The Celebrity Problem

A celebrity is a person who is known to all but does not know anyone at a party. If you go to a party of N people, find if there is a celebrity in the party or not.
A square NxN matrix M[][] is used to represent people at the party such that if an element of row i and column j is set to 1 it means ith person knows jth person. Here M[i][i] will always be 0.
Note: Follow 0 based indexing.

Example 1:

Input:
N = 3
M[][] = {{0 1 0},
{0 0 0},
{0 1 0}}
Output: 1
Explanation: 0th and 2nd person both
know 1. Therefore, 1 is the celebrity.

Example 2:

Input:
N = 2
M[][] = {{0 1},
{1 0}}
Output: -1
Explanation: The two people at the party both
know each other. None of them is a celebrity.

Your Task:
You don't need to read input or print anything. Complete the function **celebrity()** which takes the matrix M and its size N as input parameters and returns the index of the celebrity. If no such celebrity is present, return -1.

Expected Time Complexity: O(N)
Expected Auxiliary Space: O(1)

```
def celebrity(self, M, n):
        # code here
        stack = []
        for i in range(n):
            stack.append(i)
```

```
        while len(stack)>1:
            i = stack.pop()
            j = stack.pop()
            if M[i][j]==1:
                stack.append(j)
            else:
                stack.append(i)
        pop = stack.pop()
        for i in range(n):
            if i!=pop:
                if M[i][pop] == 0 or M[pop][i]==1:
                    return -1
        return pop
```