Monopol

Problem ID: monopol

Jocke and his friends usually play Monopoly with each other. After countless games, they have grown tired of the usual rules, so they have changed them a bit.

First, they choose a suitably sized country. They then look at the road network in the country and choose a sequence of **different** cities $v_1, v_2, \ldots, v_k$ that form a *cycle*. This means that there is a direct road between cities v_i and v_{i+1} for all $1 \leq i < k$ and between v_k and v_1, just like on a Monopoly board. Then they travel to the country and play by driving around the cycle in their cars to buy and sell properties with real money.

However, there is a restriction that makes it difficult to carry out the game: they must find a suitable cycle in the road network. Some countries have very large road networks. What makes it even more difficult is that the cycle must have an even number of roads, otherwise, the rules won't work ("Free Parking" does not end up in the middle, which creates an unbalanced game).

Given all the cities in the country and the roads between pairs of cities, find a cycle consisting of an even number of roads if there is one.

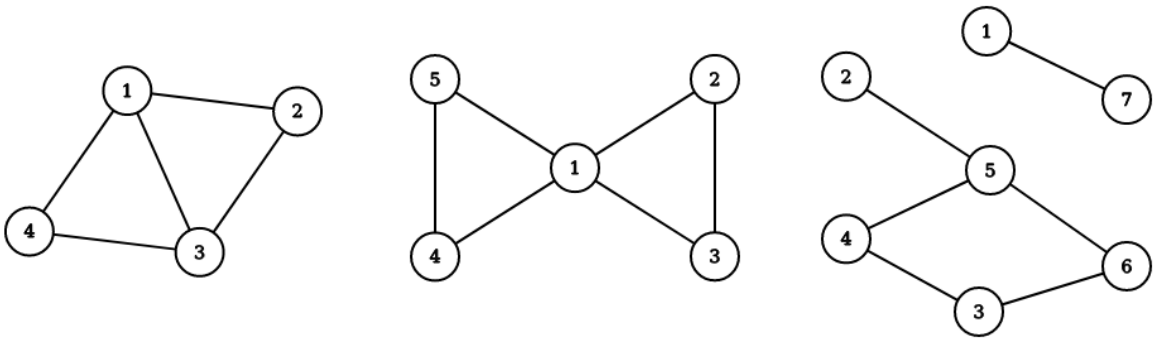

Figure 1: Illustration of the countries in the three example cases.

Input

The first line contains two integers N ($1 \leq N \leq 10^5$) and M ($0 \leq M \leq \min(2 \cdot 10^5, \frac{N(N-1)}{2})$), the number of cities and the number of roads in the road network, respectively.

Then follow M lines, each with two integers a and b, which means that there is a road between cities a and b in the country ($1 \leq a \neq b \leq N$). It is guaranteed that there are no multiple roads between the same pair of cities in the country.

Output

If there is no even cycle, print a single line with the string "`NO`".

If there is an even cycle, print a single line with the string "`YES`". Then, you should print such a cycle. First, print a line with an **even** integer k ($4 \leq k \leq N$), the number of cities in your cycle. On the next line, print k **different** integers $v_1, v_2, \ldots, v_k$ ($1 \leq v_i \leq N$) separated by spaces: the cities in your cycle. It must hold that there are roads between the cities $(v_1, v_2), (v_2, v_3), \ldots, (v_{k-1}, v_k), (v_k, v_1)$.

If there are multiple possible cycles, you can print any of them.

Points

Your solution will be tested on several test case groups. To get the points for a group, it must pass all the test cases in the group.

Group	Point value	Constraints
1	18	$N \leq 10$
2	16	$N \leq 100$ and $M \leq 200$
3	17	The cities can be divided into two parts so that no road goes between two cities in the same part.
4	13	All cities in the country have a direct road to at most 2 cities.
5	20	All cities in the country have a direct road to at most 3 cities.
6	16	No additional constraints.

Sample Input 1

```
4 5
1 2
1 3
2 3
3 4
4 1
```

Sample Output 1

```
YES
4
3 2 1 4
```

Sample Input 2

```
5 6
1 2
1 3
1 4
1 5
2 3
4 5
```

Sample Output 2

```
NO
```

Sample Input 3

```
7 6
1 7
3 4
4 5
5 6
6 3
5 2
```

Sample Output 3

```
YES
4
6 3 4 5
```