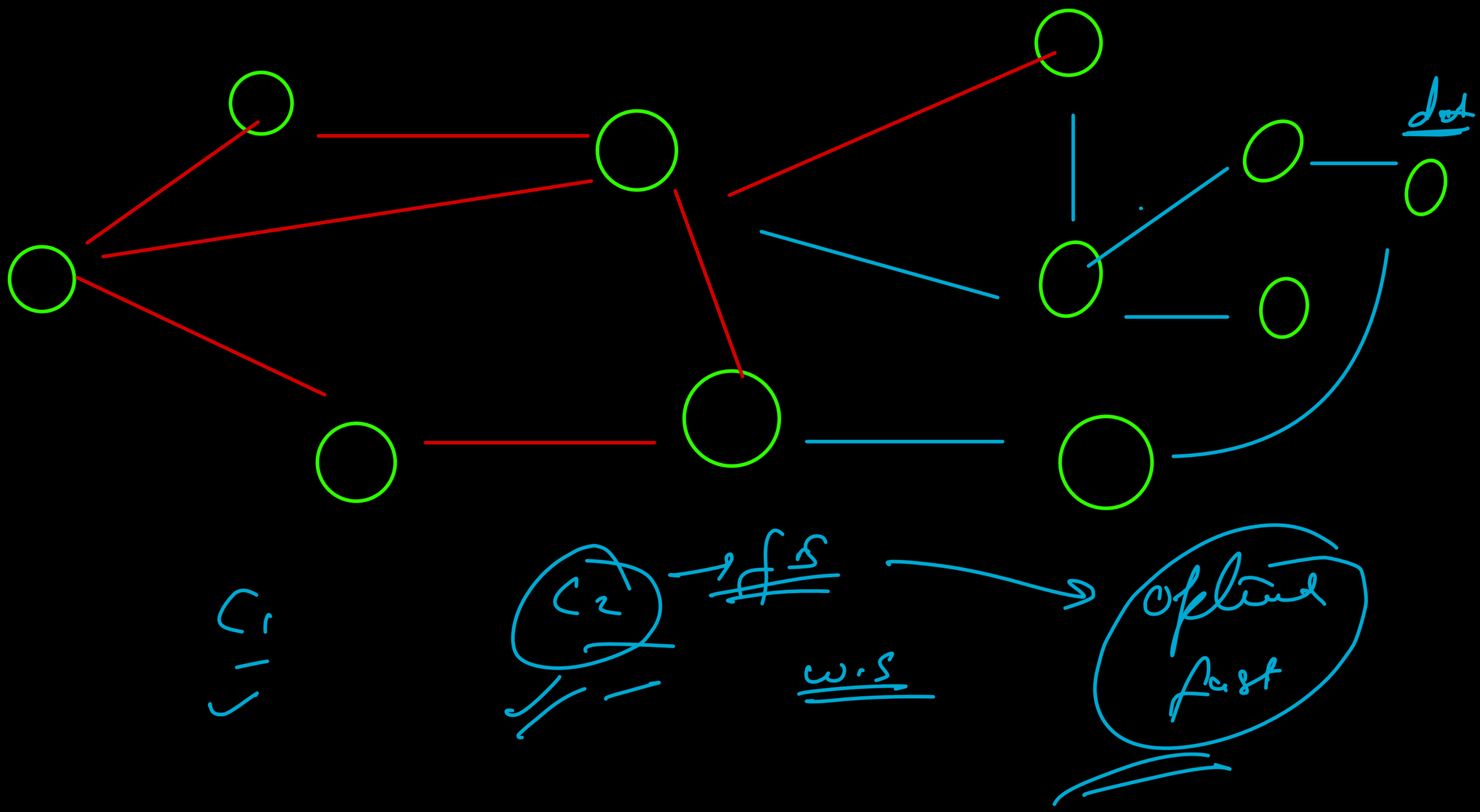
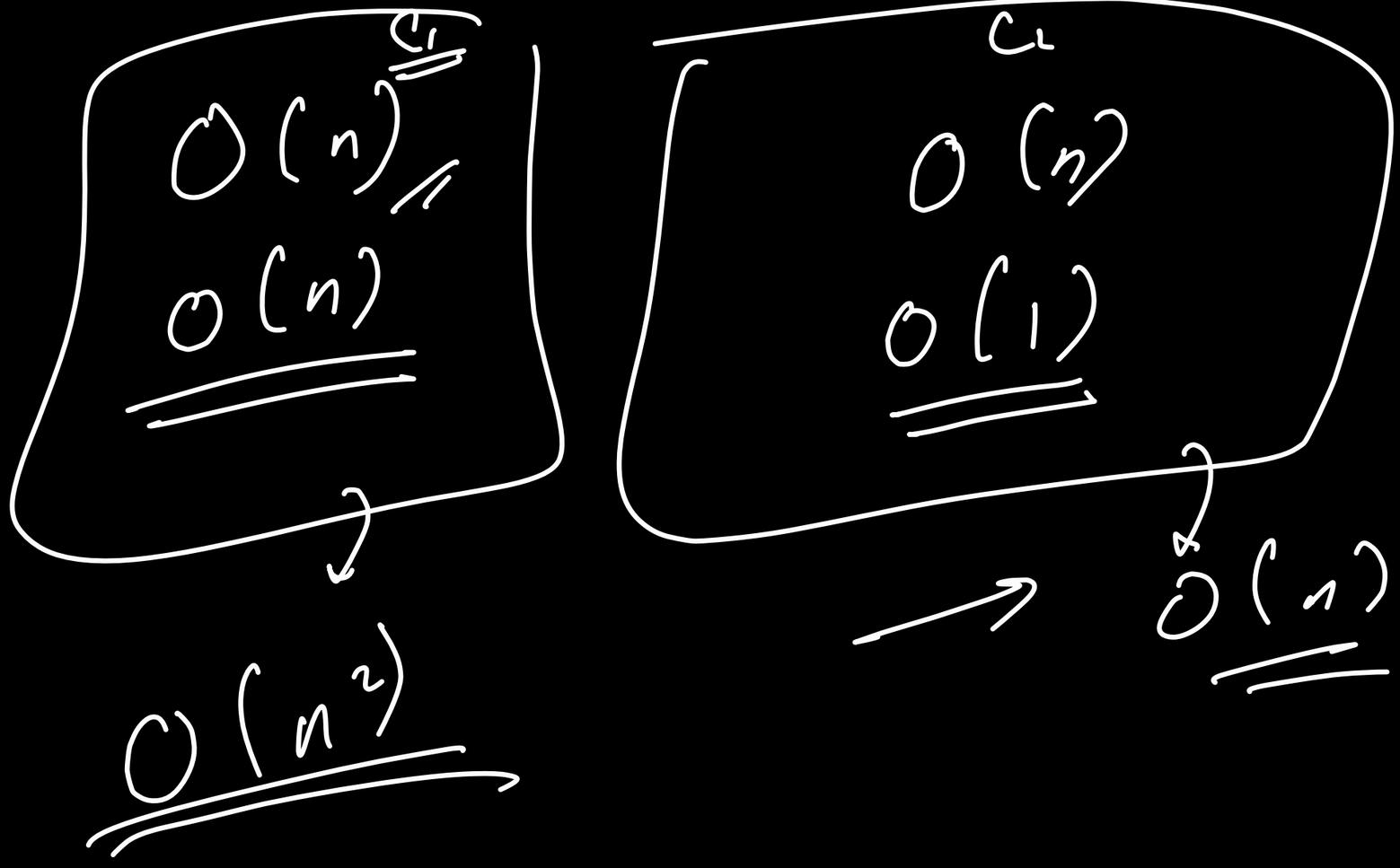
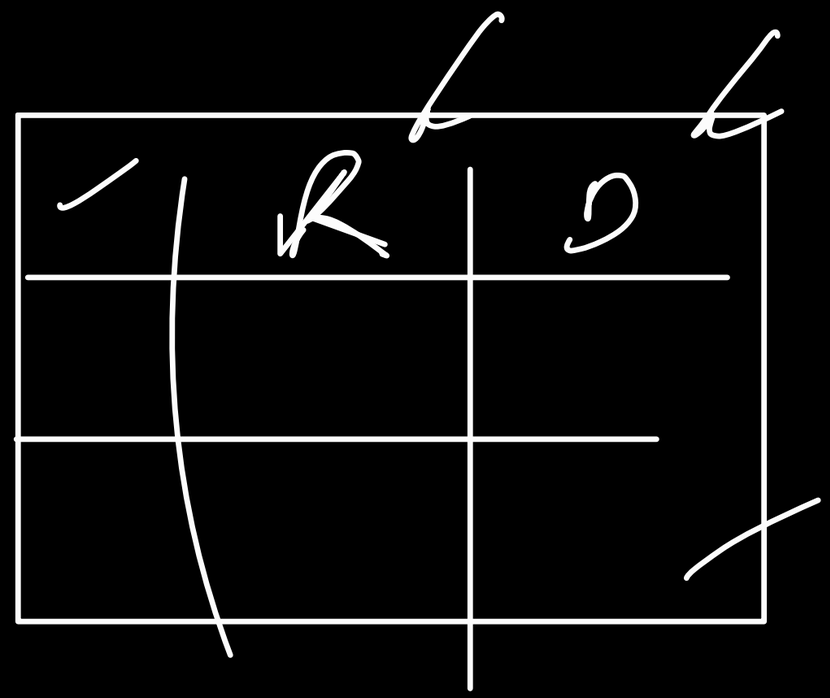






Prim's Algorithm → MST

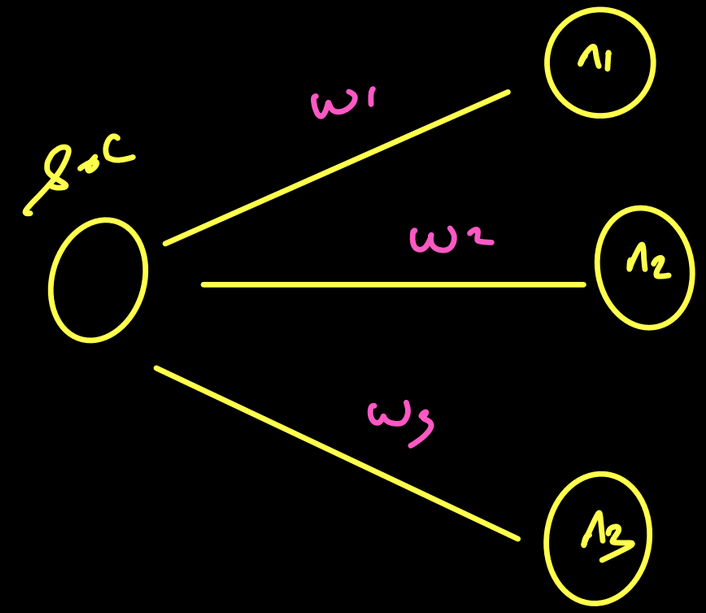
↳ greedy

↳ bfs + greedy

↳ Priority Queue

↳ source node

src



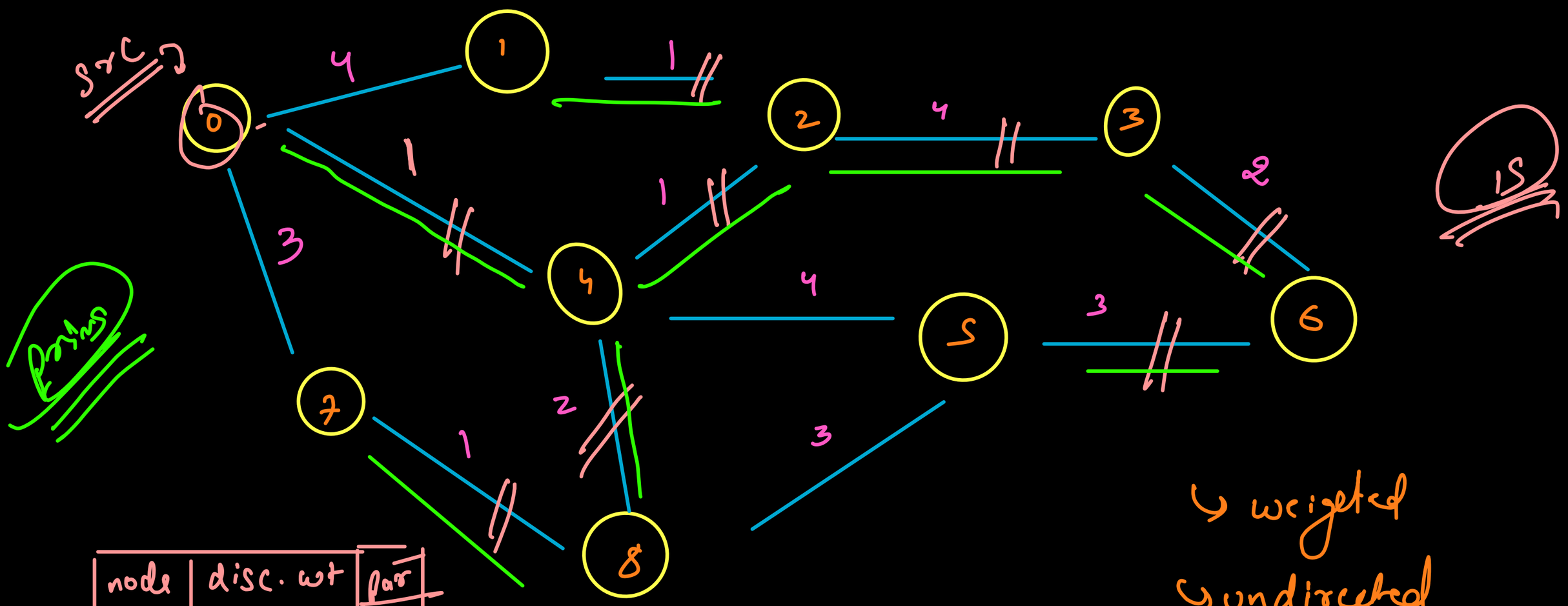
greedy

| n1 | n2 | n3 |

vis → n1, n2, n3

priority queue → min heap

queue → smallest wt edge

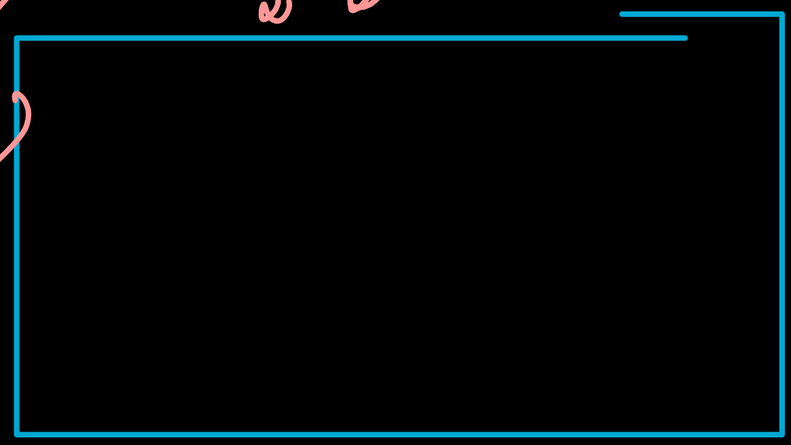


node	disc. wt	par
0	-1	^
1	1	2
2	1	4
3	4	2
4	1	0
5	3	6
6	2	3
7	1	5
8	2	4

(0,0) (1,4) (1,2) (2,8)
(1,1) (1,7) (4,3) (2,6)
(3,5)

pair < wt, vertex >

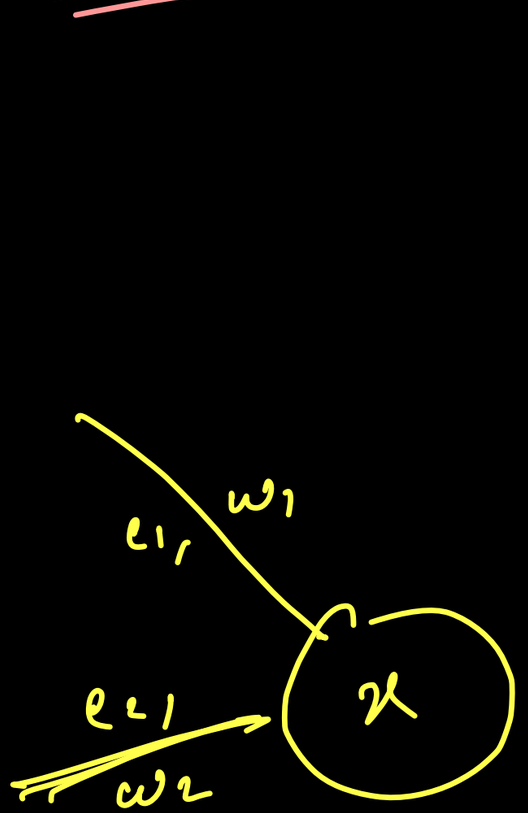
↳ weighted
↳ undirected



0,4,2,8,1,7	6,3,5

↳ visited

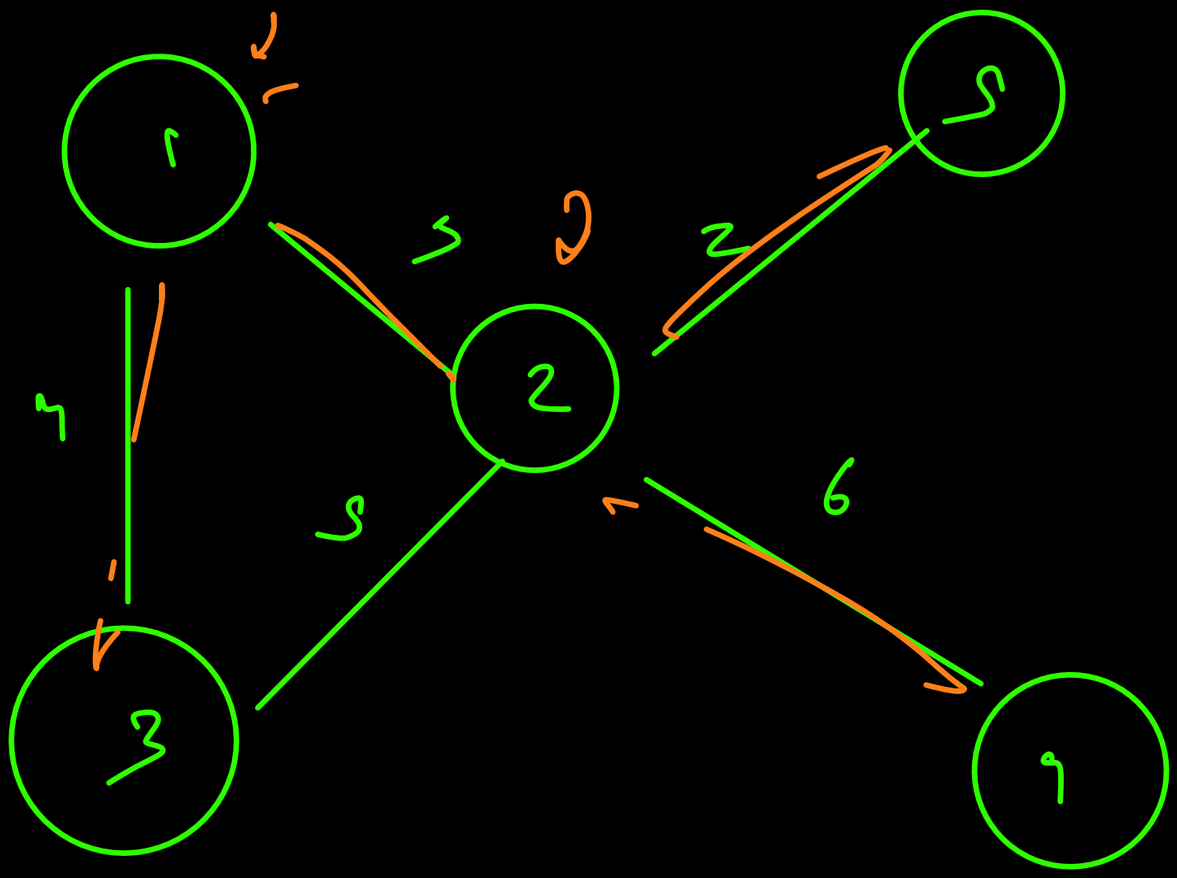
In prims we start with a src & try to discover other nodes.



$w_2 < w_1$

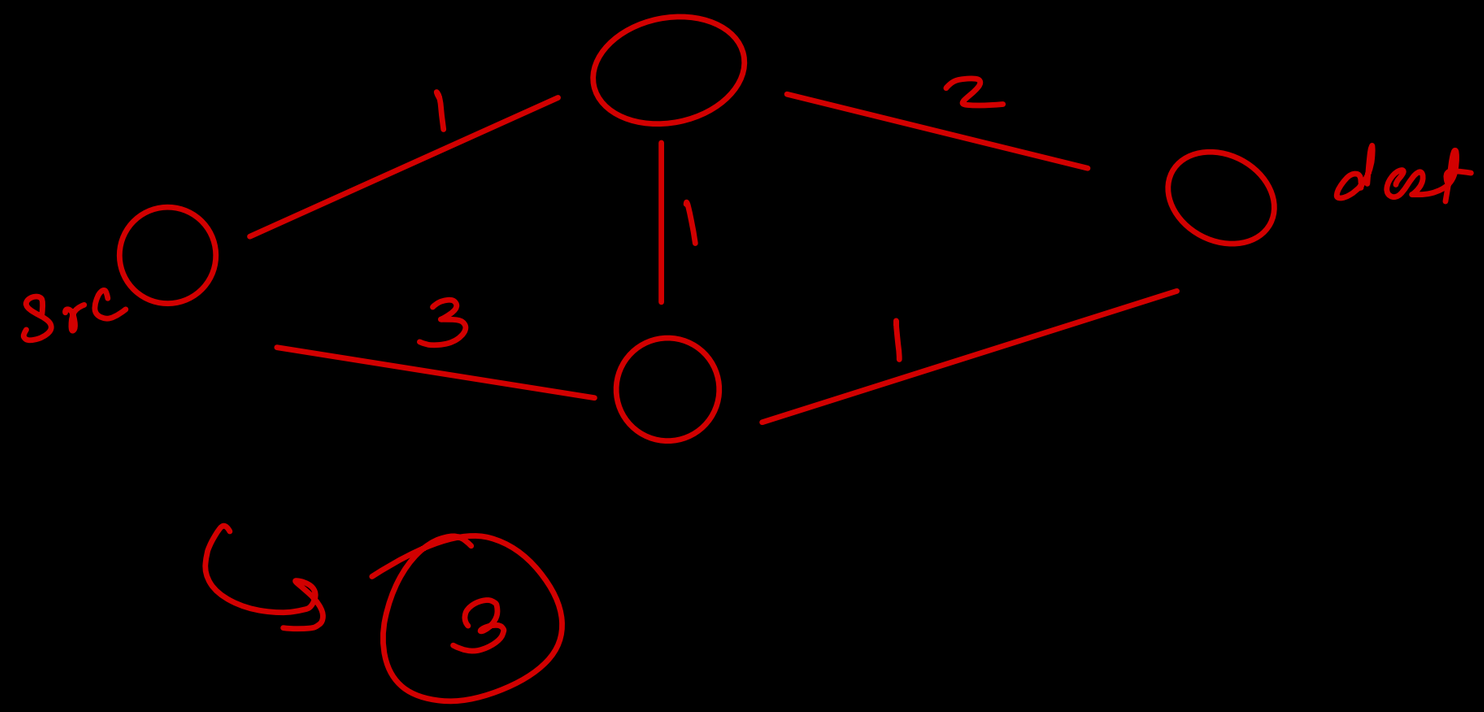
↳ we should update the dis with w_2.

at the start of the algo, dist. est of any node
except src is ∞, because we not yet discovered
any edg.



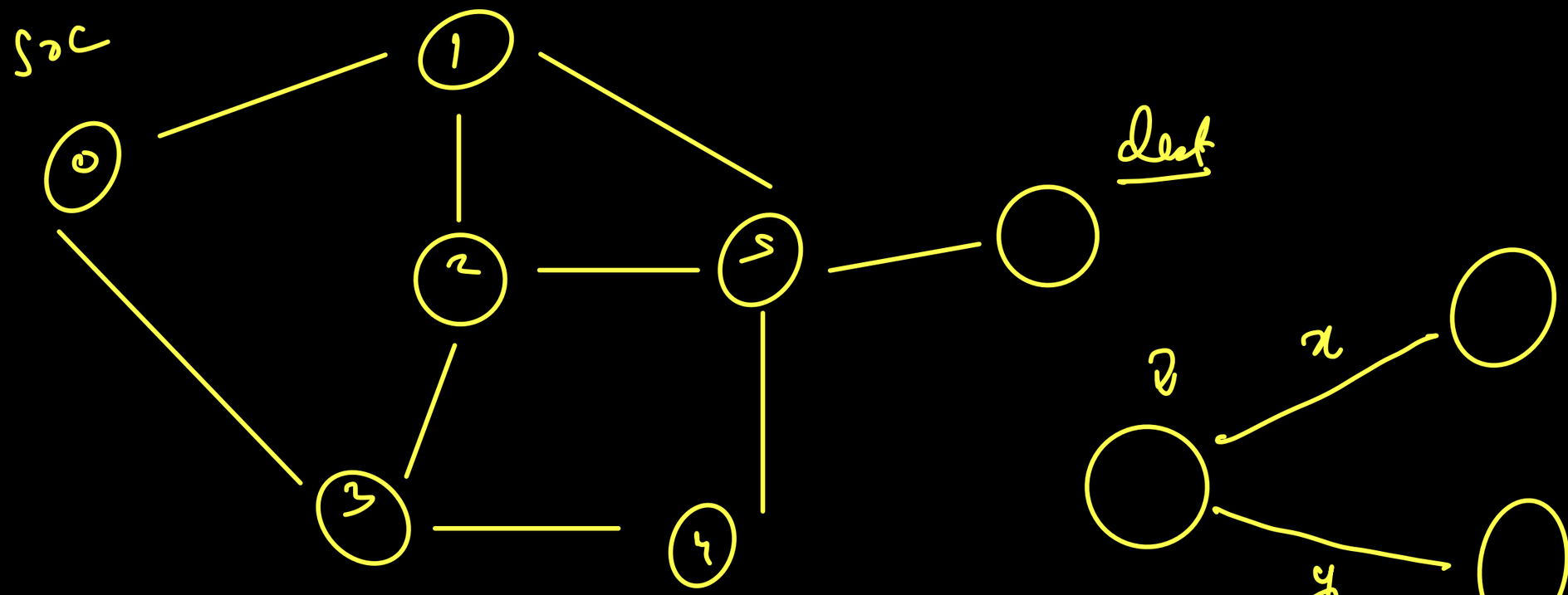
(3,2)
(2,5)
(9,3)
(6,4)

Q: Given an undirected weighted graph, along with a src and dest. find the shortest path btw the src & dest- in terms of sum of edge wts.



We already how to solve shortest path problem for unweighted graph.

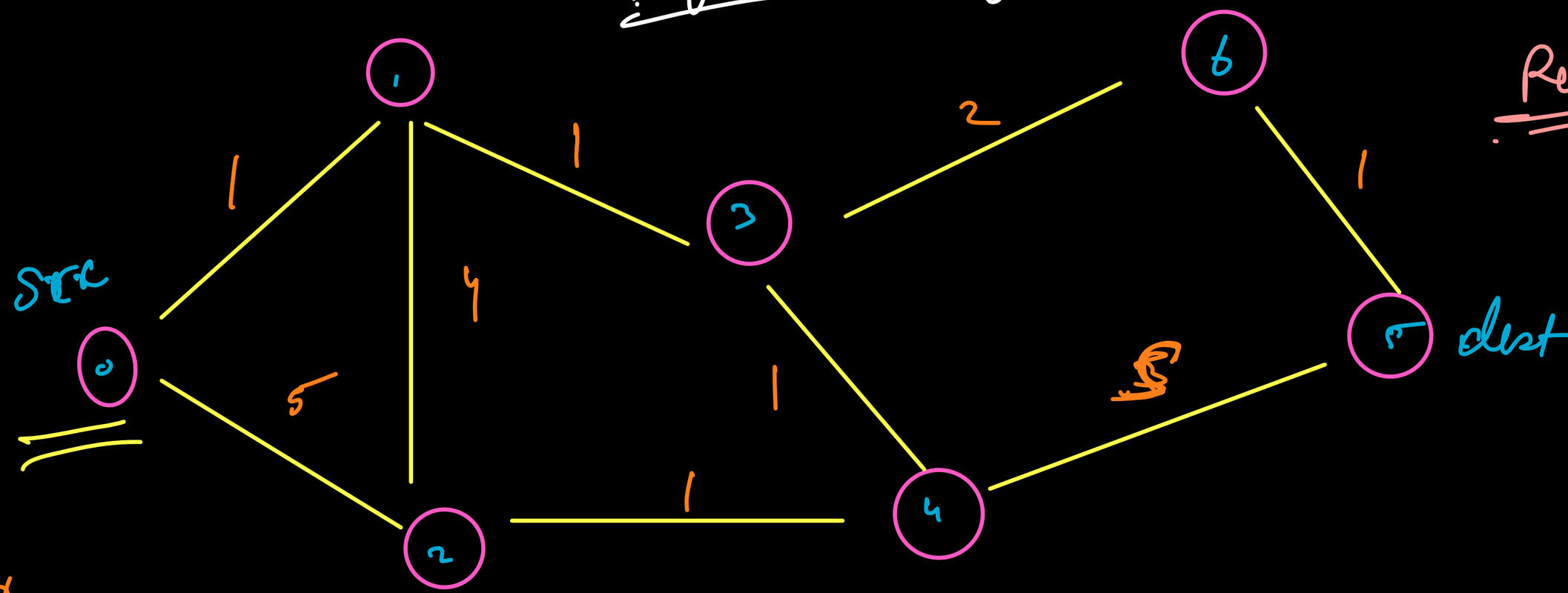
BFS → queue ⇒ FIFO



Prims

queue → priority queue

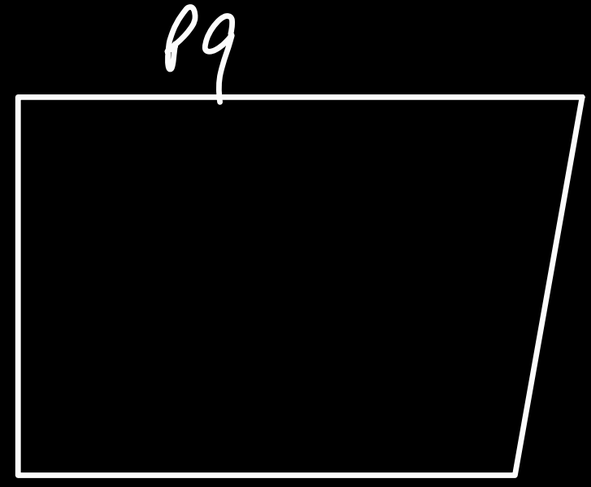
Dijkstra Algo



Relaxation

pair <dist, node>

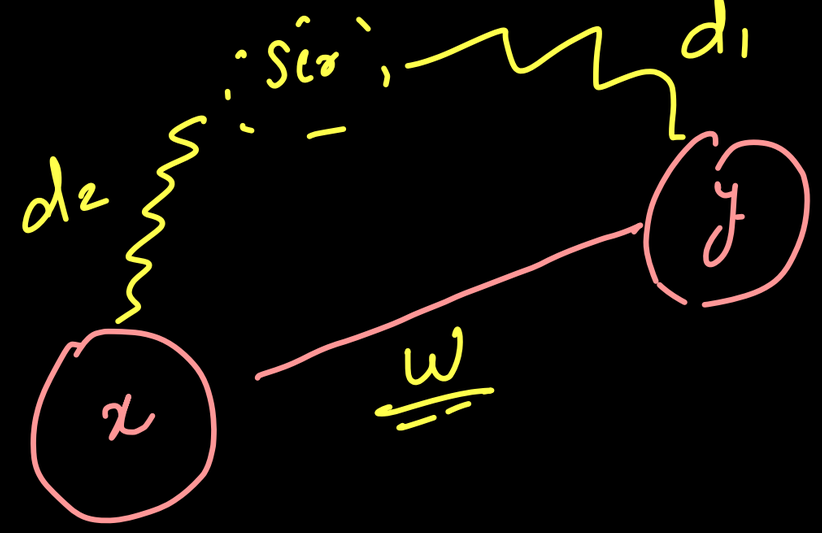
node	dist	via
0	0	x x
1	1	0
2	~~5~~ 4	~~0~~ 4
3	2	1
4	3	3
5	~~6~~ 5	~~4~~ 6
6	4	3

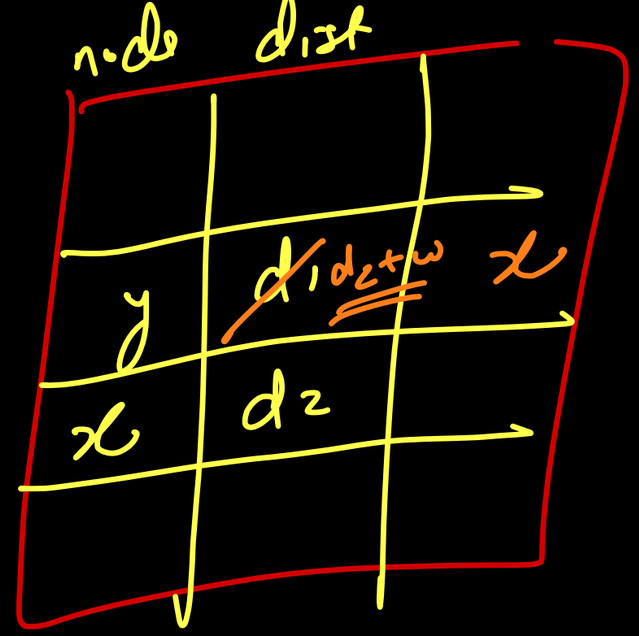


vis

what is the shortest dist from src to any node.

Given





$d_1 \rightarrow$ till now what is the shortest path from src to y.

$d_2 \rightarrow$ till now what is the shortest path from src to x

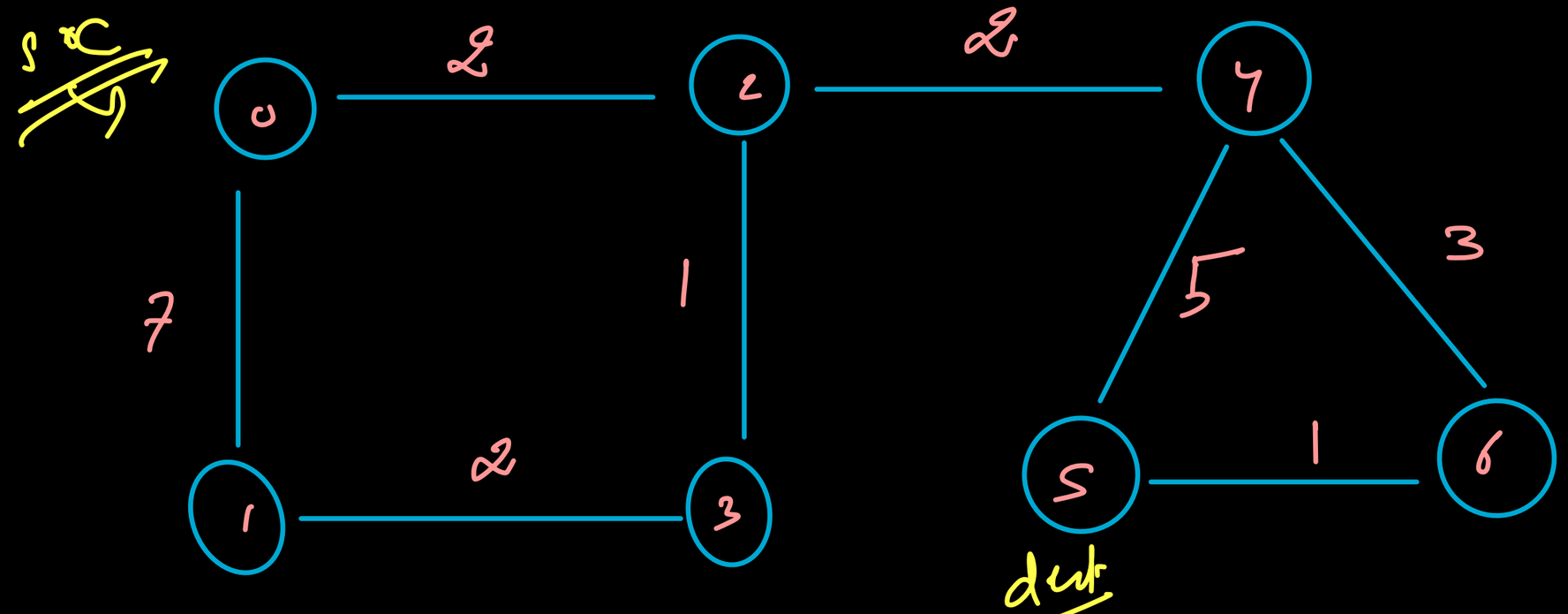
$w \Rightarrow$ edge wt of x and z

if ($d_2 + w < d_1$) {

$mp[y] = d_2 + w$

$via[y] = x$

}



<dist, node>

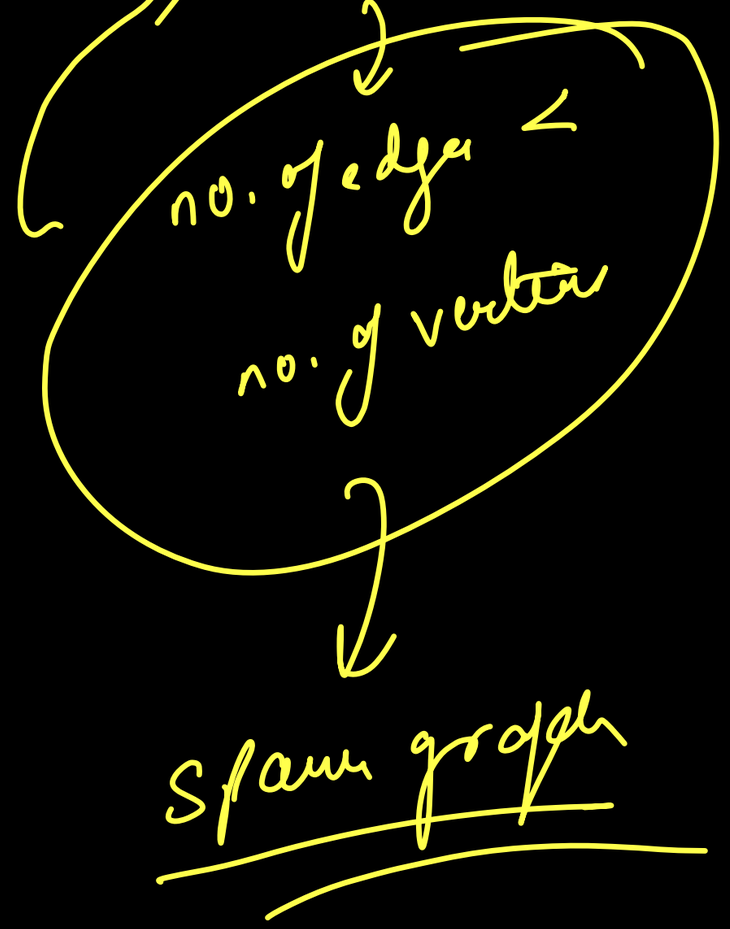
node	dist	via
0	0	-x
1	5	3
2	2	0
3	3	2
4	4	2
5	8	6
6	9	4

pq

vis

0 | 2 | 3 | 4 | 1 | 6 | 5

Kruskals

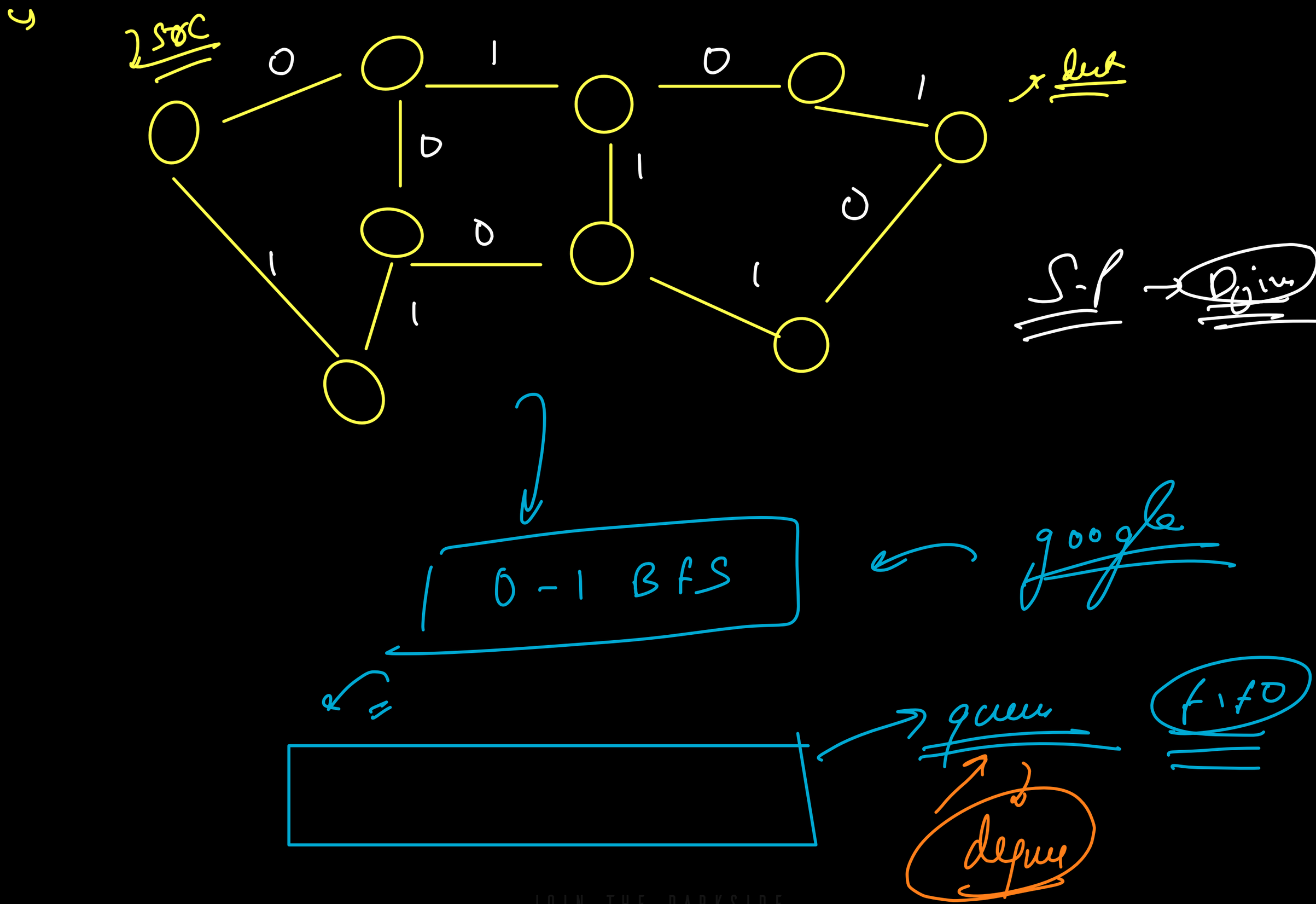


sparse graph

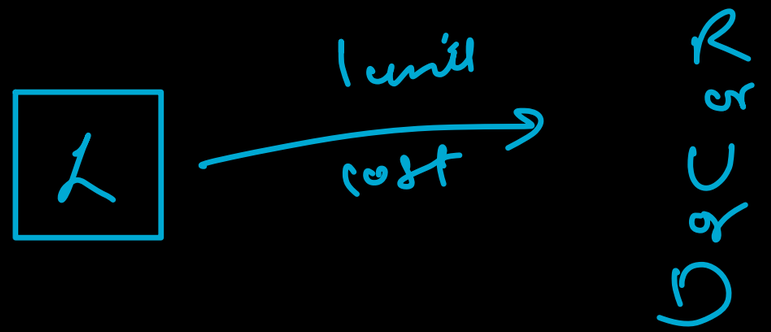
Prims

no. of edg > no. of vertex

dense graph



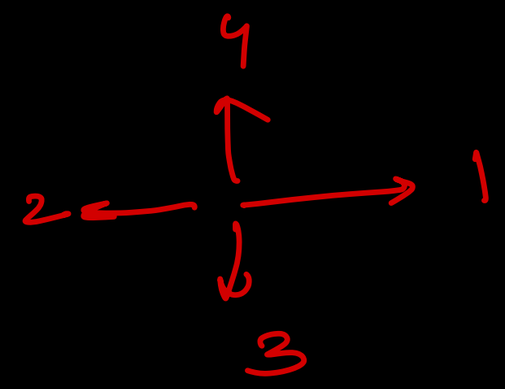
Q: Given a m×n grid, each cell is mentioned with one of the char ['L', 'R', 'U', 'D'] chars. The char represent which cell we can move to from the curr cell. You're standing from top-left. Now by spending 1 unit cost we can replace the char of the curr cell to other chars.

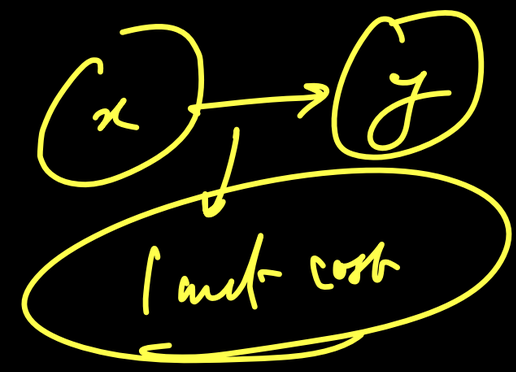


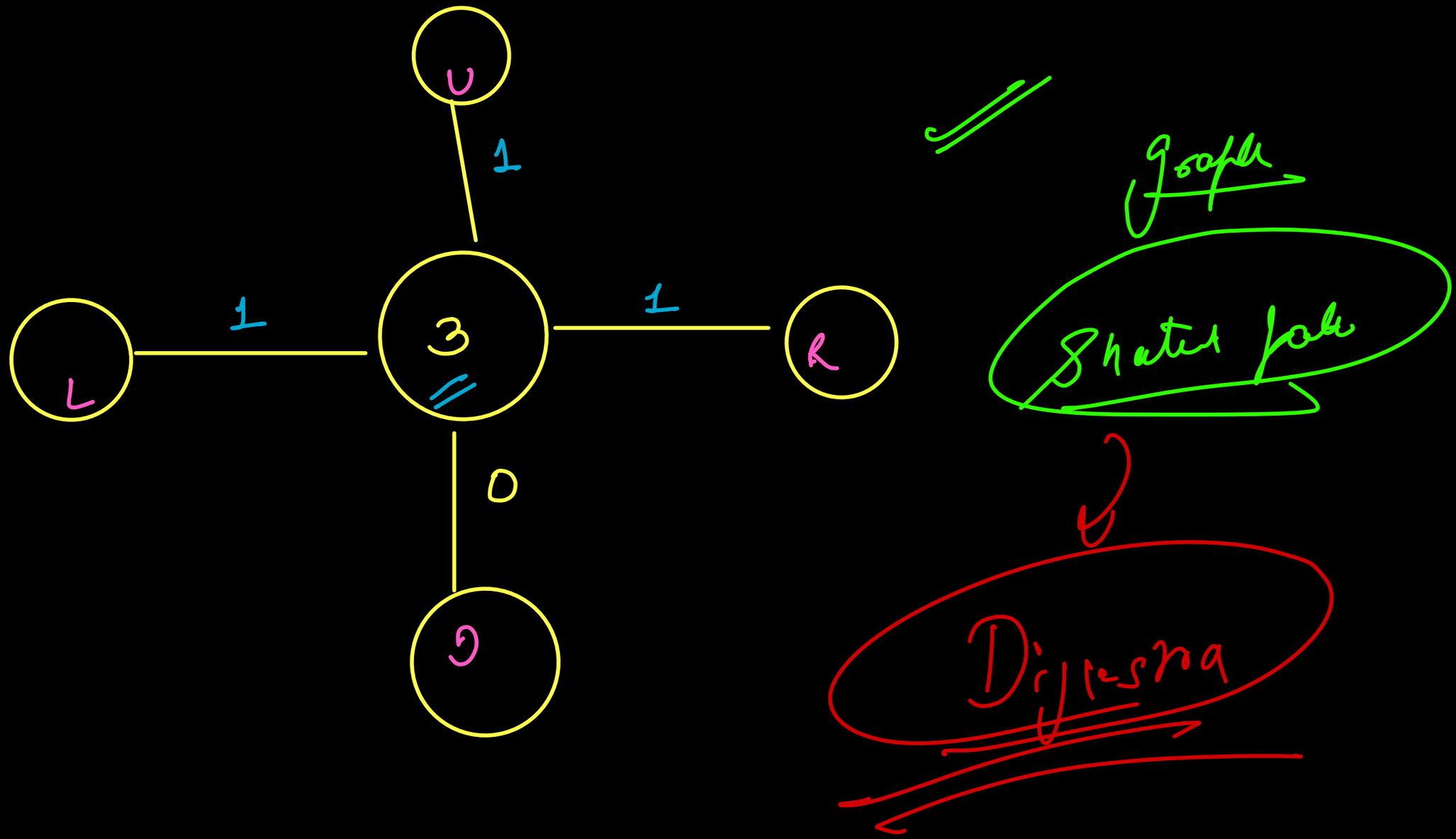
In what min cost we can reach the bottom Right

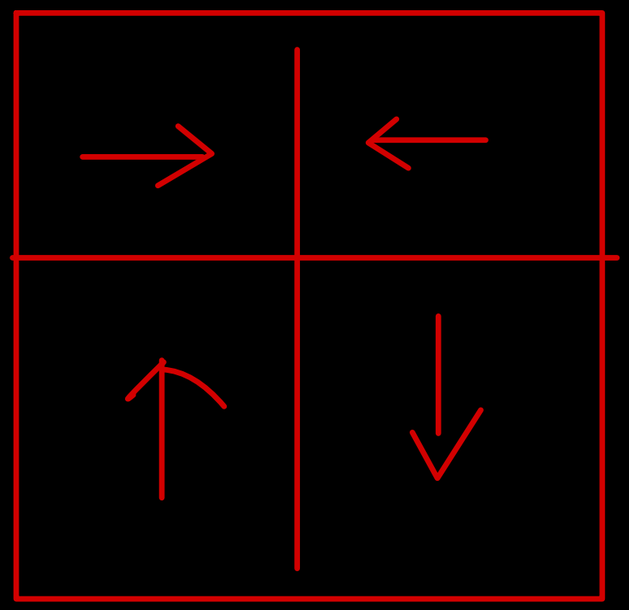
min cost

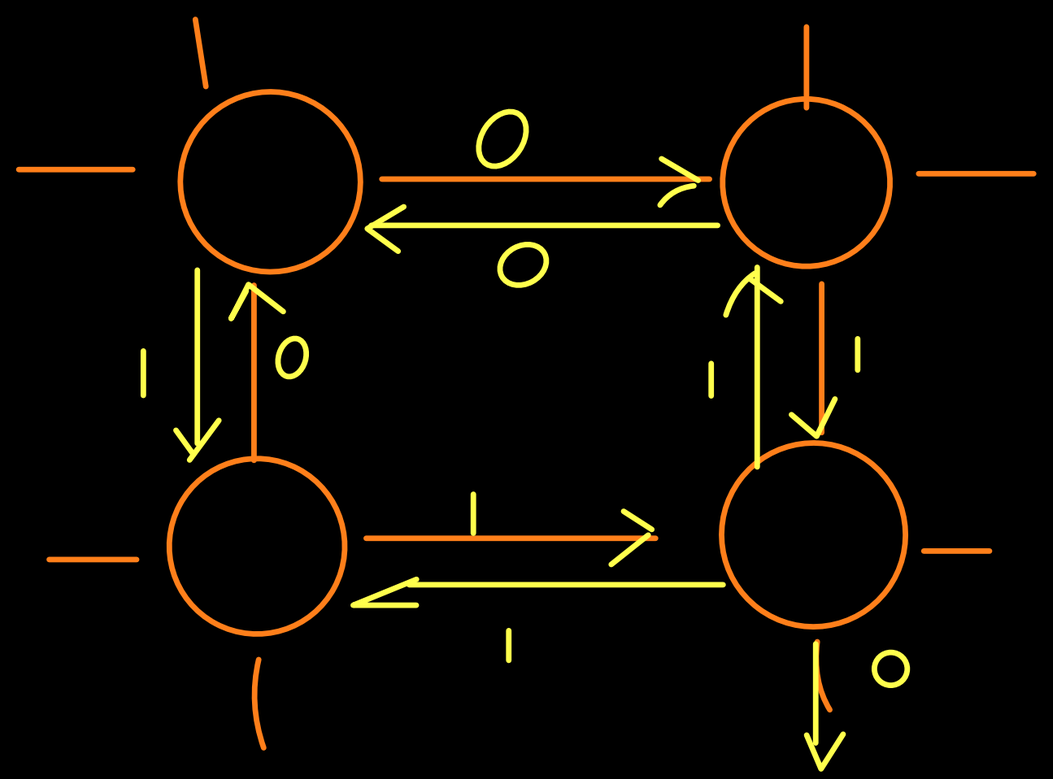
1	1	3	4	4
4	3	3	4	1
1	2	3	3	2
3	1	1	1	2
1	3	2	1	3











0,1

0-1 bfs

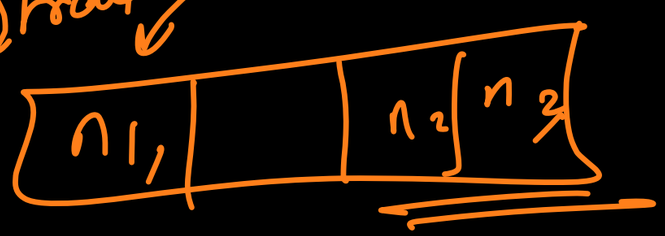
↳ queue

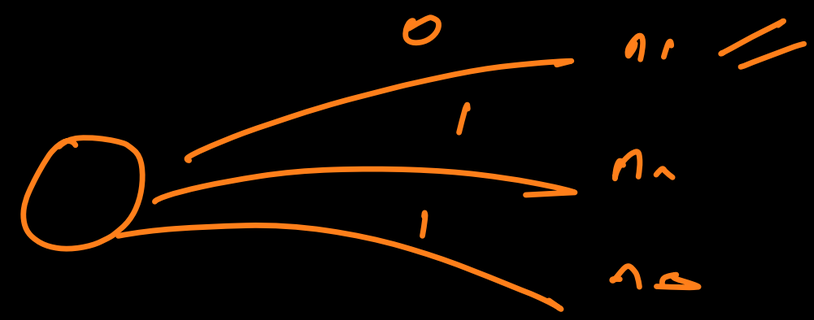
Dijkstra → PQ

deque

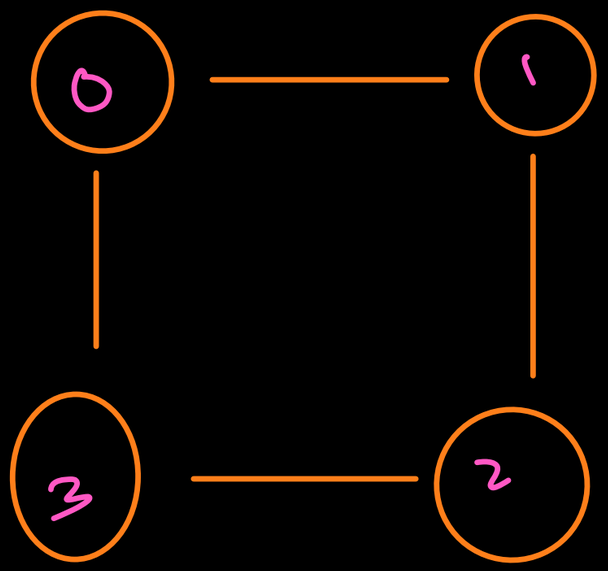
front

back





JOIN THE DARKSIDE



Set A

0, 2

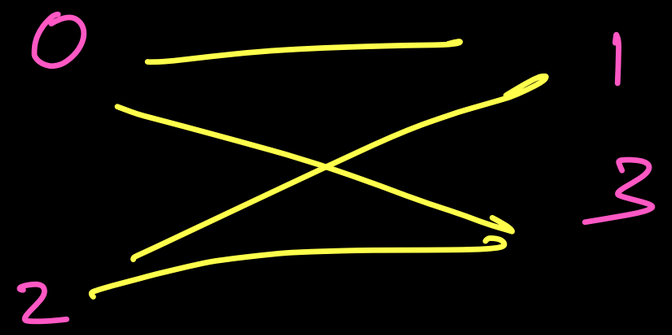
Set B

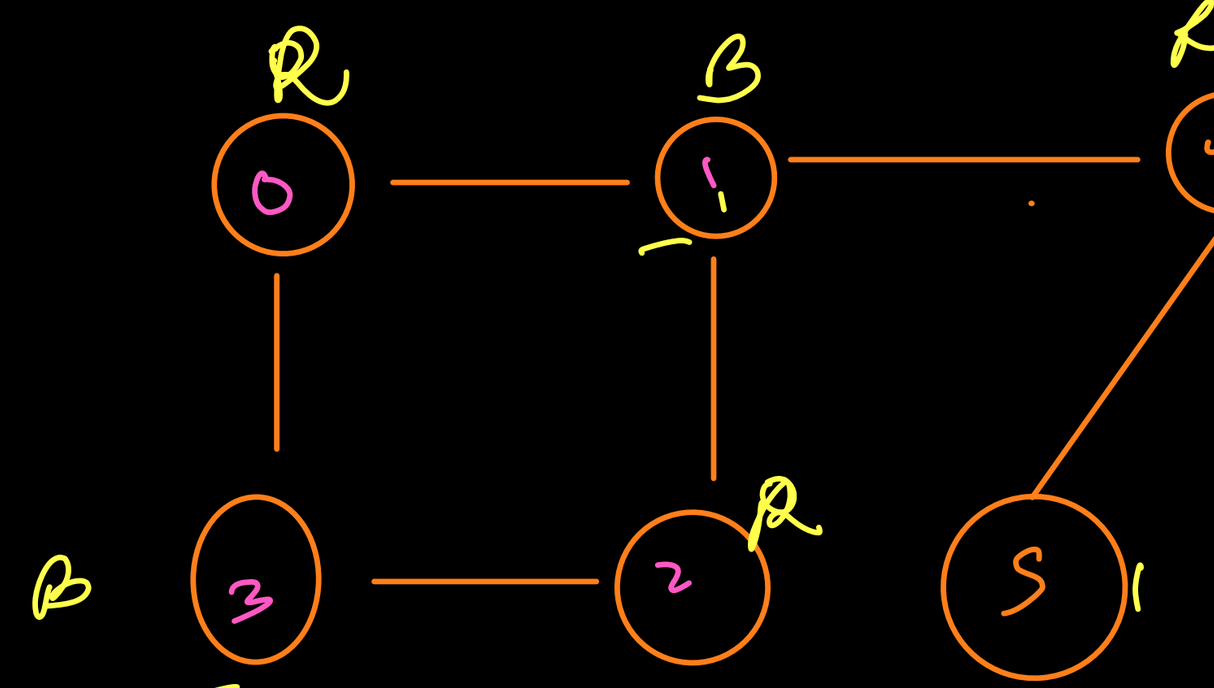
1, 3

2-colors

2^2 colors

R, B



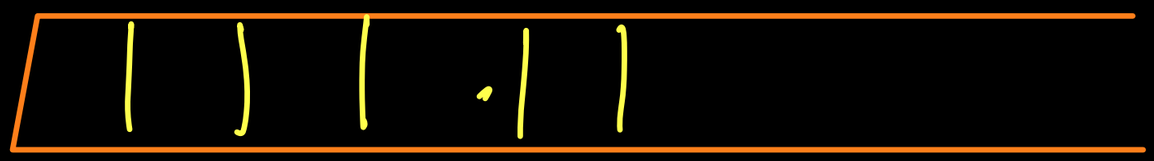


BFS

color[neigh] = !color[cur]

0 1 3 4 2 5 6

qu



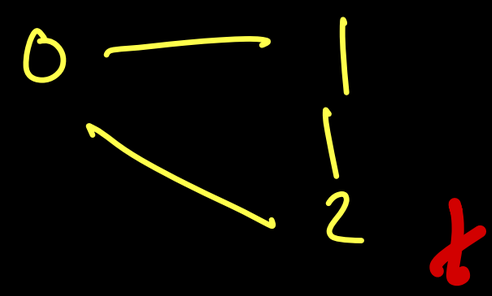
vis
0, 1, 3, 4, 2, 5, 6

color[i] = 1 Blue

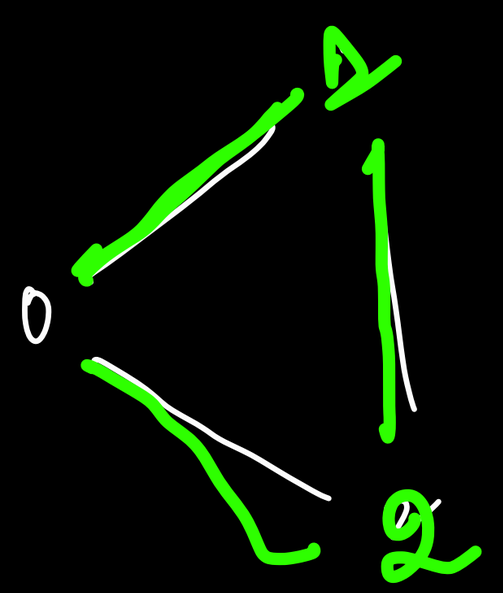
color[i] == 0 Black

0	1	2	3	4	5	6
0	1	0	1	0	1	1

color



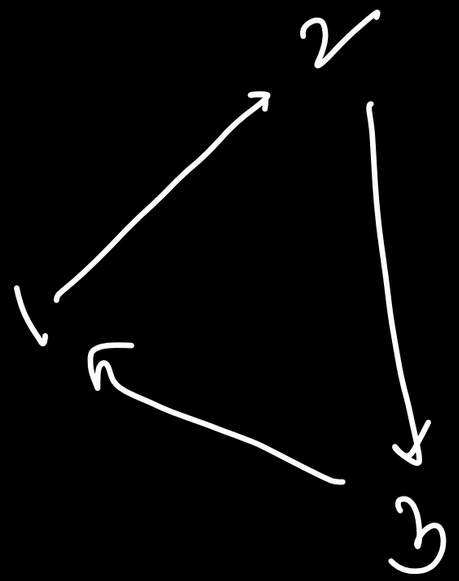
-ve



funct

3 temps

1



1 4 —

2 ——

3 ——