Desperate Electrician

`electrician.c | electrician.C | electrician.java | electrician.p`

The Czech Technical University educates not only computer engineers, but also electricians, who may sometimes face challenging problems.

One electrician has been sent to a very high building to install new electrical wiring. Before starting his work, he needed to determine the state of the old wires. There were W wire endings on the ground floor and another W endings on the 42nd floor. The electrician knew for sure that each of the lower ends was connected to exactly one of the upper ends. However, there were no signs or marks, so he had no information about which of the wires is which.

The main trouble is that elevators are not working yet (because they are still not connected to the electricity). Therefore, we absolutely need to minimize the number of trips up and down.

The electrician has connectors able to connect an arbitrary number of wire endings together (on the same floor, of course). Then he can walk to the other end and measure which wires are connected. For example, if the electrician connects two wires only, it is easy to detect this pair at the other end. Unfortunately, it is impossible to distinguish between those 2 connected wires.

Your task is to write a program that determines the best way to mark all of the wires. It is allowed to connect any number of wires together and to have any number of connected groups (there are always enough connectors) simultaneously. The electrician may perform any number of measurements on each floor. At the end, the following conditions must be met:

1. Electrician is back on the ground floor.
2. All lower ends of the wires are marked with distinct numbers 1 to W.
3. All upper ends are marked with distinct numbers 1 to W.
4. Each wire has the same number on both ends.

Input Specification

The input consists of several test cases. Each test case is given as a single line with one integer W, the number of wires, $1 \leq W \leq 200$. The input is terminated by a line containing zero.

Output Specification

For each test case, your program should output a single line with the sentence "`Electrician needs` X `trips.`", where X is the minimal possible number of times he has to climb up to the 42nd floor and descend back down.

If it is not possible to mark the wires at all, output the sentence "`Bad luck!`" instead.

Sample Input	Output for Sample Input
`2`	`Bad luck!`
`3`	`Electrician needs 1 trips.`
`0`	