

### T1. 二进制

文件名	输入文件	输出文件	时间限制	空间限制
binary.cpp	binary.in	binary.out	1s	256MB

二进制（binary）是以 2 为基数代表系统的二进位制，通常用两个不同的符号 0（代表零）和 1（代表一）来表示。8 位二进制所能表示的无符号整数范围为 0~255。

现从键盘输入 8 位二进制字符串，统计其中有多少个字符 1

样例 1:

输入:

00010100

输出:

2

样例 2:

输入:

00000000

输出:

0

## T2. ISBN 号码 (isbn.pas/c/cpp)

文件名	输入文件	输出文件	时间限制	空间限制
isbn.cpp	isbn.in	isbn.out	1s	256MB

每一本正式出版的图书都有一个 ISBN 号码与之对应，ISBN 码包括 9 位数字、1 位识别码和 3 位分隔符，其规定格式如 “x-xxx-xxxxx-x”，其中符号 “-” 是分隔符（键盘上的减号），最后一位是识别码，例如 0-670-82162-4 就是一个标准的 ISBN 码。ISBN 码的首位数字表示书籍的出版语言，例如 0 代表英语；第一个分隔符 “-” 之后的三位数字代表出版社，例如 670 代表维京出版社；第二个分隔符之后的五位数字代表该书在出版社的编号；最后一位为识别码。

识别码的计算方法如下：

首位数字乘以 1 加上次位数字乘以 2……以此类推，用所得的结果 mod 11，所得的余数即为识别码，如果余数为 10，则识别码为大写字母 X。例如 ISBN 号码 0-670-82162-4 中的识别码 4 是这样得到的：对 067082162 这 9 个数字，从左至右，分别乘以 1, 2, …, 9，再求和，即  $0 \times 1 + 6 \times 2 + \dots + 2 \times 9 = 158$ ，然后取  $158 \bmod 11$  的结果 4 作为识别码。

你的任务是编写程序判断输入的 ISBN 号码中识别码是否正确，如果正确，则仅输出 “Right”；如果错误，则输出你认为是正确的 ISBN 号码。

输入：

输入文件 isbn.in 只有一行，是一个字符序列，表示一本书的 ISBN 号码（保证输入符合 ISBN 号码的格式要求）。

输出：

输出文件 isbn.out 共一行，假如输入的 ISBN 号码的识别码正确，那么输出 “Right”，否则，按照规定的格式，输出正确的 ISBN 号码（包括分隔符 “-”）。

样例 1：

输入：

0-670-82162-4

输出：

Right

样例 2：

输入：

0-670-82162-0

输出：

0-670-82162-4

### T3. 第二大的数字

文件名	输入文件	输出文件	时间限制	空间限制
number.cpp	number.in	number.out	1s	256MB

有  $n$  个正整数，分别为  $a_1、a_2、a_3 \cdots \cdots a_n$

现在进行取模运算： $a_i \bmod a_j (1 \leq i, j \leq n \text{ 且 } i \neq j)$ ，得到一组新的数字。对新的数字进行去重操作，得到的第二大的数字即为结果。

**输入：**

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：**

若取模结果去重后剩余数字不足两个，则输出  $-1$ ，否则输出第二大的数字。

样例 1：

输入：

4

4 5 5 6

输出：

4

样例 2：

输入：

4

2 2 2 2

输出：

-1

数据范围：

40%数据	$1 \leq a_i, n \leq 100$
70%数据	$1 \leq n \leq 3000, 1 \leq a_i \leq 10^5$
100 数据	$3 \leq n \leq 2 * 10^5, 1 \leq a_i \leq 10^9$

#### T4. 钱币

文件名	输入文件	输出文件	时间限制	空间限制
coin.cpp	coin.in	coin.out	1s	256MB

##### 题目描述：

在森林深处的动物王国，流通着这样的钱币系统： $n$  种不同的钱币，每种钱币面额为  $a[i]$ ，每种钱币的数量是无限的。猴子国王想要在一年一度的王国运动会上为优秀运动选手发放奖金，他希望能够计算出最少的钱币数量，如果目前的气币系统无法拼凑出奖金的数目，则用钱币发放部分奖金，剩下的由国王开出余额支票，余额支票面额尽量最小。例如，如果钱币系统是 40，30，20，那么 50 元的最少钱币数量是 2，35 元则无法拼凑出，为了使得余额支票面额尽量最小，发放 30 元的奖金。

##### 输入数据：

第一行， $n$  和  $t$ ， $n$  是钱币系统中的钱币种类； $t$  为奖金。

第二行， $n$  个数据，为每种钱币的面额，不大于 65535。

##### 输出数据：

如果奖金  $t$  能被钱币系统中的钱币拼凑出，请输出最少的钱币数量。如果不能，则输出最小的余额支票的面额。

##### 样例 1：

输入：

```
3 50
40 30 20
```

输出：

```
2
```

##### 样例 2：

输入：

```
3 35
40 30 20
```

输出：

```
5
```

##### 数据范围

对于 20% 的数据， $n \leq 10$ ， $t \leq 200$ 。

对于 50% 的数据， $n \leq 40$ ， $t \leq 10^4$ 。

对于 100% 的数据， $n \leq 50$ ， $t \leq 10^5$ 。