

重庆市育才中学2022年 联合模拟

2022.3.13

所有题均需文件输入输出

题目名称	三角形数	扫雷	爬山	括号序列	序列谜题
输入输出文件名	num.in/out	boom.in/out	climb.in/out	bracket.in/out	seq.in/out
时间限制	1.0 秒	2.0 秒	1.0 秒	1.0 秒	1.0 秒
空间限制	512 MB	512 MB	512 MB	512 MB	512 MB
测试点数目	----	----	----	----	----

某些题时限较大，知道自己过不了的数据请 return 0，加速评测进程

三角形数 (num)

古希腊珂学家把 1, 3, 6, 10 等数称为三角形数, 因为这些数量的石头可以排成正三角形, 如下图.



一般地, 如果一个正整数 x 能表示为 $1 + 2 + \dots + m$ 的形式, 其中 m 为正整数, 则称 x 为三角形数.

给定 n 个正整数, 判断每个数是否为三角形数.

【输入格式】

输入第一行为一个正整数 n , 表示数的个数.

接下来 n 行, 每行一个正整数 x , 表示一个待判断的数.

【输出格式】

输出 n 行, 分别对应每个待判断的数是否为三角形数, 如果是则输出 `Yes`, 否则输出 `No`.

【样例 1】

```
3
1
7
36
```

```
Yes
No
Yes
```

【数据范围】

对于 20% 的数据, $n \leq 10$, 每个待判断的数 $x \leq 50$;

对于 60% 的数据, $n \leq 100$, 每个待判断的数 $x \leq 10^8$;

对于 100% 的数据, $n \leq 10^4$, 每个待判断的数 $x \leq 10^{18}$.

扫雷 (boom)



你需要编写一个扫雷交互器，获取地图信息，依次读入玩家操作并返回对应结果。

扫雷的局面是一个 $n \times m$ 的矩形，其中一些位置为地雷而另一些位置为空地，扫雷局面将以字符矩阵的形式输入。将第 i 行第 j 列的位置记作 $\langle i, j \rangle$ 。特别地，令 k 为地雷的数量，保证有 $0 < k < n \times m$ 。

一开始玩家无法得知除了 n, m, k 之外的扫雷局面的任何信息。

你需要维护名为**玩家地图**的字符矩阵，初始时矩阵中所有元素为 `_`（下划线）。

玩家将进行 q 次操作，每次将选取一个位置 $\langle x, y \rangle$ ，并用以下三种方式之一点击（若**游戏结束**，你应该忽略游戏结束后的所有玩家操作，即判定这些操作为**无效操作**，反之即为**有效操作**）：

1. 左键点击：

若 $\langle x, y \rangle$ 已经**被打开**或**玩家地图**中这个位置为 `P`（`P` 表示旗子），则不进行任何操作；否则若 $\langle x, y \rangle$ 为地雷，则**游戏失败**；否则对 $\langle x, y \rangle$ 进行**打开操作**。

2. 右键点击：

若 $\langle x, y \rangle$ 已经**被打开**，则不进行任何操作；否则

若**玩家地图**中位置 $\langle x, y \rangle$ 为 `_`，将其改为 `P`，

若**玩家地图**中位置 $\langle x, y \rangle$ 为 `P`，将其改为 `?`，

若**玩家地图**中位置 $\langle x, y \rangle$ 为 `?`，将其改为 `_`。

3. 中键点击：

若 $\langle x, y \rangle$ 未被**打开**，或**玩家地图**中这个位置周围相邻的 8 个位置的 `P` 的个数不等于**玩家地图**中该位置的数值，则不进行任何操作；否则对于 $\langle x, y \rangle$ 周围相邻 8 个未被**打开**且在**玩家地图**中不是 `P` 的位置，如果存在至少一个位置是地雷，则**游戏失败**；否则对这些位置进行**打开操作**。

打开操作：

对位置 $\langle x, y \rangle$ 进行的**打开操作**按照如下方式进行：

1. 将 $\langle x, y \rangle$ 标记为**被打开**。
2. 在**玩家地图**中位置 $\langle x, y \rangle$ 改为 $c(0 \leq c \leq 8)$, 表示这个位置周围相邻的 8 个位置的地雷数量。
3. 如果 $c = 0$, 则对其周围相邻的 8 个未被**被打开**且在**玩家地图**中不是 **P** 的位置进行**打开操作**。
4. 这个操作是递归进行的, 直到所有子操作都结束后, 本次**打开操作**才算结束。

游戏结束, 游戏结束有以下三种情况:

1. **游戏失败**: 即上述规则中的情况, 试图对地雷的位置进行打开操作就会导致**游戏失败**。
2. **游戏胜利**: 若某一次操作结束后, 未被打开的位置个数恰好为 k , 则此次操作后**游戏胜利**。
3. **退出游戏**: 若玩家操作结束, 但上述两种情况均未出现, 则视作玩家**退出游戏**。

在每一次操作后, 你需要返回结果, 具体规则如下:

• 若此次操作为**无效操作**, 返回 `INVALID`; 否则

• 若此次操作后**游戏失败**, 返回 `LOSE`; 否则

• 先返回 `RUNNING:`, 然后在同一行返回用中括号包含的, **玩家地图**中有更改的位置以及更改后的值, 格式为 `<x, y, val>`; 更改的位置按照 x 坐标为第一关键字从小到大, y 坐标为第二关键字从小到大的顺序排序, 相邻两个更改的位置用逗号隔开。例如某一次操作后可能返回: `RUNNING: [<2, 2, 2>, <2, 3, 1>, <3, 2, 1>, <3, 3, 0>]` 或 `RUNNING: []` 或 `RUNNING: [<3, 3, ?>]`。**注意其中空格的位置。**

• 若此次操作后**游戏胜利**, 再在新的一行返回 `WIN`。

• 若此次操作后**退出游戏**, 再在新的一行返回 `QUIT`。

【输入格式】

本题输入文件包含多组数据。

第一行一个正整数 T 表示数据组数, 接下来每 $n + q + 2$ 行 (意义见下) 表示一组数据。

每组数据第一行两个正整数 n, m 分别表示扫雷局面的高度和宽度。

每组数据接下来 n 行, 第 i 行一个长度为 m 的字符串, 仅包含 `_` 和 `*` 两种字符。如果第 j 个字符为 `*`, 则表示第 i 行第 j 列为地雷, 否则为空地。

每组数据接下来若干行, 每行三个正整数 op, x, y 表示玩家的一次操作,

具体操作见题目描述。

每组数据最后一行一个数 0, 表示玩家操作结束。令玩家操作次数为 q 。

【输出格式】

对于每组数据, 输出若干行, 每行表示一次操作的返回结果, 若在某一次操作后**游戏结束**, 请输出对应的结果。

相邻的两组数据之间使用一行 `=====` (10 个 `=` 字符, 不包含引号) 隔开。

【样例 1】

见选手目录下的 minesweeper/minesweeper1.in 与 minesweeper/minesweeper1.ans。

【样例 1 说明】

以下是对样例一五组数据的解释，每组数据从左到右依次描述了扫雷局面以及每一次有效操作后玩家矩阵：

第一组数据：

```

_ _ * | _ _ _ | 0 1 _
_ _ _ | _ _ _ | 0 1 1
_ _ _ | _ _ _ | 0 0 0

```

第二组数据：

```

_ * _ | _ _ _ | _ _ _ | _ _ 2 | _ P 2 | _ P 2 | 1 P 2
_ _ * | _ _ _ | 1 2 _ | 1 2 _ | 1 2 _ | 1 2 P | 1 2 P
_ _ _ | _ _ _ | 0 1 _ | 0 1 _ | 0 1 _ | 0 1 _ | 0 1 1

```

第三组数据：

```

* * * | _ _ _ | * * *
_ _ _ | _ _ _ | _ 6 _
* * * | _ _ _ | * * *

```

第四组数据（玩家矩阵中不包含 x 字符，最后一个矩阵仅作说明作用，不是真实的玩家矩阵）：

```

_ _ _ | _ _ _ | _ _ _
_ * _ | _ _ _ | _ x _
_ _ _ | _ _ _ | _ _ _

```

第五组数据（玩家矩阵中不包含 x 字符，最后一个矩阵仅作说明作用，不是真实的玩家矩阵）：

```

_ _ _ | _ _ _ | _ _ _ | _ P _ | 0 P 1
_ _ * | _ _ _ | _ 1 _ | _ 1 _ | 0 1 x
_ _ _ | _ _ _ | _ _ _ | _ _ _ | 0 1 1



```

【样例 2】

见选手目录下的 minesweeper/minesweeper2.in 与 minesweeper/minesweeper2.ans。

【测试点约定】

对于所有测试点：

$1 \leq T \leq 30, 3 \leq n, m \leq 200, 0 \leq q \leq 10000, op \in \{1, 2, 3\}, 1 \leq x \leq n, 1 \leq y \leq m$ ，保证输入的字符串仅包含  和  两种字符。

测试点编号	n	m	k	特殊限制
1	$= 9$	$= 9$	$= 10$	使用真实游戏生成
2	$= 16$	$= 16$	$= 40$	使用真实游戏生成
3	$= 16$	$= 30$	$= 99$	使用真实游戏生成
4	≤ 40	≤ 40	≤ 320	使用真实游戏生成
5 ~ 8	≤ 200	≤ 200	≤ 10000	$op = 1$
9 ~ 12	≤ 200	≤ 200	≤ 10000	$op \in \{1, 2\}$
13 ~ 16	≤ 200	≤ 200	≤ 10000	$op \in \{1, 3\}$
17 ~ 20	≤ 200	≤ 200	≤ 10000	无

爬山 (climb)

对于 n 座山峰，他们的高度为 a_1, a_2, \dots, a_n ，若存在 m 个整数 i_1, i_2, \dots, i_m ，满足 $1 \leq i_1 < i_2 < \dots < i_m$ 且 $a_{i_1} < a_{i_2} < \dots < a_{i_m}$ ，则称 i_1, i_2, \dots, i_m 是该山脉的一个**长度为 m 的升峰**。该山脉的**长度最大的升峰**称为该山脉的**最长升峰**。一个山脉可能有多个最长升峰，但最长升峰的长度是唯一的。

现在然然要去爬山捏， n 座山峰的高度为 a_1, a_2, \dots, a_n 。然然的目光高远，爬的山必须构成升峰，并且他会让升峰的长度尽可能大。

山神被然然感动了，于是给了他 k 次操作机会。具体的，每次操作然然可以任意选择两个整数 l, r ，满足 $1 \leq l \leq r \leq n$ ，然后对所有满足 $l \leq i \leq r$ 的山峰 i ，将 a_i 的值改为 $a_i + 1$ 。然然想让该山脉的最长升峰长度最大，请你帮帮然然吧。

【输入格式】

输入第一行两个由空格分隔的整数 n 和 k ，分别表示山峰座数和可进行的操作次数。

第二行 n 个由空格分隔的正整数 a_1, a_2, \dots, a_n

【输出格式】

输出仅一行一个整数，表示最后该数列的最长升峰长度的最大值。

【样例 1】

```
6 3
5 1 4 5 8 8
```

```
5
```

以下增加操作的顺序可使得最后的最长升峰长度最大：

- (1) 选择 $l = 3, r = 4$ 增加，山峰高度变为 $5, 1, 5, 6, 8, 8$;
- (2) 选择 $l = 3, r = 4$ 增加，山峰高度变为 $5, 1, 6, 7, 8, 8$;
- (3) 选择 $l = 6, r = 6$ 增加，山峰高度变为 $5, 1, 6, 7, 8, 9$ 。

这样最长升峰为第 1, 3, 4, 5, 6 座山组成的。

【数据范围】

本题共有 20 个测试点，各测试点的规模如下表所示。

测试点编号	n	k	a
1	$n = 5$	$k = 1$	$1 \leq a_i \leq 15$
2	$n = 6$	$k = 3$	
3	$n = 9$		
4	$n = 15$	$k = 15$	
5	$n = 16$	$0 \leq k \leq 100$	$1 \leq a_i \leq 500$
6	$n = 20$		
7	$n = 25$		
8	$n = 100$	$k = 0$	$1 \leq a_i \leq 10^9$
9	$n = 1000$		
10	$n = 100$	$0 \leq k \leq 10^6$	$1 \leq a_i \leq 10^6$
11			
12		$0 \leq k \leq 10^8$	$1 \leq a_i \leq 10^8$
13			
14	$n = 1000$	$0 \leq k \leq 10^6$	$1 \leq a_i \leq 10^6$
15		$0 \leq k \leq 10^8$	$1 \leq a_i \leq 10^8$
16			
17		$0 \leq k \leq 10^9$	$1 \leq a_i \leq 10^9$
18			
19			
20			

括号序列 (bracket)

给定一个仅包含左括号和右括号的序列，可以对该序列执行以下两种操作：

- 1、将序列的最后一项移动到首位
- 2、在序列任意位置添加一个左括号或者右括号

你的任务是使用上述两种操作将给定的括号序列变成一个合法的括号序列，并且使得括号序列的长度最短，如果有多个长度最短的方案，那么输出字典序最小的那一个（左括号的字典序比右括号小）

【输入格式】

一行一个字符串，表示括号序列。

【输出格式】

按照题目要求输出修复后的括号序列。

【样例 1】

```
()()
```

```
(())
```

【样例 2】

```
()(
```

```
(())
```

【数据范围】

对于前 20% 的数据，输入括号序列长度 ≤ 10

对于前 50% 的数据，输入括号序列长度 ≤ 1000

对于另外 20% 的数据，输入序列中左右括号的个数相等

对于 100% 的数据，输入序列的括号长度 ≤ 500000

序列谜题 (seq)

Diana 很聪明，擅长解决谜题。

有一天，Diana 在解决一个序列上的谜题，他得到了正整数序列 $a_1 \dots a_n$ 以及一个常数 c 。

这道谜题需要确定一个 **非负整数序列** $x_1 \dots x_n$ ，并使得这个序列与 a 的排斥值最小，排斥值的定义为 $(\sum x_i^2) + c \times (\sum_{1 \leq i \leq n} |a_i - a_{i-1} + x_i - x_{i-1}|)$ 。

Diana 还思考今晚吃什么，请你教教Diana 吧。

【输入格式】

第一行两个整数 n, c 。

第二行 n 个整数表示 $a_1 \dots a_n$ 。

【输出格式】

一行一个整数表示最小排斥值。

【样例 1】

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

```
4
```

【样例 2】

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

```
15
```

对于样例 #1, $x = \{0, 1, 0, 0\}$, 排斥值为 4。

【数据范围】

对于 20% 的数据, $1 \leq n, c \leq 10$;

另有 30% 的数据, $1 \leq n \leq 10^3, 1 \leq a_i, c \leq 100$;

另有 20% 的数据, $1 \leq n \leq 10^5, 1 \leq a_i, c \leq 100$;

对于 100% 的数据, $1 \leq n \leq 10^5, 1 \leq a_i, c \leq 10^6$ 。