

A 最佳放置

时空限制：1s 256MB

文件名

place.in/place.out/place.cpp

题意

Farmer John 沿着一条高速公路拥有一个很长的农场，可以被看作类似于一维数轴。沿着农场有 K 块草地（ $1 \leq K \leq 2 \cdot 10^5$ ）；第 i 块草地位于位置 p_i 并具有美味值 t_i （ $0 \leq t_i \leq 10^9$ ）。Farmer John 的死对头 Farmer Nhoj 已经将他的 M 头奶牛（ $1 \leq M \leq 2 \cdot 10^5$ ）放在了位置 $f_1 \dots f_M$ 。所有这些 $K + M$ 个位置均是 $[0, 10^9]$ 范围内的不同整数。

Farmer John 需要选择 N （ $1 \leq N \leq 2 \cdot 10^5$ ）个位置（不一定是整数）放置他的奶牛。这些位置必须与 Farmer Nhoj 的奶牛已经占用的位置不同，但是 Farmer John 可以将他的奶牛放在与草地相同的位置。

拥有最靠近某个草地的奶牛的农夫拥有这一草地。如果来自两方农夫的两头奶牛距这一草地相等，则 Farmer Nhoj 拥有该草地。

给定 Farmer Nhoj 的奶牛的位置以及草地的位置和美味值，求 Farmer John 的奶牛以最优方式放置时可以达到的最大总美味值。

输入格式

输出一个整数，表示最大总美味值。注意这个问题的答案可能无法用 32 位整数型存储，你可能需要使用 64 位整数型（例如，C 或 C++ 中的 "long long"）。

输入样例

```
6 5 2
0 4
4 6
8 10
10 8
12 12
13 14
2
3
5
7
11
```

输出样例

36

提示

如果 Farmer John 将奶牛放在位置 11.5 和 8 则他可以得到总美味值 $10 + 12 + 14 = 36$ 。

B 连接

时空限制：1s 256MB

文件名

connect.in/connect.out/connect.cpp

题意

Farmer John 的农场由 N 块田地 ($1 \leq N \leq 10^5$) 组成，编号为 $1 \dots N$ 。在这些田地之间有 M 条双向道路 ($0 \leq M \leq 10^5$)，每条道路连接两块田地。

农场有两个牛棚，一个在田地 1 中，另一个在田地 N 中。Farmer John 希望确保有一种方式可以沿着一组道路在两个牛棚之间行走。他愿意建造至多两条新道路来实现这一目标。由于田地的位置因素，在田地 i 和 j 之间建造新道路的花费是 $(i - j)^2$ 。

请帮助 Farmer John 求出使得牛棚 1 和 N 可以相互到达所需要的最小花费。

输入格式

每个测试用例的输入包含 T 个子测试用例 ($1 \leq T \leq 20$)，所有子测试用例必须全部回答正确才能通过整个测试用例。

输入的第一行包含 T ，之后是 T 个子测试用例。

每个子测试用例的第一行包含两个整数 N 和 M 。以下 M 行，每行包含两个整数 i 和 j ，表示一条连接两个不同田地 i 和 j 的道路。输入保证任何两个田地之间至多只有一条道路，并且所有子测试用例的 $N + M$ 之和不超过 $5 \cdot 10^5$ 。

输出格式

输出 T 行。第 i 行包含一个整数，为第 i 个子测试用例的最小花费。

输入样例

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

输出样例

```
2
1
```

样例解释

第一个子测试用例中，最优的方式是用一条道路连接田地 2 和 3，用一条道路连接田地 3 和 4。

第二个子测试用例中，最优的方式是用一条道路连接田地 3 和 4。不需要第二条道路。

数据范围

- 测试点 2 满足 $N \leq 20$ 。
- 测试点 3-5 满足 $N \leq 10^3$ 。
- 测试点 6-10 没有额外限制。

C 间隔

时空限制：1s 256MB

文件名

interval.in/interval.out/interval.cpp

题意

奶牛们正在努力尝试发明有趣的新游戏来玩。他们目前的工作之一与一组 N 个区间（ $1 \leq N \leq 2 \cdot 10^5$ ）有关，其中第 i 个区间从数轴上的 a_i 位置开始，并在位置 $b_i \geq a_i$ 结束。 a_i 和 b_i 均为 $0 \dots M$ 范围内的整数，其中 $1 \leq M \leq 5000$ 。

这个游戏的玩法是，Bessie 选择某个区间（假设是第 i 个区间），而她的表妹 Elsie 选择某个区间（假设是第 j 个区间，可能与 Bessie 所选的区间相同）。给定某个值 k ，如果 $a_i + a_j \leq k \leq b_i + b_j$ ，则她们获胜。

对范围 $0 \dots 2M$ 内的每个值 k ，请计算使得 Bessie 和 Elsie 可以赢得游戏的有序对 (i, j) 的数量。

输入格式

输入的第一行包含 N 和 M 。以下 N 行每行以整数 a_i 和 b_i 的形式描述一个区间。

输出格式

输出 $2M + 1$ 行，依次包含范围 $0 \dots 2M$ 中的每一个 k 的答案。

输入样例

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

输出样例

```
0
0
1
3
4
4
4
3
3
1
1
```

样例解释

在这个例子中，对于 $k = 3$ ，有三个有序对可以使得 Bessie 和 Elsie 获胜： $(1, 1)$ ， $(1, 2)$ ， 和 $(2, 1)$ 。

数据范围

- 测试点 1-2 满足 $N \leq 100, M \leq 100$ 。
- 测试点 3-5 满足 $N \leq 5000$ 。
- 测试点 6-20 没有额外限制。