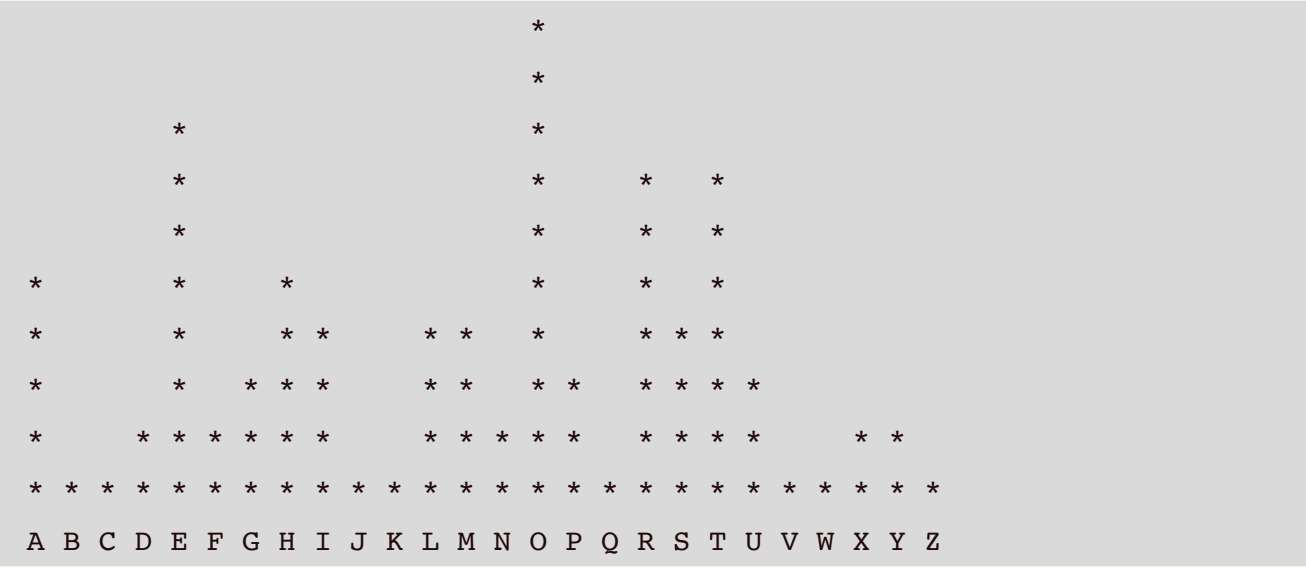
输出

由若干行组成，前几行由空格和星号组成，最后一行则是由空格和字母组成的。不要打印任何空行，具体见输出样例。

输入样例

```
THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG.  
THIS IS AN EXAMPLE TO TEST FOR YOUR  
HISTOGRAM PROGRAM.  
HELLO!
```

输出样例



关于输出的一点说明：柱状图的高度为出现次数最多的字母的 * 高度;每一列与每一列之间有一个空格，最后一列后面没有空格，每一行之间没有空行。

后记：为什么魔法少女的题最近很简单，因为我也不喜欢字符串TwT

author：魔法少女zhn

E 简单的数列维护

题目介绍

如题目所说，这是一道跟数列维护有关的题目。

*JJJ*有一个数列，一开始是空的。*JJJ*一共会下达 n 次指令，每次指令包含一个参数 a_i ，如果 a_i 不在数列中，则把 a_i 加入到数列中；如果 a_i 在数列中，则把 a_i 从数列中删除。

执行完这 n 次指令后，数列里还剩多少数呢？

输入格式

第一行，一个正整数 n ，表示指令的个数。

接下来 n 行，每行一个正整数，表示这一次指令的参数。

输出格式

一个整数，表示最后数列里数的个数。

输入样例1

```
7
23
5
3
5
12
3
6
```

输出样例1

```
3
```

数据范围

对于40%的数据： $1 \leq n \leq 1000$, $1 \leq a_i \leq 10^5$.

对于100%的数据： $1 \leq n \leq 10^5$, $1 \leq a_i \leq 10^9$.

Author: *YYY*.

F LJF帮舍友找CP

题目描述

*LJF*的舍友早已母胎*solo*整整21年，要是再不脱单就会变成魔法少女，所以他希望*LJF*帮他进行匹配，匹配规则如下：

给你两个字符串 *s1* 和 *s2*，写一个函数来判断 *s2* 是否包含 *s1* 的排列。如果是，输出 65472；否则，返回 Single Forever!。

换句话说，*s1* 的排列之一是 *s2* 的子串。

*s1*的排列即对*s1*里的元素进行重新排序。

输入

第一行为字符串 *s1*。

第二行为字符串 *s2*。

s1 和 *s2* 的长度均不超过 10^5 ，且仅包含小写字母。

输出

一行。

判断 *s2* 是否包含 *s1* 的排列。如果是，输出 65472；否则，返回 Single Forever!。

输入样例

```
acb
abcabc
```

输出样例

```
65472
```

样例解释

字符串 `acb` 经重新排序后可得到 `abc` 或 `bca` 或 `cab`，这三种都是 `abcabc` 的子串，所以输出 `65472`。

G 测测你的专属超级英雄

题目描述

小盒梓设计了一个程序，只需要输入自己的身份证号和手机号，就可以测一测你最接近哪个超级英雄。

超级英雄可能是 `Captain America, Daredevil, Iron Man, Thor, Black Widow, Deadpool, Hulk, Thing, Human Torch, Scarlet Witch, Wolverine, Storm, Spiderman, Punisher, Emma Frost, Wong, The Amazing Spider-man, Wanda Maximoff, Doctor Strange, Moon Knight, Shang-Chi` 共 21 位超级英雄中的一位，输入身份证号和手机号后，程序使用它们组成的数字对超级英雄数量自动取余，余数所对应序号（从第 0 开始计数）的超级英雄就是你最接近的超级英雄，请你实现这个程序。

输入格式

多组数据输入，每一行长度为 29 位，仅包含数字（没有 **X**！）。

输出格式

多组数据输出，每一行代表输入所对应的专属超级英雄。

输入样例

```
00000000000000000000000000000001
00000000000000000000000000000002
58554893545761446406935306226
00339168120980507254211862750
21117231938088703759512139794
```

输出样例

```
Daredevil
Iron Man
Scarlet Witch
Hulk
Hulk
```

HINT

本题为 [终极进制](#) 青春版。

为避免可能含有的不可见字符，推荐大家使用 `scanf("%s")` 进行字符串读入。如执意使用 `gets` 或 `fgets` 函数读入，可以对读入后的字符串进行如下预处理：

```
int len = strlen(input);
while(input[len - 1] < '0' || input[len - 1] > '9'){ //清除所有非数字字符
    input[len - 1] = '\0'; //添加结束符
    len--;
}
```

NOT HINT

根据真实事件改编，请同学们谨防电信诈骗

author: czy

H 图图的红包烦恼

题目描述

图图过年抢红包手速很快，每次都是第一个抢到，但少有手气最佳，为此他十分苦恼，决定探索下微信红包机制！

查找资料后图图发现，每个人能抢到的红包金额服从 **0.01 到 2 倍剩余均值**之间的随机分布；

即如果剩余红包金额为 M （以分为单位），剩余人数为 N ($N > 1$)，那么每次抢到的金额 m 是区间 $[1, M / N \times 2 - 1]$ 上的随机整数（其中 m 以分为单位， M / N 向下取整）

假设 5 个人抢 6.66 元：

- 第一个人抢时剩余 666 分钱，于是最多能抢 $666 / 5 \times 2 - 1 = 265$ 分钱，最少能抢 1 分钱，假设抢得 66 分钱；
- 第二个人抢时剩余 600 分钱，最多能抢 $600 / 4 \times 2 - 1 = 299$ 分钱，最少能抢 1 分钱，假设抢得 100 分钱；
- 第三个人抢时剩余 500 分钱，最多能抢 $500 / 3 \times 2 - 1 = 331$ 分钱，最少能抢 1 分钱，假设抢得 300 分钱；
- 第四个人抢时剩余 200 分钱，最多能抢 $200 / 2 \times 2 - 1 = 199$ 分钱，他将与第五个人分这 200 分钱。

现给出红包总额 $money$ 、红包个数 num ，请你运用随机数做 n 次实验，算出这种机制下每个人抢到的平均金额和标准差，以此来估计总体均值和方差。

注：本题采用 `Special Judge`，你的答案与标准值相差不超过 **0.1** 即为正确。

输入格式

一行，一个浮点数 $money$ （以元为单位，小数点后有且仅有两位数）和一个整数 num ，其间以一个空格分隔。

输出格式

一共 num 行，第 i （ $1 \leq i \leq num$ ）行输出以一个空格分隔的两个浮点数 avg 和 s ，分别表示 n 次实验中第 i 位抢到红包者的平均金额（以元为单位）和标准差，保留两位小数。

输入样例1

```
200.00 3
```

输出样例1

```
66.66 38.48
66.67 44.44
66.67 44.44
```

样例解释1

3 个人抢 200 元，第 1 位抢到红包者平均可以得到 66.66 元，标准差为 38.48，第 2 位抢到红包者平均可以得到 66.67 元，标准差为 44.44，第 3 位抢到红包者平均可以得到 66.67 元，标准差为 44.44。

输入样例2

```
0.05 4
```


输出样例2

```
0.01 0.00
0.01 0.00
0.01 0.00
0.02 0.00
```

样例解释2

这就是有名的“末位红包抽屉原理”，即 N 个人抢 $(N + 1)$ 分钱，不论试验多少次，结果一定是最后一位抢红包者抢到 2 分钱，因此标准差均为 0。

数据范围

$$2 < num < 10$$

$$0.01 \leq money \leq 200.00 \text{ 且满足 } 0.01 \times num \leq money$$

HINT

- 样本标准差公式为：

$$S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}}$$

- 为了使样本指标尽可能接近总体指标，推荐 n 可在 500000 ~ 1000000 范围内取值；
- 对浮点数四舍五入取整请使用 `round` 函数；
- 得到区间 $[a, b]$ 上随机整数的方法：
 - 如果使用的是 Linux 系统或 Mac OS 系统，对于本题可以直接取模：

```
int u = rand() % (b - a + 1) + a;    // 产生区间[a, b]上的随机整数
```

- 如果使用的是 Windows 系统，由于 `rand` 函数所能产生随机数的最大值为 32767，直接取模会导致产生的随机数分布不均匀，应先转为浮点随机数再作处理：

```
int u = (int)(1.0 * rand() / RAND_MAX * (b - a + 1)) +  
a;    // 产生区间[a, b]上的随机整数
```

- 由于评测系统使用的是 Linux，因此以上两种写法均可以反馈正确答案，只是在自己电脑上测试会有所不同
- 题目看似复杂实际只要模拟这一过程，做完还是感觉很有趣的嘿

出题参考资料

- 毕导《微信红包先抢和后抢差距居然这么大！春节抢红包的大数据分析》
- 《微信红包的架构设计简介》来源于QCon某高可用架构群整理，整理朱玉华
- 张志雄、张静之、赵春锋《微信红包随机金额生成算法模拟及应用》
- iqqcode《如何实现抢红包算法，怎样才能做到随机公平》

Author: wqh

I 施密特正交化

题目描述

施密特正交化是求欧氏空间正交基的一种方法，也是《线性代数》课程中为大家所熟知的概念。运用这种方法，可以将任意线性无关的向量组转变为标准正交向量组。在数学中，施密特正交化将含有多个变量的数学方程变成能够用单个变量表示的方程组；而工程意义上，施密特正交化体现了一种解耦的思想，将一个彼此影响的多变量系统化为多个独立的单变量系统，不同的输出仅由不同的输入控制，增强各自的独立存在性。

运用编程语言，可以辅助我们进行施密特正交化的计算，现给出一组线性无关的向量 $\alpha_1, \alpha_2 \dots \alpha_m$ ，按照下图流程可将其转换为一组正交向量 $\beta_1, \beta_2 \dots \beta_m$ 。

$$\begin{aligned}\beta_1 &= \alpha_1 \\ \beta_2 &= \alpha_2 - \frac{\langle \alpha_2, \beta_1 \rangle}{\langle \beta_1, \beta_1 \rangle} \beta_1 \\ \beta_m &= \alpha_m - \frac{\langle \alpha_m, \beta_1 \rangle}{\langle \beta_1, \beta_1 \rangle} \beta_1 - \frac{\langle \alpha_m, \beta_2 \rangle}{\langle \beta_2, \beta_2 \rangle} \beta_2 - \dots - \frac{\langle \alpha_m, \beta_{m-1} \rangle}{\langle \beta_{m-1}, \beta_{m-1} \rangle} \beta_{m-1}\end{aligned}$$

再令正交向量 $\beta_1, \beta_2 \dots \beta_m$ 单位化，即可得到一组标准正交基 $e_1, e_2 \dots e_m$ 。

$$e_i = \frac{\beta_i}{\|\beta_i\|} (i = 1, 2, \dots, m)$$

为了降低运算的复杂度，需要先将向量进行排序，优先处理零元素多的向量，同时避免除数较小导致增大误差，优先处理模长大的向量。

总而言之，按照下列流程完成计算。

1.将向量排序，排序优先级为：

[1]向量的零元素个数（个数大的在前，个数小的在后）

[2]向量的模长（模长大的在前，模长小的在后）

[3]向量的输入次序（输入次序前的在前，输入次序后的在后）

排序时优先满足最高优先级的条件，在最高优先级条件相同的条件下，考虑次一优先级条件。

2.将排序后的向量由前往后进行施密特正交化，得到一组线性无关的向量

3.将线性无关的向量单位化（每个元素除以模长），得到一组标准正交基

4.按照顺序依次输出标准正交基

输入

共N+1行。

第一行为一个正整数N，N具有双重含义，N既是线性无关向量组的向量个数，也是每个向量的元素个数。

接下来N行为向量组，严格保证其线性无关，每行包含N个int类型整数。

输出

共N行，每行为按照上述流程处理后的标准正交基。

每行包含标准正交向量的N个元素，保留四位小数。

数据范围

$0 < N < 11$,所有元素均在int范围内

输入样例

```
3
1 3 5
8 6 7
2 0 2
```

输出样例

```
0.7071 0.0000 0.7071
0.0828 0.9931 -0.0828
-0.7022 0.1170 0.7022
```

样例解释

首先对三个向量进行排序，排序结果为 $(2, 0, 2), (8, 6, 7), (1, 3, 5)$ 。

对三个向量依次进行施密特正交化，正交化结果为 $(2.0000, 0.0000, 2.0000), (0.5000, 6.0000, -0.5000), (-2.2192, 0.3699, 2.2192)$ 。

再对三个变量依次单位化，单位化结果为 $(0.7071, 0.0000, 0.7071), (0.0828, 0.9931, -0.0828), (-0.7022, 0.1170, 0.7022)$

Hint

可使用二维数组对数据进行存储，留出某几个特征位记录0的个数、模长和出现次序,依照这些特征位对二维数组进行排序。

ps：这学期大家所学的傅里叶级数，本质上是在函数空间中，将任意周期函数在一组标准正交基上展开，这体现了标准正交基的强大之处。《吕氏春秋》有言：“闻弦歌知雅意”，希望数学这门学科能成为大家的“弦歌”，不断求索其中的巧妙与震撼感。

Author:LNB

J 20长字符串

题目描述

现在有好多好多行字符串，请你找出前20长的字符串，并按照输入顺序输出；如果第20长的字符串有多个，输出先出现的。

输入格式

多行输入，每行一个字符串 s ；

保证字符串 s 的长度 len 满足 $1 \leq len \leq 1000$ ，保证输入的字符串总数 n 满足 $20 \leq n \leq 100000$ ；

字符串仅包含可见字符。

输出格式

按照输入的顺序，输出前20个最长的字符串。

输入样例

```
as
Ren jiangyi
Ai chizhujiaodezhujiao
cbj
Great-designer
JJJ
Lucienlrq
Shi Shangzuiqiangdexingjierxuanshou
wqhgg
Long Juanfeng
lxy
ljh
Guo Lichen
zymdka
yuruzan
Wang Wenhan
cbd
Li Nanbing
SongYou
Bodan's dog
casual name
I don't know
```

输出样例

```
Ren jiangyi
Ai chizhujiaodezhujiao
cbj
Great-designer
JJJ
Lucienlrq
Shi Shangzuiqiangdexingjierxuanshou
wqhgg
Long Juanfeng
lxy
ljh
Guo Lichen
zymdka
yuruzan
Wang Wenhan
Li Nanbing
SongYou
Bodan's dog
casual name
I don't know
```

样例解释

输入共22行字符串，输出样例为前20长的字符串；字符串 `as` 最短不予输出，字符串 `cbd` 虽然和 `cbj`、`ljh` 等字符串长度相等，但在输入顺序中它后出现，故不予输出。

HINT

注意内存限制，不能将所有字符串全部存下来。

A 星空暗流

题目描述

银河帝国在扩张的过程中，银河殖民者们需要在各个星球间进行跃迁穿梭。由于帝国的技术十分发达，飞船可以拟定一条跃迁轨迹，作为其到达某个星球的路线。例如：

$$a \longrightarrow bb \longrightarrow ccc \longrightarrow dddd \longrightarrow eeeee \longrightarrow fffffff$$

（其中 a, bb, ccc, \dots 为星球名）

然而，星空中暗流涌动，从而使得银河殖民者们有时要根据情况去调整跃迁轨迹，例如，上述的跃迁轨迹被调整为：

$$a \longrightarrow ccc \longrightarrow dddd \longrightarrow ggggggg \longrightarrow iiiiiiiii \longrightarrow fffffff$$

调整一个跃迁轨迹需要消耗飞船的能量，具体消耗规则为：

- 在轨迹中删除一个节点，需要消耗所删除节点长度的能量
- 在轨迹中添加一个节点，需要消耗所添加节点长度的能量
- 在轨迹中替换一个节点，需要消耗原节点和新节点长度平均值的能量（向上取整）

（注：规则中的“长度”指字符串长度，调整轨迹的方式只有删除，添加，替换三种，且只能对节点进行整体操作，即不能只增删改节点中部分字符）

银河殖民者们发现，利用这三种规则，有多种方式可以调整成功，但只有某一种或是几种的能量消耗是最小的。

例如，在上例中，能量消耗最低的调整策略为：

1. 删除 bb 节点，消耗2能量；
2. 添加 $ggggggg$ 节点，消耗7能量；
3. 将 $eeeeee$ 节点替换为 $iiiiiiiiii$ ，消耗7能量。

共消耗16能量。

现在，给你飞船原计划的跃迁轨迹和遇到星空暗流后的修正轨迹，请计算出飞船在调整轨迹中需消耗的最少能量。

输入

第一行，输入两个正整数 m, n ，分别为调整前后的轨迹节点数。

第二行，输入 m 个用空格隔开的字符串，表示调整前的轨迹。

第三行，输入 n 个用空格隔开的字符串，表示调整后的轨迹。

输出

输出一个正整数，即消耗的最少能量。

输入样例

```
6 6
a bb ccc dddd eeeee fffffff
a ccc dddd gggggggg iiiiiiiiii fffffff
```

输出样例

```
16
```

数据限制

- $0 < m, n \leq 500$
- 每个节点字符串的长度 $0 < l \leq 100$

HINT

可以思考一个简化的问题：

通过增删改其中的字符，将一个字符串变为另一个字符串，最少需要多少步？

“除非你了解，’琼斯柔声道，‘在一个新的银河中——一个不受恒星不稳定性威胁的银河，一个人人都有蓊蒨的银河，一个政治统一近在眼前的银河——终归有许多赢家。一千兆个赢家，整个银河的人民，他们全都是胜利者’”——《星空暗流》



Author: Arthas