ckr与平方数

众所周知，ckr的微积分水平很不怎么样，就像现在他就不会求一个很水的积分：$\int (x+1)^n \,\mathrm{d}x$ 的封闭形式而只会暴力展开一项项积分。

不过同样众所周知，ckr很擅长枚举和找规律，于是他发现 $\int (x+1)^n \,\mathrm{d}x = \dfrac{(x+1)^{n+1}}{n+1} + C$。

更进一步，他发现 $\int (x+t)^n \,\mathrm{d}x = \dfrac{(x+t)^{n+1}}{n+1} + C$，解决了这一整类的积分。

于是他想要寻找更一般的规律，开始研究 $\int (x+s)^n (x+t)^m \,\mathrm{d}x$，由于 ckr 很喜欢平方数，所以他决定先研究 n, m 是平方数的情况。

当然，由于不定积分比较麻烦，所以你只需要输出定积分：$\int_0^{x0} (x+s)^n (x+t)^m \,\mathrm{d}x$，同时 n, m 也不可能太大，所以 ckr 要求 $1 \le n, m \le N$。

由于 ckr 现在还处于探索阶段，每一个数据都是很重要的，所以你需要对每个 $1 \le n, m \le N$ 且 n, m 是完全平方数求出答案。

由于答案可能很大，你只需要求出答案关于 2147483647 取模后的值。

输入格式

输入一行四个整数 N, s, t, x_0。

输出格式

输出 $\lfloor \sqrt{N} \rfloor$ 行，每行 $\lfloor \sqrt{N} \rfloor$ 个整数，其中第 i 行第 j 个整数表示 $n = i^2, m = j^2$ 的答案。

样例一

input

```
4 3 7 1
```

output

```
1431655791 1932746596
930577433 1950302041
```

explanation

分数形式的答案为：$\begin{bmatrix} \frac{79}{3} & \frac{113137}{10} \\ \frac{35579}{30} & \frac{164819873}{315} \end{bmatrix}$。

样例二

input

```
9 432626436 222345443 0
```

output

```
0 0 0
0 0 0
0 0 0
```

样例三

input

```
9 233333 233333 666666
```

output

```
1703229151 2113117123 1350295746
2113117123 1384164355 1123817829
1350295746 1123817829 456733368
```

限制与约定

对于 10% 的测试点，保证 $1 \le N \le 10$。

对于 30% 的测试点，保证 $1 \le N \le 100$。

对于 60% 的测试点，保证 $1 \le N \le 3000$。

另有 10% 的测试点，保证 $s = t$。

对于 100% 的测试点，保证 $0 \le s, t, x_0 < 2147483647, 1 \le N \le 10^5$。

时间限制：$1\mathrm{s}$

空间限制：$512\mathrm{MB}$

Ckr 教你学数学

你**可能**会用到以下公式

1. 多项式定积分的计算方法

   对多项式 $f(x) = \sum_{i=0}^{n} c_i x^i$，定积分 $\int_a^b f(x) \mathrm{d}x$ 可以用如下公式计算：

$$\int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{n} \frac{(b^{i+1} - a^{i+1})c_i}{i+1}$$

2. 换元积分法 (https://en.wikipedia.org/wiki/Integration_by_substitution)

   设 $I \subseteq \mathrm{R}$ 为一个区间，$\varphi : [a,b] \to I$ 是一个导数可积的函数。设 $f : I \to \mathrm{R}$ 是一个连续函数，则：

$$\int_{\varphi(a)}^{\varphi(b)} f(u)\,\mathrm{d}u = \int_a^b f(\varphi(x))\varphi'(x)\,\mathrm{d}x.$$

3. 分部积分法 (https://en.wikipedia.org/wiki/Integration_by_parts)

   设 $u = u(x), \mathrm{d}u = u'(x)\mathrm{d}x$，与 $v = v(x), \mathrm{d}v = v'(x)\mathrm{d}x$，则:

$$\int_a^b u(x)v'(x)\,\mathrm{d}x = [u(x)v(x)]_a^b - \int_a^b u'(x)v(x)\,\mathrm{d}x = u(b)v(b) - u(a)v(a) - \int_a^b u'(x)v(x)\,\mathrm{d}x$$

温馨提示

在本题中，如果你希望获得全部的分数，你可能考虑由于常数的影响。只使用加法、减法和乘法运算不会有太大的常数，但请谨慎使用除法。

1. 跨越比较大的 2 的幂的数组寻址会产生较大的常数。
2. 过多的分支可以带来相当大的常数。

当然，如果你的算法在数学上是正确的，但没有考虑常数的影响，可能仍然可以获得一部分的分数。

蜡烛

ckr 最近在研究数论，他找了一个质数 p，研究模 p 的完全剩余系 $\{0, 1, 2, \cdots, p-1\}$。

ckr 发现这 p 个剩余类中有一些很优美，例如 0 在作加法和 1 在作乘法的时候就很优美；同时也有些剩余类很麻烦，如 0 在求逆元的时候就很麻烦。

这样他就有了两个集合 A, B，其中 A 中的是优美的剩余类，B 中的是麻烦的剩余类。同时还有一些可能不属于 A, B 的剩余类，他们不优美也不麻烦。

这里的 A, B 是根据具体处理的问题而决定的，所以每个剩余类在同一时刻**至多属于其中一个集合**，例如处理加法时 $0 \in A$ 而处理除法时 $0 \in B$。

由于今天是 rkc 的 2019 岁生日，ckr 需要给 rkc 准备 2019 个生日蜡烛，这些蜡烛将会由数形成。由于 ckr 在研究模 p 的完全剩余系，所以同一个剩余类中的蜡烛会被看作是一样的。由于一个剩余类中有无穷多个数，所以每种蜡烛都可以选择任意多次。由于 ckr 不希望这 2019 个蜡烛放在一起太难看，所以他会让这 2019 个蜡烛对应的数的**和是 p 的倍数**。由于 rkc 很懒，你需要帮他按顺序排好这些蜡烛，而这个顺序是可以自己选择的。

由于 rkc 不喜欢单调，所以这 2019 个蜡烛需要同时存在 A 和 B 中的剩余类的数。同时由于 rkc 也会有一些评判，所以那些不优美也不麻烦的数在他眼中也会变成 A, B 中的一个。

现在 ckr 想让你帮他估计一下有多少种不同的选择蜡烛的方式。两种选法不同当且仅当存在一个位置 i 使得两种选法中第 i 个位置的数不属于同一个剩余类。而估计中由于你不清楚 rkc 的选择方法所以你可以将剩下的数中的每一个**等概率**的分进一个集合，求出**期望下**会有多少种方案，模 2147483647 输出。

输入格式

第一行一个质数 p，表示 ckr 研究的质数。

第二行一个长度为 p 的字符串，只包含 `A`， `B`， `?` 三种字符，其中第 i 个字符为 `A` 表示 $i-1 \in A$，为 `B` 表示 $i-1 \in B$，为 `?` 表示 $i-1 \notin A, B$。

输出格式

输出期望下的方案数模 2147483647 的值。

样例一

input

```
2
AB
```

output

```
7
```

explanation

容易看出 B 中需要偶数个但不是 0 个，从而可以从 B 中选择 $2,4,6,\cdots,2018$ 个，从而共有 $\binom{2019}{2}+\binom{2019}{4}+\cdots+\binom{2019}{2018}=2^{2018}-1$ 种方案，模意义下为 7。由于 A 和 B 已经包含了所有数，所以这就是答案。

样例二

input

```
2
??
```

output

```
1073741827
```

当 $0,1$ 在同一个集合中时有 0 种方案，不同时有 $2^{2018}-1$ 种方案，期望下有 $\frac{2^{2018}-1}{2}$ 种方案，模意义下为 1073741827。

限制与约定

对于 10% 的测试点，保证 $p=2$。

对于 30% 的测试点，保证 $2\le p\le 20$。

对于 50% 的测试点，保证 $2\le p\le 100$。

对于 70% 的测试点，保证 $2\le p\le 10000$。

对于 100% 的测试点，保证 $2\le p\le 10^5$，p 为质数。

时间限制：1s

空间限制：512MB

总蒜挤核

小C、小K和小R在玩游戏。

小C在平面上画了一些红点，小K在平面上画了一些蓝点。

由于小C和小K关系不好，所以红点和蓝点是分离的。具体地说，红点都严格位于 y 轴右侧，而蓝点都严格位于 y 轴左侧。

小R想要画通过这些点画一些直线来分割红点和蓝点，具体地说，是画一条通过两个点（两个点都可以是红点或蓝点）的直线，使得这条直线的一侧严格没有红点，而另一侧严格没有蓝点。

小R想知道他能画几条**位置不同**的直线。

由于小C和小K十分的活泼，所以可能会突然画一些新的点。而在画完之后你需要给出结果。

由于小C和小K都十分喜欢新的事物，所以他们画的所有点的横坐标互不相同。

输入格式

第一行一个整数 n，表示有 n 次画点的操作。

接下来 n 行，每行两个整数 x, y，表示画的点的坐标，你可以通过点的位置判断这个点的颜色。

输出格式

输出 n 行，每行一个整数表示画完点后的答案。

样例一

input

```
0
```

output

```
```

样例二

input

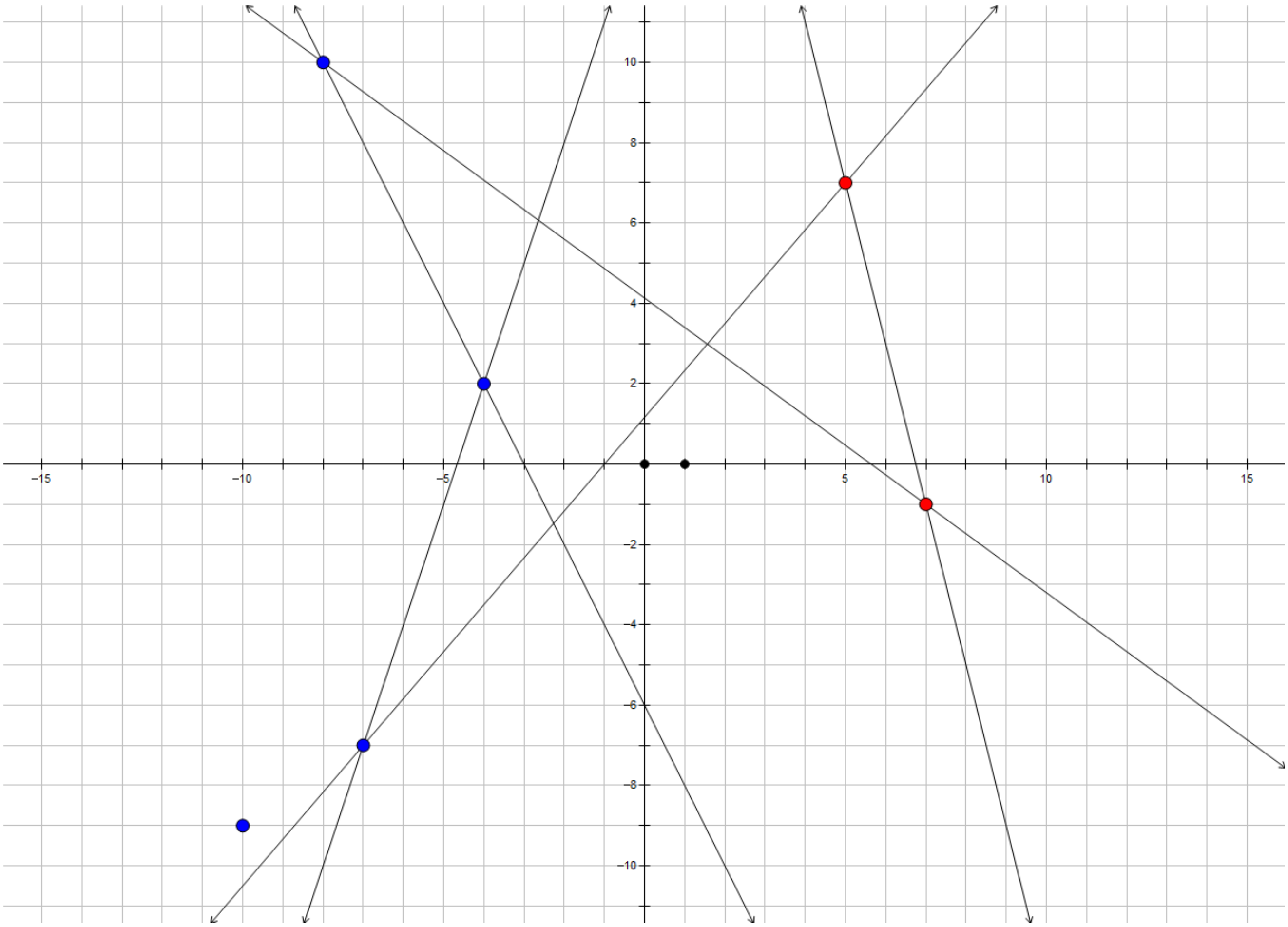
```
10
-7 -7
5 7
-8 10
-4 2
-10 -9
7 -1
-2 8
3 -10
1 -10
-5 1
```

output

```
0
1
3
4
4
5
4
4
4
4
```

explanation

画了六个点的图如下所示：



样例三

input

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

output

```
0
1
1
```

explanation

注意这三个点两两连线都是同一条直线，所以只算一个。

限制与约定

对于 20% 的测试点，保证 $n \le 100$。

对于 40% 的测试点，保证 $n \le 3000$。

另有 30% 的测试点，保证 $x > 0$。

对于 100% 的数据，保证 $0 \le n \le 300000, -10^9 \le x, y \le 10^9, x \neq 0$ 且所有的 x 互不相同。

时间限制：$\texttt{1s}$

空间限制：$\texttt{512MB}$