

NOIP 模拟题 Day2

Colin

1 题目概况

中文题目名称	排序	花花的森林	最短路
英文题目名称	sort	forest	sp
输入文件名	sort.in	forest.in	sp.in
输出文件名	sort.out	forest.out	sp.out
每个测试点时限	1 秒	1 秒	1 秒
测试点数目	20	10	10
每个测试点分值	5	10	10
附加样例文件	有	有	有
题目类型	传统	传统	传统
运行内存上限	128M	128M	128M

2 注意事项

- 考试时间：3.5 小时。
- 评测在 windows 下进行。
- 评测时将开启 -O2 优化。
- 部分题目涉及大规模文件操作，请尽可能提升程序输入输出操作的效率。

3 排序

3.1 题意描述

众所周知，熟练掌握至少一种排序算法是参加NOIP的必备技能。常见的排序算法有冒泡排序、归并排序、快速排序、奇偶排序、猴子排序、梳排序、鸡尾酒排序、臭皮匠排序等。

在这里，介绍一种利用栈进行排序的方法。例如，当数组中的元素为 1, 3, 2 时，我们可以利用栈对其进行排序：1 入栈；3 入栈；3 出栈；2 入栈；2 出栈；1 出栈。在这个例子中，出栈序列是 3, 2, 1，因而实现了对数组的排序。

遗憾的是，在不打乱入栈顺序的前提下，有时仅仅借助一个栈是不能实现对数组的完全排序。例如给定数组 2, 1, 3，借助一个栈，能获得的字典序最大的出栈序列是 3, 1, 2。（2 入栈；1 入栈；3 入栈；3 出栈；1 出栈；2 出栈）

现请你借助一个栈，在不打乱入栈顺序的情况下，对数组进行从大到小排序。当无法完全排序时，请输出字典序最大的出栈序列。

3.2 输入格式

输入共 2 行。

第一行包含一个正整数 n ，表示入栈序列长度。

第二行包含 n 个整数，表示入栈序列。输入数据保证给定的序列是 1 到 n 的全排列，即不会出现重复数字。

3.3 输出格式

输出仅一行，共 n 个整数，表示字典序最大的出栈序列。

3.4 样例输入

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

3.5 样例输出

```
5 4 3 1 2
```

3.6 样例解释

2 入栈；1 入栈；5 入栈；5 出栈；3 入栈；4 入栈；4 出栈；3 出栈；1 出栈；2 出栈

3.7 数据规模与约定

- 对于 40% 的数据： $N \leq 10$;
- 另有 20% 的数据： $N \leq 10^3$;
- 另有 20% 的数据： $N \leq 10^5$;
- 对于 100% 的数据： $N \leq 10^6$ 。

4 花花的森林

4.1 题意描述

花花有一棵带 n 个顶点的树 T ，每个节点有一个点权 a_i 。

有一天，他认为拥有两棵树更好一些。所以，他从 T 中删去了一条边。

第二天，他认为三棵树或许又更好一些。因此，他又从他拥有的某一棵树中去除了一条边。

如此往复。每一天，花花都会删去一条尚未被删去的边，直到他得到了一个包含了 n 棵只有一个点的树的森林。

定义一条简单路径¹的权值为路径上点权之和，一棵树的直径为树上权值最大的简单路径。花花认为树最重要的特征就是它的直径。所以他想请你算出任一时刻他拥有的所有树的直径的乘积。因为这个数可能很大，他要求你输出乘积对 $10^9 + 7$ 取模之后的结果。

4.2 输入格式

输入的第一行包含一个整数 n ，表示树 T 上顶点的数量。

下一行包含 n 个空格分隔的整数 a_i ，表示顶点的权值。

之后的 $n - 1$ 行中，每一行包含两个用空格分隔的整数 x_i 和 y_i ，表示节点 x_i 和 y_i 之间连有一条边，编号为 i 。

再之后 $n - 1$ 行中，每一行包含一个整数 k_j ，表示在第 j 天里会被删除的边的编号。

4.3 输出格式

输出 n 行。

在第 i 行，输出删除 $i - 1$ 条边之后，所有树直径的乘积对 $10^9 + 7$ 取模的结果。

4.4 样例输入

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

¹顶点不重复出现的路径

4.5 样例输出

6

9

6

4.6 样例解释

初始，树的直径为 6（由节点 2、1 和 3 构成的路径）。在第一天之后，得到了两棵直径都为 3 的树。第二天之后，得到了三棵直径分别为 1，2，3 的树，乘积为 6。

4.7 数据规模与约定

- 对于 40% 的数据： $N \leq 100$ ；
- 另有 20% 的数据： $N \leq 1000$ ；
- 另有 20% 的数据： $N \leq 10^4$ ；
- 对于 100% 的数据： $N \leq 10^5, a_i \leq 10^4$ 。

5 最短路

5.1 题意描述

H 国有 n 个城市。 n 个城市间有 m 条双向道路连接。 n 个城市两两连通，且每一条道路最多在一个环内。

现在花花有 q 组询问，每次询问一对城市间的最短路径长度。

5.2 输入格式

输入第一行包括 3 个非负整数 n, m, q 。

接下来 m 行，每行包括 3 个非负整数 u_i, v_i, l_i ，表示编号为 u_i 和 v_i 的城市间有一条长度为 l_i 的双向道路连接。

接下来 q 行，每行包括 2 个正整数 u_i 和 v_i ，表示询问城市 u_i 和 v_i 间的最短路径长度。

5.3 输出格式

输出包含 q 行，每行包含一个非负整数，表示城市 u_i 和 v_i 间的最短路径长度。

5.4 样例输入

```
9 10 2
1 2 1
1 4 1
3 4 1
2 3 1
3 7 1
7 8 2
7 9 2
1 5 3
1 6 4
5 6 1
1 9
5 7
```

5.5 样例输出

```
5
6
```

5.6 数据规模与约定

测试点编号	n	q	环的个数
0	≤ 100	≤ 10	0
1	≤ 100	≤ 100	0
2	≤ 100	≤ 500	0
3	≤ 100	≤ 1000	0
4	≤ 1000	≤ 500	0
5	≤ 1000	≤ 1000	1
6	$\leq 10^4$	≤ 5000	0
7	$\leq 10^4$	$\leq 10^4$	1
8	$\leq 10^5$	$\leq 10^5$	≤ 10
9	$\leq 10^5$	$\leq 10^5$	无限制

对于 100% 的数据： $1 \leq l_i \leq 10000$ 。