Isomorphic Delight

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Given a number n, create a simple undirected graph on n nodes, that is asymmetric and has the least number of edges, or output that no such graph exists.

A graph is asymmetric if there are no relabelings of the vertices (except the identity permutation), such that you obtain exactly the same graph.

Formally: For a graph (V, E) to be asymmetric, there should **not** exist a permutation π of the vertices, such that π is not the identity permutation, and it holds that: $uv \in E \Leftrightarrow \pi(u)\pi(v) \in E$.

Input

The first and only line in the input contains one integer n $(1 \leq n \leq 10^6)$ — the number of nodes the graph should have.

Output

Output "`YES`" if there exists an asymmetric graph with n nodes, otherwise print "`NO`". If the answer is "`YES`", on the following lines output a description of such a graph with the lowest number of edges.

The first line of the description is a single integer m, the number of edges in your graph. Each of the next m lines should contain 2 integers u and v, denoting an undirected edge between nodes u and v. No undirected edge should appear more than once in the output (otherwise the graph is not simple), and the graph should be asymmetric.

Examples

standard input	standard output
1	YES 0
6	YES 6 1 2 2 3 1 3 3 4 2 5 5 6
4	NO