

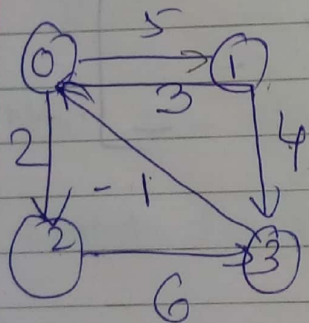
Bellman Ford algorithm
1) It works on directed & undirected graph.

2) Use to find shortest path from single source to all other vertices

3) works on graphs having -ve weights but doesn't work on graphs having -ve cycle.

4) If there are n vertices in a graph then $n-1$ iterations are required to find shortest distance.

5) n th iteration is performed to check whether there is any -ve cycle.



$u \rightarrow v$
 $v = ev$
 $u \rightarrow v$ directed edge
 $w(u, v)$

write down edges & their weights

$0 \rightarrow 1$ 5
 $0 \rightarrow 2$ 2
 $1 \rightarrow 0$ 3
 $1 \rightarrow 3$ 4
 $2 \rightarrow 3$ 6
 $3 \rightarrow 0$ -1

0 1
Relax (u, v)

$v.d > u.d + w(u, v)$
 $1.d > 0.d + w(0, 1)$
 $5 > 0 + 5$

	0	1	2	3		0	1	2	3
d	0	∞	∞	∞	p	-	-	-	-

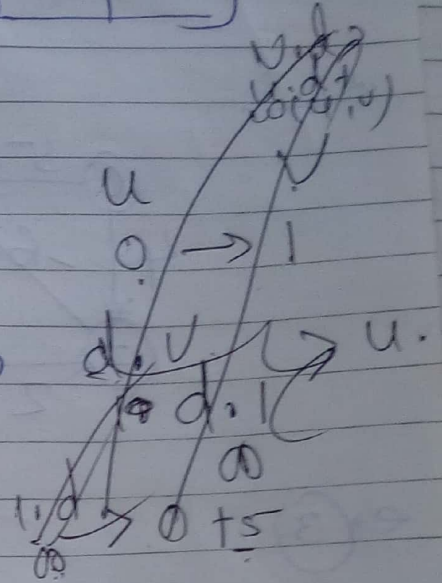
now relax all edges

relax(u, v)

if $v.d > u.d + w(u, v)$ then

$$v.d = u.d + w(u, v)$$

$$v.p = u$$



Where $v.d$ = distance from source vertex 0 to v

$u.d$ = distance from source vertex 0 to u

$w(u, v)$ = weight of the edge $u \rightarrow v$

$v.p$ = predecessor of vertex v

After Iteration 1

d	0	1	2	3	p	0	1	2	3
	0	5	2	9		-	0	0	1

After Iteration 2

d	0	1	2	3	p	0	1	2	3
	0	2	5	8		-	0	0	2

After Iteration 3

d	0	1	2	3	p	0	1	2	3
	0	5	2	8		-	0	0	2

Iteration 4. (to check whether there is -ve cycle)

ex 2 solve

