Submission deadline (on or before):

- 31st October, 2021, 10:00 PM

Policies for Submission and Evaluation:

- You must submit your assignment in the Eduserver course page, on or before the submission deadline.
- Your programs should be written in C language and should be compatible with the `gcc` compiler in Linux.
- During the evaluation, failure to execute programs without compilation errors may lead to zero marks for the evaluation.
- Detection of ANY malpractice related to the course can lead to awarding an F grade in the course.

Naming Conventions for Submission

- Submit the source file as a single C (.c) file (no other file types will be accepted on Eduserver). The file must be named as

  `DCS_<ROLLNO>_<FIRST-NAME>_B.c`

  (For example: *DCS_BxxyyyyCS_LAXMAN_B.c*). If you do not conform to the above naming conventions, your submission might not be recognized by our automated tools, and hence will lead to a score of 0 marks for the submission. So, make sure that you follow the naming conventions.

Standard of Conduct

- If we find two similar code submissions, both cases will be given zero marks, irrespective of any claim as to who wrote the original code or who submitted copied code. Please note that working with code available on the web is also considered plagiarism. Serious cases will be penalized up to F grade in the course irrespective of the marks scored in other quizzes/examinations, in accordance with the department's integrity policy (please refer to the course plan for the link).

1. Write a program to check whether a given undirected connected graph G, with no self-loops or multiple edges, is bipartite or not. If G is not bipartite, then output a cycle having odd length in G. Otherwise, output any two vertex sets, say $A, B \subseteq V(G)$, such that $A \cup B = V(G)$, $A \cap B = \varnothing$ and every edge in $E(G)$ has one endpoint each in A and B.

Input Format

The first line of the input contains an integer $n \in [1, 10^3]$, the number of vertices in G. The vertices are implicitly labeled from 0 to $n - 1$.

The second line of the input contains an integer $m \in [0, 10^6]$, the number of edges in G.

The next m lines of the input each contain two *distinct* space-separated integers, say $u, v \in [0, n - 1]$, that indicates the presence of the undirected edge (u, v) in G.

Output Format

If the graph is not bipartite, the output should contain a sequence of space-separated integers representing an arbitrary odd length cycle in G.

Otherwise, the output should contain 2 lines, listing the vertices in sets A and B, respectively, as space-separated integers.

Notes:

1. If the graph is not bipartite, it can contain multiple odd length cycles. You are required to find and print one such cycle. Any vertex on the cycle may be taken as the starting vertex, but the vertices of the cycle should be printed in the order in which they lie on the cycle. You should not repeat the starting vertex at the end of the sequence.
2. If the graph is bipartite, then there could be different pairs of sets A and B that satisfy the given specifications. You are required to find one such pair of sets and print them.

Sample Input and Output

Input 1

```
6
7
0 4
0 2
3 1
5 2
1 4
3 0
3 5
```

Possible Output

```
2 4 3
5 0 1
```

Input 2

```
6
8
0 4
0 2
3 1
5 2
1 4
3 0
3 5
1 5
```

Possible Output

```
5 2 0 4 1
```

Assignment Evaluation Policy

1. Full marks (12) will be given if the program is not plagarized and passes all the test cases of the evaluation team.
2. Partial credits (8-11) will be given if the program is not plagiarized and works satisfactorily in most test cases.
3. If the program is not plagiarized and is a genuine attempt at the problem though not working satisfactorily, up to 5 marks will be given.
4. If the plagiarism check fails, then zero marks will be given.