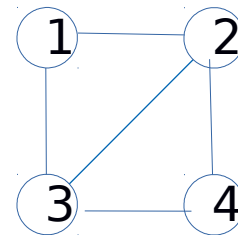
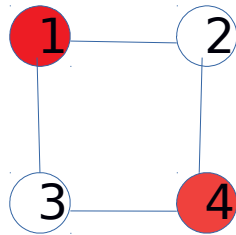


Importante: Los ejercicios deben entregarse a través de web (**Domjudge** y **Blackboard**). En un fichero con nombre:

La fecha de entrega: consultar la página de la actividad en blackboard

Grafo rojiblanco.

Realizar un programa que nos permite saber si un grafo puede pintarse de rojiblanco, es decir si sus vértices pueden colorearse utilizando dos colores (blanco y rojo) de tal forma que no exista ninguna arista que conecte dos vértices del mismo color. Por ejemplo el grafo de la izquierda es rojiblanco pero el de la derecha no puede ser pintado de rojiblanco.



El **input** estará compuesto por diversos casos de prueba. Para cada caso, la primera línea contiene el número de vértices del grafo, V (entre 1 y 100), y la segunda el número de aristas, A . A continuación aparecen A líneas, cada una con dos enteros que representan los extremos de cada una de las aristas (valores entre 0 y $V - 1$). Los grafos no contienen aristas de un vértice a sí mismo ni más de una arista que conecte un mismo par de vértices. El input se acabará cuando el número de nodos sea 0.

El **output** serán las respuesta SI o NO para indicar si cada uno de los grafos era rojiblanco o no.

Input:

Output:

4	SI
4	NO
0 1	
1 2	
2 3	
3 0	

4	
5	
0 1	
0 2	
1 2	
2 3	
3 0	
0	

NOTA: Para cada una de las funciones implementados se deberá incluir una pequeña descripción de su funcionamiento, sus precondiciones mediante `assertdomjudge` si las hubiera y el análisis de su complejidad temporal y espacial. Esta información deberá incluirse en la cabecera de cada función.