Bridge (G = (V, E))

Input: Graph G that is simple and connected

Output: edges that are bridges

Let originalPath be EXPLORE (graph)

Let res be an empty list

for all edges:

remove edge from Graph

Let newPath be EXPLORE (graph)

if len(originalPath) != len (newPath), do:

Add edge to res

add edge back to graph

return res

EXPLORE ($G = (V, E)$, v)

Input : A graph G that is simple and connected.

output : Groups of vertices that are connected to one another

---

Let all Path be an empty set

Let explored be an empty set

for all the vertex in G:

    if vertex is not in explored, do:

        Add PATH(G, vertex) to all Path

        Add all PATH(G, vertex) to explored

return all Path

PATH (G = (V, e), v)

Input : Graph G that is connected and simple.
vertex V as the starting vertex

output : A set of vertices that are traversable
from vertex v.

Let path be an empty set()

Let explored be an empty set

if v is not in explored, do

mark v as explored

add v to path

for all the neighbours of v

Let subPath be PATH (G, neighbor)

Add all vertices in subPath to path

return path

Time complexity analysis

E = # of edges

Path = $T(n) = 1 \cdot T(E) + O(1)$
$= O(E)$

Explore = $T(n) = 1 \cdot T(E) \cdot O(E) + O(1)$

$= O(E \cdot E)$

Bridge = $T(n) = 1 \cdot T(E)\, O(E \cdot E) + O(1)$

$= \boxed{O(E^2)}$

Theorem: Let $G = (V, E)$ be a connected graph. An edge $e \in E$ is a bridge if and only if there exist no cycles traversing e in G