

A. 果壳游戏

题目描述

为了消磨时光, *Bessie* 和她的朋友 *Elsie* 喜欢玩一种她们在农业展览会上看到的游戏

游戏准备阶段, *Bessie* 在桌子上放置三个倒置的坚果壳,并在其中一个坚果壳下面藏了一块小的鹅卵石。随后 *Bessie* 会两两调换坚果壳,同时 *Elsie* 试着去猜鹅卵石的位置

在农业展览会上看到的这个游戏的标准形式是玩家可以看到鹅卵石初始的位置,然后要求玩家猜所有交换完成之后鹅卵石最终的位置

然而,现在奶牛们想要去进行这样一种玩法, *Elsie* 不知道鹅卵石的初始位置,同时她可以在每一次交换之后猜一下鹅卵石的位置

Bessie 知道正确答案,在游戏结束后会给 *Elsie* 一个分数,等于她猜对的次数

给定所有的交换和 *Elsie* 的猜测,但是不给出鹅卵石的初始位置,请求出 *Elsie* 最高可能获得的分数

输入格式

输入的第一行包含一个整数 N ,为交换的次数 ($1 \leq N \leq 100$)

以下 N 行每行描述了游戏的一个回合,包含三个整数 a 、 b 和 g ,表示 *Bessie* 交换了坚果壳 a 和 b ,然后 *Elsie* 猜的是坚果壳 g

所有这三个数均为 1、2、3 之一,并且 $a \neq b$

输出格式：

输出 *Elsie* 可以得到的最高分数。

输出格式：

输出 *Elsie* 可以得到的最高分数。

输入样例

```
3
1 2 1
3 2 1
1 3 1
```

输出样例

```
2
```

样例解释

在这个例子中, *Elsie* 最多可以获得 2 分

如果鹅卵石开始时位于坚果壳 1 下面,那么她猜中了一次(最后一次)

如果鹅卵石开始时位于坚果壳 2 下面,那么她猜中了两次(开始两次)

如果鹅卵石开始时位于坚果壳 3 下面,那么她没有猜对任何一次

由于这一题只有三个果壳,所以可以假设鹅卵石在1 2 3这三个位置上,分别模拟。用一个变量计录石头的位置,另一个变量统计分数,最后取最大的分数,输出。

B.变速跑

由于xxx说不用写,这里就不写了。

C. 展开连分数

题目描述

连分数时下列形式的分式

$$a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\ddots + \frac{1}{a_n}}}}$$

其中 a_0, a_1, \dots, a_n 为非负数

给出一个分数 $\frac{a}{b}$, 请将其转换成连分数形式

输入格式

第一行输入一个正整数 T , 表示 T 组询问

接下来 T 行, 每行两个正整数 a, b

输出格式

输出 T 行, 每组询问输出一行

一行输入若干个整数分别表示 a_0, a_1, \dots, a_n

输入样例

```
2
105 38
1 114
```

输出样例

```
2 1 3 4 2
0 114
```

样例解释

$$\frac{105}{38} = 2 + \frac{1}{1 + \frac{1}{3 + \frac{1}{4 + \frac{1}{2}}}}$$
$$\frac{1}{114} = 0 + \frac{1}{114}$$

数据规模

对于 30% 的数据 $1 \leq a, b \leq 10^5$

对于 100% 的数据 $1 \leq a, b \leq 10^{18}, 1 \leq T \leq 1000$

这一题就是将一个分数拆开, 第一次是a中含b的个数, 加上 分子:a中减去所有b后所剩的数 (也就是a除b的余数, 即

$a \% b$) ; 分母: b 。↓

$$\left\lfloor \frac{a}{b} \right\rfloor + \frac{a \% b}{b}$$

而 $\frac{a \% b}{b}$ 中还要继续重复展开, 转化成 $\frac{\frac{a \% b}{b}}{b}$, 即 $\frac{b}{a \% b}$, 再将其按上面步骤展开, 这就可以用递归。并且因为对于 100% 的数据 $1 \leq a, b \leq 10^{18}, 1 \leq T \leq 1000$, 要用 long long

```
void z(long long a, long long b){
    if (b==0){ // 除数b 不能为0, 为0时结束(b==0意味着a%b==0, a是b的倍数, 则在上一次递归中输出为整数, 无需再展开
        return;
    }
    cout<<a/b<<' '; // 输出
    z(b, a%b); // 再展开
    return;
}
```

D. 猫片数量

题目描述

众所周知, 有猫的照片叫做猫片。

Mas 获得了 n 张照片, 对于每一张照片 Mas 使用一个长度不超过 10 的字符串 S 来表示照片内容。

如果字符串是 `cat` 表示这是一张猫片, 但是调皮的猫很有可能躲在一些灌木丛中, 所以你能看到部分的猫。

例如 `c`, `a`, `ca`, `at`, `t` 都是部分猫的照片, Mas 能够将部分猫的照片拼接(不允许裁剪)成一张完整的猫片。

请你告诉 Mas , 他能获得多少张完整的猫片(仅包 `cat`)。

输入描述

第一行输入一个正整数 n , 表示照片的数量。

第二行输入 n 个字符串 S_i , 表示每张照片的内容

输出描述

输出一个整数, 表示完整的猫片数量

输出描述

输出一个整数, 表示完整的猫片数量

输入样例1

```
5
a c a cat cat
```

输出样例1

```
2
```

输入样例2

```
3
ca at a
```

输出样例2

0

输入样例3

9

ca ca c c t t at at uakioi

输出样例3

4

输入样例4

3

c a t

输出样例4

1

数据规模

对于全部的数据 $1 \leq n \leq 2 \times 10^5, 1 \leq S_i \leq 10$

这道题只要输入每个有猫的照片的部分，即‘c’，‘a’，‘t’，‘ca’，‘at’，‘cat’，再在其中找出能拼出最多“cat”的情况就可以了。

输入只需要能统计出每个部分的个数就行，方法不限（怎么开心怎么来），比如每次输入一个字符串，判断是哪个部分并将其个数+1。

输入后就在其中寻找‘c’与‘at’、‘ca’与‘t’、‘c’与‘a’与‘t’三种组合，但每种组合都要以其中个数最少的来拼接。并且要最后判断‘c’与‘a’与‘t’的组合，因为同样组成“cat”，先拼接另外两种组合可以拼出更多“cat”。最后加上“cat”本身（也就是不用拼接的猫片的个数）并输出