

NOI 2014天津市代表队选拔赛 day2

May 2, 2014

注意

请各位参赛选手在 E 盘根目录建立以自己文件夹号为名字的文件夹，文件夹下建立以题目为名字的文件夹，源程序放在相应的题目文件夹下。例如 E:/TJ-001/movie/movie.cpp。除 TJ 外其余字母均为小写。除源程序外，其余无关文件不要放在该文件夹。

A 上升子序列

- 输入输出文件: subsequence.in/subsequence.out
- 源文件名: subsequence.cpp/subsequence.c/subsequence.pas
- 时间限制: 1s 内存限制: 128M

题目描述

给定一个只包含整数的序列(序列元素的绝对值大小不超过 10^9)，你需要计算上升子序列的个数，满足如下条件的称之为一个上升子序列：

1. 是原序列的一个子序列
2. 长度至少为 2
3. 所有元素都严格递增

如果两个上升子序列相同，那么只需要计算一次。例如：序列 {1,2,3,3} 有 4 个上升子序列，分别为 {1,2}, {1,3}, {1,2,3}, {2,3}。

输入

输入的第一行是一个整数 n ，表示序列长度。接下来一行是 n 个整数，表示序列。

输出

输出仅包含一行，即原序列有多少个上升子序列。由于答案可能非常大，你只需要输出答案模 $10^9 + 7$ 的余数。

样例输入

```
4
1 2 3 3
```

样例输出

4

数据范围

对于 30% 的数据, $N \leq 5000$

对于 100% 的数据, $N \leq 10^5$

B Alice and Bob

- 输入输出文件: alice.in/alice.out
- 源文件名: alice.cpp/alice.c/alice.pas
- 时间限制: 1s 内存限制: 128M

题目描述

Alice 和 Bob 发明了一个新的游戏。给定一个序列 $\{x_0, x_1, \dots, x_{n-1}\}$ 。Alice 得到一个序列 $\{a_0, a_1, \dots, a_{n-1}\}$, 其中 a_i 表示以 x_i 结尾的最长上升子序列的长度; Bob 得到一个序列 $\{b_0, b_1, \dots, b_{n-1}\}$, 其中 b_i 表示以 x_i 开头的最长下降子序列的长度。Alice 的得分是序列 $\{a_0, a_1, \dots, a_{n-1}\}$ 的和, Bob 的得分是 $\{b_0, b_1, \dots, b_{n-1}\}$ 的和。

输入

输入的第一行是 n , 第二行是序列 $\{a_0, a_1, \dots, a_{n-1}\}$ 。数据保证序列 a 可以由至少一个 1 到 n 的排列得到。

输出

输出包含一行, 表示 Bob 能得到的最高分数。

样例输入1

```
4
1 2 2 3
```

样例输出1

```
5
```

样例输入2

```
4
1 1 2 3
```

样例输出2

```
5
```

数据范围

对于 30% 的数据, $N \leq 1000$
对于 100% 的数据, $N \leq 10^5$

C 电影评分

- 输入输出文件: movie.in/movie.out
- 源文件名: movie.cpp/movie.c/movie.pas
- 时间限制: 1s 内存限制: 128M

题目描述

小 Z 发明了一套新的电影评分系统。这套系统有三种操作：发布新电影、对电影评分、以及询问电影评分的排名。具体是这样运作的：如果是发布新电影，并且这部电影的有所主演之前均没有出现，那么这部新电影的评分为 0，否则这部电影的评分为**最近一部**与该电影**至少有一个共同主演**的电影的评分；如果是对电影进行评分，那么这部电影的评分就变成之前评分与新的评分的平均数；如果是查询排名，则根据评分输出相应排名。评分最高的为第一名。如果有多部电影分数相同，那么输出最早的一部。电影的评分在 0 到 5 之间。

输入

输入的第一行是 n ，表示操作次数。接下来 n 行，每一行是以下三种操作之一：

1. $Q\ x$: 查询当前排名为 x 的电影 ID;
2. $R\ ID\ x\ actor_1\ actor_2\ \dots\ actor_x$: 发布新电影 ID ，该电影有 x 个主演分别为 $actor_1, actor_2, \dots$;
3. $C\ ID\ score$: 评分操作，表示对电影 ID 的评分为 $score$

数据保证每个电影的 ID 不相同，且每部电影至多不超过 5 名主演。

$$1 \leq actor_1, actor_2, \dots \leq 10^5$$

$$1 \leq ID \leq 10^5$$

输出

对于每一个查询操作，输出相应排名的电影的 ID 。

样例输入

```
10
R 1 1 1
R 2 2 1 2
C 2 2
R 3 1 2
Q 1
```

C 3 2
C 1 5
Q 1
Q 2
Q 3

样例输出

2
1
3
2

样例解释

Movie	1	2	3
	0	-	-
	0	0	-
	0	1	-
	0	1	1

Q 1 => 2 //Movie 2 was released before Movie 3

	0	1	1.5
	2.5	1	1.5

Q 1 => 1
Q 2 => 3
Q 3 => 2

数据范围

对于 30% 的数据, $n \leq 100$
对于 100% 的数据, $n \leq 10000$