**01. Cycle Detection**

**Condition:**

You have a directed graph, represented as a list of vertices and edges. The vertices are numbered from 0 to n-1, where n is the number of vertices in the graph. You need to check whether the graph contains a cycle.

**Input:**

* On the first line we read the number n , which represents the number of vertices.
* On the second line we read the number m , which represents the number of edges.
* And on the next n lines we read the pairs of edges, which are written with a separator between each of the numbers , two per line.

**Output:**

* If the graph contains a cycle, return **True** .
* If the graph does not contain a cycle, return **False** .

**Limitations:**

The graph will not contain duplicate edges. The number of vertices and edges can be very large, but is limited by the available memory of the computer.

**Examples:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4  4  0 1  1 2  2 0  2 3 | True |