Let's begin at 9:05 PM

L91
Bellman Ford & Floyd Warshall Algorithms

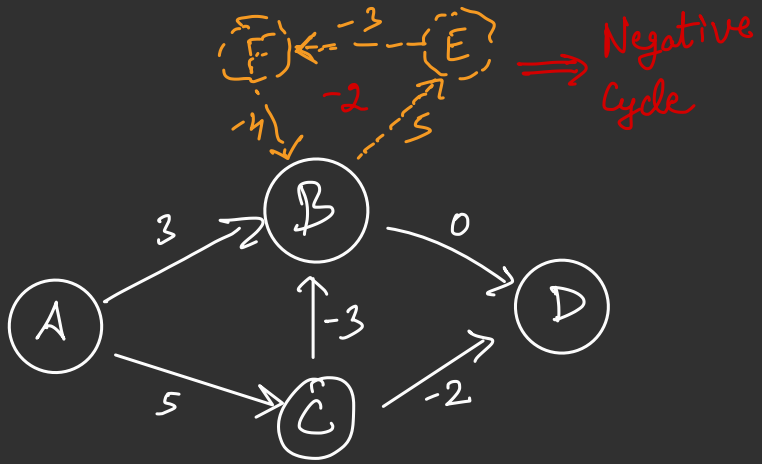
Join Discord - https://bit.ly/ly-discord

RECAP

Let’s begin with Bellman Ford Algorithm

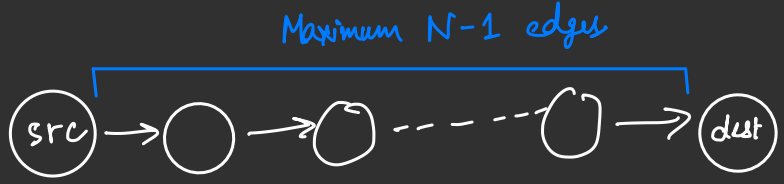
Why study Bellman Ford Algorithm?

Single source shortest path with negative weight edges



A → D ⇒ 2

given that there are NO negative cycles.



Max no. of nodes in the shortest path of src & dest ? ⟹ N

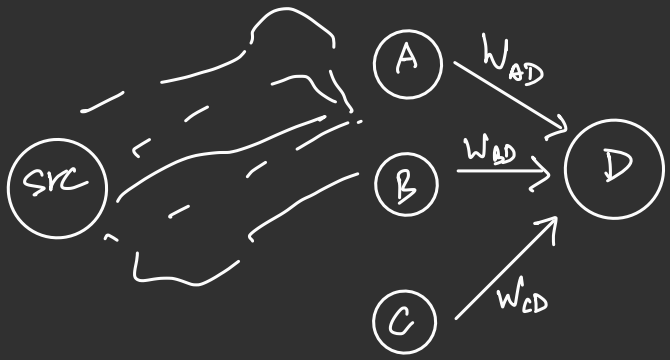
Introduction to the Algorithm

After K^{th} phase, $d[i]$ will represent shortest distance from src to i s.t. there are at-most K edges in the path.

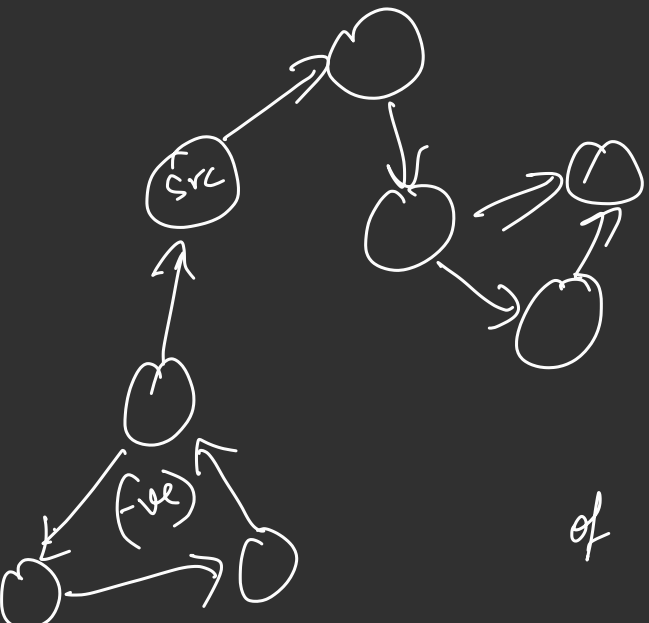
```
d[src] = 0 , d[---] = inf

for(k = 1 ; k <= N-1 ; ++k) {

    for(Edge e : edges) { // u -> v (w)

        d[v] = min( d[v] , d[u] + w);

    }

}
```

Time $\Rightarrow O(N * M)$

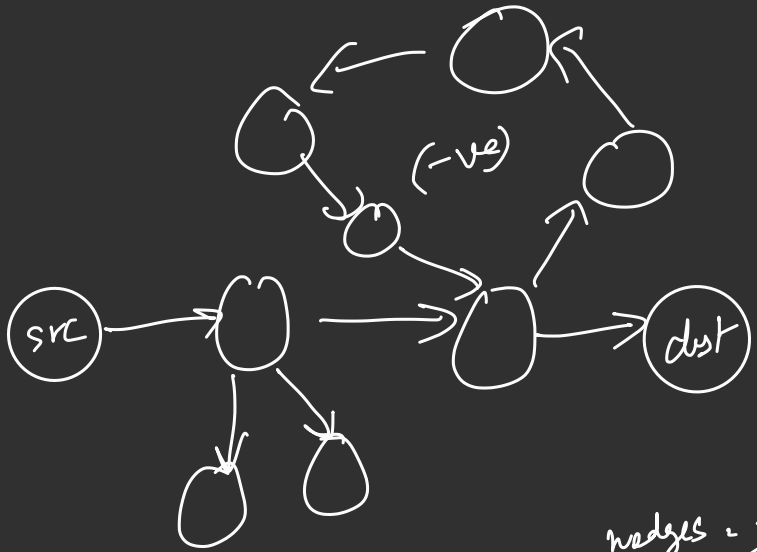
K^{th}



$$\min\left(d_{K-1}[A] + W_{AD},\ d_{K-1}[B] + W_{BD},\ d_{K-1}[C] + W_{CD}\right)$$



Bellman Ford can discover only those neg. cycles which impact the shortest distance of some node from the source.



N = 10

2N iterations

nodes = 3
nodes = 8
nodes = 13

Practice Problem
(Best Path)

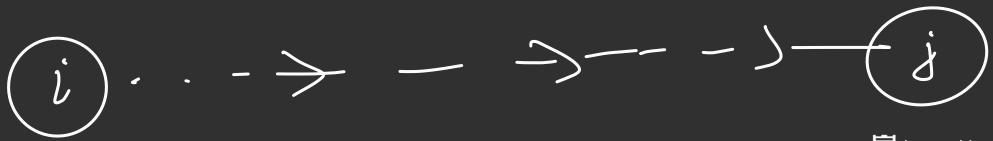
Floyd Warshall Algorithm

Why study Floyd Warshall Algorithm?

- Helps in building d[][], s.t. d[i][j] represents the shortest distance from i to j.
- Capable of dealing with neg. weights
- Easy to implement

Introduction to the Algorithm

$d_k[i][j]$ will represent s.p. from i to j s.t. only the nodes $\{1, 2, 3 \ldots k\}$ are allowed to be intermediary nodes.



$d[i][i] = 0$, $d[i][j] = inf$

```
for (e : edges)
    d[e.u][e.v] = min(d[e.u][e.v], e.w);
```

$$d_k[i][j] = \min \begin{cases} d_{k-1}[i][j] \\ d_{k-1}[i][k] + d_{k-1}[k][j] \end{cases}$$

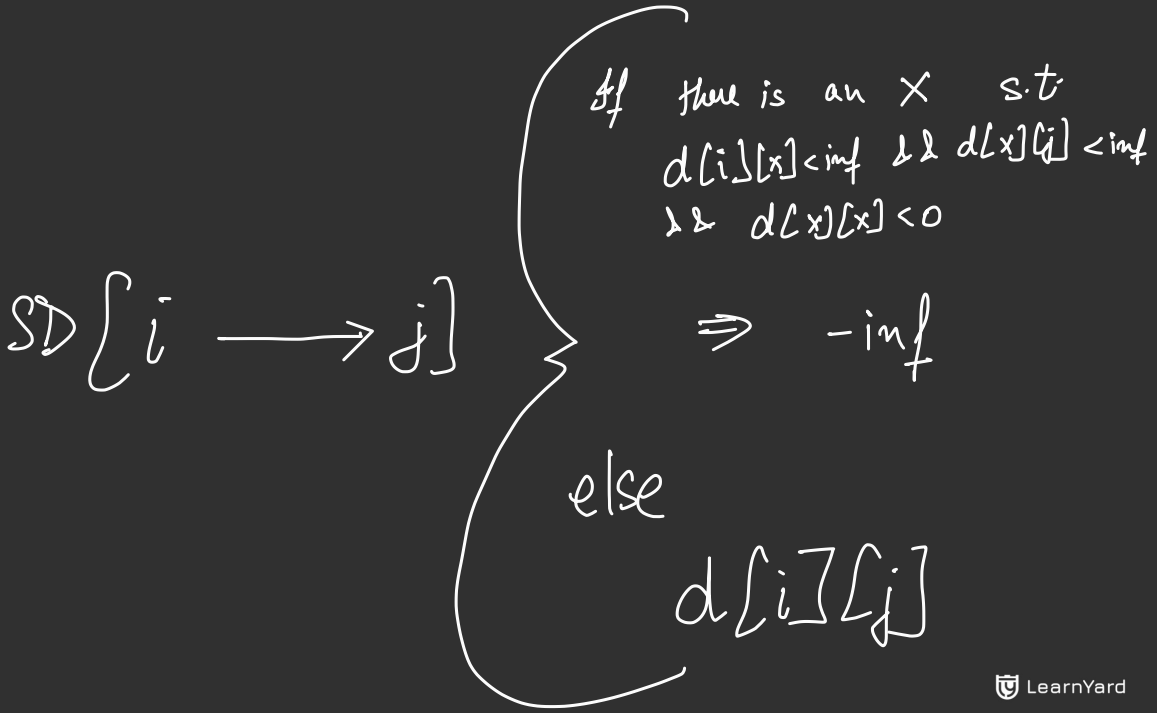
Practice Problem
(Leetcode 1334)

Check if node i is a part of neg. cycle?

Finally.

$d[i][i] < 0 \Rightarrow$ Yes

↳ otherwise, NO.



Thank You!

Reminder: Going to the gym & observing the trainer work out can help you know the right technique, but you'll muscle up only if you lift some weights yourself.

So, PRACTICE, PRACTICE, PRACTICE!