XOR Tree

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

Edward needs to find an unrooted tree which has n nodes. Each node is attached with a unique number in $1, 2, \ldots, n$. A pair (a, b) is called a good pair if the XOR sum of the numbers in the sample path between node a and node b is zero. A tree with n nodes is called a good tree if there are more than n good pairs in it. Pair (a, b) and pair (b, a) are the same, counted only once. Can you help Edward to find such a good tree?

Input

There is an integer n in a line. ($1 \leq n \leq 10000$)

Output

In the first line, print `Yes` if such trees can be found, otherwise `No`. If the first line of the output is `Yes` , then output $n - 1$ lines, each line is two integers a and b, denoting an edge of any good tree you've got.

Examples

standard input	standard output
2	No
17	Yes 9 8 9 11 9 13 9 15 9 16 11 10 13 12 15 14 16 17 8 1 1 5 5 4 4 6 6 7 7 2 2 3

Note

In the second sample, there are 18 good pairs, which are $(1, 3)$, $(1, 4)$, $(1, 9)$, $(3, 6)$, $(3, 10)$, $(3, 12)$, $(3, 14)$, $(3, 17)$, $(4, 10)$, $(4, 12)$, $(4, 14)$, $(4, 17)$, $(5, 7)$, $(7, 9)$, $(8, 10)$, $(8, 12)$, $(8, 14)$ and $(8, 17)$.