

NOIP2023 模拟赛

题目名称	枣树	对弈	括号	染色
题目类型	传统题	传统题	传统题	传统题
目录	tree	gomoku	kacbret	color
可执行文件名	tree	gomoku	kacbret	color
输入文件名	tree.in	gomoku.in	kacbret.in	color.in
输出文件名	tree.out	gomoku.out	kacbret.out	color.out
每个测试点时限	1 秒	1 秒	2 秒	2 秒
内存限制	256 MB	256 MB	1024 MB	1024 MB
子任务数目	5	4	7	7
测试点是否等分	否	否	否	否

提交源程序文件名

对于 C++ 语言	tree.cpp	gomoku.cpp	kacbret.cpp	color.cpp
-----------	----------	------------	-------------	-----------

编译选项

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

注意事项（请仔细阅读）

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考各省的具体要求。
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. 选手提交的程序源文件必须不大于 100KB。
7. 程序可使用的栈空间内存限制与题目的内存限制一致。
8. 只提供 Windows 格式附加样例文件。
9. 评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以此为准。

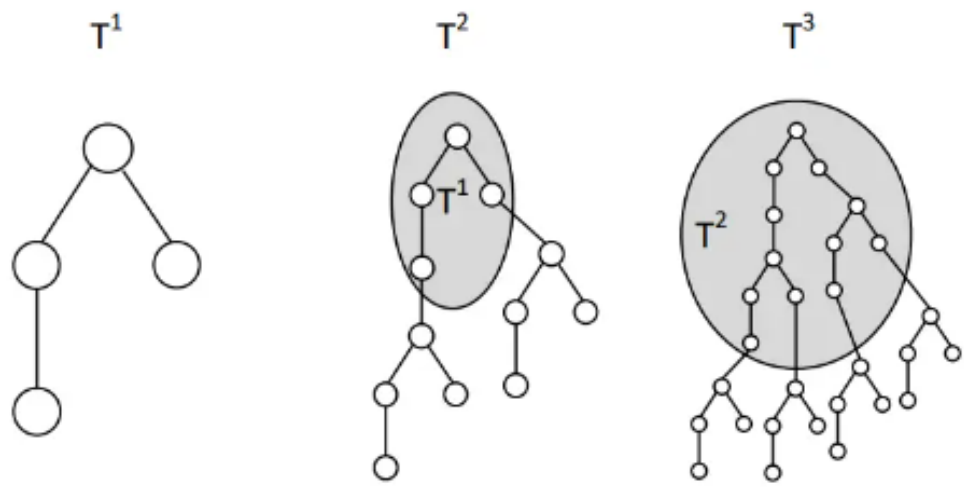
枣树 (tree)

【题目背景】

在我的后园，可以看见墙外有两株树，一株是枣树，还有一株不是枣树，是叶子是枣树的树，我称之为枣树树。

【题目描述】

给定一棵以 1 为根的有根树 T_1 ，定义 T_m 为在 T_{m-1} 的每个叶子下面接一棵 T_1 得到的树。叶子的定义是除根以外的度数为 1 的节点。



求 T_m 的直径的长度，这里长度指路径上的点数。由于答案可能很大，你只需要输出答案对 2^{64} 取模后的结果。

【输入格式】

- 从文件 `tree.in` 中读入数据。
- 第一行两个整数 n, m 。
- 第二行 $n - 1$ 个整数 p_2, p_3, \dots, p_n ，分别代表第 $2, 3, \dots, n$ 个节点的父亲。

【输出格式】

- 输出到文件 `tree.out` 中。
- 输出一行一个整数，表示 T_m 的直径长度对 2^{64} 取模的结果。

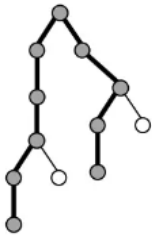
【样例 1 输入】

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

【样例 1 输出】

```
1 10
```

【样例 1 解释】



【数据范围】

对于 100% 的数据， $3 \leq n \leq 2 \times 10^5, 1 \leq m \leq 2 \times 10^5, 1 \leq p_i < i$ 。

子任务编号	$n \leq$	$m \leq$	分值
1	5000	1	19
2	2×10^5	1	10
3	5000	5000	20
4	5000	2×10^5	19
5	2×10^5	2×10^5	32

对弈 (gomoku)

【题目背景】

弈秋，通国之善弈者也，后面忘了。

【题目描述】

简要题意：

给定正整数 n, m, k ，判断

$$S_{n,m,k} = \{((a_1, b_1), (a_2, b_2), \dots, (a_t, b_t)) \mid (\forall i \in [1, t] \cap \mathbb{Z} : a_i \in [1, n] \cap \mathbb{Z}, b_i \in [1, m] \cap \mathbb{Z}) \\ \wedge (\nexists i, j \in [1, t] \cap \mathbb{Z}, i \neq j : a_i = a_j, b_i = b_j) \\ \wedge (\{(a_1, b_1), (a_2, b_2), \dots, (a_t, b_t)\} = \{(x, y) \mid x \in [1, n] \cap \mathbb{Z}, y \in [1, m] \cap \mathbb{Z}\}) \\ \wedge (\nexists i_1, i_2, \dots, i_k \in [1, t] \cap \mathbb{Z}, i_1 \equiv i_2 \equiv \dots \equiv i_k \pmod{2} : \max(|a_{i_1} - a_{i_2}|, |b_{i_1} - b_{i_2}|) = 1, \\ (\forall 3 \leq j \leq k, j \in \mathbb{Z} : a_{i_j} - a_{i_{j-1}} = a_{i_{j-1}} - a_{i_{j-2}}, b_{i_j} - b_{i_{j-1}} = b_{i_{j-1}} - b_{i_{j-2}}))\}$$

是否为空集。若否，给出 $S_{n,m,k}$ 中的任意一个元素。

(温馨提示：注意式子中的括号匹配)

非简要题意：

构造一种 $n \times m$ 棋盘上的 k 子棋平局局面或报告无解。

【输入格式】

从文件 **gomoku.in** 中读入数据。

一行三个整数，表示 n, m, k 。

【输出格式】

输出到文件 **gomoku.out** 中。

若 $S_{n,m,k}$ 为空集，输出一行 **-1** 即可。

否则输出若干行，第 i 行输出 a_i, b_i ，代表你给出的元素是 $((a_1, b_1), (a_2, b_2), \dots, (a_t, b_t))$ ，其中 t 为你输出的行数。

【样例 1 输入】

1 4 4 3

【样例 1 输出】

```
1 1 2
2 1 1
3 1 4
4 1 3
5 2 1
6 2 3
7 2 2
8 2 4
9 3 3
10 3 2
11 3 4
12 3 1
13 4 1
14 4 4
15 4 3
16 4 2
```

【样例 2 输入】

```
1 2 3 3
```

【样例 2 输出】

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

【数据范围】

对于 100% 的数据， $1 \leq n, m \leq 500, 1 \leq k \leq 10^9$ 。

子任务编号	数据范围	分值
1	$2 \leq n, m \leq 4$	30
2	$\min(n, m) = 1$	15
3	$n, m \geq 2, \min(n, m) < k$	15
4	$n, m \geq 2$	40

括号 (kacbret)

【题目背景】

题面要形式化一点。这个题面显然远远不够形式化。

【题目描述】

- 对于括号串 S ，定义

$$f(S) = \{T \mid \exists A, B, C, S = A + B + C, T = A + (+ B +) + C\}.$$

- 对于自然数 k 和括号串 S ，定义

$$f^{(k)}(S) = \begin{cases} S, & k = 0 \\ f(f^{(k-1)}(S)), & k > 0 \end{cases}.$$

- 对于括号串 S ，定义

$$\mathcal{F}(S) = \bigcup_{k \geq 0} f^{(k)}(S).$$

- 对于括号串 S ，定义

$$\mathcal{G}(S) = \{T \mid S \in \mathcal{F}(T)\}.$$

- 约定 (的字典序小于) 的字典序。
- 定义两个括号串的大小比较为它们对应的字符串的字典序的大小比较。

给定括号串 S ，求 $\min_{T \in \mathcal{G}(S)} T$ 。

【输入格式】

从文件 **kacbret.in** 中读入数据。

输入一行一个括号串 S ，保证 S 的第一个字符为)。

【输出格式】

输出到文件 **kacbret.out** 中。

输出 $\min_{T \in \mathcal{G}(S)} T$ 。

【样例 1 输入】

```
1 )((()((())))
```

【样例 1 输出】

```
1 )((()((())))
```

【样例 2】

见选手目录下的 *kacbet/kacbet2.in* 与 *kacbet/kacbet2.ans*。

【样例 3】

见选手目录下的 *kacbet/kacbet3.in* 与 *kacbet/kacbet3.ans*。

【样例 4】

见选手目录下的 *kacbet/kacbet4.in* 与 *kacbet/kacbet4.ans*。

【数据范围】

对于 100% 的数据， $2 \leq |S| \leq 3 \times 10^5$ 。

子任务编号	数据范围	特殊性质	分值
1	$ S \leq 20$	无	10
2	$ S \leq 80$	无	10
3	$ S \leq 300$	无	10
4	$ S \leq 4000$	无	10
5	$ S \leq 10^5$	A	15
6	$ S \leq 10^5$	无	15
7	$ S \leq 3 \times 10^5$	无	30

特殊性质 A：保证答案的长度不超过 200。

染色 (color)

【题目背景】

Alice 和 Bob 怎么又在玩游戏？

【题目描述】

Aob 和 Blice 又弄来了一个简单连通无向图 G ，在上面玩游戏。初始时， G 中所有节点均为白色。

游戏过程如下：

1. 游戏开始时，Aob 将 G 中一个节点染黑，然后交给 Blice。
2. Blice 拿到 G 后，进行若干轮操作直到整张图都变成黑色。每次操作如下所示：
 - 取出 G 中 1 个或 2 个与某个黑色节点相邻的白色节点，将它们染黑。如果选了两个白色节点，这两个白色节点可以与不同的黑色节点相邻。

游戏结束后，Aob 所得到的分数为 Blice 在 2. 中的操作次数。一次操作中染黑两个节点，算作一次操作。

Aob 希望最大化这个得分，Blice 希望最小化这个得分。求双方均采用最优策略的情况下 Aob 的得分是多少。

【输入格式】

从文件 `color.in` 中读入数据。

第一行输入两个整数 n, m ，分别表示 G 的点数和边数。

接下来 m 行每行输入两个整数 (x, y) ，表示 G 中的一条边 (x, y) 。

【输出格式】

输出到文件 `color.out` 中。

一行一个整数，表示所求的得分。

【样例 1 输入】

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

【样例 1 输出】

```
1 2
```

【样例 1 解释】

Aob 的最优策略是第一步染黑 1 号点或者 3 号点，此时他的得分将为 2；若他第一步染了 2 号点，则他的得分将为 1，不优。

【样例 2 输入】

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

【样例 2 输出】

```
1 2
```

【样例 2 解释】

不管 Aob 第一步染哪个点他的得分都将为 2。

【样例 3 输入】

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

【样例 3 输出】

12

【样例 4】

见选手目录下的 `color/color4.in` 与 `color/color4.ans`。

【数据范围】

对于 100% 的数据， $2 \leq n \leq 3 \times 10^5$ ， $n - 1 \leq m \leq 3 \times 10^5$ ， $1 \leq x, y \leq n$ 。保证 G 连通，无重边，无自环。

子任务编号	数据范围	特殊性质	分值
1	$n \leq 20$	无	10
2	$n, m \leq 200$	无	10
3	$n, m \leq 4000$	无	20
4	$n \leq 3 \times 10^5$	$m = n - 1$	5
5	$n \leq 3 \times 10^5$	$m = n$	10
6	$n \leq 3 \times 10^5$	A	20
7	$n \leq 3 \times 10^5$	无	25

特殊性质 **A**：保证 G 中每条边至多出现在一个简单环中。