Problem Lock

Input data	`stdin`
Output data	`stdout`

Nelu just bought a new type of digital lock which he wants to use for the locker room at school. The secret code of this lock is a sequence of N natural numbers, indexed 1 through N. Entering this code and unlocking the device is done in a special manner. The lock starts with a displayed sequence composed of N values of zero. Nelu can then use an operation called $\text{incS}(i, j)$, which increments by 1 all values with indexes between i and j (inclusive). For example, using an $\text{incS}(2, 4)$ operation on the sequence [0, 0, 0, 0] will produce the sequence [0, 1, 1, 1]. Similarly, using an $\text{incS}(2, 3)$ on the sequence [4, 1, 3, 2] will produce the sequence [4, 2, 4, 2]. The device unlocks when the displayed sequence matches the secret code.

Because the lock is new, Nelu needs to set the secret code. Being passionate about permutations, he would like the secret code to be a permutation of the numbers 1 through N, that is, a sequence of N numbers that contains each number from 1 to N exactly once. Additionally, he wants to make the code difficult to guess by his classmates. For this, Nelu wants that the minimum number of incS operations required to unlock the device be exactly equal to his favourite number M. Among all possible such codes, if any exist, he will select the minimum lexicographic one (as explained in the Restrictions). Nelu asks your help to determine what his secret code should be.

Input Data

The input consists of one line containing two space separated integers N and M, with the respective meanings from the statement above.

Output Data

Output a sequence of N numbers, separated by spaces, representing the secret code that Nelu should set for the lock. If no such sequence exist, output the message `IMPOSSIBLE`.

Restrictions

- $1 \leq N \leq 10^6$
- $1 \leq M \leq 10^{12}$
- A permutation $A_1, A_2, \ldots, A_N$ is smaller lexicographically than another permutation $B_1, B_2, \ldots, B_N$, if there is a position P for which $A_1 = B_1, A_2 = B_2, \ldots, A_{P-1} = B_{P-1}$ and $A_P < B_P$.

#	Points	Restrictions
1	3	$N \leq 6$, $M = N$
2	3	$N \leq 6$, $M = N + 1$
3	11	$N \leq 9$
4	19	$N \leq 16$
5	43	$N \leq 1\,000$
6	21	No further constraints.

Examples

Input data	Output data
3 3	1 2 3
3 4	2 1 3
3 5	IMPOSSIBLE

Explanation

The permutations for $N = 3$ are [1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 1, 2] and [3, 2, 1]. The numbers of minimum incS operations required for these permutations are, in order: 3, 3, 4, 3, 4, 3. For example, for the [2, 1, 3] permutation, Nelu can use incS$(3, 3)$, incS$(1, 3)$, incS$(1, 1)$ and incS$(3, 3)$. However, Nelu cannot obtain [2, 1, 3] with less than 4 incS operations.

For $M = 3$, the minimum lexicographic permutation, for which the minimum number of incS required to unlock the device is exactly equal to M is [1, 2, 3]. For $M = 4$, the secret code is [2, 1, 3]. For $M = 5$, there is no such permutation.