

## Problema K

# Ká entre Nós

Empates são sempre um problema em eleições ou em jogos. Recentemente, um novo jogo, chamado *Ká entre Nós*, foi inventado. O jogo é disputado por jogadores conectados numa rede social. Cada jogador tem um conjunto de amigos. A cada rodada há várias votações, mas um competidor somente pode receber votos de seus amigos. Ganha o jogador que receber o maior número de votos.

O jogo ainda está na fase de projeto, mas os desenvolvedores depararam com um problema muito comum. Dado que o número de amigos de cada jogador é em geral pequeno, empates são muito comuns, o que tira a graça do jogo. Para resolver esse problema, os desenvolvedores decidiram adicionar um novo módulo ao jogo. Esse módulo define os amigos de cada jogador, e sempre que possível dará a cada jogador um número ímpar de amigos.

O problema se mostrou mais complicado do que eles esperavam e agora estão tentando uma variação mais simples: dado um conjunto de jogadores, o módulo deverá obter uma *partição* dos jogadores em no máximo dois grupos, satisfazendo a restrição que cada jogador deve ter um número ímpar de amigos no seu grupo. Acontece que nem sempre isso é possível. Sua tarefa é decidir se é ou não possível obter a partição.

### Entrada

A primeira linha da entrada contém dois inteiros,  $P$  e  $F$ , respectivamente o número de jogadores e o número de amizades, onde  $2 \leq P \leq 100$  e  $1 \leq F \leq P \times (P - 1)/2$ . Cada uma das próximas  $F$  linhas contém dois inteiros,  $A$  e  $B$ , indicando que  $A$  e  $B$  são amigos, onde  $1 \leq A, B \leq P$  e  $A \neq B$ . Cada relação de amizade é dada no máximo uma vez, isto é, se uma linha contém os inteiros  $A$  e  $B$ , nenhuma outra linha contém tais inteiros.

### Saída

A saída contém uma única linha, contendo um único caractere. Se for possível fazer a partição em dois grupos, escreva a letra maiúscula ‘Y’; caso contrário, escreva a letra maiúscula ‘N’.

<b>Exemplo de entrada 1</b> 4 4 4 2 1 3 2 3 1 4	<b>Exemplo de saída 1</b> Y
<b>Exemplo de entrada 2</b> 4 3 4 2 2 3 1 2	<b>Exemplo de saída 2</b> Y

Exemplo de entrada 3	Exemplo de saída 3
5 5 3 5 3 1 1 4 2 5 2 4	N

## Problem K

# Between Us

Ties are usually an issue on elections and games. Recently, a new game, called *Between Us*, was created. The game is played by a number of players over a social network. On each turn, there are multiple votings, where a player may receive only votes from friends. The player getting the largest number of votes wins the game.

The game is still in its design phase, but the game developers faced a very common issue: since the number of friends of a given player is usually small, ties are very common, which makes the game less fun. To avoid this situation, they decided to add a new module to the match-making system, where it will try to form groups of players such that each player has an odd number of friends in the same group.

This problems turned out to be more challenging than they expected and they are now focusing on a simpler variation: given a set of players, the module must find a *partition* of the players in at most two groups, satisfying the restriction that each player must have an odd number of friends in his group. The problem is that this partition is not always possible. Your task is to determine whether the partition is possible.

### Input

The first line of input contains two integers,  $P$  and  $F$ , the number of players and the number of friendship relations, respectively, satisfying  $2 \leq P \leq 100$  and  $1 \leq F \leq P \times (P - 1)/2$ . Each of the next  $F$  lines will contain two integers  $A$  and  $B$ , with  $1 \leq A, B \leq P$  and  $A \neq B$ , meaning that players  $A$  and  $B$  are friends. Each friendship relation is given at most once, that is, if a line contains  $A$  and  $B$ , no other line contains both numbers.

### Output

Output a single line containing one character. If the partition is possible, then write the upper case letter ‘Y’; otherwise write the uppercase letter ‘N’.

<b>Input example 1</b> 4 4 4 2 1 3 2 3 1 4	<b>Output example 1</b> Y
<b>Input example 2</b> 4 3 4 2 2 3 1 2	<b>Output example 2</b> Y

Input example 3	Output example 3
5 5 3 5 3 1 1 4 2 5 2 4	N

## Problema K

# K personas entre nosotros

Los empates suelen ser un problema en las elecciones y los juegos. Recientemente, un nuevo juego, llamado *Between Us*, fue creado. En el juego participan varios jugadores a través de una red social. En cada turno, hay múltiples votaciones, donde un jugador puede recibir solo votos de amigos. El jugador que obtiene la mayor cantidad de votos gana el juego.

El juego aún se encuentra en su fase de diseño, pero los desarrolladores del juego se enfrentaron a un problema muy común: dado que el número de amigos de un jugador suele ser pequeño, los empates son muy comunes, lo que hace que el juego sea menos divertido. Para evitar esto, decidieron agregar un nuevo módulo al sistema de emparejamiento, donde se intentará formar grupos de jugadores de manera que cada jugador tenga un número impar de amigos en el mismo grupo.

Estos problemas resultaron ser más difíciles de lo que esperaban y ahora se están enfocando en una versión más simple: dado un conjunto de jugadores, el módulo debe encontrar una *partición* de los jugadores en como máximo dos grupos, satisfaciendo la restricción de que cada jugador debe tener un número impar de amigos en su grupo. El problema es que esta partición no siempre es posible. Su tarea es determinar si la partición es posible.

### Entrada

La primera línea de entrada contiene dos enteros,  $P$  y  $F$ , el número de jugadores y el número de relaciones de amistad, respectivamente, que satisfacen  $2 \leq P \leq 100$  y  $1 \leq F \leq P \times (P - 1)/2$ . Cada una de las siguientes  $F$  líneas contiene dos enteros  $A$  y  $B$ , con  $1 \leq A, B \leq P$  y  $A \neq B$ , indicando que los jugadores  $A$  y  $B$  son amigos. Cada relación de amistad aparece a lo sumo una vez, es decir, si una línea contiene  $A$  y  $B$ , no hay otra línea con ambos números.

### Salida

Imprima en la salida una sola línea que contenga un carácter. Si la partición es posible, escriba la letra mayúscula ‘Y’; de lo contrario, escriba la letra mayúscula ‘N’.

<b>Ejemplo de entrada 1</b> 4 4 4 2 1 3 2 3 1 4	<b>Ejemplo de salida 1</b> Y
<b>Ejemplo de entrada 2</b> 4 3 4 2 2 3 1 2	<b>Ejemplo de salida 2</b> Y

Ejemplo de entrada 3	Ejemplo de salida 3
5 5 3 5 3 1 1 4 2 5 2 4	N