

A 奶龙的数学课

题目描述

小七最近送奶龙去上数学课，奶龙在课上遇到了一道数学题，把他给难住了。题目为给定两个八进制的正整数 $L, R (L \leq R)$ ，请统计在 $[L, R]$ 区间内为 7 的倍数的数字出现的总次数（结果用八进制数表示）。奶龙一直到下课也没算出来，但他已经饿的无法思考，他于是他向你撒娇卖萌(' ')，希望你能帮他解决。



Hint

在E3的A题，我们已经知道读入十六进制数可以使用 `%x`，即 `scanf("%x", &n);`。此外，本题要求输出结果为八进制，同样可以使用 `%o`，即 `printf("%o\n", ans)`。

输入

一行，两个八进制的正整数 $L, R (L \leq R)$ ，中间用空格隔开

输出

一行一个八进制数，表示 $[L, R]$ 区间内为 7 的倍数的数字出现的总次数。

输入样例

```
17 143
```

输出样例

```
14
```

数据范围

保证本题所有数据都在int型范围内。

B 函数

题目描述

已知 $f(x) = f(x - 4) + f(x - 1)$ (x 为整数)

$f(1) = 1, f(2) = 1, f(3) = 2, f(4) = 3$

试计算 $f(n)$ 。因为 $f(n)$ 可能很大，请输出 $f(n)$ 对 $1e9 + 7$ 取模的结果。

在计算时，你可以先对计算过程中的数取模，这样并不会影响最后取模的结果。

例如，在对 Mod 取模时，有 $f(x) \% Mod = (f(x - 4) + f(x - 1)) \% Mod$ ，或者也可以使用 $f(x - 4) \% Mod + f(x - 1) \% Mod$ 计算，在输出时取余。

这样，我们就能避免计算过程中的量超过我们定义的可表示范围。

输入

一个整数 n

输出

一个整数，表示 $f(n)$ 对 $1e9 + 7$ 取模的结果

样例

输入#1

1

输出#1

1

输入#2

153

输出#2

485301345

数据范围

对于20%的数据， $1 \leq n \leq 15$

对于60%的数据， $1 \leq n \leq 200$

对于100%的数据， $1 \leq n \leq 10000$

author: void

C 朝田诗乃的压枪训练

题目描述

朝田诗乃是一个热爱各类型FPS游戏的女孩子。

这天，她接触到了一款以室内近距离战斗为主题的FPS游戏。游戏中有一把名为R4-C的自动步枪，这把枪的后坐力较低，容易压枪。但由于朝田诗乃之前一直使用的是狙击武器，适应自动武器需要一定的训练，于是她找来了你评估她的压枪训练成果。

具体地，朝田诗乃会在靶场中向圆形靶标连续射击 n 枪，你需要读取这 n 个命中点相对于靶标中心的位置，给出这 n 枪每一枪的评价，并在最后给出这一轮训练的最好成绩和平均值。

评价标准：

定义距离为命中点相较于中心方位的横纵坐标中，两个值的绝对值之间的较大者，例如(5,8)的距离为8

对于每组坐标输出一行

- 命中点与中心重合时，输出 `Perfect`
- 命中点与中心不重合，但距离不超过3的时候，输出 `Excellent`
- 命中点距离中心的距离大于 3 但不超过 5 的时候，输出 `Good`
- 命中点距中心的距离大于 5 但不超过 10 的时候，输出 `Not Bad`
- 命中点距中心的距离大于 10 的时候，输出 `oh!`

输入

第一个数为数据组数 n

接下来 n 行，每行 2 个整数 a, b ，分别为命中点的横纵坐标

输出

共有 $n + 1$ 行

对前 n 行，每行均为对应的评价

最后一行有 2 个数字，分别为：这一轮训练的最好成绩（即最小距离）、距离的平均值（只需输出小数点后两位）

输入样例

```
4
0 0
1 2
2 5
0 10
```

输出样例

```
Perfect
Excellent
Good
Not Bad
0 4.25
```

##数据范围

$n \leq 30000$ ，输入数据保证 a 、 b 以及求平均值过程中产生的加和值在 `int` 范围内

Author: 压不住R4-C的Academic Waste

D 眼镜的复数计算器

题目描述

在眼镜哥学程序设计的那个年代，还没有浮点计算器这个东西，他们只能白手起家，撸起袖子加油干，一步一步地造出浮点计算器。

如果你不知道复数是什么或者乘除法怎么算：[\(参考百度百科\) 复数定义、相关概念以及运算法则](#)

我们把形如 $z = a + bi$ (a 、 b 均为实数) 的数称为复数。其中规定 i 为虚数单位，且 $i^2 = -1$ ， a 称为实部， b 称为虚部。

运算法则

🔊 播报 ✎ 编辑

加法法则

复数的加法法则：设 $z_1=a+bi$ ， $z_2=c+di$ 是任意两个复数。两者和的实部是原来两个复数实部的和，它的虚部是原来两个虚部的和。两个复数的和依然是复数。

$$\text{即 } (a + bi) \pm (c + di) = (a \pm c) + (b \pm d)i。$$

乘法法则

复数的乘法法则：把两个复数相乘，类似两个多项式相乘，结果中 $i^2=-1$ ，把实部与虚部分别合并。两个复数的积仍然是一个复数。

$$\text{即 } (a + bi)(c + di) = (ac - bd) + (bc + ad)i。$$

除法法则

复数除法定义：满足 $(c + di)(x + yi) = (a + bi)$ 的复数 $x + yi(x, y \in \mathbf{R})$ 叫复数 $a+bi$ 除以复数 $c+di$ 的商。

运算方法：将分子和分母同时乘以分母的共轭复数，再用乘法法则运算，

$$\text{即 } \frac{a + bi}{c + di} = \frac{(a + bi)(c - di)}{(c + di)(c - di)} = \frac{(ac + bd) + (bc - ad)i}{c^2 + d^2} \quad (\text{分母实数化})。$$

现在你们将重走这一段过程，设计一个含加减乘除的简易复数计算器

输入

第一个数为数据组数 n

接下来 n 行，每行输入如下所示

Re1 Im1 op Re2 Im2

式中 Re1，Im1 分别表示第一个复数的实部和虚部，确保为整数。

op 表示运算符，为 +, -, *, / 中之一。

为了读入方便，使用 1, 2, 3, 4 分别代表 +, -, *, /

Re2, Im2 分别表示第二个复数的实部和虚部，确保为整数。

输入的 Re1 Im1 op Re2 Im2 均用空格分隔开

输出

对于每组数据，输出一行，**复数计算结果的值**，实部和虚部均四舍五入保留至小数点后第二位。

`i` 表示虚数单位 $\sqrt{-1}$ 。

输入样例

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

输出样例

```
2.00+3.00i
0.00+1.00i
0.00+9.00i
0.00+0.50i
```

数据范围

确保加减乘运算的结果在 `int` 范围内。

确保除法运算结果在 `double` 范围内，且不会出现除 0 情况（**做其他题需要关注这一点ha**）。

数据组数不多于 10 组。

HINT

```
printf("%d", x);
```

可将一个数带上符号输出

题目来源于眼镜哥曾经遇到的练习题

Author:Op zb

E 打音游

题目背景

bluebean 最近在打音游。他记下了自己的连击情况，想直接计算出自己的分数，你能帮帮他吗？

题目描述

每次成功的击打将会使 *Combo*（连击数）加一。其中成功的击打分为 *Perfect* 和 *Great* 两种情况。若击打失败则会得到一个 *Miss*，并且 *Combo* 数归零。

在输入中，使用 `P`，`G`，`M` 分别代表 *Perfect*、*Great*、*Miss* 三种情况。

分数计算方法：对于每次击打，收获的分值 = 基础分值 × 判定系数 × 连击倍率

其中：基础分值统一为 200

判定系数与击打情况相关：

Perfect : 1.0

Great : 0.5

Miss : 0

连击倍率与 *Combo* 数有关：

1 ~ 9 *Combo*: 1.0

10 ~ 19 *Combo*: 1.1

20 ~ 29 *Combo*: 1.2

30 ~ 39 *Combo*: 1.3

40 ~ 49 *Combo*: 1.4

50 *Combo* 及以上: 1.5

注：在击打成功时会先将 *Combo* 数加一再计算对应的倍率。

最终得分为所有单次击打得分的累加和。

特别地，如果所有击打都是成功的（即没有出现 *Miss*），请额外输出一行 `Full Combo!`

输入格式

一行字符串，代表击打情况。

输出格式

一个整数，代表最终得分。

如果所有击打都是成功的，额外输出一行 `Full Combo!`

输入样例1

```
PPPGGG
```

输出样例1

```
900
Full Combo!
```

输入样例2

```
PPPGGGMP
```

输出样例2

```
1100
```

数据范围

对于 30% 的数据，输入的字符串长度不超过 30

对于 100% 的数据，输入的字符串长度不超过 1000

Author : bluebean

F Cirno 的完美位运算教室

题目背景

寺子屋的教学并不能满足 Cirno 的求知欲，她找到了擅长计算的 Yakumo Ran 学习数学。众所周知，Ran 是 Yakumo Yukari 的式神，而在幻想乡式神就是电脑，所以 Ran 教给了 Cirno 一些位运算的相关知识。

题目描述

Ran 告诉了 Cirno 一个正整数 x ，让她找到一个满足下列条件的最小正整数 y ：

- $x \text{ and } y > 0$
- $x \text{ xor } y > 0$

此处 and 表示按位与， xor 表示按位异或。

输入

多组数据输入。

第一行一个整数 T ，表示数据组数。

接下来 T 行，每行一个正整数 x ，含义见上。

输出

每组数据一行，一个正整数 y ，表示答案。

样例

输入

```
6
75
36
15
78
15
34
```

输出

```
1
4
1
2
1
2
```

数据范围

$1 \leq T \leq 10^3, 1 \leq x \leq 2^{30}。$

G 魔理沙的行窃预兆-I

题目描述

雾雨魔理沙今天也一如既往地来到伏瓦鲁魔法图书馆进行『行窃预兆』，窥视着图书馆丰富的魔法书资源。

但馆主帕秋莉·诺蕾姬早有准备，在通往图书馆的路径上布下了繁杂的传送魔法阵，这些魔法阵会限制魔理沙的行动。

4	1	1	4	3	2	5	1	5	4
---	---	---	---	---	---	---	---	---	---

通向图书馆的路径是一条由正整数组成的长链，如上图。每个格子上的数字表示魔理沙从此格出发行走一步**最远**能够向右移动几格，例如 4 代表魔理沙从这一格出发一步只能向右移动一格、两格、三格或者四格。

现在，魔理沙位于路径**最左侧**的一格，目的地是位于**最右一格**的图书馆入口。

魔理沙是个聪明的魔法使，因此她能够以步数**最少**的方法前进，请你计算以此法前进的魔理沙的行走步数。

输入

输入为两行

第一行，为路径的长度 n

接下来一行，为 n 个由空格分开的正整数，表示路径上从左到右的每一个数字。

输出

一个整数，为魔理沙到达图书馆所需要的最少步数

样例

输入样例

```
10
4 1 1 4 3 2 5 1 5 4
```

输出样例

```
3
```

样例说明



如图，这是步数最少的路径之一，其长度为 3

数据范围

- 对于 62.5% 的测试点， $2 \leq n \leq 10$
- 对于 100% 的测试点， $2 \leq n \leq 1000$
- 路径上的数字不超过 10

Hint

将宏大的，繁杂的目的拆分成多个简单的，容易思考的步骤，或许能提供一些思路。

譬如说，不去直接思考到达终点的最短路线，而先考虑最后一步：我到达终点的最后一步有几种选择？分别需要多少总步数？

Author:星辰的微光

H 希卡式打孔纸带·alter

题目描述

希卡族的科技已经失传了很久。在复原遗迹中的技术时，公主复原出了古老的程序编辑器的beta版。这种编辑器可以从初始值 1 开始，编辑一个三十二位二进制数。编辑器支持两种操作：**左移位**（一位）和**按位取反**。作为公主最忠诚的随从，你要找到编辑得到某个三十二位二进制数（补码）的最少步骤。

输入

多组输入。每行一个 `int` 范围内的整数，代表编辑目标。不超过一万组输入。

输出

对于每组数据，输出一个整数，代表最少编辑次数。某个数无法通过编辑得到时，输出 -1。

输入样例

```
3
-2
```

输出样例

```
3
1
```

author: Yt

I 狼了个狼（2）

题目描述

狼族的天敌危机被化解后，狼族又遭遇了食物分配难题。作为狼巫师的你临危受命，需要合理运用魔法策略来平均地分配食物。

现在要将 n 堆肉分给狼族中的 n 个家庭，每堆肉的质量（kg）分别为 m_1, m_2, \dots, m_n 。每次操作你可以召唤一个替身并给定一个整数阈值 tv ，替身会拿走每堆肉质量**大于** tv 的部分。但由于你的法力值有限，替身每次能拿走肉的总质量不超过 M （即高于阈值部分的总质量不能超过 M ）。

不认真学习的Uanu只能再次寻求你的帮助，请你帮他算一算在保证平均分配（即每堆肉质量相等）前提下每个家庭最多能分到多重的肉和达到平均所需的最少操作次数。

输入格式

两行输入。

第一行两个整数 n 和 M ，表示家庭数（堆数）和能拿走肉的最大质量 M ，空格隔开。

第二行 n 个整数 m_1, m_2, \dots, m_n ，表示每堆肉的质量，空格隔开。

输出格式

一行输出。

一行两个整数，表示每个家庭最多能分到多重的肉和达到平均所需的最少操作次数。

输入样例

```
4 8
1 3 5 4
```

输出样例

```
1 2
```

样例解释

每个家庭最多能分到 1kg 的肉，最少需要两次操作：第一次操作阈值为 2，拿走 6kg 的肉；第二次操作阈值为 1，拿走 3kg 的肉。

数据范围

$$1 \leq n \leq 10^6, n \leq M \leq 2 \times 10^9, 1 \leq m_i \leq 10^6$$

author: Uanu

国庆快乐!!!

J 伦蒂尼姆的守卫

题目描述

在遥远的维多利亚，有一座从未被攻陷的城池——伦蒂尼姆。在伦蒂尼姆的环形城墙上，有着 n 座威力强大的城防炮，编号顺时针方向依次为 1 到 n ，两两间距相等，这正是伦蒂尼姆的防御核心。

为了守卫城池， n 位萨卡兹士兵每天在城墙上巡视。他们在每个整点到达一座炮塔，并以一定的方向（顺时针或逆时针）向另一座炮塔前进，每小时的距离恰好等于炮塔的间距。若他们在前进途中遇到另一位萨卡兹士兵，则会调转他们的方向。当且仅当萨卡兹两两相遇时，他们会改变方向

而现在，Kal'tsit 指挥罗德岛来到了伦蒂尼姆的附近，并告诉了你每座炮塔处萨卡兹当前整点的前进方向，以及第 k 位萨卡兹 Hoederer 是我方卧底的消息。作为 Doctor，你需要知晓 t 小时后，Hoederer 在哪座城防炮塔处，以帮助制订潜入策略。为了便于表示，Kal'tsit 将当前整点位于第 n 座炮塔的萨卡兹编号为 n 。

输入格式

共 2 行

第一行三个整数 n 、 k 、 t ，分别代表伦蒂尼姆炮塔数、Hoederer 的编号与经过的时间。

第二行 n 个整数（0或1），第 i 个整数代表编号为 i 的萨卡兹当前整点的前进方向，1代表顺时针，0代表逆时针。

输出格式

一行一个整数，代表赫德雷所在的塔楼编号。

样例输入

```
5 2 8
0 1 0 1 1
```

样例输出

```
3
```

数据范围

对于20%的数据, $1 \leq k \leq n \leq 10, 1 \leq t \leq 100$

对于90%的数据, $1 \leq k \leq n \leq 10^4, 1 \leq t \leq 10^6$

对于100%的数据, $1 \leq k \leq n \leq 10^5, 1 \leq t \leq 10^{16}$

同时, 对于30%的数据, $\frac{t}{n^2}$ 是一个整数

Aurthor : Red