

Railroad

Problem ID: railroad2
CPU Time limit: 1 second
Memory limit: 1024 MB
Difficulty: 1.4

Theta likes to play with her DUPLO railway set. The railway set she has consists of pieces of straight tracks, curved tracks, Y-shaped switches, and X-shaped level junctions, as well as bridges that allow one track to cross over another. There are also straight tracks that are railroad crossings to allow car traffic to cross.

Close-ups of some of the pieces are shown below:



To play, she picks a number of X-shaped level junctions and a number of Y-shaped switches and connects them with straight and curved pieces, using bridges as necessary.

Because the set doesn't include any bumpers, she wants to build a closed track, like all the examples shown in the manual that came with the set:

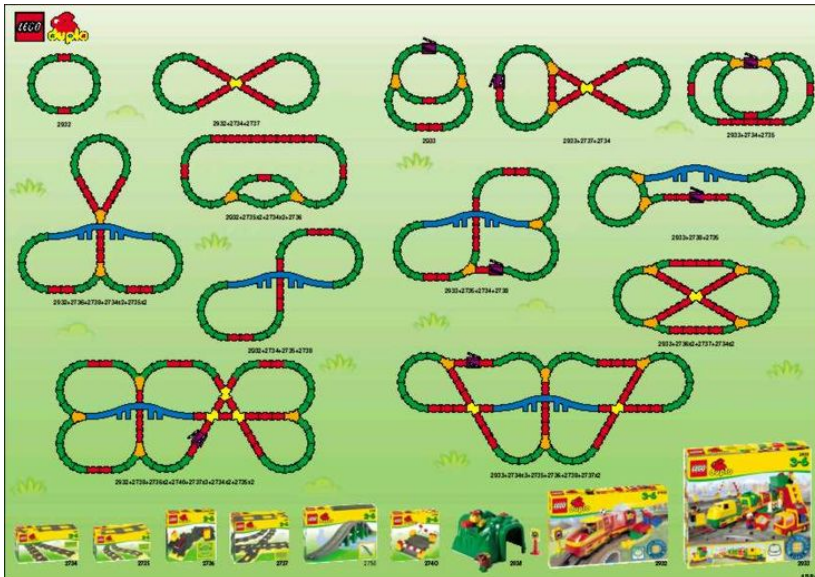


Figure 1: Various track layouts possible with the DUPLO railway system. Curved pieces are shown in green, straights in red, switches in orange, level junctions in yellow, bridges in blue, and crossings in black. DUPLO is a trademark of the LEGO Group.

Unfortunately, sometimes, this doesn't seem to work with the number of X-shaped level junctions and Y-shaped switches she starts out with.

She quickly figures out exactly when it is possible to build a closed track – can you figure it out, too?

Write a program that outputs if it is possible to build a railroad that does not require any bumpers (i.e., which does not have any dead-end tracks).

Input

The input consists of a single test case with two integer numbers X ($0 \leq X \leq 1000$) and Y ($0 \leq Y \leq 1000$) denoting the number of level junctions and switches, respectively. You may assume that Theta has sufficiently many straight and curved pieces as well as bridges.

Output

Output `possible` if she can build a closed track using all level junctions and all switches without any dead ends, or `impossible` otherwise.

Sample Input 1

1 0

Sample Output 1

possible

Sample Input 2

0 2

Sample Output 2

possible

Sample Input 3

1 3

Sample Output 3

impossible