



ACADEMY OF TECHNOLOGY

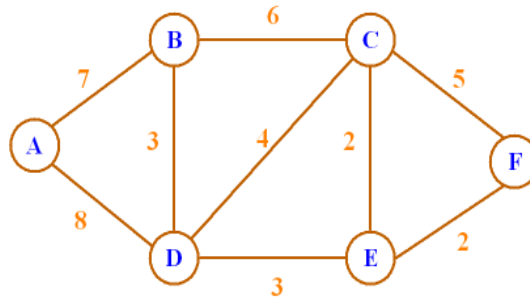
Lab Assignment (Day 8)

Paper name: Design and Analysis of Algorithms Lab
Code: PCC-CS494
Discipline: CSE

Semester: 4th
Time: 2 Hours

Date: April 10, 2023

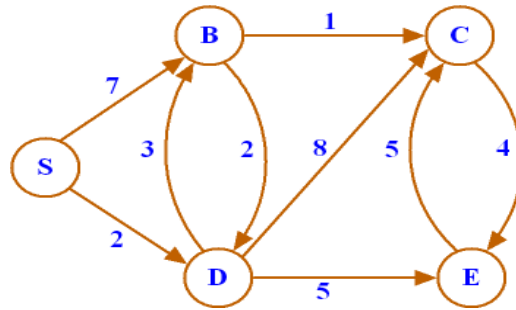
1. Given an undirected weighted graph, write a program in C or C++ to find a minimum spanning tree by applying Prim's Algorithm.



Algorithm 1: Prim (G, w, r)

```
1 for each vertex  $u \in G(V)$  do
2    $key[u] := \infty$ ; // Initialization
3    $parent[u] := nil$ ;
4    $mstSet[u] := false$ ;
5 end
6  $key[r] := 0$ ; // Start from root
7 Make a min priority Queue  $Q$ ; // min-heap is used for min priority Queue
8 while  $Q$  is not empty do
9   // Until all vertices in MST
10   $u := ExtractMin(Q)$ ; // Delete a minimum valued vertex from heap
11   $mstSet[u] := true$ ;
12  for each  $v$  adjacent from  $u$  do
13    if  $mstSet[v] = false$  and  $w[u, v] < key[v]$  then
14       $DecreaseKey(v, w[u][v])$ ; // Update heap accordingly
15       $parent[v] := u$ ;
16    end
17  end
18 end
```

2. Write a program in C or C++ to find single source shortest path applying Dijkstra Algorithm. (You may use the following graph as input graph).



Algorithm 2: Dijkstra(G, w, s)

```

1 for each vertex  $u \in G(V)$  do
2    $dist[u] := \infty$ ; // Initialization
3    $pred[u] := nil$ ;
4    $sptSet[u] := false$ ;
5 end
6  $dist[s] := 0$ ; // Start from root
7 Make a min priority Queue  $Q$ ;
  // Make a min priority Queue
8 while  $Q$  is not empty do
  // Until all vertices in MST
9    $u := extractMin(Q)$ ; // Delete a minimum valued vertex from heap
10   $sptSet[u] := true$ ;
11  for each  $v$  adjacent from  $u$  do
12    if  $sptSet[v] = false$  and  $dist[u] + cost[u, v] < dist[v]$  then
13       $decreaseKey(v, dist[u] + cost[u][v])$ ; // Update heap
        accordingly
14       $pred[v] := u$ ;
15    end
16  end
17 end

```
