Problem J. Knight's Tour Redux

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

Consider a n by n chessboard with squares labelled $(1, 1)$ through (n, n). On this chessboard lies a long knight. The long knight can move from square (x, y) to (x', y') if one of the two following conditions hold:

- $|x - x'| = 3$ and $|y - y'| = 1$
- $|x - x'| = 1$ and $|y - y'| = 3$

In essence, it is a normal chess knight, but longer.

A 'tour' of the chessboard is a sequence of squares $S_1, S_2, S_3, \ldots S_n$ such that for all $1 \leq i \leq n - 1$ the move from S_i to S_{i+1} is a valid move for a long knight. Such a tour is considered 'complete' if and only if the tour visits each row and column of the chessboard **exactly** once.

For each positive integer n, determine whether it is possible for a complete tour of an n by n chessboard to exist and construct one such tour, if possible.

Input

The first line of input consists of a single integer n ($1 \leq n \leq 10^5$) — the size of the chessboard.

Output

If it is not possible to create a 'complete' tour output the string "IMPOSSIBLE" in the only line.

Otherwise, output "POSSIBLE" in the first line.

The next n lines should contain the values x_i, y_i — the position of the i-th square in the complete tour.

Examples

standard input	standard output
1	POSSIBLE 1 1
2	IMPOSSIBLE