C. Three Sided Polygon

Limits 1s, 1 GB

A simple closed curve or a polygon formed by three sides is called a triangle. However, any three sides will not always make a triangle. The three sides must come together to form a closed region. **To make a valid triangle the sum of the lengths of any two sides of a triangle has to be greater than the length of the third side.**

Triangles are categorized into three categories based on the side values of a valid triangle: Equilateral triangles, Isosceles triangles, and Scalene triangles. But for this problem we will consider only two types of valid triangle **- Equilateral and Isosceles.**

Equilateral - All three sides of a valid triangle are equal.

Isosceles - Exactly two of the sides of a valid triangle are equal.

You will be given a rod of length N. You have to cut this rod into 3 integer pieces such that the sum of these 3 pieces is equal to N and the size of every piece must be greater than 0. You can use these 3 pieces as 3 individual sides. Using these 3 sides you have to make an Equilateral triangle. If it's not possible to make an Equilateral triangle then try to make an Isosceles triangle. If none of them are possible then declare that.

Input

The input line contains a single integer N **- the length of the rod.**

$3 \leq N \leq 10^6$

Output

On the output line print **"Equilateral",** If an Equilateral triangle can be formed. If it's impossible to make an Equilateral triangle but it's possible to make an Isosceles triangle then Print **"Isosceles".** Otherwise print **"Invalid".**

Samples

Input	Output
3	Equilateral

Input	Output
10	Isosceles

Submit

C C11 GCC 12.1 ▾

Choose File No file chosen

Submit   Open Editor