

CCF NOIP

模拟赛

题目名称	染色游戏	春风化雨	区间	天接云涛连晓雾
题目类型	传统型	传统型	传统型	传统型
目录	game	rain	range	tjytlxw
可执行文件名	game	rain	range	tjytlxw
输入文件名	game.in	rain.in	range.in	tjytlxw.in
输出文件名	game.out	rain.out	range.out	tjytlxw.out
测试点时间限制	1.0 秒	2.0 秒	4.0 秒	2.0 秒
测试点空间限制	512 MiB	512 MiB	512 MiB	512 MiB
题目总分值	100	100	100	100
是否设置子任务	否	否	否	是
测试点/子任务个数	4	17	20	10
测试点/子任务是否等分	是	否	是	是

A. 染色游戏

题目描述

Alice 和 Bob 在玩一个染色游戏。游戏在一张 N 个点 $N - 1$ 条边的连通图上进行，Bob 想要围住 Alice，而 Alice 想要逃出 Bob 的包围。

游戏开始时，Alice 将 1 号点涂成了黑色表示占领了 1 号点，Bob 将点集 S 中的所有点涂成了白色表示占领了这 $|S|$ 个点，保证 1 不在 S 中。接下来两个人轮流进行操作，由 Alice 先手，每轮中轮到的玩家可以从一个被自己占领的点出发（对于 Alice 为黑色点对于 Bob 为白色点），选择一个相邻且未被染色的点，占领该点并染上自己的颜色。如果不存在可以染色的点，那么这位玩家必须跳过这个回合。当所有点都被染完色时，游戏结束。

Alice 和 Bob 约定了一个图中的非空点集 T ，如果游戏结束时 T 中的点全都涂成白色，则代表 Bob 成功围住了 Alice，Bob 获胜。反之一定存在一个 T 中的点被涂成黑色，那么 Alice 获胜。注意这里的 T 可能会包含 S 中的点和 1 号点。

Alice 和 Bob 都会使用最优策略。Bob 注意到，在有些局面下，Alice 优势很大，如果能让 Alice 主动跳过 Alice 的一些行动回合来获得一个更加公平的局面，这个游戏会更有可玩性。Bob 想知道，如果 Alice 跳过前 k 个回合之后自己能够获胜，那么这个 k 的最小值是多少。Alice 只会跳过 Alice 的前 k 个回合，并且在剩下的回合中采用最优策略，即你可以理解为 Bob 在 Alice 的第一回合行动之前额外行动了 k 个回合。注意如果 Bob 在 Alice 跳过的一个回合中没有合法行动，那么 Bob 仍需按照规则跳过自己的回合。如果在原图上就是 Bob 获胜那么输出 0。如果无论 k 多大时 Bob 都不能取胜，则输出 `Impossible`。

由于这个图可能很大，我们用如下的方式生成。

- 首先生成一个含有标号为 1 到 n 一共 n 个点的空图。
- 接下来加入 $n - 1$ 条链，第 i 条链记作 (u_i, v_i, l_i) ，其中 $1 \leq u_i, v_i \leq n$ 且 $u_i \neq v_i$ 。
 - 首先我们加入 l_i 个点，记作 $x_1^i, x_2^i, \dots, x_{l_i}^i$ 。
 - 然后在 $(u_i, x_1^i), (x_1^i, x_2^i), (x_2^i, x_3^i), \dots, (x_{l_i-1}^i, x_{l_i}^i), (x_{l_i}^i, v_i)$ 之间连上无向边。
 - 在这次操作之后，本轮中新加入的 l_i 个点不会再与其他的点之间连边，即不同的链中的 $x_1^i \dots x_{l_i}^i$ 均为互不相同的点。特别地，如果 $l = 0$ ，那么就不添加新点，直接在 (u_i, v_i) 之间连上无向边。

保证 S 集合以及 T 集合的点均为一开始生成的 n 个点之一。

输入格式

第一行输入一个整数 C ，表示数据组数。

对于每组数据：

- 第一行输入三个整数 $n, |S|, |T|$ ($1 \leq |S| \leq n - 1, 1 \leq |T| \leq n$) 。
- 接下来 $n - 1$ 行每行输入 3 个非负整数 u_i, v_i, l_i ($1 \leq u_i, v_i \leq n, 0 \leq l_i \leq 10^6$)，表示题面中的第 i 条链。
- 接下来一行输入 $|S|$ 个数 $s_1 \dots s_{|S|}$ 表示 S 集合中的所有元素 ($2 \leq s_i \leq n$ 且不重复) 。
- 接下来一行输入 $|T|$ 个数 $t_1 \dots t_{|T|}$ 表示 T 集合中的所有元素 ($1 \leq t_i \leq n$ 且不重复) 。

即每组数据按照如下格式输入：

$$\begin{array}{l} n \ |S| \ |T| \\ u_1 \ v_1 \ l_1 \\ u_2 \ v_2 \ l_2 \\ \dots \\ u_{n-1} \ v_{n-1} \ l_{n-1} \\ s_1 \ s_2 \ \cdots \ s_{|S|} \\ t_1 \ t_2 \ \cdots \ t_{|T|} \end{array}$$

保证 $u_i \neq v_i$ （即没有自环）， 保证没有相同的 (u_i, v_i) 对（即没有重边）， 保证给出的图是一个连通图。

输出格式

输出 C 行，对于每组测试数据，输出为了让 Bob 取胜 Alice 至少要跳过的回合数 k 。如果在原图上就是 Bob 获胜那么输出 0。如果无论 k 多大时 Bob 都不能取胜，则输出 `Impossible`。

样例 #1

样例输入 #1

```
3
6 5 2
1 2 0
2 3 0
2 4 0
3 5 0
4 6 0
5 6
3 4
6 5 2
1 2 1
2 3 0
2 4 0
3 5 0
4 6 0
5 6
3 4
5 4 2
1 2 1
1 3 1
2 4 0
3 5 0
4 5
2 3
```

样例输出 #1

```
1
0
0
```

数据范围

对于所有数据， $1 \leq C \leq 10000$, $3 \leq n \leq 500$, $1 \leq |S| \leq n - 1, 1 \leq |T| \leq n$, $\sum n \leq 2.5 \times 10^5$, $0 \leq l_i \leq 10^6$ 。

测试点编号	$C \leq$	$n \leq$	$l_i \leq$	其他约定
1	10	12	0	
2				$ S = 1$
3				$ T = 1$
4				

B. 春风化雨

题目描述

Sophia 厌倦了法术世界唯实力主义的生活。

她打算回到凡间种玫瑰，于是她种了一片巨大的 $n \times n$ 的玫瑰田，然后借助雨水的浇灌让这些玫瑰自然成长。

然而雨很调皮，会首先按行浇灌，然后按列浇灌，另外有些地形不适合种玫瑰。

她很累了，所以想种的玫瑰数量尽量少。

通俗地说来，给定一个 $n \times n$ 的矩阵和一些可以填数的格子，请你在这些格子里面填正整数，要求这些正整数每行每列的最大值是给定值，求正整数之和的最小值。

输入格式

第一行三个正整数 n, m, k ，其中 k 表示所有限定数中的最大值。

第二行 n 个整数，从上到下表示每一行的限定数。

第三行 n 个整数，从左到右表示每一列的限定数。

之后的 m 行每行两个整数，表示一个可以种玫瑰的位置的坐标。

输出格式

一个整数，表示答案。

样例 #1

样例输入 #1

```
3 9 1
1 1 1
1 1 1
1 1
1 2
1 3
2 1
2 2
2 3
3 1
3 2
3 3
```

样例输出 #1

```
3
```

样例 #2

样例输入 #2

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

样例输出 #2

4

样例 #3

样例输入 #3

5 10 5
1 2 3 4 5
1 2 3 4 5
1 5
2 4
3 3
4 2
5 1
3 5
4 5
5 5
5 4
5 3

样例输出 #3

22

数据范围

测试点编号	测试点分值	$n =$	特殊性质
1	5	3	$k = 1$
2	5	4	$k = 1$
3	5	5	$k = 1$
4	5	6	$k = 1$
5	5	7	$k = 1$
6, 7, 8, 9, 10	5	100	$k = 1$
11	5	500	
12	5	1000	
13	5	1500	
14	5	2000	
15, 16, 17	10	2500	

C. 区间

题目描述

千羽是个可爱的女孩子，她在序列王国收到了人们送给她的一些区间。千羽对这些区间产生了很大的兴趣，她希望聪明的你能给予她一些帮助。

给出 n 个数轴上的区间 $[l_1, r_1], [l_2, r_2], \cdots, [l_n, r_n]$ 。千羽定义 $w(L, R)$ 表示其中第 L 个到第 R 个的并覆盖了数轴上多长。换言之， $w(L, R) = |\bigcup_{i=L}^R [l_i, r_i]|$ 。

千羽会像你发出 q 次询问，每次给出 q_l, q_r 。请你求出：如果我从 $[q_l, q_r]$ 中，等概率随机选择一个子区间 $[L, R]$ ，那么得到的 $w(L, R)$ 的期望会是多少。

千羽喜欢整数，所以请你输出答案对 998244353 取模的结果。

输入格式

第一行两个整数 n, q 。

接下来 n 行，每行两个整数 l_i, r_i 。

接下来 q 行，每行两个整数 q_l, q_r 。

输出格式

q 行，每行一个整数表示答案对 998244353 取模的结果。

样例 #1

样例输入 #1

```
2 1
1 5
4 8
1 2
```

样例输出 #1

```
5
```

样例 #2

样例输入 #2

```
5 7
11 11
11 17
6 20
5 9
8 20
1 1
1 2
3 5
1 3
3 4
2 3
4 4
```

样例输出 #2

```
0
4
499122189
9
11
332748129
4
```

样例 #3

样例输入 #3

```
15 9
910 910
600 630
910 910
60 740
570 580
60 800
260 910
910 910
670 910
450 470
590 610
```

```
60 260
800 910
60 260
260 260
7 10
1 3
8 11
5 15
3 13
6 10
2 10
11 13
9 12
```

样例输出 #3

```
362
20
164
574747259
604997180
332748635
665496868
332748316
200
```

数据范围

对于所有数据， $1 \leq n \leq 8 \times 10^5$ ， $1 \leq q \leq 2 \times 10^6$ ， $0 \leq l_i \leq r_i < 998244353$ ， $1 \leq q_l \leq q_r \leq n$ 。

测试点编号	$n \leq$	$q \leq$	特殊性质
1, 2	100	100	
3, 4	1000	1000	
5	5000	5000	
6, 7, 8	1000	1×10^5	
9, 10	1×10^5	1	$q_l = 1, q_r = n$
11, 12, 13, 14, 15	1×10^5	1×10^5	
16, 17, 18	5×10^5	5×10^5	
19, 20			

D. 天接云涛连晓雾

题目描述

有 n 株草排成一行，第 i 株草有一个原始高度 a_i ，还有一个目标高度 b_i 。我们认为高度 > 0 的草是存活的，而高度 $= 0$ 的草已经死亡。

你想要修剪这些草，有两种修剪操作：

- 选定一个区间 $[l, r]$ ，将 $[l, r]$ 内所有仍然存活的草的高度都增加 1；
 - 选定一个区间 $[l, r]$ ，将 $[l, r]$ 内所有仍然存活的草的高度都降低 1；
- 如果一株草在经过这个操作后高度变为了 0，那么就被视为已经死亡，这株草不再受后续的操作影响。

求让所有草的高度从 a_i 变成 b_i 所需要两种操作的最小数目。

输入格式

输入中包含多组测试数据。

输入的第一行包含一个整数 T ，表示数据组数。

对于每组数据，第一行一个整数 n 。

接下来一行，包含 n 个整数，描述 a_1, a_2, \cdots, a_n 。

接下来一行，包含 n 个整数，描述 b_1, b_2, \cdots, b_n 。

输出格式

对于每组数据，输出一行一个整数，表示答案。若合法方案不存在，则输出 -1 。

样例 #1

样例输入 #1

```
3
5
1 1 1 1 1
2 0 2 0 2
6
1 1 4 5 1 4
1 9 1 9 8 10
8
2 0 0 4 1 1 1 5
2 0 0 5 0 1 1 7
```

样例输出 #1

```
3
18
4
```

数据范围

对于所有数据， $1 \leq n \leq 10^6$ ， $1 \leq \sum n \leq 3 \times 10^6$ ， $0 \leq a_i, b_i \leq 10^9$ 。

子任务编号	n	a_i, b_i	特殊限制
1	≤ 3		
2	≤ 4		
3	≤ 5		
4	≤ 100	≤ 100	$T \leq 10$
5	≤ 500	≤ 500	$T \leq 10$
6	$\leq 3\,000$	≤ 500	$T \leq 10$
7	$\leq 10^5$	≤ 500	$T \leq 10$
8			$b_i > 0$
9			a_i, b_i 单调不减
10			无