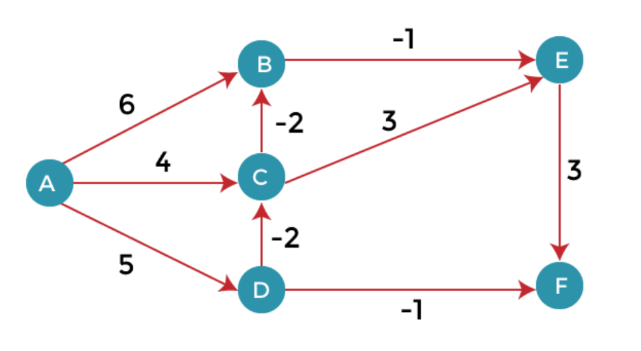
**22AIE203 – DATA STRUCTURES & ALGORITHMS 2**

**ASSIGNMENT**

Name : Anuvind MP

Roll no.: AM.EN.U4AIE22010

----------------------------------------------------------------------------------------------------------



Using Bellman ford Algorithm, find the shortest distance from source vertex ‘A’ to the remaining vertices in the given graph.

**Code :**

def bellman\_ford(nodes, edges, source=0):

    path\_lengths = {v: float('inf') for v in nodes}

    path\_lengths[source] = 0

    for i in range(len(nodes)-1):

        dup = path\_lengths.copy()

        for (u, v), w in edges.items():

            if path\_lengths[u]+w < path\_lengths[v]:

                path\_lengths[v] = path\_lengths[u] + w

        if dup == path\_lengths:

            return path\_lengths

    return path\_lengths

nodes = ['A','B','C','D','E','F']

edges = {

    ('A','B'): 6, ('A','C'): 4, ('A','D'): 5,

    ('B','E'):-1, ('C','B'): -2, ('C','E'): 3,

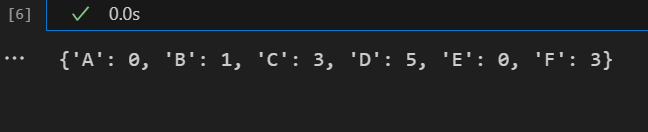
    ('D','C'):-2, ('D','F'):-1, ('E','F'): 3

}

shortestPathLengths = bellman\_ford(nodes, edges, source="A")

print(shortestPathLengths)

**Output :**

****