Just Some Bad Memory

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Relax. Let me tell you in the fastest pace what you need to do.

Give you a *simple* graph $G = (V, E)$ consisting of undirected edges. You need to tell me, what is the minimum number of edges should you add to the graph, resulting a simple graph containing at least one *odd cycle* and at least one *even cycle*.

A *simple* graph is a graph without multiple edges and self-loops, which means that each edge connects two different vertices and no two edges connect the same pair of vertices.

A *cycle* is a sequence of distinct vertices $\{v_1, v_2, \ldots, v_k\}$, such that $(v_i, v_{i \bmod k+1}) \in E$. The *odd* or *even* describes the parity of k. A smallest odd cycle is of length 3, and a smallest even cycle is of length 4.

Input

The first line contains two integers n, m $(1 \le n \le 10^5, 0 \le m \le \min\left\{2 \times 10^5, \binom{n}{2}\right\})$, denoting the number of vertices ($|V|$) and the number of edges ($|E|$).

In the next m lines, each line contains two integers u, v $(1 \le u, v \le n, u \ne v)$, denoting that there are edges connecting vertices u and v.

It's guaranteed that the input graph is a simple graph.

Output

Print one integer in a single line, denoting your answer. If the mission is impossible, print '`-1`' instead.

Examples

standard input	standard output
3 3 1 2 2 3 1 3	-1
4 0	5
5 4 1 2 2 3 3 4 4 5	2
4 6 1 2 1 3 1 4 2 3 2 4 3 4	0
4 4 1 2 2 3 3 4 4 1	1
7 7 1 2 2 3 3 4 4 1 5 6 6 7 7 5	0

Note

Here is one possible solution of sample 2. The contained odd cycles are $\{1, 2, 3\}$ and $\{1, 3, 4\}$, and the only even cycle is $\{1, 2, 3, 4\}$.