Shri Ville Parle Kelavani Mandal’s

Institute of Technology, Dhule

Department of Information Technology

Design and Analysis of Algorithms Lab

Sr No.	Student Name	Roll No.	PRN	Class
1	Ankita pravin sonawane	57	2254491246057	Second Year

- Dijkstra ‘s algorithm is a Graph algorithm that find the shortest path from a source vertex to all other vertices int the Graph (single source shortest path). It is a type of Greedy algorithm that only works on weighted Graph having positive weights.

- Dijkstra ‘s algorithm is an algorithm for finding the shortest path between nodes in a weighted graph, which may represent, for example, road networks.

- Dijkstra’s algorithm finds the shortest path from a given source node to every other node. It can also be used to find the shortest path to a specific destination node, by terminating the algorithm once the shortest path to the destination node is known.

- For example, if the node of the graph represent cities, and the costs of edges represent the average distances between pairs of cities connected by a direct road, then Dijkstra’s algorithm can be used to find the shortest route between one city and all other cities. A common application of shortest path algorithms is network routing protocols.

Algorithm:

(A)Initialize the pathlength of all vertices to infinity and predecessor of all vertices to NIL. Make the status of all vertices temporary.

(B)Make the pathlength of source vertex equal to 0.

(C)From all the temporary vertices in the graph, find out the vertex that has minimum value of pathlength, Make it permanent and now this is our current vertex.

(D)Examine all the temporary vertices adjacent to the current vertex. The value of pathlength is recalculated for all these temporary successors of current, and relabelling is done if required.

Suppose s is the source vertex, current is the current vertex and v is a temporary vertex adjacent to current.

(i)if Pathlength(current) + weight(current, v) < pathlength v)

Pathlength(v) = pathlength(current) + weight(current , v)

Predecessor(v) = current

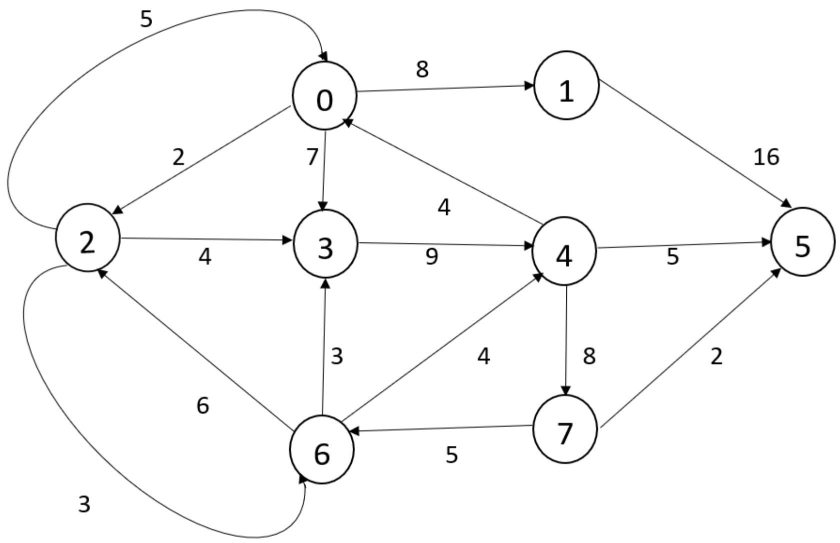
(ii)if pathlength(current) + weight(current , v)>= pathlength(v)

It means that using current vertex in the path from s to v does not offer any shorter path. So in this case vertex v is not relabelled and values of pathlength and predecessor for vertex v remain unchanged.

(E) repeat steps C and D until there is no temporary vertex left in the graph, or all the temporary vertices left have pathlength of infinity.

Problem:

Consider the graph given below. Assuming 0 as source vertex obtain the shortest paths using Dijkstra's algorithm.

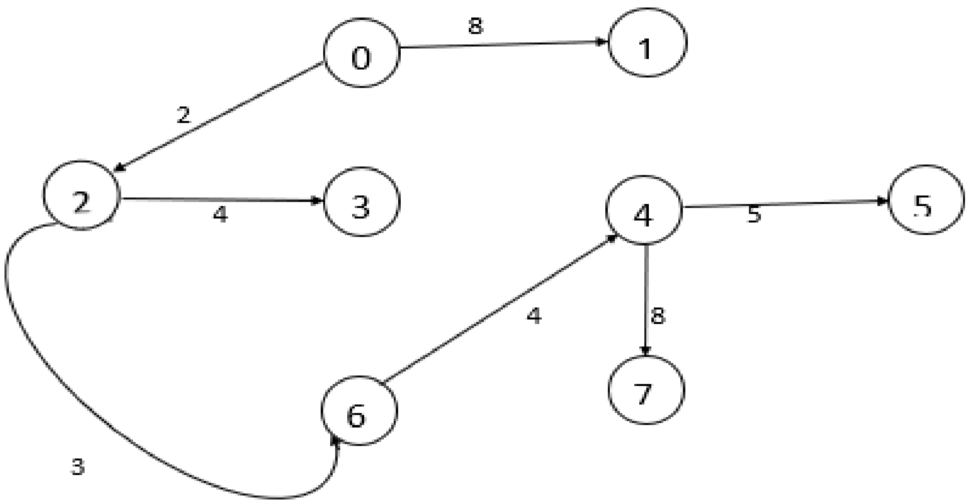


Current vertex	Adjacent Vertex	Adjacent Vertices Redefinition	Pathlength
0	1,2,3	from all the temporary Vertices, vertex 0 has the Smallest pathlength, so Make it permanent and make it current. Predecessor is NIL (0). $PL(c) + wt(c,v) < PL(v)$ V=1 $PL(0) + wt(0,1) < PL(1)$ $0 + 8 < \infty$ $\boxed{PL(1)= 8}$ Pre(1)= 0 V=2 $PL(0) + wt(0,1) < PL(1)$ $0 + 2 < \infty$ $\boxed{PL(2)= 2}$ Pre(2)= 0 V=3 $PL(0) + wt(0,3) < PL(3)$ $0 + 7 < \infty$ $\boxed{PL(3)= 7}$ pre(3)=0	1 2 3 8 2 7

current vetex	adjacent vertex	adjacent vertices redefinition	pathlength
2	3 , 6	from all the temporary Vertices, vertex 2 has the Smallest pathlength, so Make it permanent and make it current. V=3 PL(2) + wt(2,3) < PL(3) 2 + 4 < ∞ $\boxed{PL(3)=\ 6}$ Pre(3)= 2 V=6 PL(2) + wt(2,6) < PL(2) 2 + 3 < ∞ $\boxed{PL(6)=\ 5}$ Pre(6)=2	3 6 6 5
6	3 , 4	from all the temporary Vertices, vertex 6 has the Smallest pathlength, so Make it permanent and make it current. V=3 PL(6) + wt(6,3) < PL(3) 5 + 3 < ∞	3 4 6 5

current vetex	adjacent vertex	adjacent vertices redefinition	pathlength
		5 + 3 > 6 Don't relabel 3 V=4 PL(6) + wt(6,4) < PL(4) 5 + 4 < ∞ $\boxed{\text{PL(4)= 9}}$ Pre(4)=6	3 4 9
3	4	from all the temporary Vertices, vertex 3 has the Smallest pathlength, so Make it permanent and make it current. V=4 PL(3) + wt(3,4) < PL(4) 6 + 9 > 9 Don't relabel 4	4
1	5	from all the temporary Vertices, vertex 1 has the Smallest pathlength, so Make it permanent and make it current. V=5 PL(1) + wt(1,5) < PL(5)	5

current vetex	adjacent vertex	adjacent vertices redefinition	pathlength
		$8 + 16 < \infty$ $\boxed{PL(5)=24}$ Pre(5)= 1	5 24
4	5 , 7	from all the temporary Vertices, Vertex 4 has smallest pathlength, so Make it permanent and make it current. V= 5 PL(4) + wt(4,5) < PL(5) $9 + 5 < 24$ $\boxed{pL(5) = 14}$ Pre(5) = 4 PL(4) + wt(4,7) < PL(7) $9 + 8 < \infty$ $\boxed{PL(7) = 17}$ Pre(7) = 4	5 7 14 17
5	_	_	_
7	_	_	_



Vertex	Path length	predecessor	status
0	0	NIL	perm
1	8	0	perm
2	2	0	perm
3	6	2	perm
4	9	6	perm
5	14	4	perm
6	5	2	perm
7	17	4	perm