

那咤 2 系列赛

Nezha 2 Series

Nezha contest

时间：2025 年 9 月 17 日 13:40 ~ 17:40

题目名称	网球君	大战	魔丸	掌牧松
题目类型	传统型	传统型	传统型	传统型
目录	网球君	大战	魔丸	掌牧松
可执行文件名	mrwang	fight	pill	mrzhang
输入文件名	mrwang.in	fight.in	pill.in	mrzhang.in
输出文件名	mrwang.out	fight.out	pill.out	mrzhang.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	0.8 秒
内存限制	256 MB	20 MB	128 MB	256 MB
测试点数目	20	59	49	40
测试点是否等分	是	否	否	否

提交源程序文件名

对于 C++ 语言	mrwang.cpp	fight.cpp	pill.cpp	mrzhang.cpp
-----------	------------	-----------	----------	-------------

编译选项

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

注意事项与提醒（请选手务必仔细阅读）

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. 提交的程序代码文件的放置位置请参照具体要求。
4. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。
5. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
6. 程序可使用的栈内存空间限制与题目的内存限制一致。
7. 只提供 Windows 格式附加样例文件。
8. 评测在 Windows 下进行，各语言的编译器版本以其为准。

网球君 (mrwang)

【题目背景】

网球君要大战魔丸掌牧松了，他需要深入魔丸掌牧松的结界，他观察发现结界由 N 座魔法宫殿组成。这些宫殿通过魔法通道相连，形成一个神秘的魔法网络。掌牧松刻意隐藏了结界的真实结构，但还是不小心泄露了部分距离信息。

【题目描述】

结界中有 N 座宫殿，编号为 1 到 N ，这些宫殿由 $N - 1$ 条双向魔法通道连接，形成一个树形结构。每条通道都有一个正整数长度，表示通道两端宫殿的魔法距离。

网球君已经探测到：对于除宫殿 1 和宫殿 N 外的每座宫殿 i ($2 \leq i \leq N - 1$)，它到宫殿 1 的魔法距离 $d_{1,i}$ 和到宫殿 N 的魔法距离 $d_{N,i}$ 。

现在，网球君需要根据这些距离信息，还原出整个结界的真实结构，找出所有魔法通道的连接方式和长度。这将帮助他找到最优的进攻路线，直捣掌牧松的巢穴！

【输入格式】

从文件 *mrwang.in* 中读入数据。

本题开启多测

第一行包含一个正整数 T ，表示数据组数。

对于每组数据：

第一行包含一个正整数 N ，表示宫殿数量。

第二行包含 $N - 2$ 个正整数 $d(1, 2), d(1, 3), \dots, d(1, N - 1)$ ，分别表示宫殿 2 到宫殿 $N - 1$ 与宫殿 1 的距离。

第三行包含 $N - 2$ 个正整数 $d(N, 2), d(N, 3), \dots, d(N, N - 1)$ ，分别表示宫殿 2 到宫殿 $N - 1$ 与宫殿 N 的距离。

【输出格式】

输出到文件 *mrwang.out* 中。

如果距离信息存在矛盾，无法构建合理的结界结构，输出 NO

否则第一行输出 YES，接下来 $N - 1$ 行每行三个正整数 u, v, c ，表示存在一条长度为 c 的连接宫殿 u 和宫殿 v 的魔法通道。

若有多组解，输出任意一组即可。

本题使用 Special Judge 进行评测。

注意：网球君必须尽快破解结界结构，时间紧迫！请帮助他完成这个关键任务。

【样例 1 输入】

```
1 1
2 7
```

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

【样例 1 输出】

```
1 YES
2 1 5 2
3 5 7 1
4 5 2 4
5 7 3 3
6 1 4 2
7 1 6 1
```

【样例 2】

见选手目录下的 *mrwang/mrwang2.in* 与 *mrwang/mrwang2.ans*。

【数据范围】

Special Judge 下发于选手目录下的 *mrwang/spj.exe*。

使用 Special Judge 测试输出是否正确时，应将 *spj.exe* 和 *.in* 文件，*.out* 文件及 *.ans* 文件放在同一目录下，在此目录下执行命令：

`.\spj` 输入文件名 *.in* 输出文件名 *.out* 答案文件名 *.ans*。

对于所有测试数据，保证： $2 \leq N \leq 10^5$ ， $\sum n \leq 10^6$ ， $1 \leq d \leq 10^{18}$ ， $1 \leq u, v \leq N$ ， $1 \leq c \leq 10^6$ 。

数据点	特殊性质	分数
1 ~ 4	$N \leq 8$	20
5 ~ 6	保证树的结构是一条链	10
7 ~ 20	无	70

大战 (fight)

【题目背景】

网球君要大战魔丸掌牧松了，魔丸掌牧松设置了机关阻挡网球君。

【题目描述】

网球君来到了机关门前，他遇到了守卫，守卫得知网球君要去讨伐掌牧松，十分欣喜。但魔丸掌牧松十分谨慎，他只告诉了守卫一些密码的特征，守卫将这些特征转述给了网球君。

他给网球君一个长度为 N 的正整数序列 A 和一个长度为 M 的正整数序列 B ，并描述了密码序列 K 的特征

- 密码是序列 A 和序列 B 的公共子序列。
- 密码的长度为偶数。
- 对于 $i \in [1, \frac{K}{2}]$ ，都满足 $K_{i \times 2} = K_{i \times 2 - 1}$ 。

现在网球君想要知道合法的密码序列 K 可能的最大长度，他想让你帮他求出这个值。

【输入格式】

从文件 *fight.in* 中读入数据。

第一行两个整数 N, M ，分别表示序列 A, B 的长度。

接下来一行 N 个整数，表示序列 A 。

接下来一行 M 个整数，表示序列 B 。

【输出格式】

输出到文件 *fight.out* 中。

输出一个整数，表示密码序列可能的最大长度。

【样例 1 输入】

```
1 7 9
2 1 2 2 3 1 1 1
3 2 4 2 3 1 2 4 1 1
```

【样例 1 输出】

```
1 4
```

【样例 1 解释】

最长合法密码序列为 2, 2, 1, 1。

【样例 2】

见选手目录下的 *fight/fight2.in* 与 *fight/fight2.ans*。
该组测试数据满足数据范围中描述的子任务 1 的限制。

【样例 3】

见选手目录下的 *fight/fight3.in* 与 *fight/fight3.ans*。
该组测试数据满足数据范围中描述的子任务 2 的限制。

【样例 4】

见选手目录下的 *fight/fight4.in* 与 *fight/fight4.ans*。
该组测试数据满足数据范围中描述的子任务 3 的限制。

【数据范围】

本题开启捆绑测试。
对于所有测试数据，保证： $1 \leq N, M \leq 15000, 1 \leq A_i, B_i \leq 10^9$ 。

子任务	附加限制	分值
1	$n, m \leq 2000$	30
2	$n, m \leq 15000$ ，每个数字在序列中至多出现两次	20
3	$n, m \leq 15000$	50

魔丸 (pill)

【题目背景】

网球君要大战掌牧松了，他从一位 人物未解锁 那里得到一张地图。

【题目描述】

掌牧松的结界拥有 N 座宫殿，编号为 1 到 N ，这些宫殿由 $N - 1$ 条双向道路连接。

掌牧松的宫殿分为 K 种属性，编号为 1 到 K ，第 i 座宫殿的属性为 C_i 。

现在网球君想要找到一种属性，并在这种属性的宫殿里与魔丸掌牧松决战！但是网球君不熟悉魔丸族的魔法，因此他要求在具有这种属性的所有宫殿中，任意两个间相互到达不需要经过其他属性的宫殿。

现在网球君可以向 人物未解锁 求助，每次求助他可以选择两种属性 x, y ，并让人物未解锁 帮助他将所有属性为 y 的宫殿属性修改为 x 。

大战就要到来了，网球君想请你帮他求出，想要找到一种决战属性，最少要向 人物未解锁 求助多少次。

【输入格式】

从文件 *pill.in* 中读入数据。

第一行两个整数 N, K 代表宫殿数和属性种类数。

接下来 $N - 1$ 行每行两个整数 u, v 代表一条双向边。

接下来 N 行第 i 行一个整数 C_i 代表第 i 座宫殿的属性为 C_i 。

【输出格式】

输出到文件 *pill.out* 中。

一行一个整数代表最小的求助次数。

【样例 1 输入】

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

113

122

【样例 1 输出】

11

【样例 1 解释】

可以求助一次，将属性 1 和 3 合并，然后选择属性 1 作为大战属性。

【样例 2 输入】

184

241

313

436

567

672

725

858

92

104

113

121

131

142

153

164

【样例 2 输出】

11

【样例 3】

见选手目录下的 *pill/pill3.in* 与 *pill/pill3.ans*。
该组测试数据满足数据范围中描述的子任务 1 的限制。

【样例 4】

见选手目录下的 *pill/pill4.in* 与 *pill/pill4.ans*。
该组测试数据满足数据范围中描述的子任务 2 的限制。

【样例 5】

见选手目录下的 *pill/pill5.in* 与 *pill/pill5.ans*。
该组测试数据满足数据范围中描述的子任务 3 的限制。

【数据范围】

本题开启捆绑测试。

对于所有测试数据，保证： $1 \leq K, u, v \leq N \leq 2 \times 10^5$ ，保证从任何一座宫殿出发都能到达其他宫殿， $1 \leq C_i \leq K$ 。

子任务	特殊性质	分数
1	$N \leq 2000$	20
2	每座宫殿最多与两座宫殿相连	20
3	无	60

掌牧松 (mrzhang)

【题目背景】

网球君深入魔丸掌牧松的古老藏书阁，发现了一份记载着破解结界的关键知识的魔法卷轴。这份卷轴被掌牧松施加了强大的防护魔法。

【题目描述】

为了防止结界破解方法被外人获取，掌牧松会对卷轴施加扭曲魔法。这个魔法会在被施加的咒语的任意位置插入一个字符。

为了让解密过程更加困难，掌牧松选择了卷轴中的其中一些咒语并对它们施加了这种魔法。

魔丸族崇尚严格的秩序，因此要求原始卷轴中的咒语一定是按字典序非递减排列的。网球君需要从被施法的咒语中删除一个字符（以解除魔法），从而恢复出原始卷轴的某个版本。

然而，恢复古老卷轴的方法可能不止一种。为了确保不遗漏任何重要的破解信息，网球君需要尝试所有可能的原始卷轴版本。为了估算破解所需的最大时间，他需要知道所有可能的原始卷轴版本的数量。

由于答案可能非常大，请输出对 $10^9 + 7$ 取模后的结果。

注意掌牧松可能设置了空咒语（但他一定会对其施加魔法，因此空魔法现在长度为 1）。

有可能这份卷轴是掌牧松设置的陷阱，因此答案可能不存在。

字典序规则

如果咒语 a 在字典序上小于咒语 b ，当且仅当满足以下条件之一：

- a 是 b 的前缀，且 $a \neq b$ ；
- 在 a 和 b 第一个不同的位置， a 的字典序比 b 的字典序更小。

时间紧迫！ 网球君必须在掌牧松发现之前破解卷轴，请帮助他计算出所有可能的破解方案数！

【输入格式】

从文件 *mrzhang.in* 中读入数据。

第一行包含一个整数 n ($1 \leq n \leq 10^5$)，表示卷轴中的咒语数量。

接下来的 n 行，每行包含一个仅由小写英文字母组成的字符串，第 i 行表示卷轴中的第 i 个咒语。每个咒语的长度至少为 1。所有咒语长度之和不超过 10^6 。

【输出格式】

输出到文件 *mrzhang.out* 中。

输出一个整数，表示可以恢复出原始卷轴的方案数，对 $10^9 + 7$ 取模。

【样例 1 输入】

```
1 3
2 abcd
3 zaza
4 ataka
```

【样例 1 输出】

```
1 4
```

【样例 1 解释】

网球君有 4 种方法解除魔法，恢复出符合字典序要求的原始卷轴。

【样例 2 输入】

```
1 3
2 abc
3 bcd
4 a
```

【样例 2 输出】

```
1 0
```

【样例 2 解释】

这份卷轴可能是掌牧松设置的陷阱，无法恢复出符合要求的原始版本。

【样例 3】

见选手目录下的 *mrzhang/mrzhang3.in* 与 *mrzhang/mrzhang3.ans*。

该组测试数据满足数据范围中描述的子任务 1 的限制。

【样例 4】

见选手目录下的 *mrzhang/mrzhang4.in* 与 *mrzhang/mrzhang4.ans*。

该组测试数据满足数据范围中描述的子任务 2 的限制。

【样例 5】

见选手目录下的 *mrzhang/mrzhang5.in* 与 *mrzhang/mrzhang5.ans*。
该组测试数据满足数据范围中描述的子任务 3 的限制。

【样例 6】

见选手目录下的 *mrzhang/mrzhang6.in* 与 *mrzhang/mrzhang6.ans*。
该组测试数据满足数据范围中描述的子任务 4 的限制。

【数据范围】

本题开启捆绑测试。

记 L 为卷轴长度总和, 对于所有测试数据, 保证: $L \leq 10^7 \cdot 30$, 字符串仅由小写字母构成。

子任务	特殊性质	分数
1	$L \leq 300$	20
2	$L \leq 3000$	20
3	$L \leq 10^6$	30
4	$L \leq 10^7$	30