

Tower Defense Game

Memory limit: 128 MB

Bytie is playing the computer game *Tower Defense*. His aim is to construct guard towers, so that they protect all of his domain. There are multiple towns in Bytie's domain, some of which are linked by bidirectional roads. If Bytie erects a guard tower in a city, then the tower protects its city and all the cities directly linked with it by roads.

Just as Bytie was pondering over the placement of guard towers in his domain, his elder sister Bytea entered the room. She glanced at the map displayed on the screen, and after a moment exclaimed: "*Oi, what is there to think about, clearly k towers suffice!*".

Angered by his sister spoiling the fun, Bytie showed his sister the door, and began wondering what to do next. Pride will not let him construct more than k towers. He has an up his sleeve though: he can research a technology that will allow him to construct *improved* guard towers. An improved guard tower protects not only the town it was erected in and its immediate neighbors but also the towns that are further away. Formally, an improved guard tower built in the town u protects the town v if either of the following holds:

- $u = v$;
- there is a direct road from u to v ;
- or there is such a town w that there are direct roads from u to w and from w to v .

Of course, Bytie still strives to erect at most k towers, but he has no qualms about making these the improved guard towers.

Input

In the first line of the standard input, there are three integers, n , m , and k ($2 \leq n \leq 500\,000$, $0 \leq m \leq 1\,000\,000$, $1 \leq k \leq n$), separated by single spaces, that specify, respectively, the number of towns and roads in Bytie's domain, and the number k given by Bytea.

The towns in Bytie's domain are numbered from 1 to n . Next, m lines describing the roads follow. Each of those lines holds two integers, a_i and b_i ($1 \leq a_i, b_i \leq n$, $a_i \neq b_i$), indicating that the towns no. a_i and b_i are directly linked by a bidirectional road. Each pair of towns is linked by at most a single road.

Output

Your program is to print two lines, describing the placement of improved guard towers in Bytie's domain, to the standard output. The first line is to hold an integer r ($1 \leq r \leq k$), the number of improved guard towers that Bytie should construct. The second line is to specify the placement of these by providing r pairwise disjoint integers that are the numbers of town where the improved guard towers are to be built. The town numbers can be given in an arbitrary order.

If more than one solution exists, any solution can be printed. Let us remind that any placement of no more than k improved towers will do - you need not minimize their number. You may assume that Bytea was correct, i.e., that the whole domain of Bytie can indeed be protected by k plain (not improved) guard towers. Thus a solution always exists.

Example

For the input data:

```
9 8 3
1 2
2 3
3 4
1 4
3 5
4 6
7 8
8 9
```

the correct result is:

```
3
1 5 7
```

Sample grading tests:

- 1ocen: $n = m = 10$, $k = 5$, the road network forms a cycle;
- 2ocen: $n = 1\,414$, $m = 998\,991$, $k = 100$, each pair of towns is linked by a road; notice that a single tower would suffice in this case;
- 3ocen: $n = 500\,000$, $m = 499\,999$, $k = 250\,000$, the road network forms a path.