

## 1. 购物 (P1658)

### 【问题描述】

你就要去购物了，现在你手上有  $N$  种不同面值的硬币，每种硬币有无限多个。为了方便购物，你希望带尽量少的硬币，但要能组合出 1 到  $X$  之间的任意值。

### 【输入格式】

第一行两个数  $X$ 、 $N$ ，以下  $N$  个数，表示每种硬币的面值。

### 【输出格式】

最少需要携带的硬币个数，如果无解输出 -1。

### 【样例输入 1】

```
20 4
1 2 5 10
```

### 【样例输出 1】

```
5
```

### 【补充】

对于 30% 的数据，满足  $N \leq 3$ ， $X \leq 20$ ；

对于 100% 的数据，满足  $N \leq 10$ ， $X \leq 1000$ 。

## 2. 最大乘积 (P1249)

### 【问题描述】

一个正整数一般可以分为几个互不相同的自然数的和，如  $3=1+2$ ， $4=1+3$ ， $5=1+4=2+3$ ， $6=1+5=2+4$ ， $\dots$ 。

现在你的任务是将指定的正整数  $n$  分解成若干个互不相同的自然数的和，且使这些自然数的乘积最大。

### 【输入格式】

只一个正整数  $n$ ，( $3 \leq n \leq 10000$ )。

### 【输出格式】

第一行是分解方案，相邻的数之间用一个空格分开，并且按由小到大的顺序。

第二行是最大的乘积。

### 【样例输入 1】

```
10
```

### 【样例输出 1】

```
2 3 5
30
```

## 3. 次大值

### 【问题描述】

Alice 有  $n$  个正整数，数字从  $1 \sim n$  编号，分别为  $a_1, a_2, \dots, a_n$ 。Bob 刚学习取模运算，于是便拿这  $n$  个数进行练习，他写下了所有  $a_i \bmod a_j$  ( $1 \leq i, j \leq n$ ， $i \neq j$ ) 的值，其中  $\bmod$  表示取模运算。Alice 想知道所有的结果中，严格次大值是多少。将取模后得到的所有值进行去重，即相同的结果数值只保留一个，剩余数中第二大的值就称为严格次大值。

### 【输入格式】

第一行一个正整数  $n$ ，表示数字个数。第二行  $n$  个正整数表示  $a_i$ 。

### 【输出格式】

仅一行一个整数表示答案。若取模结果去重后剩余数字不足两个，则输出 -1。

### 【样例输入】

```
4
4 5 5 6
```

【样例输出】

```
4
```

【样例输入】

```
4
1 1 1 1
```

【样例输出】

```
-1
```

【样例输入】

```
7
12 3 8 5 7 20 15
```

【样例输出】

```
12
```

【样例说明】

【样例 1 解释】 所有取模的结果为 {4, 4, 4, 1, 0, 5, 1, 0, 5, 2, 1, 1}。去重后有：{0, 1, 2, 4, 5}，结果为 4。

【数据说明】

对于 40% 的数据， $1 \leq n, a_i \leq 100$ ;

对于 70% 的数据， $1 \leq n \leq 3000, 1 \leq a_i \leq 10^5$ ;

对于 100% 的数据， $3 \leq n \leq 2 \times 10^5, 1 \leq a_i \leq 10^9$ 。

## 4. 文件压缩 (P1124)

【题目背景】

提高文件的压缩率一直是人们追求的目标。近几年有人提出了这样一种算法，它虽然只是单纯地对文件进行重排，本身并不压缩文件，但是经这种算法调整后的文件在大多数情况下都能获得比原来更大的压缩率。

【问题描述】

该算法具体如下：对一个长度为  $n$  的字符串  $S$ ，首先根据它构造  $n$  个字符串，其中第  $i$  个字符串由将  $S$  的前  $i-1$  个字符置于末尾得到。然后把这  $n$  个字符串按照首字符从小到大排序，如果两个字符串的首字符相等，则按照它们在  $S$  中的位置从小到大排序。排序后的字符串的尾字符可以组成一个新的字符串  $S'$ ，它的长度也是  $n$ ，并且包含了  $S$  中的每一个字符。最后输出  $S'$  以及  $S$  的首字符在  $S'$  中的位置  $p$ 。举例：

S:example

1、构造  $n$  个字符串

example

xamplee

ampleex

mpleexa

pleexam

leexamp

eexampl

2、将字符串排序

ampleex

example

eexampl

```
leexamp
```

```
mpleexa
```

```
pleexam
```

```
xamplee
```

3、压缩结果

```
xelpame S'
```

```
7 p
```

由于英语单词构造的特殊性，某些字母对出现的频率很高，因此在  $S'$  中相同的字母有很大几率排在一起，从而提高  $S'$  的压缩率。虽然这种算法利用了英语单词的特性，然而在实践的过程中，人们发现它几乎适用于所有的文件压缩。

请你编一个程序，读入  $S'$  和  $p$ ，输出字符串  $S$ 。

#### 【输入格式】

输入文件共有三行，第 1 行是一个整数  $n$  ( $1 \leq n \leq 10000$ )，代表  $S'$  的长度，第 2 行是字符串  $S'$ ，第 3 行是整数  $p$ 。

#### 【输出格式】

仅包含一行  $S$ 。

#### 【样例输入】

```
7
xelpame
7
```

#### 【样例输出】

```
example
```

## 5. Vanya and Computer Game (CF492D)

#### 【问题描述】

Vanya 和他的朋友 Vana 正在玩一种电脑游戏，他们需要干掉  $n$  个怪物去过关，Vanya 的角色每秒攻击  $x$  次，Vana 的角色每秒攻击  $y$  次，而且每个角色抬起武器需要一次攻击的时间（也就是说 Vanya 第一次对怪物造成伤害是在  $1/x$  秒，且之后每  $1/x$  秒都会对怪物造成一次伤害），第  $i$  个怪物有  $a_i$  滴血，表示被攻击  $a_i$  次后就死了。

他们想知道对于每个怪物谁给了最后一击（如果同时给了最后一击，就认为他们都是最后一击）

#### 【输入】

第一行  $n, x, y$  如题意 接下来  $n$  行，第  $i+1$  行表示  $a_i$

#### 【输出】

每个怪物输出一行是 Vanya 还是 Vana 给了最后一击，或者输出 both 表示他们都给了最后一击。

#### 【样例输入 1】

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

#### 【样例输出 1】

```
Vanya
Vana
Vanya
Both
```

#### 【样例输入 2】

```
2 1 1
1
```

2

**【样例输出 2】**

Both

Both

**【数据规模】**

$$1 \leq n \leq 1e5, 1 \leq x, y \leq 1e6$$

## 6. 覆盖问题 (BZOJ1052)

**【问题描述】**

某人在山上种了  $N$  棵小树苗。冬天来了，温度急速下降，小树苗脆弱得不堪一击，于是树主人想用一些塑料薄膜把这些小树遮盖起来，经过一番长久的思考，他决定用 3 个  $L \times L$  的正方形塑料薄膜将小树遮起来。我们不妨将山建立一个平面直角坐标系，设第  $i$  棵小树的坐标为  $(X_i, Y_i)$ ，3 个  $L \times L$  的正方形的边要求平行与坐标轴，一个点如果在正方形的边界上，也算作被覆盖。当然，我们希望塑料薄膜面积越小越好，即求  $L$  最小值。

**【输入】**

第一行有一个正整数  $N$ ，表示有多少棵树。接下来有  $N$  行，第  $i+1$  行有 2 个整数  $X_i, Y_i$ ，表示第  $i$  棵树的坐标，保证不会有 2 个树的坐标相同。。

**【输出】**

一行，输出最小的  $L$  值。

**【样例输入】**

4  
0 1  
0 -1  
1 0  
-1 0

**【样例输出】**

1

**【数据规模】**

100%的数据， $N \leq 20000$