Problem M. Most Annoying Constructive Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

The array $a_1, a_2, \ldots, a_m$ of integers is called **odd** if it has an odd number of inversions, and **even** otherwise. Recall that an inversion is a pair (i, j) with $1 \le i < j \le m$ such that $a_i > a_j$. For example, in the array $[2, 4, 1, 3]$, there are 3 inversions: $(1, 3), (2, 3), (2, 4)$ (since $a_1 > a_3, a_2 > a_3, a_2 > a_4$), so it is **odd**.

Given n, k, determine if there exists a permutation of integers from 1 to n, which has exactly k odd subarrays.

An array b is a subarray of an array c if b can be obtained from c by the deletion of several (possibly, zero or all) elements from the beginning and several (possibly, zero or all) elements from the end.

Input

The first line contains a single integer t $(1 \le t \le 10^4)$ — the number of test cases. The description of the test cases follows.

The only line of each test case contains two integers n, k $(1 \le n \le 1000, 0 \le k \le \frac{n(n-1)}{2})$.

It's guaranteed that the sum of n^2 over all test cases doesn't exceed $4 \cdot 10^6$.

Output

For every test case, if there is no such permutation, output `NO`.

Otherwise, output `YES`. In the next line, output n integers $p_1, p_2, \ldots, p_n$ $(1 \le p_i \le n$, all p_i are distinct$)$ — the elements of your permutation.

Example

standard input	standard output
4 1 0 3 3 4 1 6 15	YES 1 YES 3 2 1 YES 1 3 4 2 NO

Note

In the first test case, the permutation is (1); all its subarrays are even.

In the second test case, the permutation is $(3, 2, 1)$. It has 3 odd subarrays: $[3, 2], [2, 1]$ with 1 inversion each, and $[3, 2, 1]$ with 3 inversions.

In the third test case, the permutation is $(1, 3, 4, 2)$. It has exactly 1 odd subarrays: $[4, 2]$ with 1 inversion.

It can be shown that no such permutation exists for the fourth test case.