

Problem A. 璃月的桥梁

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

原神中，主角旅行者擅长借助传送锚点进行快速移动，具体地说旅行者可以从任何位置传送到指定传送锚点的位置。

璃月的华光林有几个几乎无法攀爬的极高的山峰，它们之间用一些木桥梁连接。这些山峰中有且仅有一个拥有传送锚点。



华光林

最近，旅行者接到一份委托，要求旅行者去检修这些桥梁，为了使得检修时间最短，旅行者希望找到一条路径，这条路径从传送锚点出发，恰好经过每条桥梁一次，可以在任意山峰结束。

你的任务是判断旅行者想要的路径是否存在。

Input

第一行两个整数 n 、 m ，山峰的数量和桥梁的数量，其中令带有传送锚点的山峰为1号。
($3 \leq n \leq 200$, $2 \leq m \leq \frac{n(n-1)}{2}$)

接下来 m 行，每行一个两个整数 u 、 v ，表示一个桥梁连接 u 、 v 两座山峰。($1 \leq u < v \leq n$)

保证从传送锚点可以到达任意山峰，保证不存在两座桥梁连接相同的两座山峰。

Output

如果路径存在，输出“YES”，否则输出“NO”，不区分大小写。

Examples

standard input	standard output
9 8 1 2 2 5 2 3 5 6 6 7 8 9 3 4 6 8	NO
9 10 1 2 2 5 2 3 5 6 6 7 8 9 3 4 6 8 4 7 2 9	YES
9 10 1 2 2 5 2 3 5 6 6 7 8 9 3 4 6 8 4 7 1 9	NO

Problem B. 枫丹的预言

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

本题是交互题，如果你不熟悉此题型，可以先查看指南。概括地说，你每次交互都必须刷新标准输出的缓冲区才能从标准输入读入裁判程序的回答。<https://codeforces.com/blog/entry/45307>

枫丹有一则预言，它内容大致为：枫丹的水位上升，所有人溶于原始胎海之水，留下芙宁娜在神座上哭泣。

现在我们知道，这一切都是因为原始胎海中一只巨大的鲸鱼，吞星之鲸，等到它足够大的时候，便会使得原始胎海之水泄漏。



吞星之鲸

现在，芙卡洛斯掌握了吞星之鲸的体型随时间变化的函数 $f(t)$ ，和使得原始胎海之水泄漏的最小体型 k 。现在已知 $f(t)$ 是一个严格递增且可微，在 $(0, 100)$ 上某处的函数值等于 k 的函数，然而芙卡洛斯并不掌握函数的闭合形式，仅能根据某个 t 计算 $f(t)$ ，无法求导或者做更多操作。

Input

本题的裁判程序不支持科学计数法。

一个有限小数 k ，表示使得原始胎海之水泄漏的最小体型。（ $1 \leq k \leq 100$ ）

Interaction Protocol

你的程序需要与裁判程序交互来完成任务，利用标准输出输出内容到裁判程序，并用标准输入从裁判程序读入数据。

你输出到裁判程序的内容有两种：

- $? t$ ，其中 t 是一个有限小数，表示询问 $f(t)$ 的值。（ $0 \leq t \leq 100$ ）
裁判程序将回应一个有限小数 $f(t)$ ，误差不超过 10^{-12} 。（ $0 \leq f(t) \leq 1000$ ）

- ! t_0 , 其中 t_0 是一个有限小数, 表示 $f(t) = k$ 的根。 ($0 \leq t_0 \leq 100$)
在向裁判程序发送此消息后, 你的程序应当立即退出, 裁判程序将判断你的答案是否正确。

你的答案 t_0 会被接受当且仅当它足下列条件种至少一个: (t^* 表正解)

- $|t_0 - t^*| \leq 10^{-4}$
- $\frac{|t_0 - t^*|}{|t^*|} \leq 10^{-4}$
- $|f(t_0) - k| \leq 10^{-4}$
- $\frac{|f(t_0) - k|}{|k|} \leq 10^{-4}$

交互格式不正确或者最终答案不被接受的, 评测结果为Wrong Answer。

除此之外, 你的程序至多交互100次, 否则评测结果为Partial Correct, 交互超过100次扣除该测试点20%的分数, 此后每10次扣除10%的成绩, 扣完即止。

Example

standard input	standard output
50	? 71
50.41	? 70.75
50.055625	? 70.7107
50.0000309449	! 70.7107

Problem C. 蒙德的卡牌

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

七圣召唤是风摩提瓦特的卡牌游戏，这个游戏中每个角色牌有三种技能，在每回合的开始，玩家通过投掷获取一定数量的骰子，每个骰子可能是七种元素之一，也可能是无色骰子，玩家可以消耗骰子来发动一些技能。

蒙德的猫尾酒馆经常举行热斗模式的七圣召唤对决，这种模式下一般会有许多离谱的规则，比如减少技能所需骰子，所有骰子均为无色等等。

现在我们考虑一种简单的热斗模式对决，所有骰子均为无色，元素战技和普通攻击仅需1骰子，元素爆发仅需2骰子，且不再需要充能，场上只有一位角色。

为了避免浪费，我们总是希望用完所有骰子，你要求出用完所有骰子的方案数。

具体地说，你一开始有 k 个骰子，每次你可以进行以下三个操作，直到剩余0个骰子为止：

- 普通攻击，消耗1骰子。
- 元素战技，消耗1骰子。
- 元素爆发，消耗2骰子。

两种操作方案被认为是不同的当且仅当它们总的操作次数不同，或者存在一个 i 使得两种操作的第 i 步操作不同。

由于这个数可能特别大，你只需要输出对998244353取模的结果。



猫尾酒馆

Input

一个整数 k ，一开始的骰子数量。（ $1 \leq k \leq 10^{15}$ ）

Output

一个整数，用完所有骰子的方案数，对998244353取模。

Examples

standard input	standard output
1	2
3	12
87842506579379	893188283

Note

一些选手可能不知道如何计算 x^n (n 是自然数)，这里给出一种算法：

注意到：

$$x^0 = 1, x^n = x^{n \bmod 2} \cdot (x^2)^{\lfloor n/2 \rfloor}$$

在编程中实际上将递归展开成循环，有算法：

```

POWER( $x, n$ ) :
     $r = 1$ 
    while  $n > 0$  :
        if  $n \bmod 2 == 1$  :
             $r = r \times x$ 
         $x = x \times x$ 
         $n = \lfloor n/2 \rfloor$ 
    return  $r$ 

```

时间复杂度: $O(\lg n)$

Problem D. 稻妻的谜题

Input file: standard input
 Output file: standard output
 Time limit: 1.5 seconds
 Memory limit: 256 megabytes

神樱大祓任务中，巫女花散里带旅行者来到井底。

在这里有多个祝祷座，有唯一一个祝祷座上面勾玉数量为1，这个祝祷座不可修改，其他的祝祷座可以通过修改使得它上面的勾玉数量为任何大于等于2的整数。此外有一个鸟居，上面有一幅关于谜题的图案。



神樱大祓任务

修改完成后，旅行者回到勾玉数量为1的祝祷座进行祝祷，此时，所有勾玉数量为1的祝祷座会连接所有勾玉数量为2的祝祷座，所有勾玉数量为2的祝祷座会连接所有勾玉数量为3的祝祷座，以此类推，所有勾玉数量为 n 的祝祷座会连接所有勾玉数量为 $n + 1$ 的祝祷座。

祝祷后，祝祷座之间的连接情况必须于鸟居上的图案完全一致，即两个祝祷座图案上连接，旅行者祝祷后也必须连接，反之亦然。鸟居上的图案没有勾玉数量，并且保证图案上所有的祝祷座直接或间接与勾玉数量为1的祝祷座相连。

因为旅行者需要多次在此任务中解出类似谜题，他/她希望你编写一个程序帮他/她解决这个谜题，或者表明这个谜题无解。

Input

两个整数 n 、 m ，图案上祝祷座的数量和连接的数量，其中勾玉数量为1的祝祷座为1号。
 $(2 \leq n \leq 10^5, 1 \leq m \leq \max\{\frac{n(n-1)}{2}, 10^5\})$ 接下来 m 行，每行两个整数 u 、 v ，表示 u 与 v 相连。
 $(1 \leq u < v \leq n)$

Output

如果有解，输出“YES”，接下来一行 $n - 1$ 个整数，表示除了1号祝祷座以外每个祝祷座上勾玉的数量。
 如果无解，输出“NO”。

Examples

standard input	standard output
5 6 1 2 1 4 2 3 3 4 2 5 4 5	YES 2 3 2 3
5 4 1 2 2 3 3 4 4 5	YES 2 3 4 5
5 5 1 2 2 3 1 3 3 4 3 5	NO

Problem E. 须弥的山头

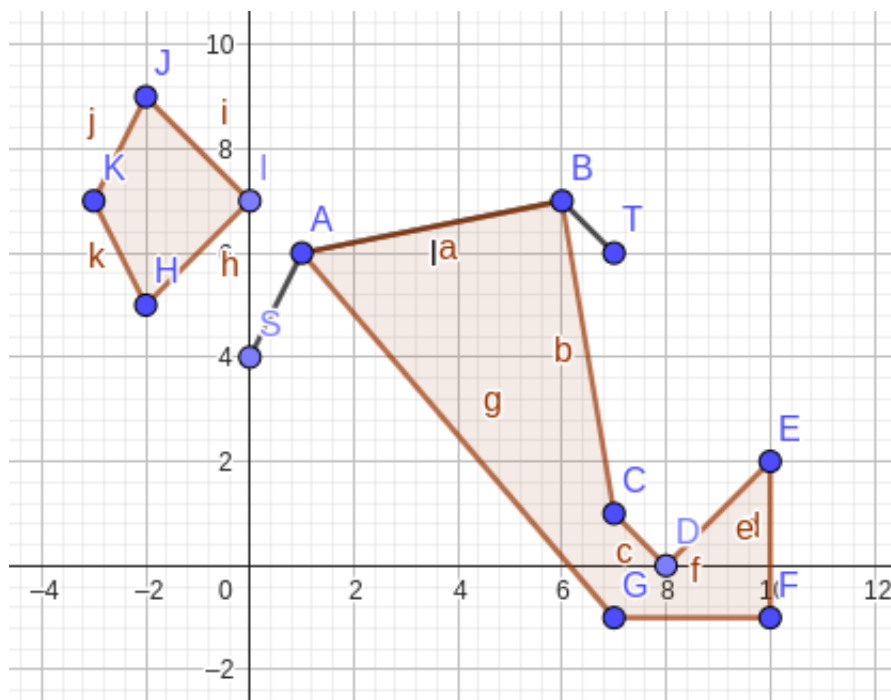
Input file: standard input
 Output file: standard output
 Time limit: 4.5 seconds
 Memory limit: 256 megabytes

须弥沙漠有许多山，这些山大都无法攀爬，让旅行者感觉很不方便。

众所周知，蒙德城的温迪曾经削掉了蒙德的山头丢进海里以平整地块，并形成了金苹果群岛。

然而温迪不能削掉这些山头，但是他可以帮助你完成**至多一次**上山。

这些山头都可以被近似认为是多边形，如下图所示，最短路的顶点一定在障碍物多边形的顶点上。



一条从S到T的最短路

温迪觉得这道题对你来说太难了，所以他帮你处理了几何关系，确切地说，他将告诉你起点，终点，各个多边形的顶点之间是所有无需上山可达的对，恰好一次上山可达的对，以及这些点对之间的距离。

看不懂没关系，形式化地，给出一个有红蓝两种边的无向带权图，求至多经过一条蓝边的从起点到终点的最短路。或者断言无论如何都无法到达终点。



温迪可爱！

Input

第一行三个正整数 n 、 m 、 k ($3 \leq n \leq 2 \cdot 10^5$, $2 \leq m + k \leq \min\{\frac{(n+1)(n+2)}{2}, 2 \cdot 10^5\}$)，表示多边形顶点的个数（不含起点和终点，编号 $3, 4, \dots, n+2$ ），无需上山可达的点对的数量（红边数量），恰好一次上山可达的点对的数量（蓝边数量）。

接下来 m 行，每行三个整数 i 、 j 、 w ($1 \leq i < j \leq n+2$, $0 \leq w \leq 10^4$)，表示点 i 和点 j 之间无需上山可达（红边），距离为 w 。

接下来 k 行，每行三个整数 i 、 j 、 w ($1 \leq i < j \leq n+2$, $0 \leq w \leq 10^4$)，表示点 i 和点 j 之间恰好一次上山可达（蓝边），距离为 w 。

Output

一个整数，表示起点1到终点2的最短距离，或者-1表示无法到达。

Examples

standard input	standard output
3 5 2 2 3 1 3 4 2 4 5 3 1 5 3 1 4 9 1 3 10 3 5 4	8
3 0 4 2 3 1 3 4 1 4 5 1 1 5 1	-1
4 4 1 1 6 1 2 3 0 4 5 0 3 4 0 2 6 3	4