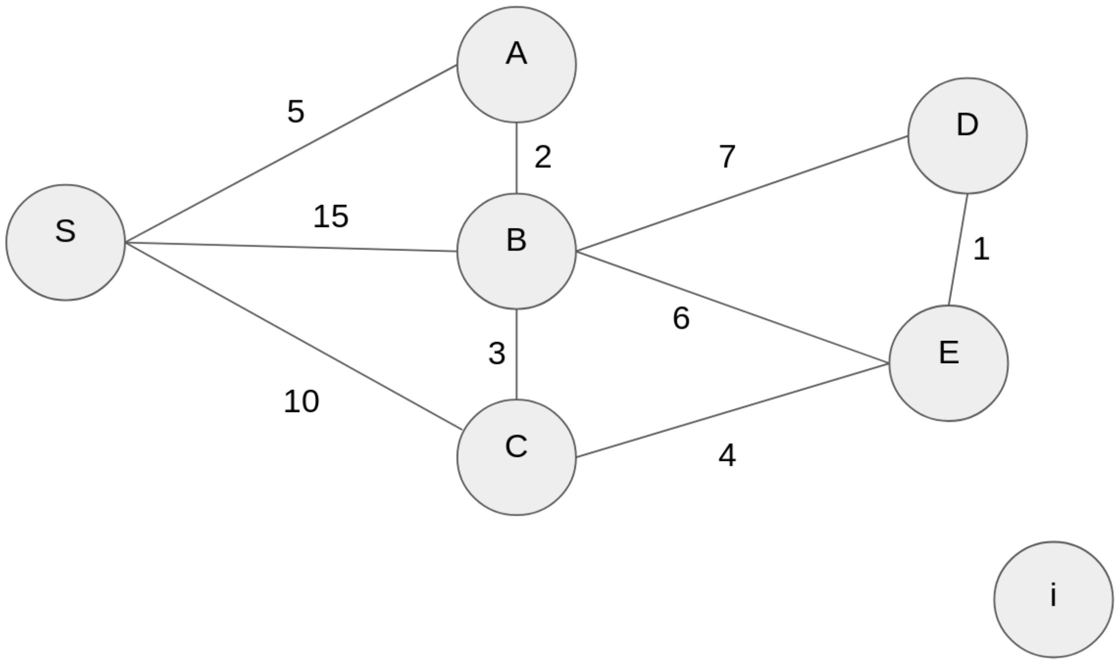
Cole Stephens
211 Data Structures
Dijkstra’s Algorithm - GYHD worksheet



```
function Dijkstra(Graph, source):

    create vertex set Q

    source.dist = 0
    for each vertex v in Graph:
        if(v is not source)
            v.dist = INFINITY
        v.prev = null
        Q.addWithPriority(v, v.dist)

    while Q is not empty:

        u = Q.extractMin() // O(log(n))


        for each neighbor v of u:           // only v that are still in Q
            alt = u.dist + weight(u, v)
            if alt < v.dist:
                v.dist = alt
                v.parent = u
                Q.decreasePriority(v, v.dist) //O(log(n)) if we can find it quickly

    // return depends on the application
```

Nodes should be of the form (Name, Cost, Parent) i.e. (B, 15, S), (C, 10, S) (i, INF, null). Remember to not consider finished neighbors!

Finished Nodes (in order of when they are finished):

Initial PQ: (S, 0, null) (A, INF, null) (B, INF, null) (C, INF, null) (D, INF, null) (E, INF, null) (I, INF, null)

Removed: (S, 0, null)

PQ after considering:

Removed:

PQ a/c:

Removed:

PQ a/c:

Removed:

PQ a/c:

Removed:

PQ a/c:

Removed:

PQ a/c:

Removed:

PQ a/c: