

大连二十四中NOIP2015模拟赛Day1

saffah

2015 年 8 月 6 日

中文题目名称	最小质因子	楼层跳跃	大鱼吃小鱼
英文题目名称	factor	jump	eat
输入文件名	factor.in	jump.in	eat.in
输出文件名	factor.out	jump.out	eat.out
每个测试点时限	1秒	1秒	1秒
测试点数目	10	10	10
每个测试点分值	10	10	10
内存限制	128MB	128MB	128MB
是否有部分分	否	否	否
题目类型	传统	传统	传统

阿斯大法好

1 最小质因子

1.1 题目描述

不要题目背景了。

给出 l 和 r ，求区间 $[l, r]$ 中每个数的最小质因子。

1.2 输入格式

输入文件为factor.in。

第一行一个整数 l ，第二行一个整数 r 。

1.3 输出格式

输出文件为factor.out。

输出 $r - l + 1$ 行，第一行表示 l 的最小质因子，第二行表示 $l + 1$ 的最小质因子……最后一行表示 r 的最小质因子。

1.4 样例输入

6
10

1.5 样例输出

2
7
2
3
2

1.6 样例解释

不解释。

1.7 数据规模与约定

对于20%的数据， $l = 2, r \leq 10$ 。

对于40%的数据， $l = 2, r \leq 10^5$ 。

对于70%的数据, $l = 2, r \leq 10^6$ 。

对于100%的数据, $2 \leq l \leq r \leq 10^{12}, r - l \leq 10^6$ 。

2 楼层跳跃

2.1 题目描述

不要题目背景……这道题就说不明白了。

话说你身处一个 n 层大楼中, 这个大楼有一个电梯。所有楼层从1到 n 编号, 现在你在第 a 层, 然后你心血来潮想坐 k 次电梯。

然而电梯也是有规则的。第 b 层是机密场所, 所以你不能去。(保证 $a \neq b$) 并且, 假设你当前在第 x 层, 想去第 y 层, 你必须保证 y 比 b 相对于你当前的位置更近。(也就是说, 要保证 $|x - y| < |x - b|$)

每次你从电梯出来时, 你会记录下你当前所在的楼层。坐完 k 次电梯后, 这些数字会形成一个序列。现在要求的就是所有的可能出现的序列的种类数。答案可能很大, 模1,000,000,007输出。

2.2 输入格式

输入文件为jump.in。

只有一行, 包含四个整数 n, a, b, k , 以空格隔开。

2.3 输出格式

输出文件为jump.out。

输出一个整数, 代表可能的序列数对1,000,000,007取模的结果。

2.4 样例输入

5 2 4 2

2.5 样例输出

2

2.6 样例解释

第一步，你可以选择到达第1层或第3层。但是如果选择第3层，第二步将无路可走，所以只能选择去第1层。

第二步，你可以选择去第2层或第3层。

总共有2种不同的序列：[1, 2]和[1, 3]。

2.7 数据规模与约定

对于40%的数据， $n \leq 10, k \leq 5$ 。

对于70%的数据， $n, k \leq 200$ 。

对于100%的数据， $1 \leq a, b \leq n \leq 2000, k \leq 2000, a \neq b$ 。

3 大鱼吃小鱼

3.1 题目描述

从前有 n 条鱼排成一列，第 i 条鱼的大小为 a_i 。为了方便，保证 $1 \leq a_i \leq n$ 并且互不相同。

现在这些鱼要开始互相吃了！在每轮当中，每条鱼如果比它右边的第一条鱼大，就会将它吃掉。一条鱼可能在吃别的鱼的同时被别的鱼吃掉，而吃掉别的鱼不会增加自己的大小。

一轮结束过后，如果还能存在互相吃的情况，则再进行一轮，直到没有互相吃的情况发生。

请你编程输出需要多少轮整个过程才能结束？

3.2 输入格式

输入文件为eat.in。

第一行有一个正整数 n 。

接下来 n 行，依次表示每一个 a_i 。

3.3 输出格式

输出文件为eat.out。

输出一个整数，表示所需的轮数。

3.4 样例输入1

10
10
9
7
8
6
5
3
4
2
1

3.5 样例输出1

2

3.6 样例输入2

6
1
2
3
4
5
6

3.7 样例输出2

0

3.8 样例解释

对于第一组样例，[10, 9, 7, 8, 6, 5, 3, 4, 2, 1]将在一轮后变成[10, 8, 4]，在第二轮后变成[10]，之后不会发生吃鱼的情况，所以输出2。

对于第二组样例，一开始就不会有任何吃鱼的情况，所以输出0。

3.9 数据规模与约定

对于40%的数据， $n \leq 1000$ 。

对于100%的数据， $1 \leq n \leq 10^5$ 。

均匀分布着一半的数据保证所有的 a_i 是完全随机生成的。