

NOIP2022模拟赛20221119C组

7:50--11:50，提前交卷请报告老师

CWOI提交，需要文件操作，以第一次提交为准

A. Prime

`prime.cpp/.in/.out` 1S 512M

题目描述

Cuber QQ 会给你两个数 l 和 k ，你需要确定的是在区间 $[l, l + 2k)$ 中是不是有超过 k 个质数。

输入

输入包含一行，两个整数 $l, k (1 \leq l, k \leq 10^8)$ 。

输出

输出一行包含 `Yes` 或者 `No`，表示问题的答案。

样例

输入1

3 3

输出1

No

输入2

2 1

输出2

Yes

数据范围

本题采用捆绑测试。

第一个捆绑测试点满足 $1 \leq l, k \leq 10^3$ ，30 分。

第二个捆绑测试点满足 $1 \leq l, k \leq 10^6$ ，30 分。

第三个捆绑测试点满足 $1 \leq l, k \leq 10^8$ ，40 分。

B. Algebra

algebra.cpp/.in/.out 1S 512M

题面描述

Cuber QQ 最近沉迷抽象代数，他打算给你出一个题。

给定一个乘法群 $M = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix} \mid a, b, c, d \in \mathbb{Z}, ad - bc = 1 \right\}$ 。

这个乘法群拥有非常有意思的性质：

即，存在两个生成元 $A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ 和 $B = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ ，如果有 $x \in M$ ，那么一定满足 $x = \prod_i x_i$ 其中 $x_i \in \{A, B, A^{-1}, B^{-1}\}$ 。

现在 Cuber QQ 会给你一个 $x \in M$ ，你需要找到一个序列 x_i 满足 $x = \prod_i x_i$ 其中 $x_i \in \{A, B, A^{-1}, B^{-1}\}$ 。

Cuber QQ 也不想太为难你，为了让题目更加简单，保证 $x = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ 满足 $abcd = 0$ ，即 a, b, c 和 d 中至少有一个是 0。

输入

输入第一行包含一个整数 T ($1 \leq T \leq 10$)，表示测试数据组数。

接下来的 T 行，每行包含四个整数 a, b, c 和 d ($-10^5 \leq a, b, c, d \leq 10^5, ad - bc = 1, abcd = 0$)，表示 $x = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ 。

输出

对于每一组测试数据，第一行输出一个整数 m ($1 \leq m \leq 4$)，表示你所需要的序列长度。

在接下来的 m 行，每一行输入一个字符 c_i 和一个整数 a_i ($c_i \in \{ 'A', 'B' \}, -10^5 \leq a_i \leq 10^5$)。

你需要保证 $x = c_1^{a_1} c_2^{a_2} \cdots c_m^{a_m} = \prod_i c_i^{a_i}$ 。

如果有多个可能的结果，你只需要输入任意一个。

我们保证给定的输入总是有解的。

样例

输入

```
3
1 1 0 1
-1 1 -1 0
-1 -1 0 -1
```

输出

```
1
A 1
2
A 1
B 1
2
A 1
B 2
```

数据范围

本题采用捆绑测试。

第一个捆绑测试点满足 $a = 0$ ，25 分。

第二个捆绑测试点满足 $b = 0$ ，25 分。

第三个捆绑测试点满足 $c = 0$ ，25 分。

第四个捆绑测试点满足 $d = 0$ ，25 分。

C. Convert

convert.cpp/.in/.out 1S 512M

题面描述

在采用进位计数的数字系统中，如果使用的数码依次为 $0, 1, 2, \dots, R - 1$ ，总共 R 个数码，则称其为基 R 数制或 R 进制。 R 称为该数制的“基数”(Radix)，例如，十进制的基数是 10（数码为： $0, 1, 2, \dots, 9$ ），二进制的基数是 2（数码为： $0, 1$ ），八进制的基数是 8（数码为： $0, 1, 2, \dots, 7$ ）。

现有一种特殊的按位计数系统，该系统采用的基数为复数 $-1 + i$ （其中 i 表示 $\sqrt{-1}$ ）。在 $-1 + i$ 进制系统中，所有“复数整数”（实部和虚部都为整数的复数，这种复数在数学上称为高斯整数）都可以表示成一个“数”，该“数”只含有 0, 1 两个数码，不需要正负符号或其他常规手段，而且每个“复数整数”只有唯一一种表示方法。例如：整数 2 用 $-1 + i$ 进制表示出来是 1100。

任意一个“复数整数” $a + bi$ （ a, b 均为整数）可采用如下算法转换为 $-1 + i$ 进制表示：

1、 $a + bi$ 除以 $-1 + i$ 得到商 q 和余数 r ，商 q 也为“复数整数”，余数 r 为 0 或 1。

令 $q = q_r + q_i i$ (q_r 与 q_i 均为整数，分别表示商 q 的实部与虚部)，

q, r 必满足下式： $a + bi = (q_r + q_i i)(-1 + i) + r$

如果 a, b 均为偶数或均为奇数，则令 r 为 0。

此外，如果 a, b 一奇一偶，则令 r 为 1。

2、重复第一步用商 q 除以 $-1 + i$ 记录下余数 r ，直到商为 0，运算过程结束。

3、从下往上读取余数，就可得到“复数整数” $a + bi$ 的 $-1 + i$ 进制表示。

例如：整数 $2 = 2 + 0i$ 的运算过程：

整数 2 的实部和虚部都是偶数，所以余数为 0， $\frac{2}{-1+i} = (-1 - i)$ 余 0

$-1 - i$ 的实部和虚部均为奇数，所以余数为 0， $\frac{-1-i}{-1+i} = i$ 余 0

i 的实部为偶数，虚部为奇数，所以余数为 1， $\frac{i}{-1+i} = 1$ 余 1

1 的实部为奇数，虚部为偶数，所以余数为 1， $\frac{1}{-1+i} = 0$ 余 1

商为 0，运算结束，从下往上读取，得到整数 2 用 $-1 + i$ 进制表示为 1100。

下表列出了 $-1 + i$ 进制的位串 0000 至 1111 对应的十进制下的“复数整数”。

$-1 + i$ 进制	十进制下的复数整数
0	0
1	1
10	$-1 + i$
11	i
100	$-2i$
101	$1 - 2i$
110	$-1 - i$
111	$-i$
1000	$2 + 2i$
1001	$3 + 2i$
1010	$1 + 3i$
1011	$2 + 3i$
1100	2
1101	3
1110	$1 + i$
1111	$2 + i$

输入一个十进制下的“复数整数” $a + bi$ ，输出其用 $-1 + i$ 进制的表示。

输入

在一行中输入一个“复数整数” $a + bi$ （ a, b 均为整数， $-10^{18} \leq a, b \leq 10^{18}$ ）。

注意：输入中不包含多余的空格，但当某项为 0 或 ± 1 时，采用与手写时相同的表示方式，可以参考上表中的例子。

输出

在一行中输出“复数整数” $a + bi$ 用 $-1 + i$ 进制的表示。

样例

输入1

2

输出1

1100

输入2

-1+i

输出2

10

输入3

-2

输出3

11100

输入4

3+2i

输出4

1001

数据范围

本题采用捆绑测试，完成以下子任务，你可以获得相应的分数。

$|a|, |b| \leq 5$ ，20 分。

$|a|, |b| \leq 10^6$ ，20 分。

$|a|, |b| \leq 10^9$ ，30 分。

$|a|, |b| \leq 10^{18}$ ，30 分。

D. Dye

dye.cpp/.in/.out 1S 512M

题面描述

Cuber QQ 有一个 n 个节点的连通图，用 1 到 n 的整数标记。

图中包含 $n-1$ 条边。

Cuber QQ 现在在和 Little Fang 玩游戏，游戏开始之前，Cuber QQ 将一些边涂成蓝色，Little Fang 将一些边涂成红色。

如果有某条边被同时涂了两种颜色，最终颜色为黑色。保证所有的边至少由其中一种颜色着色。

Cuber QQ 在节点 a 开始游戏，Little Fang 在节点 b 开始游戏。

两个人会交替移动他们所在的位置，游戏从 Cuber QQ 开始。每一次轮到其中一个人移动的时候，他们需要将他们的位置移动到任意相邻的位置上，但是不能是对手所在的位置。

另外，Cuber QQ 不能使用红色边，Little Fang 不能使用蓝色边，而他们都可以使用黑色的边。无法采取行动的 player 就输了。

游戏中，Cuber QQ 和 Little Fang 都按照最优的策略进行游戏。

如果他们意识到游戏可以永远运行，他们会宣布游戏平局，并结束游戏。

输入

第一行包含一个整数 $n(2 \leq n \leq 10^5)$ ，表示节点数量。

第二行包含整数 a 和 b ($1 \leq a, b \leq n, a \neq b$)，表示 Cuber QQ 和 Little Fang 的初始位置。

接下来的 $n - 1$ 行表示边每一行的格式都是 `x y color`，表示边 $(x, y), 1 \leq x, y \leq n$ ，color 是 blue/red/black 的其中一个。

输出

输出包含一行字符串，输出 `Cuber QQ` 表示 Cuber QQ 获胜，如果 Little Fang 获胜则输出 `Little Fang`。如果是平局，则输出 `Draw`。

样例

输入1

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

输出1

```
Cuber QQ
```

输入2

```
5
3 5
1 2 black
1 3 black
2 4 blue
2 5 red
```

输出2

```
Little Fang
```

数据范围

本题采用捆绑测试。

对于其中 33分 的数据，保证 $2 \leq n \leq 100$ ；

对于其中另外 33分 的数据，保证所有边的染色都是 black 。

对于其中 100% 的数据，保证 $2 \leq n \leq 10^5$ 。