**Algorithm 2:** Algorithm to Find the total number of spanning trees and minimum spanning trees in a Incomplete Graph

---

**Data:** An undirected weighted graph $G(V, E, W)$

**Result:** Number of Spanning trees and Minimum Spanning Trees

$A[m][n] = AdjacencyMatrix(G(V, E))$;

$D[m][n] = DegreeMatrix(G(V, E))$;

$L[m][n] = D[m][n] - A[m][n]$;

$N = (-1)^{(1+1)}|M_{11}|$;

$P = store(edgeList)$;

$print("Number of Spanning Tree is", N)$;

$minList = []$;

**for** $i = 1$ to $|N|$ **do**

    $S[i] = \sum_{n=1}^{|V|-1} P[i].W[n]$;

    $i = i+1$;

    $minList.append(S[i])$;

**end**

$minMST = min(minList)$;

$countMST = count(minList, minMST)$;

$print("Total number of MST is", countMST)$;