

标题：最大括号深度 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【最大括号深度】现有一字符串仅由 '(', ')', '[', ']', '{', '}' 六种括号组成。

若字符串满足以下条件之一，则为无效字符串：

①任一类型的左右括号数量不相等；

②存在未按正确顺序（先左后右）闭合的括号。

输出括号的最大嵌套深度，若字符串无效则输出0。

0≤字符串长度≤100000

输入描述：

一个只包括 '(', ')', '[', ']', '{', '}' 的字符串

输出描述：

一个整数，最大的括号深度

示例1：

输入

[]

输出

1

标题：最长连续子序列 | 时间限制：2秒 | 内存限制：102400K | 语言限制：不限

【最长连续子序列】

有N个正整数组成的一个序列。给定整数sum，求长度最长的连续子序列，使他们的和等于sum，返回此子序列的长度，如果没有满足要求的序列，返回-1。



输入描述：

序列：1,2,3,4,2
sum：6

输出描述：

序列长度：3

备注：

输入序列仅由数字和英文逗号构成，数字之间采用英文逗号分隔；
序列长度：1 ≤ N ≤ 200；
输入序列不考虑异常情况，由题目保证输入序列满足要求。

示例1：

输入

1,2,3,4,2
6

输出

3

标题：寻找相同子串 | 时间限制：1秒 | 内存限制：65536K | 语言限制：不限

【寻找相同子串】给你两个字符串 t 和 p，要求从 t 中找到一个和 p 相同的连续子串，并输出该子串第一个字符的下标。

输入描述：

输入文件包括两行，分别表示字符串 t 和 p，保证 t 的长度不小于 p，且 t 的长度不超过1000000，p 的长度不超过10000。

输出描述：

如果能从 t 中找到一个和 p 相等的连续子串，则输出该子串第一个字符在t中的下标（下标从左到右依次为1,2,3,...）；如果不能则输出"No"；如果含有多个这样的子串，则输出第一个字符下标最小的。

示例1：

输入

AVERDXIVYERDIAN
RDXI

输出

4

标题：滑动窗口最大和 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【滑动窗口最大和】有一个N个整数的数组，和一个长度为M的窗口，窗口从数组内的第一个数开始滑动直到窗口不能滑动为止，每次窗口滑动产生一个窗口和（窗口内所有数的和），求窗口滑动产生的所有窗口和的最大值。

输入描述：

第一行输入一个正整数N，表示整数个数。(0<N<100000)
第二行输入N个整数，整数的取值范围为[-100,100]。
第三行输入一个正整数M，M代表窗口大小，M<=100000，且M<=N。

输出描述：

窗口滑动产生的所有窗口和的最大值。

示例1：

输入

6
10 20 30 15 23 12
3

输出

68

标题：最长子字符串的长度（一） | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【最长子字符串的长度（一）】给你一个字符串 s，字符串s首尾相连成一个环形，请在环中找出 'o' 字符出现了偶数次最长子字符串的长度。

输入描述：

输入是一串小写字母组成的字符串

输出描述：

输出是一个整数

备注：

$1 \leq s.length \leq 5 \times 10^5$
s 只包含小写英文字母。

示例1：

输入

alolobo

输出

6

标题：字符串变换最小字符串 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【字符串变换最小字符串】

给定一个字符串s，最多只能进行一次变换，返回变换后能得到的最小字符串（按照字典序进行比较）。

变换规则：交换字符串中任意两个不同位置的字符。

输入描述：

一串小写字母组成的字符串s

输出描述：

按照要求进行变换得到的最小字符串

备注：

s是都是小写字符组成
 $1 \leq s.length \leq 1000$

示例1：

输入

abcdef

输出

abcdef

标题：整数对最小和 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【整数对最小和】

给定两个整数数组array1、array2，数组元素按升序排列。假设从array1、array2中分别取出一个元素可构成一对元素，现在需要取出k对元素，并对取出的所有元素求和，计算和的最小值

注意：两对元素如果对应于array1、array2中的两个下标均相同，则视为同一对元素。

输入描述：

```
输入两行数组array1、array2，每行首个数字为数组大小size(0 < size <= 100);
0 < array1[i] <= 1000
0 < array2[i] <= 1000
接下来一行为正整数k
0 < k <= array1.size() * array2.size()
```

输出描述：

```
满足要求的最小和
```

示例1：

输入

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

输出

```
4
```

标题：幻方修复1 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【幻方修复1】幻方 (Magic Square) 是一个由1~N²共N²个整数构成的N*N矩阵，满足每行、列和对角线上的数字和都相等。

老师在上课的时候讲了幻方的概念和例子，好学的小明将它们抄在了小本本上，但是不小心抄错了一个数字。

你可以帮小明把这个抄错的数字找出来并且改正吗？

输入描述：

```
第一行输入一个整数N，代表待检验幻方的阶数 ( 3 ≤ N < 50 )
接下来的N行，每行N个整数，空格隔开 ( 0 ≤ 每个整数 ≤ N2 )
```

输出描述：

```
输出空格隔开的三个整数，分别是：出错行号、出错列号、应填入的数字 ( 末尾无空格 )
```

示例1：

输入

```
3
8 1 6
3 5 7
4 0 2
```

输出

```
3 2 9
```

【流浪地球】

流浪地球计划在赤道上都部署了N个转向发动机，按位置顺序编号为0~N。

- 1) 初始状态下所有的发动机都是未启动状态；
- 2) 发动机启动的方式分为“手动启动”和“关联启动”两种方式；
- 3) 如果在时刻1一个发动机被启动，下一个时刻2与之相邻的两个发动机就会被“关联启动”；
- 4) 如果准备启动某个发动机时，它已经被启动了，则什么都不用做；
- 5) 发动机0与发动机N-1是相邻的；

地球联合政府准备挑选某些发动机在某些时刻进行“手动启动”。当然最终所有的发动机都会被启动。哪些发动机最晚被启动呢？

输入描述：

第一行两个数字N和E，中间有空格
N代表部署发动机的总个数，E代表计划手动启动的发动机总个数
 $1 < N \leq 1000, 1 \leq E \leq 1000, E \leq N$
接下来共E行，每行都是两个数字T和P，中间有空格
T代表发动机的启动时刻，P代表此发动机的位置编号。
 $0 \leq T \leq N, 0 \leq P < N$

输出描述：

第一行一个数字N，以回车结束
N代表最后被启动的发动机个数
第二行N个数字，中间有空格，以回车结束
每个数字代表发动机的位置编号，从小到大排序

示例1：

输入

8 2
0 2
0 6

输出

2
0 4

标题：判断字符串是否有处理价值 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【判断字符串是否有处理价值】

小明的领导交给他一组文本字符串，需要他初步筛选出有意义的字符串，以便进一步分析。

已知字符串中会随机出现3组符号<,>,[,],{,}，文本字符串中如果每组符号能够左右匹配，则有进一步分析的价值，否则丢弃；分隔符需要左右匹配，并可以多重嵌套，例如：<[{}>,{<>},[]<>均为分隔符匹配的字符串；{<<],[<<均为分隔符不匹配的串
配对返回配对的符号数，不完全配对返回0。

输入描述：

包含<,>,[,],{,}的字符串
例如：<abc>ss[012]{vv}

输出描述：

配对返回配对的符号数，不完全配对返回0。

示例1：

输入

ss[012]{vv}

输出

3

【最长连续子序列】



有N个正整数组成的一个序列。给定整数sum，求长度最长的连续子序列，使他们的和等于sum，返回此子序列的长度，如果没有满足要求的序列，返回-1。

输入描述：

```
序列：1,2,3,4,2
sum：6
```

输出描述：

```
序列长度：3
```

备注：

```
输入序列仅由数字和英文逗号构成，数字之间采用英文逗号分隔；
序列长度：1 <= N <= 200；
输入序列不考虑异常情况，由题目保证输入序列满足要求。
```

示例1：

输入

```
1,2,3,4,2
6
```

输出

```
3
```

标题：用连续自然数之和来表达整数 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限
【用连续自然数之和来表达整数】一个整数可以由连续的自然数之和来表示。给定一个整数，计算该整数有几种连续自然数之和的表达式，且打印出每种表达式。

输入描述：

```
一个目标整数T (1 <= T <= 1000)
```

输出描述：

```
该整数的所有表达式和表达式的个数。如果有多种表达式，输出要求为：
1.自然数个数最少的表达式优先输出
2.每个表达式中按自然数递增的顺序输出，具体的格式参见样例。在每个测试数据结束时，输出一行"Result:X"，其中X是最终的表达式个数。
```

示例1：

输入

```
9
```

输出

```
9=9
9=4+5
9=2+3+4
Result:3
```

标题：相对开音节 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【相对开音节】

相对开音节构成的结构为辅音+元音 (aeiou) +辅音(r除外)+e，常见的单词有bike、cake等。

给定一个字符串，以空格为分隔符，反转每个单词中的字母，若单词中包含如数字等其他非字母时不进行反转。

反转后计算其中含有相对开音节结构的子串个数（连续子串中部分字符可以重复）。

输入描述：

字符串，以空格分割的多个单词，字符串长度<10000，字母只考虑小写

输出描述：

含有相对开音节结构的子串个数，注：个数<10000

示例1：

输入

ekam a ekac

输出

2

他的代码：

空

标题：正则表达式替换 | 时间限制：1秒 | 内存限制：32768K | 语言限制：[C, Python, C++, Javascript, Python 3, Java, Go]

【正则表达式替换】

为了便于业务交互，约定一个对输入的字符串中的下划线做统一替换

具体要求：对于给定的输入字符串，将其中包含的每一个下划线“_”，使用特殊字符串“^\$[,]”替换，并输出替换后的结果

其中的例外场景不做替换，场景如下：

- 1、在一对方括号之内的下划线则不替换
- 2、为转义的下划线，即“_”则不替换

输入描述：

输出描述：

备注：

- 1、调用者已经保证，输入表达式的方括号会成对出现，并且方括号会存在嵌套，即存在“[xx[yyyy]xxx]”这种形式的字符串
- 2、输入字符串的长度范围：[0, 10000]

示例1：

输入

“^_9494”

输出

“^((^\$[,]9494)”

标题：找朋友 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【找朋友】

在学校中，N个小朋友站成一队，第i个小朋友的身高为height[i]，

第i个小朋友可以看到的第一个比自己身高更高的小朋友j，那么j是i的好朋友(要求j > i)。

请重新生成一个列表，对应位置的输出是每个小朋友的好朋友位置，如果没有看到好朋友，请在该位置用0代替。

小朋友人数范围是 [0, 40000]。

输入描述：

第一行输入N，N表示有N个小朋友
第二行输入N个小朋友的身高height[i]，都是整数

输出描述：

输出N个小朋友的好朋友的位置

示例1：

输入

2
100 95

输出

0 0

标题：最大矩阵和 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【最大矩阵和】给定一个二维整数矩阵，要在这个矩阵中选出一个子矩阵，使得这个子矩阵内所有的数字和尽量大，我们把这个子矩阵称为和最大子矩阵，子矩阵的选取原则是原矩阵中一块相互连续的矩形区域。

输入描述：

输入的第一行包含2个整数n, m(1 <= n, m <= 10)，表示一个n行m列的矩阵，下面有n行，每行有m个整数，同一行中，每2个数字之间有1个空格，最后一个数字后

面没有空格，所有的数字的在[-1000, 1000]之间。

输出描述：

输出一行一个数字，表示选出的和最大子矩阵内所有的数字和。

示例1：

输入

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

输出

20

标题：按身高和体重排队 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【按身高和体重排队】

某学校举行运动会，学生们按编号(1、2、3...n)进行标识，现需要按照身高由低到高排列，对身高相同的人，按体重由轻到重排列；对于身高体重都相同的人，维持原有的编号顺序关系。请输出排列后的学生编号。

输入描述：

两个序列，每个序列由n个正整数组成 (0 < n <= 100)。第一个序列中的数值代表身高，第二个序列中的数值代表体重。

输出描述：

排列结果，每个数值都是原始序列中的学生编号，编号从1开始

示例1：

输入

```
4
100 100 120 130
40 30 60 50
```

输出

```
2 1 3 4
```

标题：最大矩阵和 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【最大矩阵和】给定一个二维整数矩阵，要在这个矩阵中选出一个子矩阵，使得这个子矩阵内所有的数字和尽量大，我们把这个子矩阵称为和最大子矩阵，子矩阵的选取原则是原矩阵中一块相互连续的矩形区域。

输入描述：

输入的第一行包含2个整数n, m (1 <= n, m <= 10)，表示一个n行m列的矩阵，下面有n行，每行有m个整数，同一行中，每2个数字之间有1个空格，最后一个数字后面没有空格，所有的数字的在[-1000, 1000]之间。

输出描述：

输出一行一个数字，表示选出的和最大子矩阵内所有的数字和。

示例1：

输入

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

输出

```
20
```

标题：数据分类 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限
【数据分类】

对一个数据a进行分类，分类方法为：此数据a（四个字节大小）的四个字节相加对一个给定的值b取模，如果得到的结果小于一个给定的值c，则数据a为有效类型，其类型为取模的值；如果得到的结果大于或者等于c，则数据a为无效类型。

比如一个数据a=0x01010101，b=3，按照分类方法计算（0x01+0x01+0x01+0x01）%3=1，所以如果c=2，则此a为有效类型，其类型为1，如果c=1，则此a为无效类型；

又比如一个数据a=0x01010103，b=3，按照分类方法计算（0x01+0x01+0x01+0x03）%3=0，所以如果c=2，则此a为有效类型，其类型为0，如果c=0，则此a为无效类型。

输入12个数据，第一个数据为c，第二个数据为b，剩余10个数据为需要分类的数据，请找到有效类型中包含数据最多的类型，并输出该类型含有多少个数据。

输入描述：

输入12个数据，用空格分隔，第一个数据为c，第二个数据为b，剩余10个数据为需要分类的数据。

输出描述：

输出最多数据的有效类型有多少个数据。

示例1：

输入

3 4 256 257 258 259 260 261 262 263 264 265

输出

3

标题：简单的数学游戏 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【简单的数学游戏】有一个正整数数组，每次从该数组中取出两个最大的数字做减法（如果两个数字不相等，用大数减去小数，保证相减后的结果大于等于0），如果相减的结果不为0，则将相减后的结果数字放回数组，然后再从数组中取两个最大的数字做减法，不断循环重复。如果最后数组中只剩一个数字，请给出该数字的值，如果没有数字剩下给出0值即可。

输入描述：

输入的第一行为数组的长度；第二行为数组元素值，用空格分隔。

输出描述：

输出一个整数，表示数组中最后数字的值。

示例1：

输入

3
1 2 3

输出

0

标题：回文素数 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限
【回文素数】

给定一个整数N，生成不大于N的所有回文素数，第一个数字是素数的个数，之后是生成的素数，素数按从小到大的顺序排列。如果没有生成任何素数，则输出0。
如果一个数从左往右读与从右往左读是一样的，那么这个数是回文数，例如101是回文数。

输入描述：

给定一个整数。

输出描述：

输出不大于该整数的所有素数。

示例1：

输入

50

输出

5,2,3,5,7,11

标题：靠谱的车 | 时间限制：1秒 | 内存限制：262144K | 语言限制：不限

【靠谱的车】

程序员小明打了一辆出租车去上班。出于职业敏感，他注意到这辆出租车的计费表有点问题，总是偏大。
出租车司机解释说他不喜欢数字4，所以改装了计费表，任何数字位置遇到数字4就直接跳过，其余功能都正常。
比如：

- 1. 23再多一块钱就变为25；
- 2. 39再多一块钱变为50；
- 3. 399再多一块钱变为500；

小明识破了司机的伎俩，准备利用自己的学识打败司机的阴谋。
给出计费表的表面读数，返回实际产生的费用。

输入描述：

只有一行，数字N，表示里程表的读数。
(1<=N<=888888888)。

输出描述：

一个数字，表示实际产生的费用。以回车结束。

示例1：

输入

5

输出

4