Problem F
Incremental Double Free Strings

A string is called **double free** if no two adjacent letters are the same.

A string is called k**-incremental** if for all values of j in the range $[1, k]$, there exists exactly one character with j occurrences, and the string's length is $1+2+3+\ldots+(k-1)+k$. For example, if $k = 3$, then a 3**-incremental** string should have one character appear once, another twice, another three times, in any order, for a total string length of 6.

A string is both k**-incremental** and **double free** if it meets both these criteria. Now consider examining all such strings of lowercase letters for a given k in alphabetical order. Consider the following examples.

$k = 2$: **aba**, **aca**, **ada**, ..., **aya**, **aza**, **bab**, **bcb**, **bdb**, ..., **zxz**, **zyz**

$k = 3$: **ababac**, **ababad**, ..., **ababay**, **ababaz**, **ababca**, ..., **zyzyzx**

What is the n^{th} string in an alphabetized list of all k**-incremental**, **double free** strings?

Input

Each input will consist of a single test case. Note that your program may be run multiple times on different inputs. There will be exactly one line of input. It will contain two integers, k and n ($1 \leq k \leq 26, 1 \leq n \leq 10^{18}$), which is asking for the n^{th} string in the alphabetically sorted list of all k**-incremental**, **double free** strings.

Output

Output the n^{th} k**-incremental**, **double free** string in the alphabetized list. If no such string exists, output -1.

Sample Input 1	Sample Output 1
2 650	zyz

Sample Input 2	Sample Output 2
2 651	-1

Sample Input 3	Sample Output 3
5 12345678901234	yuzczuyuyuzuyci