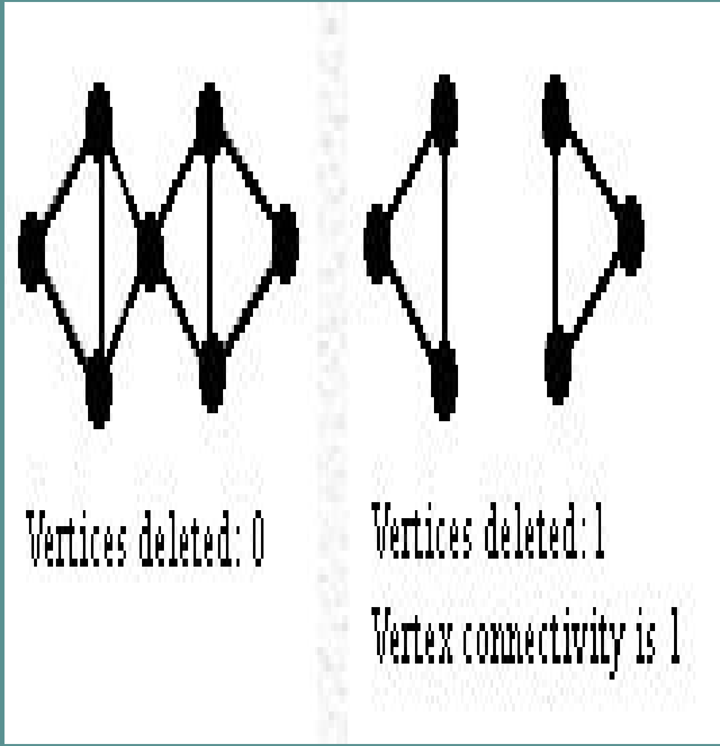
Find out the vertices contributing to vertex connectivity

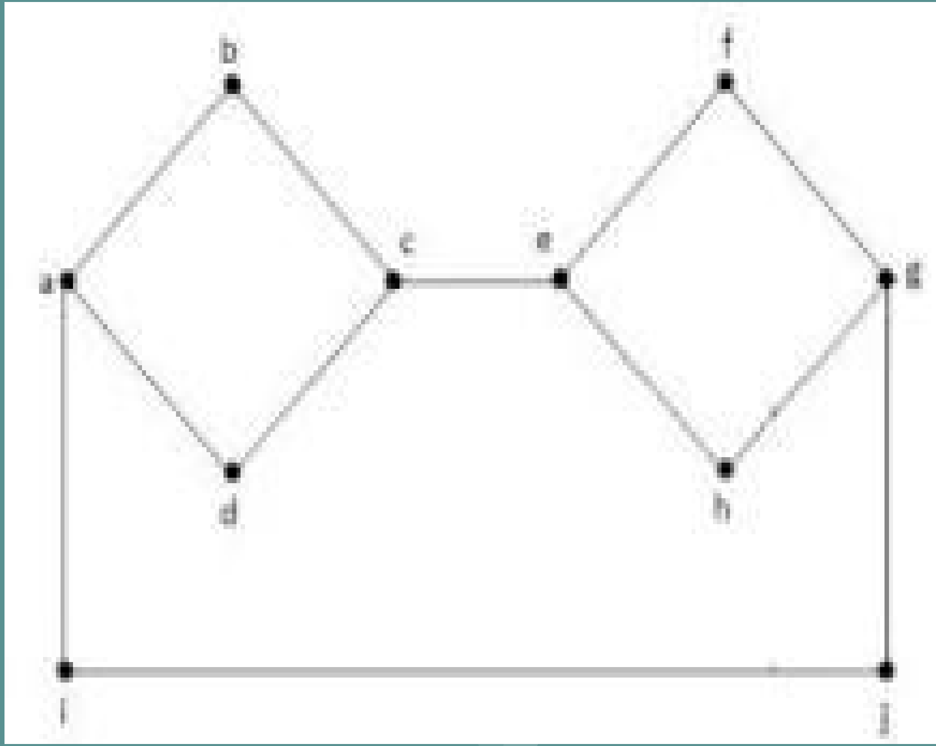
Submitted by:

Shiv Pratap Singh
IIT2014121

Nazish Tabassum
ICM2014504

Arqum Ahmed
IWM2014008





Vertex Connectivity is 2

Algorithm 1

A. Brute Force Approach:

Input : A graph G with cardinality equal to n.
Output : A set of vertices S whose size is equal to vertex connectivity of graph G and it contains all vertices enlisted in vertex connectivity.

```
FindVerticesInVertexConnectivity(G)
    RemoveSelfLoop
    RemoveParallelEdges
    S = emptySet
    If graph is disconnected or n = 1
      return S
    If graph is complete
      Select any n-1 vertices and add it to S
      Return S
    For i = 1 to (n-2)
       Generate every combination of vertices of size i
       remove those vertices and add it to set S
       If graph is disconnected
          return S
       empty set S
```

Algorithm 2

```
RemoveSelfLoop
RemoveParallelEdges
S = emptySet
If graph is disconnected or n = 1
  return S
If graph is complete
  Select any n-1 vertices and add it to S
  Return S
Replace each edge (x, y) ∈ E with arcs (x, y) and
(y, x), and call the resulting digraph D.
For each pair v, w in G which are non-adjacent
    For each vertex u other than v and w in G,
    replace u with two new vertices u1 and u2, and
    then add the new arc (u1, u2). Connect all the
    arcs that were coming to u in G to u1, and
    similarly, connect allthe arcs that were going
    out of u in G to u2 in D.
    Assign v as the source vertex and w as the sink
    vertex.
     Assign the capacity of each arc to 1, and call
    the resulting network H.
     T = ComputeMinCutUsingMaxFlow(H, v, w)
     if (|T|< |S|)
       S = T
     Restore  network H back to D.
return S
```

ComputeMinCutUsingMaxFlow(H, v, w)

1. Run Ford-Fulkerson algorithm and consider the final residual graph
2. Find the set of vertices that are reachable from source in the residual graph.
3. All edges which are from a reachable vertex to non-reachable vertex are minimum cut edges.
4. Find the corresponding vertices to edges all the edges that form minimum cut edges.
5. Add all those vertices to Set T
6. Return T

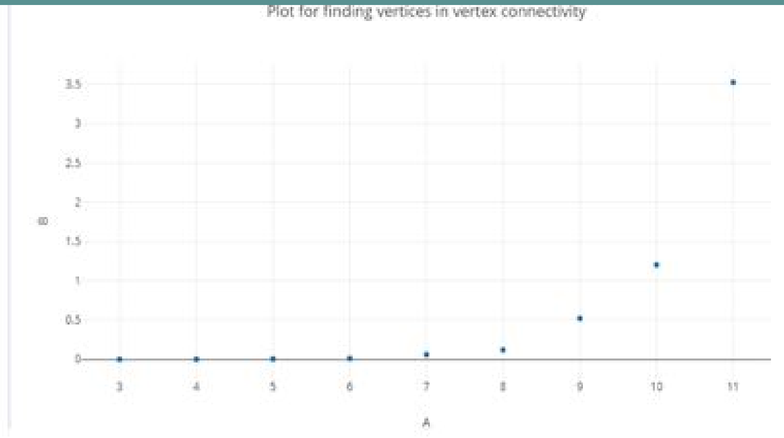


Fig. 1. Algorithm1 graph

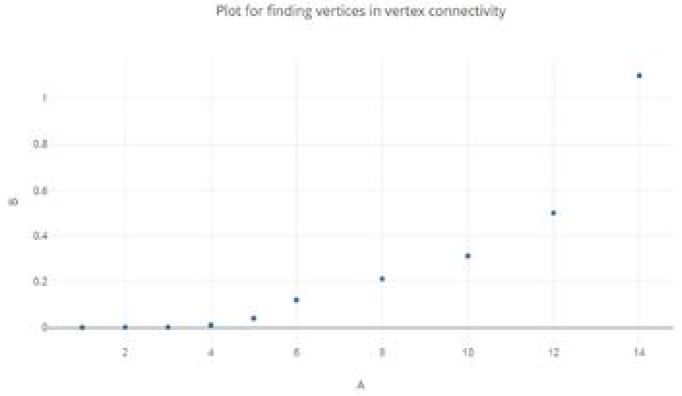


Fig. 2. Algorithm2 graph