

Problem K. Petya's Cryptography

Time limit: 1 second
Memory limit: 512 megabytes

Petya received a failing grade in cryptography, but it was the teachers who did not understand his genius, not him being lazy. To convince the entire world community of his genius, Petya created a new public key encryption system — *PSA*. Any self-respecting public key cryptosystem has a public key and a private key.

As a private key, Petya chose a tree T , and as a public key — two numbers (n, p) , where n is the number of vertices in the tree T , and p is the number of paths of length 2 in T . Recall that a tree is an undirected connected graph that does not contain cycles.

The peculiarity of Petya's cryptosystem is that any private key corresponding to the public key will suffice to break it. But that's not a problem; Petya chose quite a complex task, right? Restore any private key of the cryptosystem *PSA* or state that such a public key could not have been produced.

Input

The input consists of a single line containing two numbers n and p ($1 \leq n \leq 1000$, $0 \leq p \leq 10^9$).

Output

If a solution exists, output "Yes" in the first line. In the next $n - 1$ lines, output two distinct integers from 1 to n — the edges of the tree.

If no solution exists, output "No" in a single line.

Examples

standard input	standard output
7 11	Yes 1 2 2 3 3 4 3 5 3 6 3 7
5 5	No