

1 比较版本号

题目描述

给定两个版本字符串`version1`和`version2`，比较两个版本的大小，
如果`version1 > version2`， 则返回1；
如果`version1 < version2`， 则返回-1；
如果`version1 = version2`， 则返回0。
`version`非空，只包含数字和字符'.'

输入格式

共2行输入，第一行为`version1`，第二行为`version2`。

输出格式

共1行输出，包含一个整数，表示比较结果。

示例 1

输入

0.1

1.1

输出

-1

示例 2

输入

1.0.1

1

输出

1

2 纸箱收纳

题目描述

小多有 N 个空纸箱，每个纸箱是一个边长为 A_i 的正方体。第 i 个纸箱可以装进第 j 个纸箱里当且仅当以下 3 个条件同时满足：

1. 纸箱 j 中没有装其他纸箱
2. 纸箱 i 没有被装到其他纸箱中
3. 纸箱 i 的边长小于纸箱 j ，即 $A_i < A_j$ 。

现在小多希望将一些纸箱装进其他纸箱中，使得肉眼可见的纸箱个数最少(被装到纸箱里的纸箱肉眼不可见)。

输入格式

第一行包含一个正整数 N ，表示纸箱个数。

第二行包含 N 个正整数 A_i ，分别表示每个纸箱的边长。

数据范围：

$$1 \leq N \leq 100000$$

$$1 \leq A_i \leq 1000000000$$

输出格式

输出一行，包含一个整数，表示肉眼可见的最少纸箱个数。

示例 1

输入

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

输出

```
2
```

3 打乱字符串树

题目描述

给定一个字符串 s ，可以将它表示成二叉树结构，比如下面就是 `order` 的一种二叉树形式。

```
    order
   /    \
  or     der
 /  \   /  \
o   r d er
      /  \
      e   r
```

针对这个二叉树，我们可以任意交换一个节点的子节点来重新变化字符串的形式，将 `order` 变为 `roder`。

```
    roder
   /    \
  ro     der
 /  \   /  \
r   o d er
      /  \
      e   r
```

如果再继续交换，可以得到 `rored`。

```
    rored
   /    \
  ro     red
 /  \   /  \
r   o re d
      /  \
      r   e
```

给定两个字符串 s_1 和 s_2 ，判断 s_1 能否通过上面提到的交换子节点的方式得到 s_2 。

输入格式

共2行输入

第一行为字符串s1，

第二行为字符串s2。

输出格式

共1行输出，整数，1表示可以，0表示不可以。

示例 1

输入

order

rored

输出

1

示例 2

输入

order

darre

输出

0

4 小熊与狼

题目描述

N 匹狼要给小熊利马一次考验。每匹狼选择一个大于等于 0 小于等于 $2^{30}-1$ 的数字，并向其他所有狼展示这个数字。没有狼会直接把自己的数字告诉利马。有的狼会完全保持沉默，有的会给利马一些信息。一匹狼会告诉利马的唯一信息是其它所有狼的数字进行位运算 `xor` 后的结果。

利马的目标是找出所有狼选出数字的总和，如果有多种可能，找出最小的那个。

给出长度为 N 的 `int` 数组 `x`，`x[i]` 表示第 `i` 匹狼提供的信息，没有给出信息用 -1 表示。

求最小的可能值之和，如果不存在这种可能，输出 -1。

输入格式

第一行是一个长度为 N 的数字数组 `x`

$2 \leq N \leq 40, -1 \leq x[i] \leq 2^{30} - 1$

输出格式

输出一个数字表示最小和。若不可能则输出 -1。

示例 1

输入

{1, 4, 13, -1}

输出

21

示例 2

输入

{0,0,7}

输出

-1