B. Obsession with Robots

time limit per test: 2 seconds
memory limit per test: 64 megabytes

The whole world got obsessed with robots,and to keep pace with the progress, great Berland's programmer Draude decided to build his own robot. He was working hard at the robot. He taught it to walk the shortest path from one point to another, to record all its movements, but like in many Draude's programs, there was a bug — the robot didn't always walk the shortest path. Fortunately, the robot recorded its own movements correctly. Now Draude wants to find out when his robot functions wrong. Heh, if Draude only remembered the map of the field, where he tested the robot, he would easily say if the robot walked in the right direction or not. But the field map was lost never to be found, that's why he asks you to find out if there exist at least one map, where the path recorded by the robot is the shortest.

The map is an infinite checkered field, where each square is either empty, or contains an obstruction. It is also known that the robot never tries to run into the obstruction. By the recorded robot's movements find out if there exist at least one such map, that it is possible to choose for the robot a starting square (the starting square should be empty) such that when the robot moves from this square its movements coincide with the recorded ones (the robot doesn't run into anything, moving along empty squares only), and the path from the starting square to the end one is the shortest.

In one movement the robot can move into the square (providing there are no obstrutions in this square) that has common sides with the square the robot is currently in.

Input

The first line of the input file contains the recording of the robot's movements. This recording is a non-empty string, consisting of uppercase Latin letters `L`, `R`, `U` and `D`, standing for movements left, right, up and down respectively. The length of the string does not exceed `100`.

Output

In the first line output the only word `OK` (if the above described map exists), or `BUG` (if such a map does not exist).

Examples

input	Copy
LLUUUR	

output	Copy
OK	

input	Copy
RRUULLDD	

output	Copy
BUG	

→ **Problem tags**

constructive algorithms | graphs | implementation | *1400

No tag edit access

→ **Contest materials**

- Codeforces Beta Round #8
- Tutorial (en)