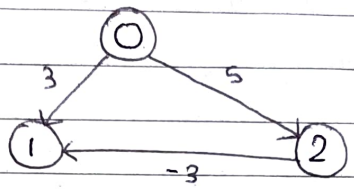
Date _____ Page _____

Day - 210

Graph - 14

* **Bellman Fand Algorithm:**



⇉ This algo is used for finding single source shortest path.

⇉ But for this, we also have Dijkstra, but it will not work for -ve weights.

⇉ So, here come our today's algo.

⇉ Now, if we use dijkstra then we did'n get the correct answer that is —

$0 \to 1 = 0 \xrightarrow{5} 2 \xrightarrow{-3} 1 = 2$

⇉ But it will give 3.

⇉ It is due to the explore array that we used.

⇉ But this logic will also failed.

⇉ Also, if the graph contain -ve cycle then it's impossible to find out the shortest path.

Negative Cycle: Sum all the weights of graph if the weight will be -ve then the graph contain -ve cycle.

Date _____ Page _____

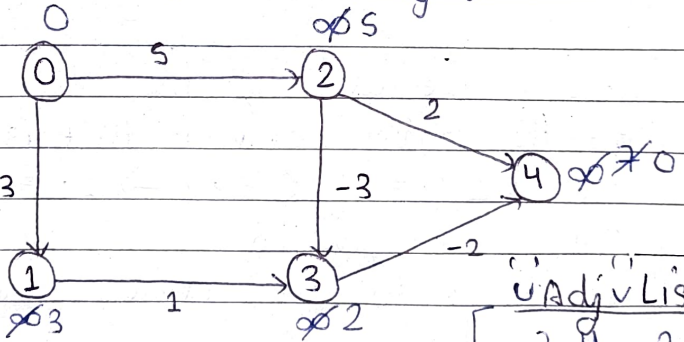
⇒ So, Bellman-Ford helps us in this type questions.

⇒ It works for directed graph and if the graph contain -ve cycle then it will tell us about ~~that~~ it.

Algo:

1. (Relax your all edges) * (v-1) times
   ↳ dist[u] + wt < dist[v]
   dist[v] = dist[u] + wt

2. Relax your edges one more time —
   ↳ to detect -ve cycle.



u	Adj v List	w
	2, 4	, 2
	3, 4	, -2
	2, 3	, -3
	0, 2	, 5
	1, 3	, 1
	0, 1	, 3

⇒ We have to follow adj. list order.

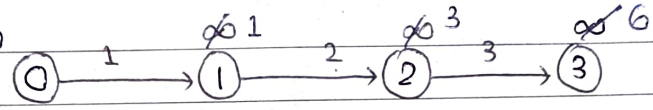
⇒ dist[2] + 2 < dist[4] ✗
dist[3] + (-2) < dist[4] ✗
⋮ ⋮ ⋮ ⋮

⇒ After doing this process (5-1) = 4 times.

Date _____ Page _____

⇒ we will do again one more time & if any change in the ~~weig~~ distance then the cycle present.

⇒ Now, we will understand, why this algo works —



	0	1	2	3
1st →	0	∞1	∞	∞
2nd →	0	1	∞3	∞
3rd →	0	1	3	∞6

2, 3, 3
1, 2, 2
0, 1, 1

⇒ So, as we can see, we are doing this process (v-1) times because in the (v-1)th iteration, it will propagate to the last vertex.

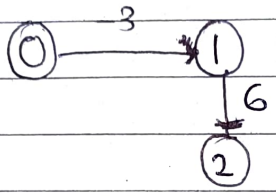
⇒ So, Dij. works with vertex & B.F works with edges.

⇒ Also, if you have relaxed all your edges & if there is no change in dist, then this is our answer.

Date _____ Page _____

⇉ T.C → O(V.*E) = O(VE)

⇉ If the graph is undirected & contain -ve weight then it is impossible to find out the shortest path.



⇉ So, if we find the shortest path b/w 0 & 1 then it will be -3 but if travel back to 0 then come back to 1 again then it will be -9.

⇉ So, we can't find the shortest distance.