5) Algorithm dfs (~~root~~ u, v, G)

Input : edge (u, v) and graph G

Output : True if edge is a bridge, false otherwise.

If u or v have no adjacent vertices:

    return false

path := empty stack

push u to path

mark u and v as visited

while stack != empty :

    ~~node~~ node = path.pop()

    if node is not u and adjacent vertex is v:

        return false

    for adjacent vertex of node :

        if adjacent vertex not visited :

            mark adjacent vertex visited

            push adjacent vertex to path

return true

$O(m+n)$