SINGLE SOURCE SHORTEST PATHS:

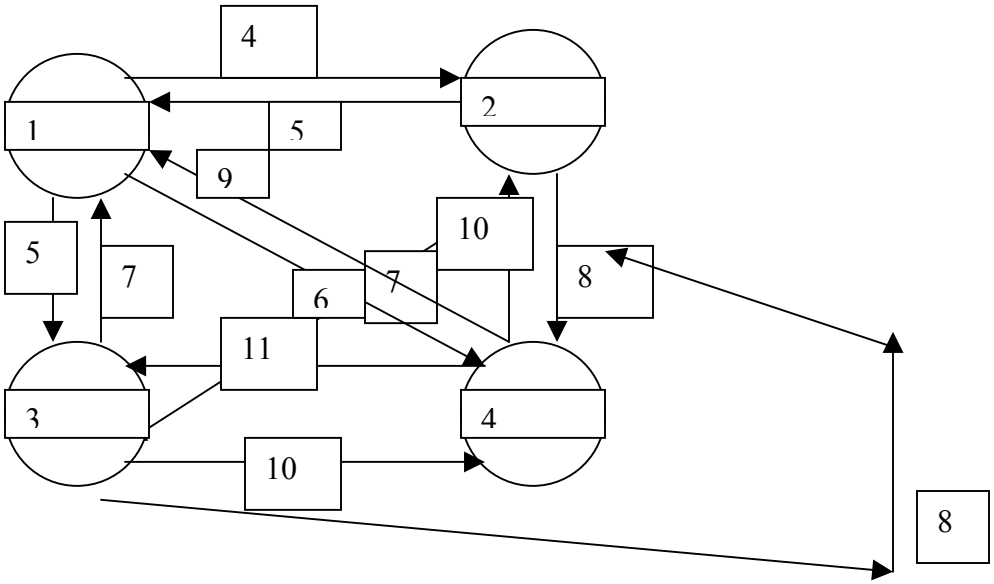
DIJKSTRA'S ALGORITHM

Consider a complete directed graph of 4 nodes, where the vertices are v_i for i between 1 and 4 and the weight of an edge (v_i,v_j) is 2i+j, for I not = j.. Obtain shortest paths from node 1 to all the other nodes using **Dijkstra's single source shortest path algorithm**. What is the time complexity of your algorithm? Discuss. What happens when some of the weights of the edges are negative?

COMPLEXITY
$T(n) = O(n^2)$

WORKS ONLY WHEN EDGE WEIGHTS ARE POSITIVE

PICTORIAL VIEW OF THE GRAPH



Initialisation

Iteration	S	Vertex selected	1	2	3	4
initial	---	1	∞	4	5	6
Path			1	1-2	1-3	1-4

Shortest from node 1 is now node 2

Iteration	S	Vertex selected	1over	2over	3	4
initial	---	1	∞	4	5	6
			1	1-2	1-3	1-4
1	{1}	2			5	6
					1-3	1-4

(1 to 2) to 3 is 11, so (1 to 3) remains shortest.

(1 to 2) to 4 is 12, so (1 to 4) remains shortest.

Least distance is now to node 3, of length 5.

Iteration	S	Vertex selected	1over	2over	3	4
initial	---	1	∞	4	5	6
			1	1-2	1-3	1-4
1	{1}	2			5	6
					1-3	1-4
2	{1,2}	3				6
						1-4

(1 to 3) to 4 is 15 so 6 remains shortest above.

Iteration	S	Vertex selected	1over	2over	3	4

initial	---	1	∞	4	5	6
			1	1-2	1-3	1-4
1	{1}	2			5	6
					1-3	1-4
2	{1,2}	3				6
						1-4