**Task 1:**

For a number n, we divide it by 2 if n is even; or we change it to 3\*n+1 if it is odd. It can be shown that transforming a number through this steps can reduce it to 1. For example, **n=11 → n= 34 -->n=17 → n=52-->n=26-->n=13-->n=40-->n=20-->n=10-->n=5-->n=16-->n=8-->n=4-->n=2-->n=1---14 steps to turn n into 1.**

You will be given a number **n. Print the number of steps to turn it into 1.**

**Sample input:**

**34**

**Sample Output:**

**14**

**----------------------------**

**Task 2:**

Write a Java program which will take two integers **n and m**  and print “Yes, they have X divisors both” (X is the number of divisors) if they have the same number of divisors or “No” if they do not.

**Sample Input:**

**24**

**30**

**Sample Output:**

**Yes, they have 8 divisors both**

**Explanation:**

**24’s divisors = {1,2,3,4,6,8,12,24}**

**30’s divisors = {1,2,3,5,6,10,15,30}**

**Both have 8 divisors**