**Chitti Robot Movement**

Consider a robot in the position (0, 0). The robot can move in all the four directions i.e left, right, up and down. With the given robot moves print whether the robot moved in a circle or not.

**Input Format**

First line will contain a String pattern ie Moves of Robot

**Constraints**

pattern can contain only L R U D (Characters String)

**Output Format**

Second line will print a boolean value

**Sample Input 0**

LR

**Sample Output 0**

true

**Sample Input 1**

UU

**Sample Output 1**

false

## Adjacent characters

Given a string and we have to swap its adjacent characters(pairs).

Here, to swap adjacent characters of a given string - we have a condition, which is "string length must be EVEN i.e. string must contains even number of characters".

**Input Format**

1. Take a string from the user.

**Constraints**

1. String length should not exceed 1000.

**Output Format**

1. It should print the swapped string.
2. if string length is odd then print "Odd length."

**Sample Input 0**

Hello

**Sample Output 0**

Odd length.

**Sample Input 1**

help

**Sample Output 1**

ehpl

## Write a function 1

We add a Leap Day on February 29, almost every four years. The leap day is an extra, or intercalary day and we add it to the shortest month of the year, February. In the Gregorian calendar three criteria must be taken into account to identify leap years:

The year can be evenly divided by 4, is a leap year, unless: The year can be evenly divided by 100, it is NOT a leap year, unless: The year is also evenly divisible by 400. Then it is a leap year. This means that in the Gregorian calendar, the years 2000 and 2400 are leap years, while 1800, 1900, 2100, 2200, 2300 and 2500 are NOT leap years

You are given the year, and you have to write a function to check if the year is leap or not.

Note that you have to complete the function and remaining code is given as template.

**Input Format**

Read y, the year that needs to be checked.

**Constraints**

1900<=y<=2019

**Output Format**

Output is taken care of by the template. Your function must return a boolean value (True/False)

**Sample Input 0**

2018

**Sample Output 0**

False

**Sample Input 1**

2000

**Sample Output 1**

True

## Valid Substring (ST2)

Given a string consisting of opening and closing parenthesis, find length of the longest valid parenthesis substring.

Example

Input: )()())

Output : 4

Explanation: ()()

**Input Format**

1. Take the parenthesis string from the user.

**Constraints**

1. String length should not be greater than 1000.

**Output Format**

1. It should print the length of longest valid substring.

**Sample Input 0**

)()())

**Sample Output 0**

4