

NOIP 模拟赛 Day1

By tsy

题目名称	游戏	轰炸	统计
题目类型	传统型	传统型	传统型
目录	game	bomb	count
可执行文件名	game	bomb	count
输入文件名	game.in	bomb.in	count.in
输出文件名	game.out	bomb.out	count.out
每个测试点时限	1s	1s	3s
内存限制	512 MB	512 MB	512 MB
测试点数目	10	10	20
每个测试点分值	10	10	5

提交源程序文件名

对于C++ 语言	game.cpp	bomb.cpp	count.cpp
对于C 语言	game.c	bomb.c	count.c
对于Pascal 语言	game.pas	bomb.pas	count.pas

编译选项

对于C++ 语言	-lm	-lm	-lm
对于C 语言	-lm	-lm	-lm
对于Pascal 语言			

注意事项:

除非特殊说明,结果比较方式均为忽略行末空格及文末回车的全文比较。

C/C++中函数main()的返回值类型必须是int,程序正常结束时的返回值必须是0。

每道题有三个样例,请注意查看simple文件夹。

评测在Windows 10下进行,评测软件为Lemon。

编译时不打开任何优化选项。

游戏 (game)

【题目描述】

A 和 B 在玩游戏。桌子上有 n 个格子，其中一些格子放有棋子 ‘x’，空格子用 ‘o’ 表示。A 和 B 轮流进行以下操作中的任何一种。

- 1) 当某一个棋子右边有空格时，将该棋子向右移动一位。

ooxoo \rightarrow oooxo

- 2) 当某一个棋子右边恰好是连续的两个棋子时，将该棋子直接跳过连续的两个棋子。

ooxxxoo \rightarrow oooxxxo

当棋子到达终点(最右端的格子)时，棋子将消失。当某一方不能移动时，这方输。A 先走。请问 A 是否必胜？

【输入格式】

多组测试数据。

对于每一组测试数据：

第一行输入一个整数 n ，表示格子数量。

第二行输入长度为 n 的 ‘x’ ‘o’ 串。

【输出格式】

对于每一组测试数据，输出一行 “YES” 或者 “NO”。

【样例输入】

```
4
ooxo
4
xxxo
5
oxxxo
```

【样例输出】

```
YES
NO
NO
```

【数据范围】

对于 30% 的数据， $n \leq 20$

对于 100% 的数据， $n \leq 10^6$ ，测试数据组数 $T \leq 5$ ，最后一个格子为空。

轰炸 (bomb)

【题目描述】

A 国和 B 国开战。B 国国王痴迷于树，所以 B 国的城市道路是一棵树的形状。A 国情报局通过 B 国地下组织获取了该消息……给力的地下党还窃取了 B 国每一个城市的炸药储备量 $m[i]$ ，即轰炸城市 i 可以同时炸掉距离城市 i 在 $m[i]$ 之内的其他城市。

由于 A 国国力有限，A 国情报局局长想知道至少需要炸几次才能把 B 国所有城市都炸掉……

【输入格式】

第一行输入一个整数 n ，表示 B 国的城市数量。

第二行输入 n 个整数，第 i 个整数 $m[i]$ 表示城市 i 的炸药储备量。

接下来的 $n-1$ 行，每一行输入两个整数 u, v ，表示一条连接城市 u, v 长度为 1 的城市道路。

【输出格式】

输出一行一个整数，即最少轰炸次数。

【样例输入】

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

【样例输出】

```
2
```

【数据范围】

对于 10% 的数据， $n \leq 10$ 。

对于 30% 的数据， $n \leq 1000$ 。

对于 100% 的数据， $n \leq 10^5, m[i] \leq 100$ 。

统计 (count)

【题目描述】

定义一个整数集合 S , S 中的元素满足在十进制表示下只有 5 和 7, 例如 5、7、57、75。

对于一个有 n 个元素的数列 $\{a\}$ 进行以下两种操作:

操作一: `add l r v`, 表示对于数列中加上 v 。

操作二: `query l r`, 表示询问数列中有多少个 $i \in [l, r]$ 满足 $a_i \in S$ 。

【输入格式】

第一行输入两个整数 n 、 m , 分别表示数列长度和操作次数。

第二行输入 n 个整数, 表示这个数列。

接下来的 m 行, 每一行输入一个操作, 操作格式如上。

【输出格式】

对于每一个询问操作输出一行一个整数。

【样例输入】

```
10 5
1 5 7 2 5 75 8 7 9 57
query 2 7
add 1 5 2
query 1 5
add 3 8 -1
query 1 10
```

【样例输出】

```
4
2
3
```

【数据范围】

测试点编号	n	m	约定
1	= 100	= 100	无
2	= 9987	= 98706	
3	= 9987	= 10^5	保证修改操作只有 100 次
4	$\leq 10^5$	≤ 1000	保证询问操作只有 100 次
5			无
6	$\leq 10^5$	$\leq 10^5$	保证修改操作 ≤ 1000 次
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			无

对于所有数据保证任意时刻的 $a_i \in [1,10000]$ 。

【提示】

注意常数因子对效率的影响。