

NOIP2017 Simulation

dy0607

August 1, 2017

题目名称	世界线	时间机器	密码
可执行文件名	worldline	machine	password
输入文件名	worldline.in	machine.in	password.in
输出文件名	worldline.out	machine.out	password.out
题目类型	传统型	传统型	传统型
每个测试点时限	1.5s	2s	1s
内存限制	256MB	256MB	256MB
测试点数目	5	10	10
每个测试点分值	20	10	10
编译选项	-O2 -lm -Wl,-stack=67108864		

1 世界线

1.1 问题描述

时间并不是一条单一的线，而是由许多世界线构成的流。

在一些时刻，世界线会发生分裂；同样的，它们也有可能在一些时刻收束在一起。如果将这些时刻抽象成点，那么这些世界线构成的网络，实际上是一张有向无环图。

Okabe 想要改变世界线的构造，他认为世界线是优美的，当且仅当其中不存在三个点 u, v, t ，其中 u 到 v 有连边， v 到 t 有连边，而 u 到 t 没有连边。

作为世界的观测者，Okabe 已经知道了世界线的构成。现在他想知道，在不删除边的情况下，至少要连接多少条边，才能得到优美的世界线。

1.2 输入格式

从文件 *worldline.in* 中读入数据。

第一行两个整数 n, m ，表示点数和边数。

接下来 m 行，每行两个整数 u, v ，表示 u 到 v 有一条有向边。

1.3 输出格式

输出到文件 *worldline.out* 中。

仅包含一个整数，表示答案。

1.4 样例1输入

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

1.5 样例1输出

5

1.6 样例解释1

还需要连边 $(1, 4), (1, 5), (2, 4), (2, 5), (3, 5)$ 。

1.7 样例2

见选手目录下的`worldline2.in/worldline2.ans`。

1.8 数据范围及约定

保证 $1 \leq u, v \leq n$ ，给出的图是一张有向无环图，且没有重边和自环。
各个测试点还满足如下约束：

测试点	n	m	其他约束
1	$\leq 10^2$	$\leq 10^2$	无
2	$\leq 2 \times 10^3$	$\leq 2 \times 10^3$	
3	$\leq 4 \times 10^4$	$\leq 10^5$	
4	$\leq 6 \times 10^4$	$= n - 1$	只有1个连通块
5		$\leq 10^5$	无

2 时间机器

2.1 问题描述

作为一名天才科学家，Kurusu已经设计出了时间机器的构造。

根据Kurusu的构想，时间机器中有 n 种需要放置电阻的节点，第 i 种节点有 s_i 个，其电压 U 的变动范围是 $low_i \leq U \leq high_i$ 。

现在有 m 种电阻，第 i 种电阻有 k_i 个，第 i 种电阻能正常工作时，电压 U' 需要满足 $l_i \leq U' \leq r_i$ 。第 i 种电阻能放置在第 j 种节点上，当且仅当 $l_i \leq low_j \leq high_j \leq r_i$ 。

现在Kurusu想要确认她的构想能不能实现，即是否能在所有节点上放置合适的电阻。

2.2 输入格式

从文件`machine.in` 中读入数据。

第一行一个整数 T ，表示数据组数。

每组数据的第一行有两个整数 n, m ，表示节点和电阻的种类数。

接下来 n 行，每行三个整数 $low_i, high_i, s_i$ ，描述一种节点。

接下来 m 行，每行三个整数 l_i, r_i, k_i ，描述一种电阻。

2.3 输出格式

输出到文件`machine.out` 中。

输出共 T 行，每行一个字符串，若可以实现则输出"Yes" (不包含引号)，否则输出"No"。

2.4 样例1输入

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

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

2.5 样例1输出

```
Yes
No
Yes
```

2.6 样例2

见选手目录下的`machine2.in/machine2.ans`。

2.7 数据范围及约定

保证 $1 \leq low_i \leq high_i \leq 10^9, 1 \leq l_i \leq r_i \leq 10^9, 1 \leq k_i, s_i \leq 10^9, T \leq 50, 1 \leq n, m \leq 5 \times 10^4$ 。

记 $\sum n$ 为一个测试点中所有 n 的总和，记 $\sum m$ 为一个测试点中所有 m 的总和，则 $\sum n, \sum m \leq 4 \times 10^5$ 。

各个测试点还满足如下约束：

测试点	n	m	其他约束
1	$= 1$	$\leq 5 \times 10^4$	无
2	≤ 6	≤ 6	$s_i = 1$
3			
4	≤ 300	≤ 300	$s_i, k_i \leq 2$
5			无
6			
7			
8	$\leq 5 \times 10^4$	$\leq 5 \times 10^4$	$s_i, k_i \leq 2$
9			无
10			

2.8 提示

输入量巨大，建议读入优化。

3 密码

3.1 问题描述

为了揭穿SERN的阴谋, Itaru黑进了SERN的网络系统。

然而, 想要完全控制SERN, 还需要知道管理员密码。Itaru从截获的信息中发现, SERN的管理员密码是两个整数 $l, s, 0 \leq s \leq l$, 并且一旦得知了管理员密码, 就可以生成出SERN各个网络接口的密码: 各个网络接口的密码均是若干个长为 l 的0/1串, 且每个串中1的个数恰为 s 。不难发现, 生成的密码串个数是一个组合数。

SERN的网络系统是由 p^k 个网络接口构成的, SERN为了保证网络系统的稳定性, 保证了 p 为质数, 且所有生成的密码串被平均分配到了 p^k 个网络接口, 这意味着生成的密码串个数能被 p^k 整除。为了网络通讯的方便, SERN的网络接口的密码不会太长, 即可以保证 $l \leq N$ 。

作为一名Super Hacker, Itaru已经想到了破解密码的绝妙方法, 然而在这之前, 他需要确认管理员密码的可能情况有多少。由于答案可能很大, 答案对 $10^9 + 7$ 取模。

3.2 输入格式

从文件`password.in` 中读入数据。

仅包含一行三个整数 N, p, k 。

3.3 输出格式

输出到文件`password.out` 中。

仅包含一个整数, 表示答案。

3.4 样例1输入

4 2 2

3.5 样例1输出

2

3.6 样例解释1

有2种可能的情况， $l = 4, s = 1; l = 4, s = 3$ 。

3.7 样例2

见选手目录下的 *password2.in/password2.ans*。

3.8 数据范围及约定

保证 $1 \leq p, k \leq 10^9$, $1 \leq N \leq 10^{1000}$, p 为质数。

各个测试点还满足如下约束：

测试点	N	p	k
1	$\leq 10^2$	$\leq 10^2$	≤ 2
2	$\leq 2 \times 10^3$	$\leq 2 \times 10^3$	
3			$\leq 10^9$
4	$\leq 10^6$	$= 2$	$= 1$
5		$= 3$	
6	$\leq 10^{100}$	≤ 10	$\leq 10^9$
7			
8			
9	$\leq 10^{1000}$	$\leq 10^9$	
10			