

TQOI Round 1

时间: 2025 年 13 月 32 日 01:14 ~ 05:14

题目名称	火	捐赠	染色	孤独旅者
题目类型	传统型	传统型	传统型	传统型
目录	fire	donate	color	loneliness
可执行文件名	fire	donate	color	loneliness
输入文件名	fire.in	donate.in	color.in	loneliness.in
输出文件名	fire.out	donate.out	color.out	loneliness.out
每个测试点时限	1.0 秒	2.0 秒	2.0 秒	1.0 秒
内存限制	512 MB	512 MB	512 MB	512 MB
测试点数目	10	4	5	4
测试点是否等分	是	否	否	否

提交源程序文件名

对于 C++ 语言	fire.cpp	donate.cpp	color.cpp	loneliness.cpp
-----------	----------	------------	-----------	----------------

编译选项

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

注意事项与提醒 (请选手务必仔细阅读)

1. 选手请直接提交源程序至 becoder.com.cn 上的对应比赛。
2. 输入输出文件名必须使用英文小写。
3. 选手提交的源程序必须存放在**已建立好的，且带有样例文件和下发文件**的文件夹中，文件夹名称与对应试题英文名一致。
4. 文件名 (包括程序名和输入输出文件名) 必须使用英文小写。
5. C++ 中函数 `main()` 的返回值类型必须是 `int`，值必须为 0。
6. 若无特殊说明，结果比较方式为**忽略行末空格、文末回车后的全文比较**。
7. 程序可使用的栈空间大小与该题内存空间限制一致。
8. 在终端中执行命令 `ulimit -s unlimited` 可将当前终端下的栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
9. 每道题目所提交的**代码文件大小限制为 100KB**。
10. 若无特殊说明，输入文件与输出文件中同一行的相邻整数均使用一个空格分隔。
11. 输入文件中可能存在行末空格，请选手使用更完善的读入方式 (例如 `scanf` 函数) 避免出错。

12. 直接复制 PDF 题面中的多行样例，数据将带有行号，建议选手直接使用对应目录下的样例文件进行测试。
13. 使用 `std::deque` 等 STL 容器时，请注意其内存空间消耗。
14. 请务必使用题面中规定的的编译参数，保证你的程序在本机能够通过编译。此外不允许在程序中手动开启其他编译选项，一经发现，本题成绩以 0 分处理。
15. 对于因未遵守以上规则对成绩造成的影响，相关申诉不予受理。

火 (fire)

【题目描述】

paper 种了一片人工林，用于生产纸张。树林中的 n 棵树排成了一条直线。

某一天发生了火灾，初始时每棵树都有概率被点燃。并且因为树排成了一条直线，每一棵燃着的树每过一秒都会使相邻没有燃着的树木燃着。

paper 给出了初始时每棵树燃烧的概率，请你求出 k 秒后燃烧的树木数量的期望值。这个期望本身是一个有理数，但为了避免精度误差，我们要求输出这个有理数对 $10^9 + 7$ 取模的结果。

形式化地说，可以证明答案可以被表示为既约分数 $\frac{y}{x}$ ，其中 $x \not\equiv 0 \pmod{10^9 + 7}$ 。可以证明存在一个唯一的整数 z ($0 \leq z < 10^9 + 7$)，使得 $xz = y$ ，你只需要输出 z 的值。

【输入格式】

从文件 *fire.in* 中读入数据。

第一行两个整数 n, k ，分别表示树的棵数与秒数。

接下来 n 个在 0 到 100 之间的整数，表示每一棵树在 0 时刻时燃着的百分率。

【输出格式】

输出到文件 *fire.out* 中。

输出一个整数，表示 $(P \cdot Q^{-1}) \bmod (10^9 + 7)$ 的值。

【样例 1 输入】

```
1 3 1
2 0 50 0
```

【样例 1 输出】

```
1 500000005
```

【样例 2 输入】

```
1 5 2
2 10 40 90 0 12
```

【样例 2 输出】

```
1 613760009
```

【数据范围】

对于 20% 的数据, $1 \leq n, k \leq 20$ 。

对于 50% 的数据, $1 \leq n, k \leq 5000$ 。

对于 100% 的数据, $1 \leq n, k \leq 2 \times 10^5$ 。

捐赠 (donate)

【题目描述】

有 A 、 B 两类物品。

paper 打算每类各选 k 个 (k 可自由决定, 可取 0) 一起捐出。捐赠的总贡献为所选物品的价值总和。

初始时 paper 没有物品, 但是 paper 可以通过一些操作改变物品的数量, 一共进行了 m 次操作, 分为三种类型:

- 增加或减少若干个价值相同的 A 类物品。
- 增加或减少若干个价值相同的 B 类物品。
- 询问当前能获得的最大捐赠总贡献。

请你帮助 paper 处理这些操作。

保证在减少操作中物品个数一定充足。

【输入格式】

从文件 *donate.in* 中读入数据。

第一行有一个整数, 表示操作的总数 m 。

接下来 m 行, 每行表示一次操作, 首先有一个整数 op 。

- 若 op 为 1, 则后面有两个整数 x, y 。若 $y \geq 0$ 则表示增加 y 个价值为 x 的 A 类物品, 否则表示减少 $-y$ 个。
- 若 op 为 2, 则后面有两个整数 x, y 。若 $y \geq 0$ 则表示增加 y 个价值为 x 的 B 类物品, 否则表示减少 $-y$ 个。
- 若 op 为 3, 代表询问目前最大可能贡献。

【输出格式】

输出到文件 *problem.out* 中。

对于每一个 $op = 3$, 输出此时的最大贡献。

【样例 1 输入】

```
1 5
2 1 2 3
3 2 -1 3
4 3
5 1 2 -2
6 3
```

【样例 1 输出】

```
1 3
2 1
```

【数据范围】

本题采用捆绑测试。

- Subtask 0 (10 points): $m \leq 200$ 。
- Subtask 1 (30 points): $m \leq 5000$ 。
- Subtask 2 (20 points): $y = 1$ 。
- Subtask 3 (40 points): 无特殊限制。

对于所有测试数据, $1 \leq m \leq 10^6, -10^6 \leq x, y \leq 10^6$ 。

染色 (color)

【题目描述】

paper 在玩飞行棋。

飞行棋的棋盘是一棵 n 个节点的树，根为 1，有且仅有 k 个叶子，编号为 2 至 $k+1$ ，每个叶子上都有一颗棋子。

paper 要用 m 种颜色为每个节点染色（也可不染），但必须满足：对于任意叶子 i ($2 \leq i \leq k+1$)，从 i 跳到根节点 1 的路径上（包括 i ），**必须存在染色节点，且遇到的第一个染色节点的颜色恰好为 c_i 。**

paper 想要知道满足条件的染色方案数，由于这个数很大，你只需要求出答案对 998244353 取模的结果。

【输入格式】

从文件 *color.in* 中读入数据。

第一行两个整数 n, m, k ，含义同题目描述。

第二行 k 个整数，第 i 个表示 c_{i+1} ，含义同题目描述。

接下来 $n-1$ 行，每行两个整数 u, v 表示树上有一条边连接 u, v 。

【输出格式】

输出到文件 *color.out* 中。

一行一个整数，表示方案数对 998244353 取模的结果。

【样例 1 输入】

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

【样例 1 输出】

```
1 13
```

【数据范围】

本题采用捆绑测试。

- Subtask 0 (5 points): $1 \leq n, m \leq 7$ 。

- Subtask 1 (15 points): $1 \leq n, m \leq 300$ 。
- Subtask 2 (30 points): $1 \leq n, m \leq 5000$ 。
- Subtask 3 (10 points): $m = 2$ 。
- Subtask 4 (40 points): $1 \leq n, m \leq 2 \times 10^5$ 。

对于所有测试数据, $1 \leq n, m \leq 2 \times 10^5$, $1 \leq c_i \leq m$, $1 \leq k < n$ 。

孤独旅者 (loneliness)

【题目描述】

这是一个孤独旅者的故事。

这个世界由 n 个地点和 m 条单向道路组成。第 i 条道路从 a_i 通往 b_i ，孤独旅者只会在这些道路上行走。

他的笔记本上留着 k 篇游记，如今字迹漫漶，无人能识。

考古学家们只知道他独特的记录方式：每经过一条路，他**可能会**写下一篇游记——

- **追忆**：当孤独旅者走上道路 i 并且未来**不可能**再经过 a_i 时，孤独旅者**可能会**写下一篇描写 a_i 的游记用以追忆曾经在 a_i 发生过的事。
- **期许**：当孤独旅者走上道路 i 并且未来依然**可以**经过 a_i 时，孤独旅者**可能会**写下一篇描写任意一个 a_i 可达的地点的游记用以期许接下来的旅程。当且仅当存在一条起点为 u ，终点是 v 的简单路径时， u 可达 v 。

当然，他也可能不写。我们只知游记共 k 篇，却不知每篇描绘何地。

现在，考古学家向你提出 q 个询问：

每次给出起点 s 、终点 t ，他们想知道所有从 s 走到 t 并写下 k 篇游记的旅行中，游记内容可能有多少种不同的描写情况？答案对 $10^9 + 7$ 取模。

两种描写情况不同，当且仅当至少有一篇游记的描写方式不同。

两种描写方式相同当且仅当两篇游记所在的**边的类型（追忆边或期许边）相同且描写的地点相同且该边的起点可到达的点的集合相同**。

【输入格式】

从文件 *loneliness.in* 中读入数据。

共 $m + q + 1$ 行。

第一行有四个整数 n, m, k, q ，含义同题目描述。

第 $2 \sim m + 1$ 行，每行各有两个整数 a_i, b_i ，表示存在一条 a_i 到 b_i 的单向路径。

第 $m + 2 \sim m + q + 1$ 行，每行各有两个整数 s_i, t_i ，表示第 i 个询问中孤独旅者的起点与终点。

【输出格式】

输出到文件 *loneliness.out* 中。

q 行，每行一个整数，表示第 i 个询问的答案。

【样例 1 输入】

```
1 4 4 2 1
2 1 2
```

```
3 2 3
4 3 1
5 2 4
6 1 4
```

【样例 1 输出】

```
1 20
```

【样例解释】

若两篇游记都是在 $1 \rightarrow 2 \rightarrow 3 \rightarrow 1$ 的环上记录的，那么就有 $4 \times 4 = 16$ 种情况。
若第一篇游记在环上记录而第二篇在 $2 \rightarrow 4$ 记录，那么就有 $4 \times 1 = 4$ 种情况。
则答案为 $4 + 16 = 20$ 。

【数据范围】

本题采用捆绑测试。

- Subtask 0 (5 points): $n, m \leq 10$ 。
- Subtask 1 (30 points): $n \leq 200$, $k \leq 2000$ 。
- Subtask 2 (15 points): 保证给定图为有向无环图。
- Subtask 3 (50 points): 无特殊限制。

对于所有测试数据， $1 \leq n \leq 300$, $1 \leq m \leq 5000$, $1 \leq q \leq 9 \times 10^4$, $1 \leq k \leq 10^{18}$ 。
保证图中没有重边和自环。