

全国青少年信息学奥林匹克竞赛

NOIP2023模拟

时间：8:00-12:20

题目名称	数数奶牛	没时间烘干	最小化边	计数图形
题目类型	传统型	传统型	传统型	传统型
目录	count	notime	edges	graphs
可执行文件名	count	notime	edges	graphs
输入文件名	count.in	notime.in	edges.in	graphs.in
输出文件名	count.out	notime.out	edges.out	graphs.out
每个测试点时限	1.0秒	1.0秒	1.0秒	1.0秒
内存限制	256 MB	256 MB	256 MB	256MB
子任务数目	20	20	20	20
测试点是否等分	是	是	是	是

提交源程序文件名

对于C++语言	count.cpp	notime.cpp	edges.cpp	graphs.cpp
---------	-----------	------------	-----------	------------

编译选项

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

注意事项与提醒（请选手务必仔细阅读）

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++ 中主函数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. 提交的程序代码文件的放置位置请参照各省的具体要求。
4. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。
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. 程序可使用的栈内存空间限制与题目的内存限制一致。
7. 全国统一评测时采用的机器配置为：Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz，内存 32GB。上述时限以此配置为准。
8. 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。
9. 终评测时所用的编译命令中不含编译选项之外的任何优化开关。

数数奶牛 (count)

【问题描述】

如同平常一样，FJ 的奶牛们分散在他的最大的草地上。草地可以看作是一个由正方形方格组成的巨大的二维方阵（想象一个巨大的棋盘）。

奶牛分布在草地上的方式相当迷人。对于每一个满足 $x \geq 0$ 以及 $y \geq 0$ 的方格 (x, y) ，当对于所有整数 $k \geq 0$ ， $\lfloor \frac{x}{3^k} \rfloor$ 和 $\lfloor \frac{y}{3^k} \rfloor$ 除以三的余数的奇偶性均相同时，有一头奶牛位于 (x, y) 。换言之，两个余数均为奇数（均等于 1），或均为偶数（均等于 0 或 2）。例如，满足 $0 \leq x, y < 9$ 的方格中，包含奶牛的方格在下图中用 1 表示。

```
      x
    012345678
y 0 101000101
  1 010000010
  2 101000101
  3 000101000
  4 000010000
  5 000101000
  6 101000101
  7 010000010
  8 101000101
```

FJ 对他的草地上的某个特定区域内的奶牛数量感兴趣。他进行了 Q 个询问，每个询问由三个整数 x_i, y_i, d_i 组成。对每个询问，FJ 想要知道有多少奶牛位于 (x_i, y_i) 至 $(x_i + d_i, y_i + d_i)$ 的对角线上的方格内（包括两端）。

【输入格式】

输入的第一行包含 Q ($1 \leq Q \leq 10^4$)，为询问的数量。

以下 Q 行每行包含三个整数 d_i, x_i 和 y_i ($0 \leq x_i, y_i, d_i \leq 10^{18}$)。

【输出格式】

输出 Q 行，每个询问输出一行。

【样例输入1】

```
8
10 0 0
10 0 1
9 0 2
8 0 2
0 1 7
1 1 7
2 1 7
1000000000000000000 1000000000000000000 1000000000000000000
```

【样例输出1】

```
11
0
4
3
1
2
2
100000000000000000001
```

【数据范围及约定】

- 对于另外 10% 的数据，满足对于每一个询问有 $d_i \leq 100$ 。
- 对于另外 40% 的数据，满足对于每一个询问有 $x + d = 3^{30} - 1$ 以及 $y = 0$ 。
- 对于另外 50% 的数据，没有额外限制。

没时间烘干 (notime)

【问题描述】

Bessie 最近收到了一套颜料, 她想要给她的牧草地一端的栅栏上色。栅栏由 N 个 1 米长的小段组成 ($1 \leq N \leq 2 \cdot 10^5$)。Bessie 可以使用 N 种不同的颜色, 她将这些颜色由浅到深用 1 到 N 标号 (1 是很浅的颜色, N 是很深的颜色)。从而她可以用一个长为 N 的整数数组来描述她想要给栅栏的每一小段涂上的颜色。

初始时, 所有栅栏小段均未被上色。Bessie 一笔可以给任意连续若干小段涂上同一种颜色, 只要她不会在较深的颜色之上涂上较浅的颜色 (她只能用较深的颜色覆盖较浅的颜色)。

例如, 一段长为 4 的未被涂色的栅栏可以按如下方式上色:

```
0000 -> 1110 -> 1122 -> 1332
```

不幸的是, Bessie 没有足够的时间等待颜料变干。所以, Bessie 认为她可能需要放弃为栅栏上某些小段上色! 现在, 她正在考虑 Q 个候选的区间 ($1 \leq Q \leq 2 \cdot 10^5$), 每个区间用满足 $1 \leq a \leq b \leq N$ 的两个整数 (a, b) 表示, 为需要上色的小段 $a \dots b$ 的两端点位置。

对于每个候选区间, 将所有区间内的栅栏小段都涂上所希望的颜色, 并且区间外的栅栏小段均不涂色, 最少需要涂多少笔? 注意在这个过程中 Bessie 并没有真正进行任何的涂色, 所以对于每个候选区间的回答是独立的。

【输入格式】

输入的第一行包含 N 和 Q 。

下一行包含一个长为 N 的整数数组, 表示每个栅栏小段所希望的颜色。

以下 Q 行, 每行包含两个空格分隔的整数 a 和 b , 表示一个需要涂色的候选区间。

【输出格式】

对于 Q 个候选区间的每一个, 输出一行, 包含答案。

【样例输入1】

```
8 4
1 2 2 1 1 2 3 2
4 6
3 6
1 6
5 8
```

【样例输出1】

```
2
3
3
3
```

【样例1解释】

在这个样例中，对应颜色为 1 1 2 的子段涂上颜色需要两笔。

对应颜色为 2 1 1 2 的子段涂上颜色需要三笔。

对应颜色为 1 2 2 1 1 2 的子段涂上颜色需要三笔。

对应颜色为 1 2 3 2 的子段涂上颜色需要三笔。

【数据范围及约定】

- 对于 10% 的数据，满足 $N, Q \leq 100$ 。
- 对于另外 15% 的数据，满足 $N, Q \leq 5000$ 。
- 对于另外 25% 的数据，输入数组不包含大于 10 的数。
- 对于另外 50% 的数据，没有额外限制。

最小化边 (edges)

【问题描述】

Bessie 有一个连通无向图 G 。 G 有 N 个编号为 $1 \dots N$ 的结点, 以及 M 条边 ($2 \leq N \leq 10^5, N-1 \leq M \leq \frac{N^2+N}{2}$)。 G 有可能包含自环 (一个结点连到自身的边), 但不包含重边 (连接同一对结点的多条边)。

令 $f_G(a, b)$ 为一个布尔函数, 对于每一个 $1 \leq a \leq N$ 和 $0 \leq b$, 如果存在一条从结点 1 到结点 a 的路径恰好经过了 b 条边, 则函数值为真, 否则为假。如果一条边被经过了多次, 则这条边会被计算相应的次数。

Elsie 想要复制 Bessie。具体地说, 她想要构造一个无向图 G' , 使得对于所有的 a 和 b , 均有 $f_{G'}(a, b) = f_G(a, b)$ 。

Elsie 想要进行最少数量的工作, 所以她想要构造最小可能的图。所以, 你的工作是计算 G' 的边数的最小可能值。

每个输入包含 T ($1 \leq T \leq 5 \cdot 10^4$) 组独立的测试用例。保证所有测试用例中的 N 之和不超过 10^5 , 且所有测试用例中的 M 之和不超过 $2 \cdot 10^5$ 。

【输入格式】

输入的第一行包含 T , 为测试用例的数量。

每个测试用例的第一行包含两个整数 N 和 M 。

每个测试用例的以下 M 行每行包含两个整数 x 和 y ($1 \leq x \leq y \leq N$), 表示 G 中存在一条连接 x 与 y 的边。

【输出格式】

对每个测试用例, 输出一行, 为 G' 中的边数的最小可能值。

【样例输入1】

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

【样例输出1】

```
4
5
```

【样例1解释】

在第一个测试用例中，Elsie 可以通过从 G 中移除 $(2, 5)$ 来构造得到 G' 。或者，她也可以构造一张包含以下边的图，因为她并未被限制只能从 G 中移除边：

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

Elsie 显然不能得到比 $N - 1$ 更优的解，因为 G' 一定也是连通的。

【样例2】

见下发文件中。

【样例2解释】

在以上这些测试用例中，Elsie 都不能做得比 Bessie 更优。

【数据范围及约定】

- 对于另外 5% 的数据，满足 $N \leq 5$ 。
- 对于另外 10% 的数据，满足 $M = N$ 。
- 对于另外 20% 的数据，如果并非对于所有的 b 均有 $f_G(x, b) = f_G(y, b)$ ，则存在 b 使得 $f_G(x, b)$ 为真且 $f_G(y, b)$ 为假。
- 对于另外 30% 的数据，满足 $N \leq 10^2$ 。
- 对于另外 25% 的数据，没有额外限制。

计数图形 (graphs)

【问题描述】

Bessie 有一个连通无向图 G 。 G 有 N 个编号为 $1 \dots N$ 的结点，以及 M 条边 ($1 \leq N \leq 10^2, N-1 \leq M \leq \frac{N^2+N}{2}$)。 G 有可能包含自环 (一个结点连到自身的边)，但不包含重边 (连接同一对结点的多条边)。

令 $f_G(a, b)$ 为一个布尔函数，对于每一个 $1 \leq a \leq N$ 和 $0 \leq b$ ，如果存在一条从结点 1 到结点 a 的路径恰好经过了 b 条边，则函数值为真，否则为假。如果一条边被经过了多次，则这条边会被计算相应的次数。

Elsie 想要复制 Bessie。具体地说，她想要构造一个无向图 G' ，使得对于所有的 a 和 b ，均有 $f_{G'}(a, b) = f_G(a, b)$ 。

你的工作是计算 Elsie 可以构造的图 G' 的数量，对 $10^9 + 7$ 取模。与 G 一样， G' 可以包含自环而不能包含重边 (这意味着对于 N 个有标号结点共有 $2^{\frac{N^2+N}{2}}$ 个不同的图)。

每个输入包含 T ($1 \leq T \leq \frac{10^5}{4}$) 组独立的测试用例。保证所有测试用例中的 N^2 之和不超过 10^5 。

【输入格式】

输入的第一行包含 T ，为测试用例的数量。

每个测试用例的第一行包含整数 N 和 M 。

每个测试用例的以下 M 行每行包含两个整数 x 和 y ($1 \leq x \leq y \leq N$)，表示 G 中存在一条连接 x 与 y 的边。

【输出格式】

对每个测试用例，输出一行，为不同的 G' 的数量，对 $10^9 + 7$ 取模。

【样例输入1】

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

【样例输出1】

```
3
```

【样例1解释】

在第一个测试用例中， G' 可以等于 G ，或以下两个图之一：


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

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

【样例2】

见下发文件中。

【样例2解释】

有一些较大的测试用例。确保你的答案对 $10^9 + 7$ 取模。注意倒数第二个测试用例的答案为 $2^{45} \pmod{10^9 + 7}$ 。

【数据范围及约定】

- 对于另外 5% 的数据，满足 $N \leq 5$ 。
- 对于另外 10% 的数据，满足 $M = N - 1$ 。
- 对于另外 30% 的数据，如果并非对于所有的 b 均有 $f_G(x, b) = f_G(y, b)$ ，则存在 b 使得 $f_G(x, b)$ 为真且 $f_G(y, b)$ 为假。
- 对于另外 45% 的数据，没有额外限制。