There are some candies that need to be distributed to some children as fairly as possible (i.e. the variance of result needs to be as small as possible), but I don't know how to distribute them, so I need your help. Your assignment is to write a function distribute(m, n) in which m represents how many candies there are, while n represents how many children there are. The function should return a List (or Array etc. depending on the specific language) which contains the number of candies each child gains.

Notice
The candy can't be divided into pieces.
The list's order doesn't matter.

Requirements
The case m < 0 is equivalent to m == 0.
If n <= 0 the function should return an empty list.
If there isn't enough candy to distribute, you should fill the corresponding number with 0.

Examples
distribute(-5, 0) # should be []
distribute(0, 0) # should be []
distribute(5, 0) # should be []
distribute(10, 0) # should be []
distribute(15, 0) # should be []
distribute(-5, -5) # should be []
distribute(0, -5) # should be []
distribute(5, -5) # should be []
distribute(10, -5) # should be []
distribute(15, -5) # should be []
distribute(-5, 10) # should be [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
distribute(0, 10) # should be [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
distribute(5, 10) # should be [1, 1, 1, 1, 1, 0, 0, 0, 0, 0]
distribute(10, 10) # should be [1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
distribute(15, 10) # should be [2, 2, 2, 2, 2, 1, 1, 1, 1, 1]

Input
m: Integer (m <= 100000)
n: Integer (n <= 1000)

Output
[Integer]