

NOIP2019 模拟题 day1

一. 题目概况

中文题目名称	冲锋	炸弹轮胎	午时已到	
英文题目名称	rush	bomb	noon	pool
可执行文件名	rush	bomb	noon	
输入文件名	rush.in	bomb.in	noon.in	
输出文件名	rush.out	bomb.out	noon.out	
时间限制	1s	1s	1s	
空间限制	128MB	128MB	128MB	
测试点数目	10	10	20	
单测试点分值	10	10	5	
比较方式	全文比较（忽略行末空格和结尾回车）	全文比较（忽略行末空格和结尾回车）	全文比较（忽略行末空格和结尾回车）	
题目类型	传统	传统	传统	

二. 提交源程序文件名

对于 Pascal 语言	rush	bomb	noon
对于 C 语言	rush	bomb	noon
对于 C++语言	rush	bomb	noon

三. 编译命令

对于 Pascal 语言	fpc rush.pas	fpc bomb.pas	fpc noon.pas
对于 C 语言	gcc -o rush rush.c -lm	gcc -o bomb bomb.c -lm	gcc -o noon noon.c -lm
对于 C++语言	g++ -o rush rush.cpp -lm	g++ -o bomb bomb.cpp -lm	g++ -o noon noon.cpp -lm

四. 注意事项:

1. 文件名（程序名和输入输出文件名）必须使用小写。

2. 选手提交以自己中文名字命名的文件夹，文件夹内包含三个源文件(.c, .cpp, .pas,)，并在文件夹下建立 3 个相应的子目录，并将 3 个对应的源程序分别放入对应的子文件夹中，所有名字必须使用小写；
3. 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。
4. 题目简单，请认真对待，争取 AK。
5. 每道题源代码长度限制均为 100KB。
6. 每道题的数据都有一定梯度。请尽量优化算法，争取拿高分。
7. 题目若无特殊说明，所有输入的数均为 `int` 范围内的非负整数。
8. 评测机为 `linux` 系统下全国评测机。
9. 题目不一定按照难度顺序，但保证第一题最简单。
10. 如果发现原题，偷偷 A 掉，不要告诉别人。

背景

小 X，小 Y，小 Z 是三个游戏王，他们每天都开黑玩一款叫做《学习先锋》（简称 OW）的游戏，因此日渐消瘦。突然有一天晚上，他们听到一个声音：

“这个世界需要更多的英雄。”

他们发现自己穿越到了 OW 的世界。而这个世界的英雄都面临各种各样的问题，英雄们希望他们能解决这些问题。

可由于沉迷游戏，他们已经荒废了学习，因此，他们只好寻求你的帮助。

冲锋（rush）

题目描述

“正义必将得到伸张！”



小 X 在

训练场遇到了莱因哈特。

莱因哈特是一位英雄，他有一个特别的技能，叫冲锋——即向前方笔直的冲过去，并对撞到的单位造成伤害（真正的冲锋还有更多功效，在这里不做讨论）。

由于 OW 克隆技术的发展，莱因哈特可以同时存在很多个，有一天他们决定展开一个训练：

训练场地是一个 $n*n$ 的平台，其中有一些格子是空的。训练开始时每个莱因哈特会从平台的某一个边缘同时开始向对面冲锋，每一秒冲锋一格。如果莱因哈特冲到了空的格子上就会掉下去摔伤；如果某一时刻，两个莱因哈特撞到了一起（包括正对着撞到一起和从侧边撞到一起），则他们会同时受伤。每个莱因哈特会被安排在一行或者一列的某一端开始冲锋，如果他成功到达了另一端，那么就算通过了训练。

由于这个世界需要更多的英雄，所以小 X 不希望让任何莱因哈特受伤。他想知道，他应该怎样安排莱因哈特的位置使得在没有任何莱因哈特受伤的情况下能有尽可能多的莱因哈特通过训练。

然而小 X 是个游戏王，他已经不会解决这个问题了，于是他请你来帮忙。

输入格式

输入第一行 2 个数 n, m ，表示平台大小为 $n*n$ ，其中有 m 个空格子。

接下来 m 行，每行两个数 x, y ，表示第 x 行，第 y 列的格子是空的。（可能有重复）

输出格式

输出一行一个数，表示最多能有多少莱因哈特通过训练。

输入输出样例

rush.in	rush.out
---------	----------

2 2 2 1 2 2	1
-------------------	---

数据范围

对于 30%的数据， $n \leq 5, m \leq 25$ ；对于 60%的数据， $n \leq 1000, m \leq 1000$ ；
对于 100%的数据， $n \leq 100000, m \leq 100000$ 。




炸弹轮胎 (bomb)

题目描述






“女士们，先生们，炸弹轮胎滚起来啦！”





小 Y 在训练场遇到了狂鼠 .




 是一名雇佣兵，也是一位英雄。他拥有一个终极技能——炸弹轮胎 。即释放出可以自爆的轮胎。 会在道路上移动，并且可以在靠近敌人身边时自爆。造成的伤害巨大伤害可以立即摧毁敌人



现在， 接到了一个训练任务——摧毁敌方堡垒 .






训练场可以看成一张 n 个点， m 条边的带权无向图， 在 1 号节点， 在 n 号节点。如果   的能够到达  所在的节点并且  没受到保护，炸弹轮胎就能立



即爆炸并摧毁 ，这时任务完成。

可训练任务不会这么简单，敌方在每个节点派出了一个查莉娅  来保护 .

  和所在的节点没有 .

 拥有一个技能——充能护盾，可以为一个友方英雄（包括自己）套上一个护盾。被护盾保护的队友可以看做无敌，而护盾会在  的瞬间消失。也可以保


护另一个。由于本身的血量较少，当一个节点的或没有死亡时，强大的火力压制会让无法经过这个节点走到其它道路。



由于科技的发展，同一时间可以释放出任意多的，他想知道他最早什么时候能完成任务。如果不能完成任务，输出“Mission Failed”。

然而小 Y 是个游戏王，他已经不会解决这个问题了，于是他请你来帮忙。

输入格式

输入第一行两个数 n,m ，表示有 n 个节点 m 条边。

接下来 M 行，每行三个正整数 u_i, v_i, w_i ，表示有一条从节点 u_i 到节点 v_i 的道路，通过这条道路需要 w_i 的时间。

接下来 $N-2$ 行，第 i 行描述处于第 $i+1$ 个节点的的技能目标。每行若干个数。第一个数为 k ，表示这个有 k 个保护目标，接下来 k 个 $[2,n]$ 之间的数，表示保护的目標所在节点。

输出格式

输出一行一个数，表示完成任务的最短时间。

如果无法完成任务输出“Mission Failed”

输入输出样例

输入输出样例 1

bomb.in	bomb.out
2 0	Mission Failed

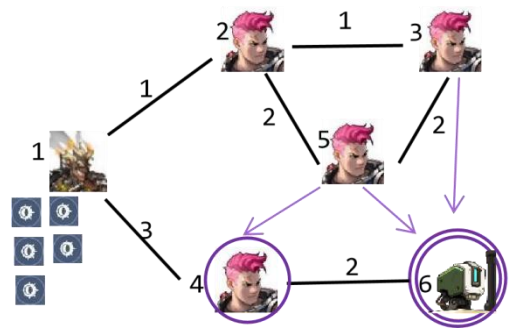
输入输出样例 2

bomb.in	bomb.out
6 6 1 2 1	5
1 4 3 2 3 1 2 5 2 4 6 2 5 3 2 0 1 6 0 2 4 6	

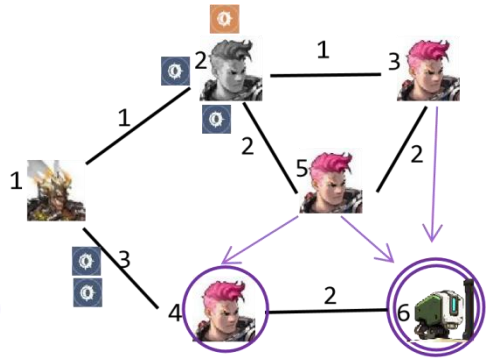
样例解释

样例 2 中，训练场即最快方案如图所示：

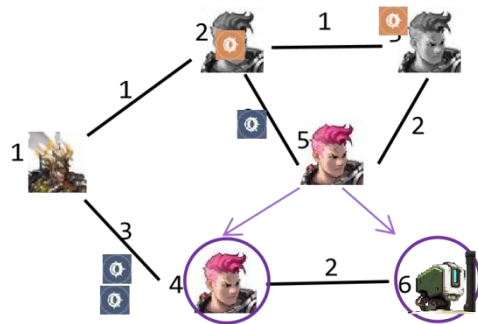
0 时刻：



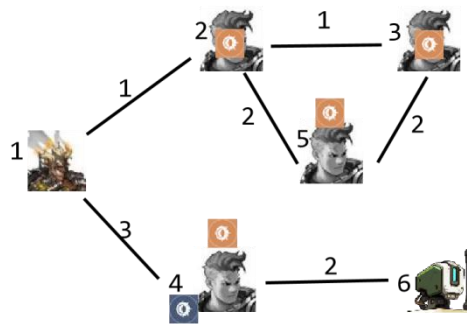
1 时刻：



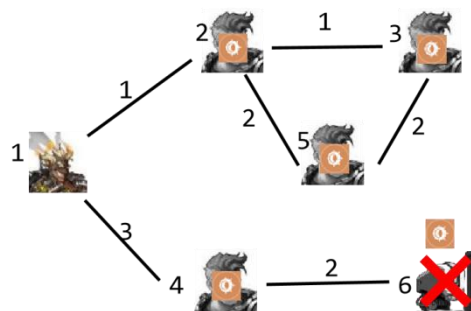
2 时刻：



3 时刻：



5 时刻：



数据范围

对于 20%的数据， $n \leq 20, m \leq 60$ ；对于 50%的数据， $n \leq 500, m \leq 10000$ ；
对于 100%的数据， $n \leq 5000, M \leq 100000, w_i \leq 10^7$ 。

午时已到 (noon)

题目描述

“午时已到。”



小 Z 来

到了一个训练场，他遇见了麦克雷。

麦克雷是一个牛仔，也是一位英雄，他的招牌技能是“神枪手”，可以将可视范围内的所有敌人瞬间击杀。

现在麦克雷来到了一个训练场前，他面前的一条直线上有 n 个平台，相邻两个平台相距 1 米，麦克雷所站的位置到第 1 个平台也恰好是 1 米。这些平台是可以升降的，而且平台之下的所有部分都是实心的。也就是说如果一个敌人与麦克雷的连线上有其他平台（的柱子）或敌人，那么麦克雷就看不到他。

现在训练开始了，初始时，所有的敌人头顶高度都是 -1，也就是说麦克雷看不到任何敌人。麦克雷想知道，在每一时刻，他最多能击杀多少敌人。（麦克雷并不会真的击杀他们）

然而小 Z 是个游戏王，他已经不会解决这个问题了，于是他请你来帮忙。

输入格式

第一行两个数 n, m ，表示有 n 个平台， m 次平台升降。

接下来 m 行，每行两个数 x, y ，表示第 x 个平台敌人头顶的高度变为 y 。

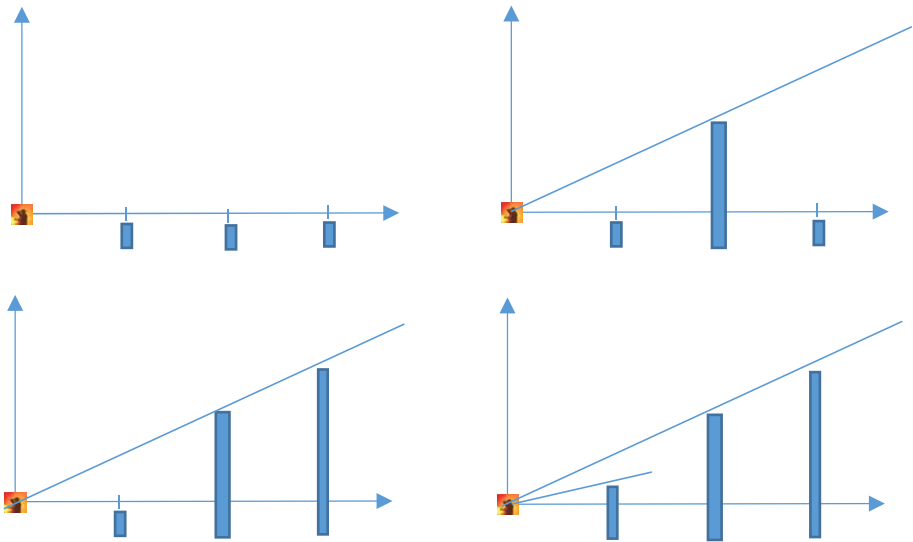
输出格式

输出共 m 行，每行一个数，第 i 行的数为第 i 次平台升降后麦克雷能击杀的敌人数量。

输入输出样例

noon.in	noon.out
3 4	1
2 4	1
3 6	1
1 1000000	2
1 1	

样例解释



数据范围

对于 30%的数据， $n, m \leq 1000$ ；

另有 20%的数据，输入的 x 严格递增；对于 100%的数据， $n, m \leq 100000$ 。

注意常数优化。

鱼塘（pool）

题目描述

钓鱼是一种玄学仪式，往往需要施术者全神贯注，从而导致他们忽略了周遭的事物。这让他人感到不满。

XPB 就是一个喜欢钓鱼的玄学家，而且他总是在上课的时候钓鱼，以至于老师经常请他喝茶。可 **XPB** 却不在乎，因为他想钓的鱼，不是一般的鱼，而是上古神兽——鲲。

钓鲲的姿势很特别，需要施术者想象出嘴里有一根吊杆，并让头部有节奏的上下晃动，晃动的姿势有两种：长点和短点。而晃动的节奏也是有讲究的——长点不能连续。

由于某种玄学原因，短点耗时 1 秒，长点耗时 k 秒。由于老师的存在，**XPB** 钓鱼的持续时间是有限的，第 i 堂课允许的钓鱼持续时间为 $L_i \sim R_i$ 秒。如果晃动的节奏满足一定的要求，就能钓到鲲。但是不同的节奏可能有很多，因此他想知道有多少种不同的钓鱼方法。需要对 $1e9+7$ 取模。

输入格式

第一行两个数 T, k ，分别表示有 T 堂课和长点耗时 k 秒。
接下来 T 行，每行 2 个数 L_i, R_i ，表示这堂课允许的钓鱼时间

输出格式

输出共 T 行，每行一个数，表示该堂课钓鱼方法数对 $1e9+7$ 取模后的值。

输入输出样例

pool.in	pool.out
3 3	2
3 3	7
1 4	11
1 5	

数据范围

对于 30%的数据， $T=1, k \leq 10, 0 < L_i \leq R_i \leq 10$ ；对于 60%的数据，输入中所有数 ≤ 1000 ；对于 100%的数据，输入中所有数 $\leq 100000, L_i \leq R_i$ ；

后记

小 X，小 Y，小 Z 是三个游戏王。

经过 3.5 个小时的穿越历险之后，他们重新回到了现实世界。

在冒险中，他们逐渐意识到了学习的重要性——不学习，什么问题也没办法解决.....

于是他们下定决心开始认真学习，若干年后，他们果然成就了一番大事业。当然，他们永远不会忘记，在这短短的 3.5 个小时中，你对他们的帮助改变了他们的一生。