DIJKSTRA'S ALGORITHM

(SINGLE-SOURCE SHORTEST PATH ALGORITHM)

- an algorithm that is used for finding the shortest path from starting node to target node in a weighted graph
- The algorithm makes a tree of the shortest path from the starting node (source node) to all the other nodes in the graph.

Formula used to calculate the shortest distance

$$dist[i][j] = min(dist[i][j], dist[i][K] + dist[K][j])$$

i ⇒ source vertex

j ⇒ destination vertex

K ⇒ intermediate vertex

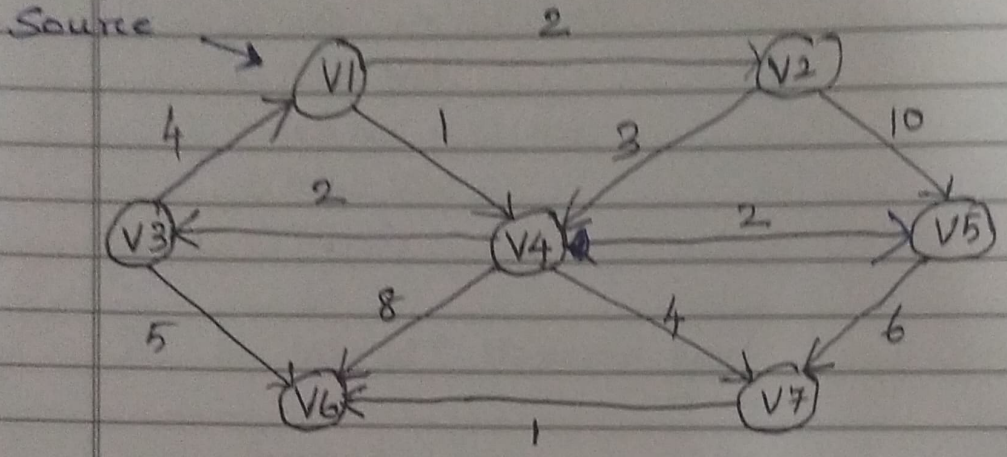
Conditions

- Dijkstra's algorithm works only for connected graphs.
- The alg. works only for those graphs that do not contain any negative weight edges.

Date

DIJKSTRA'S ALGORITHM

- single source shortest path algorithm
- used to find the shortest path from source vertex to destination in a graph.



Vertices.	Kv	Dv	Pv
V1	0	0	0
V2	0	∞	0
V3	0	∞	0
V4	0	∞	0
V5	0	∞	0
V6	0	∞	0
V7	0	∞	0

Vertices	Kv	Dv	Pv
V1	1	0	0
V2	0	2	V1
V3	0	4	V1
V4	0	1	V1
V5	0	∞	0
V6	0	∞	0
V7	0	∞	0

Vertices	Kv	Dv	Pv
V1	1	0	0
V2	0	2	V1
V3	0	3	V4
V4	1	1	V1
V5	0	3	V4
V6	0	9	V4
V7	0	5	V4

Vertices	Kv	Dv	Pv
V1	1	0	0
V2	1	2	V1
V3	0	3	V4
V4	1	1	V1
V5	0	3	V4
V6	0	9	V4
V7	0	5	V4

Vertices	Kv	Dv	Pv
V1	1	0	0
V2	1	2	V1
V3	1	3	V4
V4	1	1	V1
V5	0	3	V4
V6	0	8	V3
V7	0	5	V4

Vertices	Kv	Dv	Pv
V1	1	0	0
V2	1	2	V1
V3	1	3	V4
V4	1	1	V1
V5	1	3	V4
V6	0	8	V3
V7	0	5	V4

Vertices	Kv	Dv	Pv
V1	1	0	0
V2	1	2	V1
V3	1	3	V4
V4	1	1	V1
V5	1	3	V4
V6	0	6	V7
V7	1	5	V4

Vertices	Kv	Dv	PV
V1	1	0	0
V2	1	2	V1
V3	1	3	V4
V4	1	1	V1
V5	1	3	V4
V6	1	6	V7
V7	1	5	V4

The shortest path from source to other vertices are

V1 → V2 = 2
V1 → V3 = 3
V1 → V4 = 1
V1 → V5 = 3
V1 → V6 = 6
V1 → V7 = 5

Algorithm

```
Void Dijkstras (Graph G, Table T)
{
 int i;
 Vertex v, w;
 Read Graph (G,T);
 for (i=0; i< no.of vertices; i++)
 {
   T[i].Known = FALSE; // False = 0
   T[i].Dist = INFINITY;
   T[i].Path = FALSE;  // False = Not a vertex
 }.
 T[Start].dist = 0

 V = vertex with smallest distance
 T[V].Known = TRUE   // True = 1

 for each vertex u adjacent to v
 if (!T[u].Known)
 {
  T[u].dist = min (T[u].dist,
                   T[v].dist + C_vw)

 T[u].Path = v;
 }.
```