

CCF 全国信息学奥林匹克联赛 NOIP2018 复赛（模拟赛）

提高组 day 1

（请选手务必仔细阅读本页内容）

一、题目概况

中文题目名称	梦境	玩具	飘雪圣域
英文题目名称	dream	toy	icekingdom
可执行文件名	dream	toy	icekingdom
输入文件名	dream.in	toy.in	icekingdom.in
输出文件名	dream.out	toy.out	icekingdom.out
每个测试点时限	1s	1s	1s
测试点数目	10	10	10
每个测试点分值	10	10	10
附加样例文件	有	有	有
结果比较方式	全文比较（过滤行末空格及文末回车）		
题目类型	传统题	传统题	传统题
运行内存上限	256M	256M	256M

二、提交源程序文件名

对于 C++ 语言	dream.cpp	toy.cpp	icekingdom.cpp
对于 C 语言	dream.c	toy.c	icekingdom.c
对于 pascal 语言	dream.pas	toy.pas	icekingdom.pas

三、编译命令（不包含任何优化开关）

对于 C++ 语言	g++ -o dream dream.cpp -lm	g++ -o toy toy.cpp -lm	g++ -o icekingdom icekingdom.cpp -lm
对于 C 语言	gcc -o dream dream.c -lm	gcc -o toy toy.c -lm	gcc -o icekingdom icekingdom.c -lm
对于 pascal 语言	fpc dream.pas	fpc toy.pas	fpc icekingdom.pas

注意事项

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. 只提供 Linux 格式的样例文件。
4. 特别提醒：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其为准。
5. 真正有用的特别提醒：题目难度和顺序无关。

1. 梦境 (dream.cpp/c/pas)

【问题描述】

智者奥尔曼曾说过：有缘的人即使相隔海角天涯，也会在梦境中相遇。

IcePrince_1968 和 IcePrincess_1968 便是如此。有一天 IcePrincess_1968 突发奇想：为什么不用梦境操控仪器来增加她和 IcePrince_1968 的缘分呢？

IcePrincess_1968 的梦境可以用 n 个区间来表示，第 i 个区间 $[l_i, r_i]$ 表示她的第 i 个梦境会在 l_i 时刻开始，在 r_i 时刻结束（包含 l_i 和 r_i 两个时刻）。因为 IcePrincess_1968 经常做白日梦，所以 n 可能很大。

两个人的梦境不是什么时候都能融合的。只有在一些关键的与另一个人相关的梦境转折点两个人的梦境相遇，才能完成融合，形成浪漫的梦境。IcePrincess_1968 探测到 IcePrince_1968 近期的 m 个与 IcePrincess_1968 相关的梦境转折点，第 i 个转折点 t_i 表示他的第 i 个梦境转折点会在 t_i 时刻出现。因为 IcePrince_1968 和 IcePrincess_1968 很有缘，IcePrince_1968 经常梦到 IcePrincess_1968，所以 m 可能会很大。

当 IcePrincess_1968 的一个梦境包含了 IcePrince_1968 的一个梦境转折点时，两个人的这两段梦境就能得到融合。但要注意 IcePrincess_1968 的每段梦境只能和 IcePrince_1968 的一个梦境转折点融合，类似的，IcePrince_1968 的每个梦境转折点只能和 IcePrincess_1968 的一段梦境融合，否则会引发时空混乱。

IcePrincess_1968 很喜欢做和 IcePrince_1968 相关的梦。所以她想算出她的这些梦境最多能和 IcePrince_1968 的梦境转折点融合出多少个浪漫的梦境。IcePrincess_1968 擅长文学但不擅长计算机，所以只能找你帮忙。

【输入格式】

输入文件名为 `dream.in`。

文件的第一行为有两个正整数 n, m ，表示 IcePrincess_1968 的梦境个数和 IcePrince_1968 的与 IcePrincess_1968 相关的梦境转折点个数。

第 2 至第 $n+1$ 行，每行两个正整数 l_i, r_i ，第 $i+1$ 行的两个数刻画了 IcePrincess_1968 的第 i 段梦境，含义如题面中所述。

第 $n+2$ 至第 $n+m+1$ 行，每行一个正数 t_i ，第 i 行的两个数刻画了 IcePrince_1968 的第 $i-n-1$ 个梦境转折点，含义如题面中所述。

【输出格式】

输出文件名和 `dream.out`。

输出文件仅一行，一个非负整数 N 表示 IcePrincess_1968 最多能获得多少段浪漫的梦境。

【输入输出样例 1】

dream.in	dream.out
2 2	2
1 3	
2 4	
1	
3	

【输入输出样例 1 解释】

IcePrincess_1968 可以将自己的第一段梦境和第一个梦境转折点匹配, 第二段梦境和第二个梦境转折点匹配, 从而获得两段浪漫的梦境。因为 IcePrincess_1968 一共只做了两个梦, 所以这一定是最多的数量。

【输入输出样例 2】

见选手目录下的 dream/dream2.in 和 dream/dream2.ans。

【数据规模和约定】

对于 30% 的数据, $n \leq 10$, $m \leq 10$;

对于 50% 的数据, $n \leq 100$, $m \leq 100$;

对于 70% 的数据, $n \leq 2000$, $m \leq 2000$;

对于 100% 的数据, $n \leq 200,000$, $m \leq 200,000$, $1 \leq l_i, r_i \leq 1,000,000,000$, $1 \leq t_i \leq 1,000,000,000$, 保证对于每段梦境, $l_i \leq r_i$ 。

2. 玩具

(toy.cpp/c/pas)

【问题描述】

这个故事发生在很久以前，在 IcePrincess_1968 和 IcePrince_1968 都还在上幼儿园的时候。

IcePrince_1968 最近迷上了一种玩具，这种玩具中有两种零件：圆球和棍子。棍子的两头可以插在两个圆球上的各一个空洞中，从而将两个圆球连接起来。为了保证玩具的娱乐性，任意一个圆球上的空洞个数总是多于玩具套装中的棍子数。你可以认为圆球是没有体积的，所有棍子的长度均为 1。

IcePrince_1968 喜欢这样玩这种玩具：他先摸出玩具袋里的一个圆球放在地上，然后重复下面的操作 $n-1$ 次：每次从袋中取出一个圆球和一根棍子，然后等概率的从地上的圆球中选择一个，将该圆球和选择的圆球用棍子连起来，使得新的圆球在选中圆球的正上方。

IcePrince_1968 对自己搭出的艺术品很满意，便决定把这个物品送给 IcePrincess_1968 作为生日礼物。然而生日礼物是需要包装的，因为默认圆球没有体积，所以 IcePrince_1968 不用考虑包装盒的长和宽，但是包装盒的高是需要确定的，这里我们假设 IcePrince_1968 是一个非常节俭的孩子，所以包装盒的高总是等于艺术品的高度。IcePrince_1968 想知道自己需要的包装盒的高的期望对质数 p 取模后的值，但他还在上幼儿园，怎么会算呢，于是就请你来帮助他。

【输入格式】

输入文件名为 toy.in。

输入数据仅一行，包含两个正整数 n, p ，表示最终的艺术品中圆球的个数和模数 p 。

【输出格式】

输出文件名为 toy.out。

输入文件仅一行，一个正整数，表示包装盒的高的期望对质数 p 取模后的值。

【输入输出样例 1】

toy.in	toy.out
3 998244353	499122178

【输入输出样例 1 解释】

三个圆球组成的艺术品，高度只可能是 1 或者 2，所以高度的期望是 1.5，在模 998244353 下的期望是 499122178。

【输入输出样例 2】

见选手目录下的 toy/toy2.in 和 toy/toy2.ans。

【数据规模与约定】

对于 30% 的数据，满足 $n \leq 10, p \leq 1,000,007$ ；

对于 50% 的数据，满足 $n \leq 20$ ；

对于 70% 的数据，满足 $n \leq 50$ ；

对于 100% 的数据，满足 $n \leq 200, p \leq 1,000,000,007, p$ 是质数。

3. 飘雪圣域 (icekingdom.cpp/c/pas)

【问题描述】

IcePrincess_1968 和 IcePrince_1968 长大了，他们开始协助国王 IceKing_1968 管理国内事物。

IcePrincess_1968 和 IcePrince_1968 住在一个宁静悠远的王国：IceKingdom —— 飘雪圣域。飘雪圣域有 n 个城镇，编号 $1,2,3\dots n$ 。有些城镇之间有道路，且满足任意两点之间有且仅有一条路径。飘雪圣域风景优美，但气候并不是太好。根据 IcePrince_1968 的气候探测仪，将来会发生 q 场暴风雪。每场暴风雪可以用两个整数 l_i, r_i 刻画，表示这场暴风雪之后，只有编号属于 $[l_i, r_i]$ 的城市没有受到暴风雪的影响。

在暴风雪的影响下迅速确定王国的农业生产方案是非常重要的事情。IceKing_1968 认为，一个农业生产地域应该是一个极大连通块，满足每个节点都没有被暴风雪影响。这里极大连通块的定义是：不存在一个不属于该点集的未被暴风雪影响的点与该连通块连通。

IcePrincess_1968 要负责算出每次暴风雪后，王国能拥有多少个农业生产地域。注意这里每次暴风雪是独立的，即每次暴风雪过后，直到每个城镇重新焕发生机，下一次暴风雪才会到来。

正如上文所述，IcePrincess_1968 擅长文学但不擅长计算机，于是请你帮忙。

【输入格式】

输入文件名为 icekingdom.in。

第一行包含两个正整数 n, q ，表示 IceKingdom 的城镇个数和暴风雪次数。

第 2 至第 n 行，每行两个正整数 x, y ，表示城镇 x 和城镇 y 之间有一条道路。

第 $n+1$ 至第 $n+q$ 行，每行两个正整数 l_i, r_i ，描述一场暴风雪，含义如题面所述。

【输出格式】

输出文件名 icekingdom.out。

输出文件共有 q 行，第 i 行表示在第 i 场暴风雪之后农业生产地域的个数。

【输入输出样例 1】

icekingdom.in	icekingdom.out
4 3	1
1 2	1
2 3	2
2 4	
1 2	
1 3	
3 4	

【输入输出样例 1 解释】

第一次询问，只有 (1,2) 一个连通块。

第二次询问，只有 (1,2,3) 一个连通块。

第三次询问，有 3 和 4 两个连通块。

【输入输出样例 2】

见选手目录下的 icekingdom/icekingdom2.in 和 icekingdom/icekingdom2.ans。

【数据规模和约定】

对于 30%的数据: $n \leq 100, q \leq 100$;

对于 50%的数据: $n \leq 2,000, q \leq 2,000$;

对于 100%的数据: $n \leq 200,000, q \leq 200,000$, 对于所有的暴风雪, $l_i \leq r_i$ 。

【后记】

IcePrincess_1968 和 IcePrince_1968 其实只是一对兄妹而已。