

NOIP 2021 模拟

Dragon_in_Bed 2021.10

时间：2021 年 11 月 4 日 13:00-17:00

题目概况

题目名称	塔纳	托萨妮娅	底夫斯拉夫	托萨卡
题目类型	传统	传统	传统	传统
目录	tanao	tosania	dfslover	tosaka
可执行文件名	tanao	tosania	dfslover	tosaka
输入文件名	tanao.in	tosania.in	dfslover.in	tosaka.in
输出文件名	tanao.out	tosania.out	dfslover.out	tosaka.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	512MiB	512MiB	1GiB	512MiB
测试点数目	10	20	20	25
每个测试点分值	10	5	5	4
附加样例文件	无	有	有	有

提交源程序文件名

对于 C++语言	tanao.cpp	tosania.cpp	dfslover.cpp	tosaka.cpp
----------	-----------	-------------	--------------	------------

编译选项

对于 C++语言	-lm -O2 -std=c++14
----------	--------------------

注意事项

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. 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`。
3. 若题目没有特殊说明，则测试数据中所有在同一行的元素间用一个空格隔开，无多余空格，且在文件末尾有且只有一行换行。
4. 结果比较方式为全文比较（过滤行末空格及文末回车）。
5. 程序可使用的栈内存空间限制与题目的内存限制一致。

塔纳 (tanao)

【问题背景】

话说那 tanao，是天选之子，出生时便散发金光，转眼间，四周的电脑全部自己开机！他从小便表现惊人的天赋，3 岁自学平衡树，5 岁 AC 猪国杀，6 岁破格录取为 BNDS 学生，并在 NOIP 赛场中叱咤风云，各种奖拿到手软。有词为证：

牛逼神仙塔纳，幼年爆切猪国杀。

考场键盘声浩大，AK 奖杯随便拿。

13 岁时，他的能力已经达到惊人的水平。他做的题目从三连击，到猪国杀，再到 TopTree，再到自己研发强大数据结构，以惊人的速度秒杀全国其他 OI 选手。中国已经容不下他了！他要到国外去，去 AK！

在 IOI 赛场上，他的手在键盘上飞快地游走，其他选手看去，只能看到 tanao 电脑面前有一团幻影，和他一题一题的 AC 记录。

考场有外国友人举报他键盘声太大影响其他选手做题，当 tourist 来提醒他的时候，他立刻站起身来放倒了他。他已无人可挡！

——《tanao 传》

【问题描述】

tanao 是计数大师，虽然他自己不承认。

对于一个排列 p ，若有 $i, j (i < j)$ ，满足 $p_i > p_j$ ，则称 (i, j) 为 p 的一个逆序对。

tanao 最擅长逆序对计数。给定 n ，tanao 想请你求出 $1 - n$ 的所有排列的逆序对数之和。

【输入格式】

输入文件名为 tanao.in。

本题有多组测试数据。

输入共 $T + 1$ 行。

第 1 行包含 1 个正整数 T ，表示数据组数。

接下来 T 行，每行包含一个正整数 n 。

【输出格式】

输出文件名为 tanao.out。

输出共 T 行，每行包含 1 个非负整数，表示 $1 - n$ 的所有排列的逆序对数之和。答案对 998244353 取模。

【样例 1】

tanao.in	tanao.out
2	1
2	9
3	

【样例 1 解释】

当 $n = 2$ 时:

排列	逆序对
$[1, 2]$	无
$[2, 1]$	$(2, 1)$

当 $n = 3$ 时:

排列	逆序对
$[1, 2, 3]$	无
$[1, 3, 2]$	$(2, 1)$
$[2, 1, 3]$	$(2, 1)$
$[2, 3, 1]$	$(2, 1), (3, 1)$
$[3, 1, 2]$	$(3, 1), (3, 2)$
$[3, 2, 1]$	$(3, 2), (3, 1), (2, 1)$

【数据范围】

对于 20% 的数据, $n \leq 10$ 。

对于 40% 的数据, $n \leq 10^3$ 。

对于 60% 的数据, $n \leq 10^5$ 。

另外 20% 的数据, $T = 1$ 。

对于 100% 的数据, $1 \leq T \leq 10^4, 1 \leq n \leq 10^7$ 。

托萨妮娅 (tosania)

【问题描述】

Tosania 相信存在平行宇宙。人们说平行宇宙的名字也叫 **Tosania**。

在 k 个平行宇宙中有对应相同的 n 颗星星。在每个平行宇宙中，它们都由 $n - 1$ 条轨道连接着，任何两颗星星之间都存在唯一一条路径。不同平行宇宙中的轨道可能不同。

Tosania 时常在夜晚想象其中的一对星星 p, q 相连。据说，如果一颗星星在所有平行宇宙中都在 p 和 q 之间的简单路径上，那它就会闪烁。特殊地， p 和 q 都会闪烁；当 $p = q$ 时，只有 p 这一颗星星会闪烁。

现在，**Tosania** 想知道，对于她想象的任意一对相连的星星，夜晚会有几颗星星闪烁。

形式化地讲，一条简单路径对应一个经过该路径的点的集合，**Tosania** 想知道任意点对在 k 棵树上的简单路径对应的 k 个点集的交的大小。

【输入格式】

输入文件名为 **tosania.in**。

输入共 $k(n - 1) + 1$ 行。

第 1 行包含 2 个正整数 n, k 。

接下来 $k(n - 1)$ 行，每 $n - 1$ 行描述一个平行宇宙的所有轨道。其中每行包含两个正整数 u, v ，表示星星 u, v 之间有一条轨道连接。

【输出格式】

输出文件名为 **tosania.out**。

由于本题的输出量可能较大，你不需要也不应该直接输出答案。

记当 $p = i, q = j$ 时，夜晚闪烁的星星数量为 $ans_{i,j}$ 。记 $p_i = \sum_j j \cdot ans_{i,j}$ 。

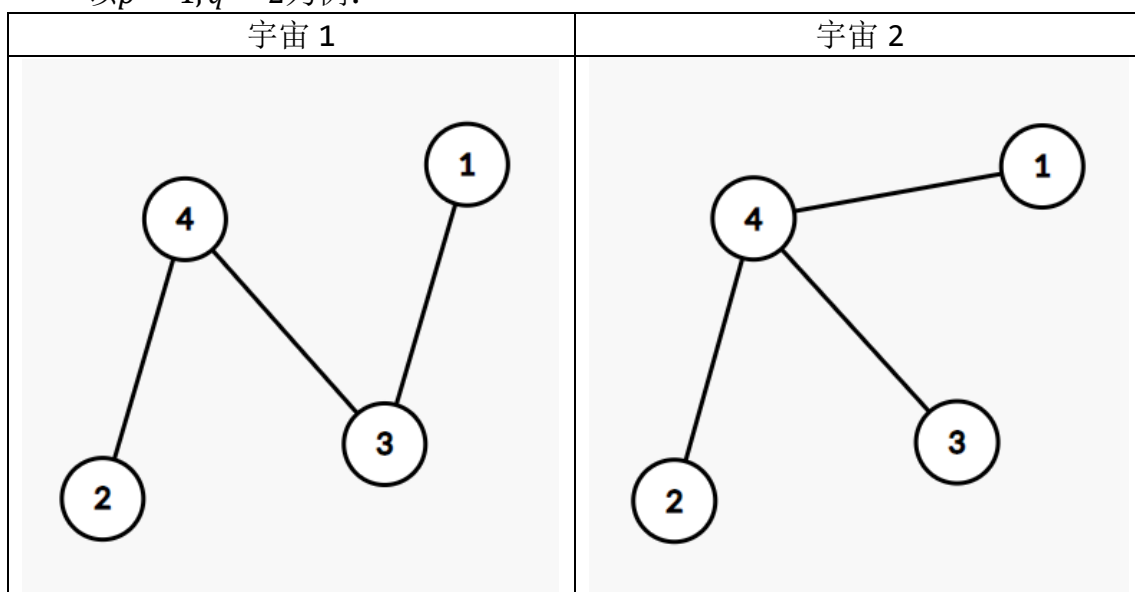
输出共 n 行，每行 1 个非负整数。其中第 i 行的数表示 p_i 。

【样例 1】

tosania.in	tosania.out
4 2	21
1 3	22
4 2	19
3 4	16
1 4	
4 3	
2 4	

【样例 1 解释】

以 $p = 1, q = 2$ 为例：



在宇宙 1 中，星星 1 和星星 2 之间的路径为 $1 - 3 - 4 - 2$ ；在宇宙 2 中，星星 1 和星星 2 之间的路径为 $1 - 4 - 2$ 。因此，星星 1, 2, 4 会闪烁，答案为 3。

以下第 i 行第 j 个数表示当 $p = i, q = j$ 时，夜晚闪烁的星星数量：

1	3	2	2
3	1	3	2
2	3	1	2
2	2	2	1

以第一行为例， $1 \times 1 + 2 \times 3 + 3 \times 2 + 4 \times 2 = 21$ ，所以第一行输出 21。

【样例 2】

见选手目录下的 `tosania/tosania2.in` 与 `tosania/tosania2.ans`。

【数据范围】

数据点编号	$n \leq$	$k \leq$
1-2	50	
3-4	100	
5-6	200	
7-8	250	
9-10	300	
11-12	350	
13-14	400	
15-16	500	1
17-20	500	

对于 100% 的数据， $1 \leq n, k \leq 500$ 。

【提示】

Tosania，请务必展现你的卡常技巧。

底夫斯拉夫 (dfslover)

【问题背景】

猴子排序是一种著名的排序算法。众所周知，基于比较的传统排序算法的最优时间复杂度不低于 $O(n \log n)$ 。而猴子排序算法突破了这一桎梏，拥有 $O(n)$ 的优秀最优时间复杂度。

【问题描述】

DFSlover 擅长使用 DFS 解决各类问题，包括但不限于计数题。

DFSlover 热爱 DFS，因而同样热爱指数级的时间复杂度。他最喜欢的排序算法就是猴子排序，因为它的期望时间复杂度是 $O(n \cdot n!)$ 的。他最讨厌的排序算法就是快速排序，因为它的期望时间复杂度是 $O(n \log n)$ 的。

我们定义一个序列 a 中的一个元素 a_i 是混乱的，当且仅当 $\exists j(j < i), a_j > a_i$ 或 $\exists k(k > i), a_k < a_i$ ，也就是说，在 a_i 前存在大于 a_i 的数或在 a_i 后存在小于 a_i 的数。众所周知，快速排序的一个关键步骤就是把序列中某个元素变成非混乱的。

出于对快速排序的讨厌，DFSlover 讨厌非混乱的元素。我们定义序列 b 为将序列 a 中的元素重新排列后所得的序列（ a, b 可能相同）。现在，DFSlover 想知道，对于给定的序列 a ，存在多少个本质不同的序列 b ，使得 b 中的所有元素都是混乱的。

两个序列 a, b 是本质不同的，当且仅当 $\exists i, a_i \neq b_i$ 。

【输入格式】

输入文件名为 `dfslover.in`。

输入共 2 行。

第 1 行包含 1 个正整数 n 。

第 2 行包含 n 个正整数，其中第 i 个数表示 a_i 。

【输出格式】

输出文件名为 `dfslover.out`。

输出共 1 行，包含 1 个非负整数，表示满足条件的本质不同的序列 b 的数量。答案对 998244353 取模。

【样例 1】

dfslover.in	dfslover.out
4 1 2 2 3	7

【样例 1 解释】

满足条件的序列有 $[2, 1, 3, 2]$ ， $[2, 2, 3, 1]$ ， $[2, 3, 1, 2]$ ， $[2, 3, 2, 1]$ ， $[3, 1, 2, 2]$ ， $[3, 2, 1, 2]$ 和 $[3, 2, 2, 1]$ 。

【样例 2】

见选手目录下的 `dfslover/dfslover2.in` 与 `dfslover/dfslover2.ans`。
该样例满足 $a_i = i$ 。

【样例 3】

见选手目录下的 `dfslover/dfslover3.in` 与 `dfslover/dfslover3.ans`。

【数据范围】

对于 20% 的数据， $n \leq 10$ 。

对于 40% 的数据， $n \leq 200$ 。

对于 60% 的数据， $n \leq 1000$ 。

另外 30% 的数据，保证 $a_i = i$ 。

对于 100% 的数据， $1 \leq a_i \leq n \leq 8 \cdot 10^3$ 。

【提示】

为什么不试试 DFS？

——DFShater

托萨卡 (tosaka)

【问题背景】

雀魂是一款日式麻将游戏。它采用日式竞技立直麻将的规则。麻将牌可以分为字牌（东、南、西、北、白、发、中七种）和序数牌（条子、筒子、万子三种花色，每种花色各有一到九这九种点数的牌），每种牌各四张。三张相同的牌称为刻子，三张同花色组成顺序数字的牌称为顺子，顺子和刻子合称面子。两张相同的牌称为雀头。当 14 张牌可以分为 4 组面子和 1 组雀头时，即可和牌。

然而，在立直麻将中，和牌还需要一个条件，即役。常见的役有断幺九、立直、役牌等。当最终和牌的 14 张牌中不存在字牌和点数为 1 或 9 的牌时，称这种和牌为断幺九。

【问题描述】

Tosaka 是雀魂大师，虽然只有初心一星。他擅长作弊，总能在自己装有 n 张牌的口袋里摸出 14 张牌和牌。

Tosaka 作为雀魂大师，只会使用断幺九这一种役。也就是说，只有当摸出的 14 张牌能和断幺九时，他才能和牌。

现在，在 zjjws 的注视下，Tosaka 慌乱地从口袋里拿出了 14 张牌。他想知道自己能和牌的概率是多少。如果某一种牌超过了 4 张，他会直接因为作弊被发现而无法和牌。

【输入格式】

输入文件名为 `tosaka.in`。

输入共 2 行。

第 1 行包含 1 个正整数 n 。

第 2 行包含 n 个字符串，每个字符串描述一张牌。若该牌为字牌，则东、南、西、北、白、发、中分别用 `east`, `south`, `west`, `north`, `white`, `green`, `red` 表示。若该牌为序数牌，则条子、筒子、万子分别用 `bamboo`, `circle`, `character` 表示，并在后面加上一个数字表示点数，例如九条用 `bamboo9` 表示。

【输出格式】

输出文件名为 `tosaka.out`。

输出共 1 行，包含 1 个非负整数 p ，表示 Tosaka 的和牌概率对 $k = 998244353$ 取模的结果。也就是说，如果和牌概率的最简分数表示为 $\frac{a}{b}$ ，则 p 为满足 $b \times p \equiv a \pmod{k}$ 的最小非负整数。

【样例 1】

tosaka.in	tosaka.out
16 bamboo2 bamboo2 bamboo2 bamboo2 bamboo3 bamboo3 bamboo4 bamboo4 bamboo6 bamboo6 bamboo6 bamboo8	856826403

bamboo8 bamboo8 green green

【样例 1 解释】

给定的 16 张牌为 4 张 2 条, 2 张 3 条, 2 张 4 条, 3 张 6 条, 3 张 8 条和 2 张发。组牌方法一共有 120 种。其中, 4 张 2 条, 2 张 3 条, 2 张 4 条, 3 张 6 条和 3 张 8 条构成一组顺子 (2 条, 3 条, 4 条), 一组顺子 (2 条, 3 条, 4 条), 一组刻子 (3 张 6 条), 一组刻子 (3 张 8 条) 和一组雀头 (2 张 2 条), 可以和牌。因此, 和牌概率为 $\frac{1}{120}$ 。

这是 Tosaka 从来没有和过的役满牌型绿一色。

【样例 2】

见选手目录下的 `tosaka/tosaka2.in` 与 `tosaka/tosaka2.ans`。
该样例满足特殊性质 A, B, C。

【样例 3】

见选手目录下的 `tosaka/tosaka3.in` 与 `tosaka/tosaka3.ans`。

【数据范围】

数据点编号	$n \leq$	特殊性质
1-3	14	无
4-7	23	A, B, C
8-9	10^3	A, B, C
10-11		B, C
12-13		C
14		A
15-16		无
17-18	10^5	A, B, C
19-20		B, C
21-22		C
23		A
24-25		无

特殊性质 A: 不存在字牌和点数为 一或九 的牌。

特殊性质 B: 不存在筒子牌。

特殊性质 C: 不存在万子牌。

对于 100% 的数据, $14 \leq n \leq 10^5$ 。

【提示】

棒听即立, 速攻速攻!