

Dijkstra's Algorithm

Formulate The Problem: Dijkstra's Algorithm is an algorithm used to determine the shortest path between two nodes. In this algorithm we have a graph (G) which is a data structure for holding nodes or (V) vertices, and also has Edges (E). E's have a weight or cost associated with them (W). The source and destination are both nodes in the graph G. V's have a value associated with them is the cost to get to that node from the source node. This means that V_{Source} has a value of 0, and all other V have a value of ∞ . At the start of the Algorithm.

Pseudo Code:

Algorithm 1 Dijkstra

```
1: procedure DIJKSTRA( $G, V_{Source}, V_{Destination}$ )
2:   for each  $v \in V$  do
3:     if  $v \equiv V_{source}$  then
4:        $v.cost \leftarrow 0$ 
5:        $v.cost \leftarrow \infty$ 
6:        $P\text{-}Queue \leftarrow v$ 
7:   while  $P\text{-}Queue$  has elements do
8:      $V_{cur} \leftarrow P\text{-}Queue.pop$ 
9:     if  $V_{cur} \equiv V_{Destination}$  then
10:      break
11:     for each neighbor  $n$  of  $V_{cur}$  do
12:        $dist \leftarrow V_{cur}.cost + length(V_{cur}, n)$ 
13:       if  $n.cost > dist$  then
14:          $dArray[n.cost] \leftarrow dist$ 
15:          $pArray[n] \leftarrow V_{cur}$ 
16:   return  $dArray, pArray$ 
```

Proof of Correctness: