Floyd Warshell Algorithm →

This algorithm considers the intermediate vertices of a shortest path where an intermediate vertex of a simple path $P=\langle v_1, v_2, \ldots v_m\rangle$ is any vertex of P other than v_1 and v_m.

The floyd warshell algo is based on the following:

Let vertices of G be $V=\{1,2,3,\ldots n\}$

Consider a subset of vertices $\{1,2,\ldots k\}$

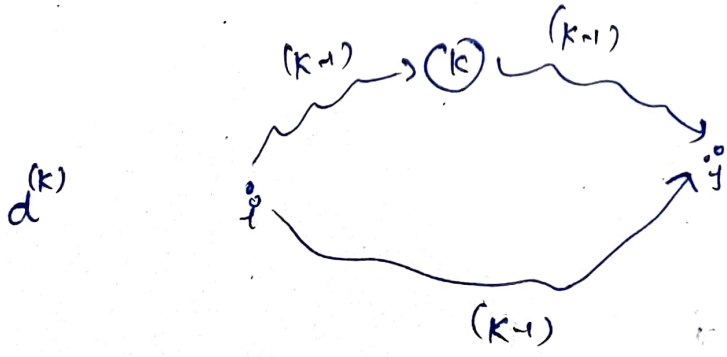
For any pair of vertices $i,j \in V$, consider all paths from i to j whose intermediate vertices are all drawn from $\{1,2,\ldots,k\}$ and let p be the minimum weight path among them.

→ If k is not an intermediate vertex of path P, then all intermediate vertices of path P are in the set $\{1,2,\ldots k-1\}$.

→ If k is an intermediate vertex of path P, then we break P down into $i \overset{P_1}{\rightsquigarrow} k \overset{P_2}{\rightsquigarrow} j$

→ Let d_{ij}^{k} be the weight of a shortest path from vertex i to vertex j with all intermediate vertices in the set $\{1,2,\ldots k\}$

$$d_{ij}^{k} = \begin{cases} w_{ij} & \text{if } k=0 \\ \min\left(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{k-1}\right) & \text{if } k \geq 1 \end{cases}$$



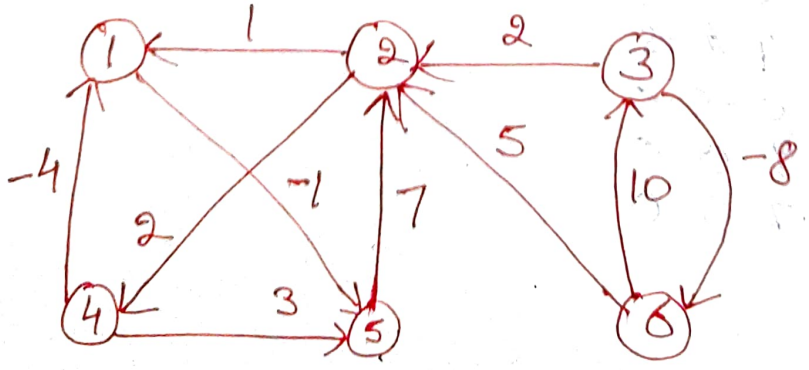
Algorithm:

```
Floyd_warshell(w)
1.  n ← rows(w)
2.  D^0 ← w
3   For k ← 1 to N
4       do for u ← 1 to N
5           do for j ← 1 to N
6               do d_uj^k = min(d_uj^(k-1), d_uk^(k-1) + d_kj^(k-1))
7   return D^n
```

The strategy adopted by Floyd warshell algo is called Dynamic Programming.

The running time is $O(n^3)$

Q. Apply Floyd warshell algorithm for constructing shortest path. Show the matrix D^k that results in each iteration. Also find the minimum cost from node 3 to node 1 and the corresponding path also.



Sol^{M}.

$D^0 =$

	1	2	3	4	5	6
1	0	∞	∞	∞	-1	∞
2	-1	0	∞	2	∞	∞
3	∞	2	0	∞	∞	-8
4	-4	∞	∞	0	3	∞
5	∞	7	∞	∞	0	∞
6	∞	5	10	∞	∞	0

$\pi_0 =$

	1	2	3	4	5	6
1	NIL	NIL	NIL	NIL	1	NIL
2	2	NIL	NIL	2	NIL	NIL
3	NIL	3	NIL	NIL	NIL	3
4	4	NIL	NIL	NIL	4	NIL
5	NL	5	NIL	NIL	NIL	NIL
6	NIL	6	6	NIL	NIL	NIL

$D^1 =$

	1	2	3	4	5	6
1	0	∞	∞	∞	-1	∞
2	1	0	∞	2	0	∞
3	∞	2	0	∞	∞	-8
4	-4	∞	∞	0	-5	∞
5	∞	7	∞	∞	0	∞
6	∞	5	10	∞	∞	0

$\pi^1 =$

	1	2	3	4	5	6
1	N	N	N	N	1	N
2	2	N	N	2	1	N
3	N	3	N	N	N	3
4	4	N	N	N	1	N
5	N	5	N	N	N	N
6	N	6	6	N	N	N

$D^2 =$

	1	2	3	4	5	6
1	0	∞	∞	∞	-1	∞
2	1	0	∞	2	0	∞
3	3	2	0	4	2	-8
4	-4	∞	∞	0	-5	∞
5	8	7	∞	9	0	∞
6	6	5	10	7	5	0

$\pi^2 =$

	1	2	3	4	5	6
1	N	N	N	N	1	N
2	2	N	N	2	1	N
3	2	3	N	2	2	3
4	4	N	N	N	1	N
5	2	5	N	2	N	N
6	2	6	6	2	2	N

$D^3 =$

	1	2	3	4	5	6
1	0	∞	∞	∞	-1	∞
2	-1	0	∞	2	0	∞
3	3	2	0	4	2	-8
4	-4	∞	∞	0	-5	∞
5	8	7	∞	9	0	∞
6	6	5	10	7	5	0

$D^4 =$

	1	2	3	4	5	6
1	0	∞	∞	∞	-1	∞
2	-2	0	∞	2	-3	∞
3	0	2	0	4	-1	-8
4	-4	∞	∞	0	-5	∞
5	5	7	∞	9	0	∞
6	3	5	10	7	2	0

$D^5 =$

	1	2	3	4	5	6
1	0	6	∞	8	-1	∞
2	-2	0	4	2	-3	∞
3	0	2	0	4	-1	-8
4	-4	2	∞	0	-5	∞
5	5	7	∞	9	0	∞
6	3	5	10	7	2	0

$D^6 =$

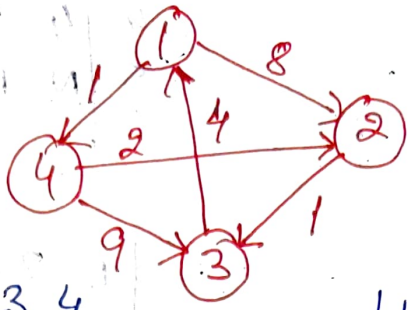
	1	2	3	4	5	6
1	0	6	∞	8	-1	∞
2	-2	0	4	2	-3	∞
3	-5	-3	0	-1	-6	-8
4	-4	2	∞	0	-5	∞
5	5	7	∞	9	0	∞
6	3	5	10	7	2	0

The minimum cost path from Node 3 to Node 1 is -5

and Path is $3 \to 6 \to 2 \to 4 \to 1$

Ans

Q Apply Floyd warshall Algo to find the shortest path between every pair of vertices in the follow Graph G.



D^0 =

	1	2	3	4
1	0	8	∞	1
2	∞	0	1	∞
3	4	∞	0	∞
4	∞	2	9	0

π^0 =

	1	2	3	4
1	NIL	1	NIL	1
2	NIL	NIL	2	NIL
3	3	NIL	NIL	NIL
4	NIL	4	4	NIL

D^1 =

	1	2	3	4
1	0	8	∞	1
2	∞	0	1	∞
3	4	12	0	5
4	∞	2	9	0

π^1 =

	1	2	3	4
1	NIL	1	NIL	1
2	NIL	NIL	2	NIL
3	3	1	NIL	1
4	NIL	4	4	NIL

D^2 =

	1	2	3	4
1	0	8	9	1
2	∞	0	1	∞
3	4	12	0	5
4	∞	2	3	0

π^2 =

	1	2	3	4
1	NIL			
2				
3				
4				

D^3 =

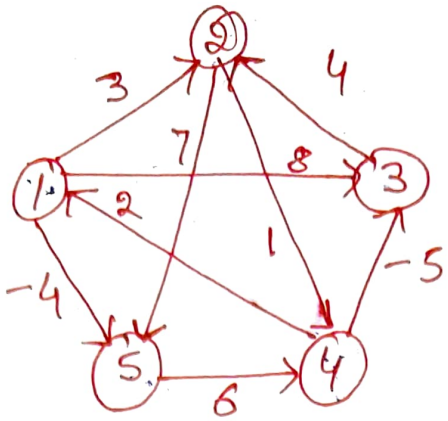
	1	2	3	4
1	0	8	9	1
2	5	0	1	6
3	4	12	0	5
4	7	2	3	0

D^4 =

	1	2	3	4
1	0	3	4	1
2	5	0	1	6
3	4	7	0	5
4	7	2	3	0

Ans

Q Apply Floyd Warshell algo to find shortest path.



Sol^n.

D^0 =

	1	2	3	4	5
1	0	3	8	∞	-4
2	∞	0	∞	1	7
3	∞	4	0	∞	∞
4	2	∞	-5	0	∞
5	∞	∞	∞	6	0

π^0 =

	1	2	3	4	5
1	N	1	1	N	1
2	N	N	N	2	2
3	N	3	N	N	N
4	4	N	4	N	N
5	N	N	N	5	N

D^1 =

	1	2	3	4	5
1	0	3	8	∞	-4
2	∞	0	∞	1	7
3	∞	4	0	∞	∞
4	2	5	-5	0	-2
5	∞	∞	∞	6	0

π^1 =

	1	2	3	4	5
1	N	1	1	N	1
2	N	N	N	2	2
3	N	3	N	N	N
4	4	1	4	NIL	1
5	N	N	N	5	N

Solid → 1, 2 — Look in π^0

D^2 =

	1	2	3	4	5
1	0	3	8	4	-4
2	∞	0	∞	1	7
3	∞	4	0	5	11
4	2	5	-5	0	-2
5	∞	∞	∞	6	0

π^2 =

	1	2	3	4	5
1	N	1	1	2	1
2	N	N	N	2	2
3	N	3	N	2	2
4	4	1	4	N	1
5	N	N	N	5	N

$D^3 =$

	1	2	3	4	5
1	0	3	8	4	-4
2	∞	0	∞	1	7
3	∞	4	0	5	11
4	2	-1	-5	0	-2
5	∞	∞	∞	6	0

$\pi^3 =$

	1	2	3	4	5
1	N	1	1	2	1
2	N	N	N	2	2
3	N	3	N	2	2
4	4	3	4	N	1
5	N	N	N	5	N

$D^4 =$

	1	2	3	4	5
1	0	3	-1	4	-4
2	3	0	-4	1	-1
3	7	4	0	5	3
4	2	-1	-5	0	-2
5	8	5	1	6	0

$\pi^4 =$

	1	2	3	4	5
1	N	1	4	2	1
2	4	N	4	2	1
3	4	3	N	2	1
4	4	3	4	N	1
5	4	3	4	5	N

$D^5 =$

	1	2	3	4	5
1	0	1	-3	2	-4
2	3	0	-4	1	-1
3	7	4	0	5	3
4	2	-1	-5	0	-2
5	8	5	1	6	0

$\pi^5 =$

	1	2	3	4	5
1	N	3	4	5	1
2	4	N	4	2	1
3	4	3	N	2	1
4	4	3	4	N	1
5	4	3	4	5	N

The cost from node 4 to node 5 is -2 and required path is $4 \to 1 \to 5$

Path 4—1—5 Ans

2—5

2—1—5

2—4—1—5

Path: $2 \to 1$

$2 \to 4 \to 1 =$