A | Numbering Roads!

Input: Standard Input
Output: Standard Output

In my country, streets don't have names, each of them are just given a number as name. These numbers are supposed to be unique but that is not always the case. The local government allocates some integers to name the roads and in many case the number of integers allocated is less that the total number of roads. In that case to make road names unique some single character suffixes are used. So roads are named as 1, 2, 3, 1A, 2B, 3C etc. Of course the number of suffixes is also always limited to 26 (A, B, ..., Z). For example if there are 4 roads and 2 different integers are allocated for naming then some possible assignments of names can be:

1, 2, 1A, 2B
1, 2, 1A, 2C
3, 4, 3A, 4A
1, 2, 1B, 1C

Given the number of roads (**R**) and the numbers of integers allocated for naming (**N**), your job is to determine minimum how many different suffixes will be required (of all possible namings) to name the streets assuming that no two streets can have same names.

Input

The input file can contain up to **10002** lines of inputs. Each line contains two integers **R** and **N** (0<N,R<10001). Here **R** is the total number of streets to be named and **N** denotes number integers allocated for naming.

Output

For each line of input produce one line of output. This line contains the serial of output followed by an integer **D** which denotes the minimum number of suffixes required to name the streets. If it is not possible to name all the streets print "impossible" instead (without the quotes).

Sample Input

Sample Input	Output for Sample Input
8 5 100 2 0 0	Case 1: 1 Case 2: impossible

Problem Setter: Shahriar Manzoor, Special Thanks: Sohel Hafiz