

# Dominating Set

Input file:            `standard input`  
Output file:          `standard output`  
Time limit:           2 seconds  
Memory limit:        64 megabytes

Bobo had a bipartite graph  $G = (V, E)$  with  $n$  vertices and  $m$  edges. He would like to choose a subset  $D$  of vertices, such that for every vertex  $v$ , either  $v$  or one of its neighbours is in  $D$ . Find the number of possible subsets bobo might choose.

Note:

1.  $G$  is bipartite if and only if  $G$  contains no cycles of odd-length.
2.  $w$  is neighbour of  $v$  if and only if  $u$  and  $v$  are connected by an edge.

## Input

The first line contains 2 integers  $n, m$  ( $1 \leq n \leq 30, 0 \leq m \leq 225$ ).

The  $i$ -th of the following  $m$  lines contains 2 integers  $a_i, b_i$ , which denotes an edge between the  $a_i$ -th and  $b_i$ -th vertices ( $1 \leq a_i, b_i \leq n$ ).

It is guaranteed that there is no self loops and multiple edges.

## Output

An integer denotes the number of subsets bobo might choose.

## Examples

| standard input                  | standard output |
|---------------------------------|-----------------|
| 4 4<br>1 2<br>2 3<br>3 4<br>4 1 | 11              |
| 4 0                             | 1               |