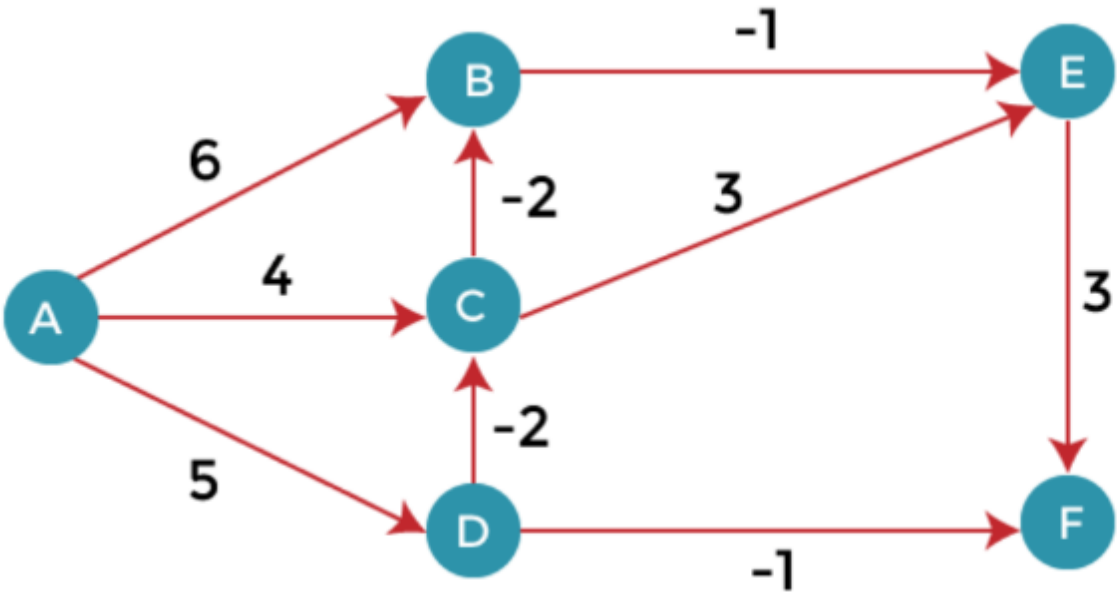
# 22AIE203 – DATA STRUCTURES & ALGORITHMS 2

# 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 :**