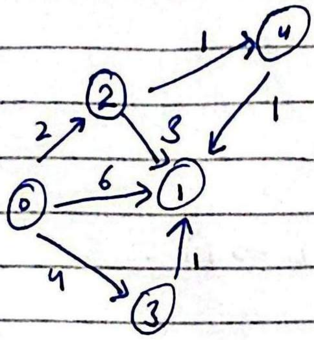
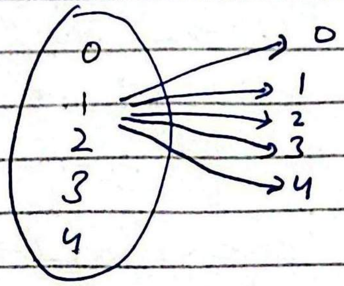
Floyd Warshall Algorithm :



5 nodes



→ It is a multisource shortest path algorithm we are going to store the distance of every node to other node.

→ It detect negative cycle as well.

So shortest path from 0 → 1 is

eg. 0 → 2 → 4 → 1 = 4

Note: go via every vertex / node

eg:

distance from 0->1

dist [0][1] -> (0->1)

i can go from

(0->2) + (2->1) = 5

2 + 3

Also

(0->3) + (3->1) = 5

4 + 1

Also

~~(0->2) + (0->4)~~

(0->4) + (4->1) = 4

3 + 1

it assumes that (0->2) is calculated.

So apparently (0->4) -> (4->1) = 4

which is shortest of them all.

So we can say

dist [i][j] -> (0->1)

dist [0][1]

and

(0->2) + (2->1) = 5

(0->3) + (3->1) = 5

(0->4) + (4->1) = 4

i k j

So the path is

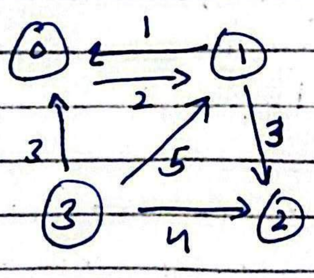
$$\min(dist[i][k] + dist[j][k])$$

So min of them all is ans.

But in this case we are assuming that $0 \to 2$ is calculated something which is computed like from $0 \to 4 \Rightarrow (0 \to 2), (2 \to 4)$

---

till now in all problems we were using adj list but now we are going to use adj Matrix

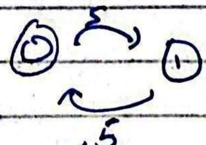


adj Matrix

$$\begin{array}{c|cccc} & 0 & 1 & 2 & 3 \\ \hline 0 & 0 & 2 & \infty & \infty \\ 1 & 1 & 0 & 3 & \infty \\ 2 & \infty & \infty & 0 & \infty \\ 3 & 3 & 5 & 4 & 0 \end{array}$$

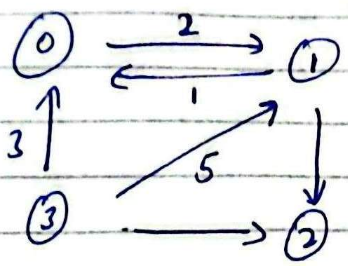
initial cost matrix

if that was undirected just do

$0 \overset{5}{—} 1 \Rightarrow$ 

If we want to apply floyd warshal to an undirected graph we just make dual direction of nodes.

lets moves from vesten 0.



$$\begin{array}{c|cccc} & 0 & 1 & 2 & 3 \\ 0 & 0 & 2 & \infty & \infty \\ 1 & 1 & 0 & 3 & \infty \\ 2 & \infty & \infty & 0 & \infty \\ 3 & 3 & 5 & 4 & 0 \end{array}$$

initial cost matrix

moving via 0 $\rightarrow$

$$\begin{array}{c|cccc} & 0 & 1 & 2 & 3 \\ 0 & \boxed{0} & 2 & \infty & \infty \\ 1 & 1 & 0 & 3 & \infty \\ 2 & \infty & \infty & 0 & \infty \\ 3 & 3 & 5 & 4 & 0 \\ 3. & & & & \end{array}$$

it is $\rightarrow$ same as initial coz if dst is 0 and src as well.

0 -> 1

means [0][0] + [0][1]

0 -> 2

[0][0] + [0][2]

⋮

1 -> 0

[1][0] + [0][0]

1 -> 2

[1][0] + [0][2]

1 -> 3

[1][0] + [0][3]

1 + ∞

$2 \to 1$

$[2][1] + [2][0]$ ∞

$2 \to 2$ $[2][1]$ ∞

$2 \to 3$ ∞

$3 \to 1$

$[3][0] + [0][1]$

$3 + 2 = 5$

$\overset{i}{3} \to \overset{j}{2}$

$[3][0] + [0][2]$

$0 + \infty = \infty$

Now go via 1

$$\begin{array}{c|cccc} & 0 & 1 & 2 & 3 \\ \hline 0 & 0 & 2 & 5 & \infty \\ 1 & 1 & 0 & 3 & \infty \\ 2 & \infty & \infty & 0 & \infty \\ 3 & 6 & 5 & 8 & 0 \end{array}$$

initial cost

$0 \to 2$

$[0][1] + [1][2]$

$2 + 3 = 5$

$0 \to 3$

$[0][1] + [1][3]$

$2 + \infty = \infty$

$2 \to 0$

$[2][1] + [1][0]$ ∞

$2 \to 3$

$[2][1] + [1][3]$

$3 \to 0$

$[3][1] + [1][0]$

$5 + 1 = 6$

$3 \to 1$

$[3][1] + [1][1]$

$5 + 0 = 5$

$3 \to 2$

$[3][1] + [1][2]$

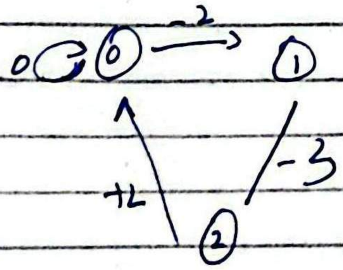
$5 + 3$

we are going to do it for every node.

```
for (via(k) = 0 → via→last node)
    for (i = 0 → last node)
        for (j = 0 → last node)
```

$$cost[i][j] = \min(cost[i][j],\ cost[i][\underset{k}{via}] + cost[\underset{k}{via}][j])$$

How it detects a negative cycle?



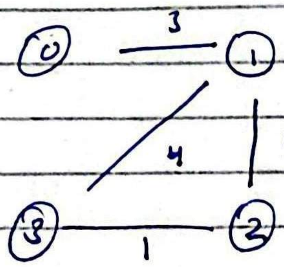
if costing of any node

$dist[i][i] < 0$ then

negative cycle exists.

City with Smallest number of Neighbors at a threshold Distance.

~~edges = [[~~
Threshold = 4



cities → cities

$\boxed{0 \rightarrow 1, 2}$

1 → 0, 2, 3

2 → 0, 1, 3

$\boxed{3 \rightarrow 1, 2}$

2 cities have lowest no. of cities.
as there are multiple so we are going to choose larger which is 3

$$\begin{array}{c|c} & \begin{matrix} 0 & 1 & 2 & 3 \end{matrix} \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} 0 & 3 & 4 & 5 \\ 3 & 0 & 1 & 2 \\ 4 & 1 & 0 & 1 \\ 5 & 2 & 1 & 0 \end{bmatrix} \end{array}$$

count = 0