

NOI2025 广东省队集训

GDOI

时间：2025 年 4 月 xx 日

题目名称	平衡树	上升树	并行程序
题目类型	传统题	传统题	传统题
目录	balance	lis	program
可执行文件名	balance	lis	program
输入文件名	balance.in	lis.in	program.in
输出文件名	balance.out	lis.out	program.out
每个测试点时限	2 秒	2 秒	9 秒
内存限制	512 MB	512 MB	2048 MB
子任务数目	20	20	7
测试点是否等分	是	是	否

提交源程序文件名

对于 C++ 语言	balance.cpp	lis.cpp	program.cpp
-----------	-------------	---------	-------------

编译选项

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

注意事项（请仔细阅读）

1. 测试机器：CPU(AMD Ryzen 5 3600 6-Core Processor *12), RAM 8.0G。
2. 系统环境：NOI Linux 2.0(基于 Ubuntu 20.04.1)。
3. 文件名（程序名和输入输出文件名）必须使用英文小写。
4. C/C++ 中函数 `main()` 返回类型必须是 `int`，程序正常结束返回值必须是 0。
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。

平衡树 (balance)

【题目描述】

小 A 有一棵 n 个点的无根树，树上的每个点有一个权值，点 i 的权值是 a_i 。树上还有一些边是关键边。定义一条关键边的不平衡度为：将其删去后，得到的两棵子树的点权和之差的绝对值。定义整棵树的不平衡度为所有关键边的不平衡度的最大值。

给定 k, L, R ，以及 k 条关键边。小 A 每次可以将一个点的权值增加 1，他至少要进行 L 次操作，至多进行 R 次操作。小 A 想知道，进行若干次操作后，这棵树的不平衡度最小能是多少？

小 A 有 Q 次询问，每次都会给定 k, L, R 以及 k 条这次询问中的关键边，你需要对每次询问输出答案。

【输入格式】

从文件 `balance.in` 中读入数据。

第一行 2 个整数 n, Q 。

第二行 n 个以空格分隔的整数 a_1, \dots, a_n 表示点的权值。

接下来 $n - 1$ 行每行两个以空格分隔的整数 u_i, v_i 表示编号为 i 的边。

接下来有 Q 次询问，每个询问的第一行是 3 个整数 k, L, R ，第二行是 k 个空格分开的整数表示关键边的编号（从小到大给出且不重复）。

【输出格式】

输出到文件 `balance.out` 中。

输出 Q 行，第 i 行一个整数表示第 i 次询问中树的最小不平衡度。

【样例 1 输入】

```
1 5 2
2 1 4 2 6 10
3 1 4
4 2 3
5 2 4
6 4 5
7 3 0 5
8 2 3 4
```

9

2 10 20

10

1 3

【样例 1 输出】

1

14

2

16

【样例 2】

见选手目录下的 *balance/balance2.in* 与 *balance/balance2.ans*。
该样例满足测试点 3 ~ 5 的约束条件。

【样例 3】

见选手目录下的 *balance/balance3.in* 与 *balance/balance3.ans*。
该样例满足测试点 6 ~ 9 的约束条件。

【样例 4】

见选手目录下的 *balance/balance4.in* 与 *balance/balance4.ans*。
该样例满足测试点 10 ~ 12 的约束条件。

【样例 5】

见选手目录下的 *balance/balance5.in* 与 *balance/balance5.ans*。
该样例满足测试点 13 ~ 15 的约束条件。

【样例 6】

见选手目录下的 *balance/balance6.in* 与 *balance/balance6.ans*。
该样例满足测试点 16 ~ 20 的约束条件。

【数据范围】

对于所有测试点， $2 \leq n \leq 3 \times 10^5$ ， $1 \leq Q \leq n$ ， $0 \leq a_i \leq 10^6$ ， $0 \leq L \leq R \leq 10^9$ ， $1 \leq k \leq n - 1$ ， $\sum k \leq n$ 。

测试点	$n \leq$	特殊性质
1 ~ 2	5	$v_i, L, R \leq 10$
3 ~ 5	10^5	$u_i = i, v_i = i + 1$
6 ~ 9	10^5	$u_i = 1, v_i = i + 1$
10 ~ 12	3×10^5	$L = R, Q = 1$
13 ~ 15	3×10^5	$a_i = 0, Q = 1$
16 ~ 20	3×10^5	无

上升树 (lis)

【题目描述】

小 A 又有一棵 n 个点的无根树，树上的每个点还是有一个权值 a_i 。对于树上的两个点 $u \rightarrow v$ ，定义 $L(u, v)$ 为将树上 $u \rightarrow v$ 的最短路径经过的点的点权列成序列，其最长上升子序列 (LIS) 的长度。

现在小 A 需要删掉树上的一个点 u ，删掉 u 后这棵树会被分为若干棵子树 S_1, \dots, S_k 。定义 $F(u) = \max_{i=1}^k (\max_{u,v \in S_i} L(u, v))$ ，即删除点 u 后森林中 LIS 最长的路径，小 A 想要你求出 $\min_u F(u)$ ，即删除一个点使得森林中 LIS 最长的路径的 LIS 长度最小。

【输入格式】

从文件 **lis.in** 中读入数据。

第一行一个整数 n 。

第二行 n 个整数 a_1, \dots, a_n 表示点的权值。

接下来 $n - 1$ 行每行两个整数 u, v 表示一条边。

【输出格式】

输出到文件 **lis.out** 中。

输出一行一个整数表示答案，即 $\min_u F(u)$ 。

【样例 1 输入】

```
1 10
2 1 3 5 7 9 8 9 2 3 4
3 1 2
4 2 3
5 3 4
6 1 5
7 5 6
8 1 7
9 7 8
10 7 9
11 7 10
```

【样例 1 输出】

12

【样例 2】

见选手目录下的 *lis/lis2.in* 与 *lis/lis2.ans*。
该样例满足测试点 1 ~ 2 的约束条件

【样例 3】

见选手目录下的 *lis/lis3.in* 与 *lis/lis3.ans*。
该样例满足测试点 5 ~ 8 的约束条件。

【样例 4】

见选手目录下的 *lis/lis4.in* 与 *lis/lis4.ans*。
该样例满足测试点 9 ~ 10 的约束条件。

【样例 5】

见选手目录下的 *lis/lis5.in* 与 *lis/lis5.ans*。
该样例满足测试点 11 ~ 14 的约束条件。

【数据范围】

对于所有测试点， $1 \leq n \leq 10^5$ ， $1 \leq a_i \leq n$ 。

测试点	$n \leq$	特殊性质
1 ~ 2	1000	无
3 ~ 4	5000	无
5 ~ 8	10^5	$u_i = i, v_i = i + 1$
9 ~ 10	10^5	$a_i \leq 10$
11 ~ 14	3×10^4	无
15 ~ 20	10^5	无

并行程序 (program)

【题目描述】

有 n 个程序。所有程序共享一个全局整数型变量 x ，此外，每个程序都有一个私有的计数器 y 。每个程序都由一连串的指令组成，每个指令都属于以下四种类型之一：

- **W**: 将全局变量的值 x 载入私有计数器 y 。
- **Z**: 将私有计数器 y 的值写入全局变量 x 。
- **+** c : 将 y 的值加一正常数 c 。
- **-** c : 将 y 的值减一正常数 c 。

这些程序将会并行运行。所有计数器 y 和变量 x 的初始值都是 0。这些程序的指令交错执行，即所有程序的所有指令都是一个接一个地执行，对于每个时刻，每个程序满足它的指令的一个前缀以一定顺序被执行。

计算所有程序并行执行后变量 x 的最小可能值是多少。

【输入格式】

从文件 `program.in` 中读入数据。

第一行一个整数 t ，表示该测试点的数据组数。

每组数据第一行一个整数 n ，表示程序的数量。每组数据的第 2 到 $2n + 1$ 行，每两行表示一个程序。

每个程序的第一行一个整数 l_i ，表示该程序的指令数量。每个程序的第二行是 l_i 个空格分隔的指令，每个指令的格式是如下四种之一：

- 一个字符 **W**: 表示载入指令；
- 一个字符 **Z**: 表示写入指令；
- 空格分隔的一个字符 **+** 和一个数字 c_{ij} : 表示给私有计数器加 c_{ij} ；
- 空格分隔的一个字符 **-** 和一个数字 c_{ij} : 表示给私有计数器减 c_{ij} 。

【输出格式】

输出到文件 `program.out` 中。

每组数据输出一行一个整数，表示答案。

【样例 1 输入】

```
1 2
2 2
3 12
4 W + 2 Z W + 2 Z W + 2 Z W + 2 Z
5 12
6 W + 3 Z W + 3 Z W + 3 Z W + 3 Z
7 3
8 3
9 W W - 5
10 5
11 + 9 Z + 1 Z W
12 8
13 + 10 Z - 2 Z - 5 W - 1 Z
```

【样例 1 输出】

```
1 5
2 7
```

【样例 1 解释】

对于第一组数据，得到最小的 x 程序指令执行顺序如下表。

			W				+ 2					Z	W		+ 2	Z	W	+ 2		Z	W	+ 2	Z	
	W	+ 3		Z	W	+ 3		Z	W	+ 3	Z		W		+ 3		Z	W	+ 3		Z	W	+ 3	Z
y_1	0	0	0	0	0	0	2	2	2	2	2	2	2	2	4	4	4	6	6	6	6	8	8	8
y_2	0	3	3	3	3	6	6	6	9	9	9	9	9	2	2	2	2	2	5	5	5	5	5	5
x	0	0	0	3	3	3	3	6	6	6	9	2	2	2	2	4	4	4	4	6	6	6	8	5

该样例满足子任务的约束。

【样例 2】

见选手目录下的 `program/program2.in` 与 `program/program2.ans`。

【样例 3】

见选手目录下的 `program/program3.in` 与 `program/program3.ans`。

【样例 4】

见选手目录下的 `program/program4.in` 与 `program/program4.ans`。

【数据范围】

- $1 \leq t \leq 10^5$
- $1 \leq n \leq 10^5$
- $1 \leq l_i \leq 10^5$
- $1 \leq c_{ij} \leq 10^9$
- 一个测试点中的所有数据的 $\sum l_i$ 之和不超过 10^6

子任务的列表如下：

子任务	额外约束	分数
1	$t \leq 10$ ，每组数据中 $\sum l_i \leq 10$	3
2	$n = 2$ ，每个测试点中的所有数据的 $\sum l_i$ 之和不超过 10^4	10
3	$n = 2$	15
4	$n \leq 50$ ，每个测试点中的所有数据的 $\sum l_i$ 之和不超过 10^4	13
5	每个测试点中的所有数据的 $\sum l_i$ 之和不超过 10^4	21
6	$l_i \leq 10$	22
7	无	16