Problem C
Knapsack in a Globalized World

Globalization stops at nothing, not even at the good old honest profession of a burglar. Nowadays it is not enough to break in somewhere, take everything you can carry and dart off. No! You have to be competitive, optimize your profit and utilize synergies.

So, the new game rules are:

- break only into huge stores, so there is practically endless supply of any kind of items;
- your knapsack should be huge;
- your knapsack should be full (there should be no empty space left).

Damn you, globalization, these rules are not easy to follow! Luckily, you can write a program, which will help you decide whether you should loot a store or not.

Input

The input consists of:

- one line with two integers n ($1 \leq n \leq 20$) and k ($1 \leq k \leq 10^{18}$), where n is the number of different item types and k is the size of your knapsack;
- one line with n integers $g_1, \ldots, g_n$ ($1 \leq g_i \leq 10^3$ for all $1 \leq i \leq n$), where $g_1, \ldots, g_n$ are the sizes of the n item types.

Output

Output "`possible`" if it is possible to fill your knapsack with items from the store (you may assume that there are enough items of any type), otherwise output "`impossible`".

Sample Input 1	Sample Output 1
`2 10000000000` `3 6`	`impossible`

Sample Input 2	Sample Output 2
`2 10000000000` `4 6`	`possible`

This page is intentionally left (almost) blank.