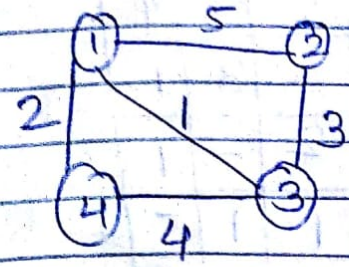
floyd - warshall algorithm

(1)



Iteration = 0 $k = 0$

d_0

	1	2	3	4
1	-	5	1	2
2	5	-	3	∞
3	1	3	-	4
4	2	∞	4	-

distance table

S_0

	1	2	3	4
1	-	2	3	4
2	1	-	3	4
3	1	2	-	4
4	1	2	3	-

sequence table

(1) Put - in cells having same row & column name

(2) Now fill the distance table using graph

(3) fill the cell C_{ij} in distance table D_k using foll condition is

$d_{ij} > d_{ik} + d_{kj}$ [in distance table D_{k-1}]

If yes, then fill the cell C_{ij} in D_k table with the value $d_{ik}+d_{kj}$ of D_{k-1} table

If no, then fill the cell C_{ij} in D_k table with the value d_{ij} of D_{k-1} table

Where i = row no
j = Col no
k = Iteration no.

i.e we

2)

D_1

	1	2	3	4
1	–	5	1	2
2	5	–	3	7
3	1	3	–	3
4	2	7	3	–

S_1

	1	2	3	4
1	–	2	3	4
2	1	–	3	1
3	1	2	–	1
4	1	1	1	–

As we are in Iteration 1 Copy values in first row & first column from previous table

Do & so

Now fill cell c_{23} using foll cond.

$i=2, j=3, k=1$

Is $d(i,j) > d(i,k) + d(k,j)$

$d(2,3) > d(2,1) + d(1,3)$

$3 > 5 + 1$

$3 > 7$

? NO

of C_{23} & C_{32}

so, Copy values from previous tables in D_1 & S_k for

Now cell C_{24}

$i=2, j=4, k=1$

$d(2,4) > d(2,1) + d(1,4)$

$\infty > 5 + 2$

$\infty > 7$?

Yes

& C_{42}

So fill C_{24} with $d(2,1)+d(1,4)$ in D_1 table and store iteration no (k) in S_1 table

at loc^n S_{24} & S_{42}

Now $i=3, j=4, k=1$

$d_{i,j} > d_{i,k} + d_{k,j}$

$d_{3,4} > d_{3,1} + d_{1,4}$

$4 > 1+2$

$4 > 3$

⇓ yes

so

fill C_{34} & C_{43} with updated values

so D_1

	1	2	3	4
1	–	5	1	2
2	5	–	3	7
3	1	3	–	3
4	2	7	3	–

S_1

	1	2	3	4
1	–	2	3	4
2	1	–	3	1
3	1	2	–	1
4	1	1	1	–

Now Iteration 2 fill col 2 & row 2 using previous tables

	1	2	3	4
1	–	5	1	2
2	5	–	3	7
3	1	3	–	3
4	2	7	3	–

D_2

	1	2	3	4
1	–	2	3	4
2	1	–	3	1
3	1	2	–	1
4	1	1	1	–

S_2

$i=1, j=3, k=2$

$d(1,3) > d(1,2) + d(2,3)$

$1 > 5+3$

$1 > 8$

2)

Now, Iteration 3, k=3

	1	2	3	4
1	–	4	1	2
2	4	–	3	6
3	1	3	–	3
4	2	6	3	–

D3

	1	2	3	4
1	–	3	3	4
2	3	–	3	3
3	1	2	–	1
4	1	3	1	–

S3

Now, Iteration 4, k=4

	1	2	3	4
1	–	4	1	2
2	4	–	3	6
3	1	3	–	3
4	2	6	3	–

D4

	1	2	3	4
1	–	3	3	4
2	3	–	3	3
3	1	2	–	1
4	1	3	1	–

S4

Now use these tables to find shortest distance

Shortest dist between 1 & 2 = d_{12}

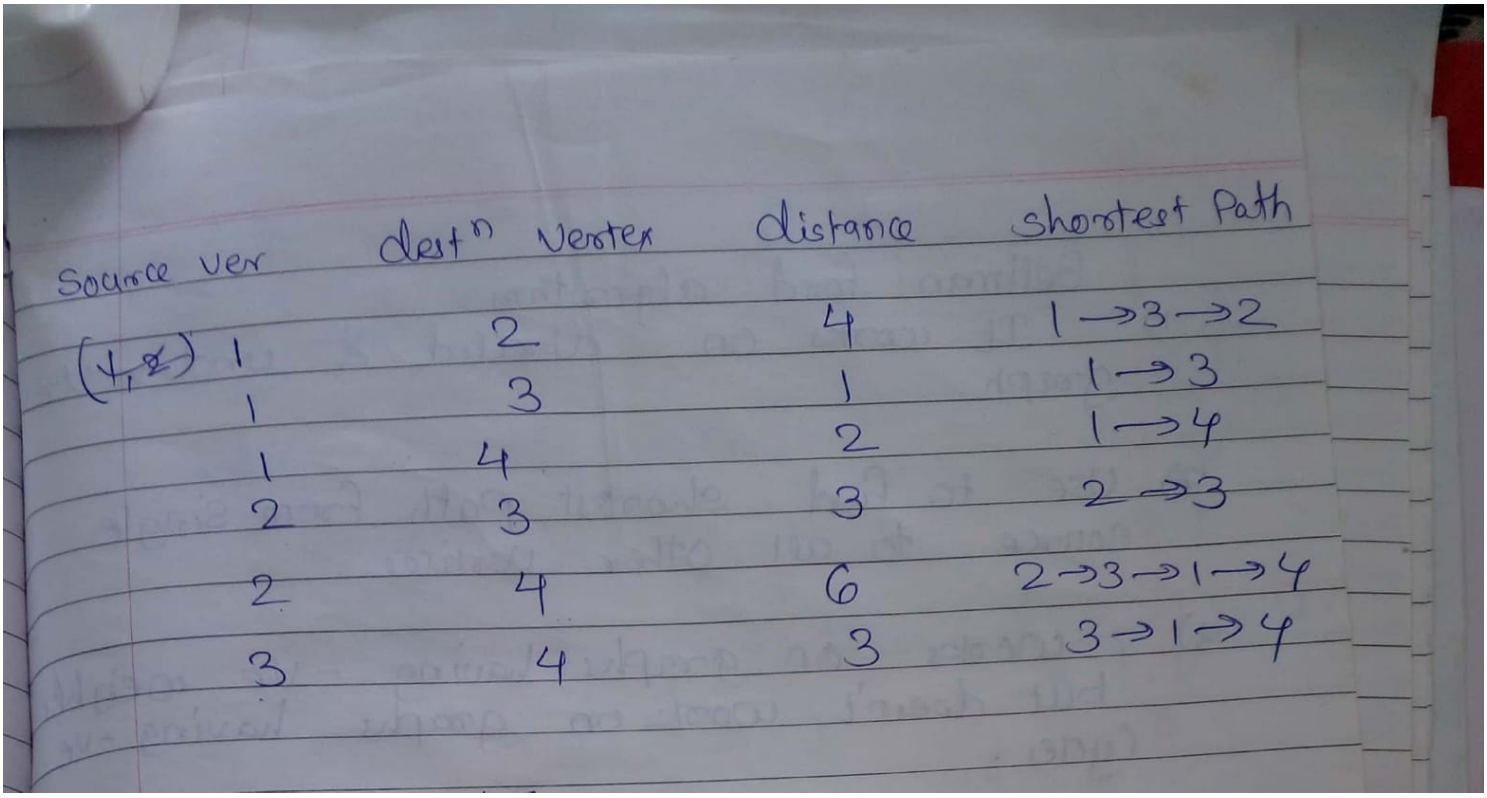
= 4

1 → 2

Now, Check the value of S_{12} in sequence table It is 3 means vertex 3 needs to be visited On the path

1 → 3 → 2

Now, Check S_{32} in seq table It is 2 It means vertex 2 is directly Connected to three



Source ver	dest^n Vertex	distance	shortest Path
~~(1, 2)~~ 1	2	4	1→3→2
1	3	1	1→3
1	4	2	1→4
2	3	3	2→3
2	4	6	2→3→1→4
3	4	3	3→1→4