

道路

N 个以 $1 \dots N$ 标号的城市通过单向的道路相连:。每条道路包含两个参数: 道路的长度和需要为该路付的通行费 (以金币的数目来表示)

Bob and Alice 过去住在城市 1.在注意到 Alice 在他们过去喜欢玩的纸牌游戏中作弊后, Bob 和她分手了, 并且决定搬到城市 N 。他希望能够尽可能快的到那, 但是他囊中羞涩。我们希望能够帮助 Bob 找到从 1 到 N 最短的路径, 前提是他能够付得起通行费。

时间限制: 1000

内存限制: 65536

输入

第一行包含一个整数 K , $0 \leq K \leq 10000$, 代表 Bob 能够在他路上花费的最大的金币数。

第二行包含整数 N , $2 \leq N \leq 100$, 指城市的数目。第三行包含整数 R , $1 \leq R \leq$

10000, 指路的数目. 接下来的 R 行, 每行具体指定几个整数 S, D, L 和 T 来说明关于道路的一些情况, 这些整数之间通过空格间隔: S is 道路起始城市, $1 \leq S \leq N$ D is 道路终点城市, $1 \leq D \leq N$ L is 道路长度, $1 \leq L \leq 100$ T is 通行费 (以金币数量形式度量), $0 \leq T \leq 100$ 注意不同的道路可能有相同的起点和终点。

输出

输入结果应该只包括一行, 即从城市 1 到城市 N 所需要的最小的路径长度 (花费不能超过 K 个金币)。如果这样的路径不存在, 结果应该输出 -1。

样例输入

```
5
6
7
1 2 2 3
2 4 3 3
```

3 4 2 4

1 3 4 1

4 6 2 1

3 5 2 0

5 4 3 2

样例输出

11

Rainbow 的商店

Rainbow 开了一家商店，在一次进货中获得了 N 个商品。

已知每个商品的利润和过期时间。

Rainbow 每天只能卖一个商品，并且过期商品不能再卖。

Rainbow 也可以选择每天出售哪个商品，并且一定可以卖出。

由于这些限制，Rainbow 需要制定一份合理的售卖计划。请你计算一下，Rainbow 最终可以获得的最大收益。

时间限制：1000

内存限制：262144

输入

第一行两个整数 N 。接下来 N 行每行两个整数，分别表示每个商品的利润、过期时间。

$1 \leq N, \text{利润}, \text{时间} \leq 10000$ 。

输出

输出一个整数，表示 Rainbow 最终可以获得的最大收益。

样例输入

```
7
20 1
2 1
10 3
100 2
8 2
5 20
50 10
```

样例输出

```
185
```

提示

第 1 天卖出 20 第 2 天卖出 100 第 3 天卖出 10 第 4 天卖出 50（实际上只要在第 10 天卖就可以） 第 5 天卖出 5（实际上只要在第 20 天前卖就可以） 总计 185 其它 2 件商品由于过期、每天只能卖一个的限制，在最优策略下应该不出售。

冰阔落 I

老王喜欢喝冰阔落。

初始时刻，桌面上有 n 杯阔落，编号为 1 到 n。老王总想把其中一杯阔落倒到

另一杯中，这样他一次性就能喝很多很多阔落，假设杯子的容量是足够大的。

有 m 次操作，每次操作包含两个整数 x 与 y。

若原始编号为 x 的阔落与原始编号为 y 的阔落已经在同一杯，请输出 "Yes"；

否则，我们将原始编号为 y 所在杯子的所有阔落，倒往原始编号为 x 所在的

杯子，并输出 "No"。

最后，老王想知道哪些杯子有冰阔落。

时间限制：10000

内存限制：65536

输入

有多组测试数据，少于 5 组。每组测试数据，第一行两个整数 n, m ($n, m \leq 50000$)。

接下来 m 行，每行两个整数 x, y ($1 \leq x, y \leq n$)。

输出

每组测试数据，前 m 行输出 "Yes" 或者 "No"。第 $m+1$ 行输出一个整数，表示有阔落的杯子数量。第 $m+2$ 行有若干个整数，从小到大输出这些杯子的编号。

样例输入

3 2

1 2

2 1

4 2

1 2

4 3

样例输出

No

Yes

2

1 3

No

No

2

1 4

青蛙的约会

两只青蛙在网上相识了，它们聊得很开心，于是觉得很有必要见一面。它们很高兴地发现它们住在同一条纬度线上，于是它们约定各自朝西跳，直到碰面为止。可是它们出发之前忘记了一件很重要的事情，既没有问清楚对方的特征，也没有约定见面的具体位置。不过青蛙们都是很乐观的，它们觉得只要一直朝着某个方向跳下去，总能碰到对方的。但是除非这两只青蛙在同一时间跳到同一点上，不然是永远都不可能碰面的。为了帮助这两只乐观的青蛙，你被要求写一个程序来判断这两只青蛙是否能够碰面，会在什么时候碰面。我们把这两只青蛙分别叫做青蛙 A 和青蛙 B，并且规定纬度线上东经 0 度处为原点，由东往西为正方向，单位长度 1 米，这样我们就得到了一条首尾相接的数轴。设青蛙 A 的出发点坐标是 x ，青蛙 B 的出发点坐标是 y 。青蛙 A 一次能跳 m 米，青蛙 B 一次能跳 n 米，两只青蛙跳一次所花费的时间相同。纬度线总长 L 米。现在要你求出它们跳了几次以后才会碰面。

时间限制：1000

内存限制：65536

输入

输入只包括一行 5 个整数 x, y, m, n, L ，其中 $x \neq y < 2000000000$ ， $0 < m, n < 2000000000$ ， $0 < L < 2100000000$ 。

输出

输出碰面所需要的跳跃次数，如果永远不可能碰面则输出一行 "Impossible"

样例输入

```
1 2 3 4 5
```

样例输出

```
4
```