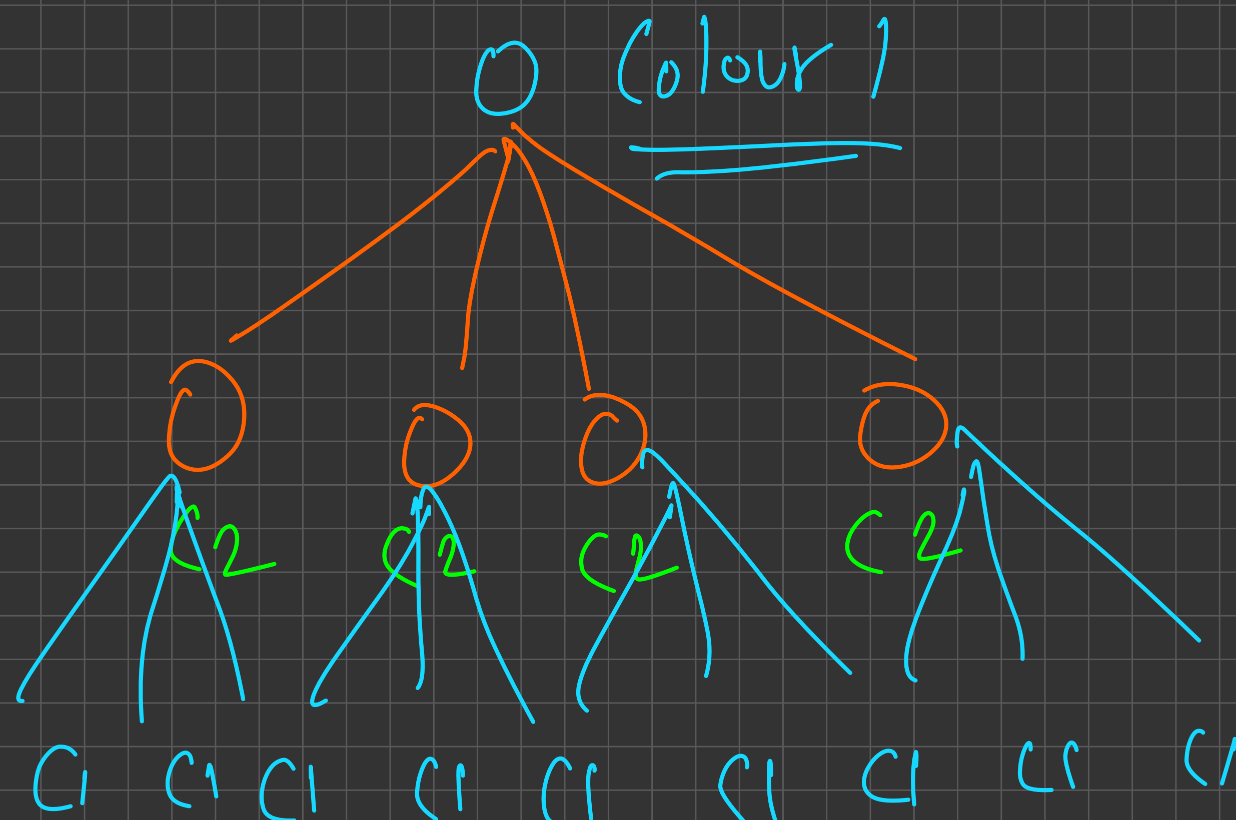
Graphs 2
Bipartite Graphs, Dijkstra, Bellman Ford

-Priyansh Agarwal

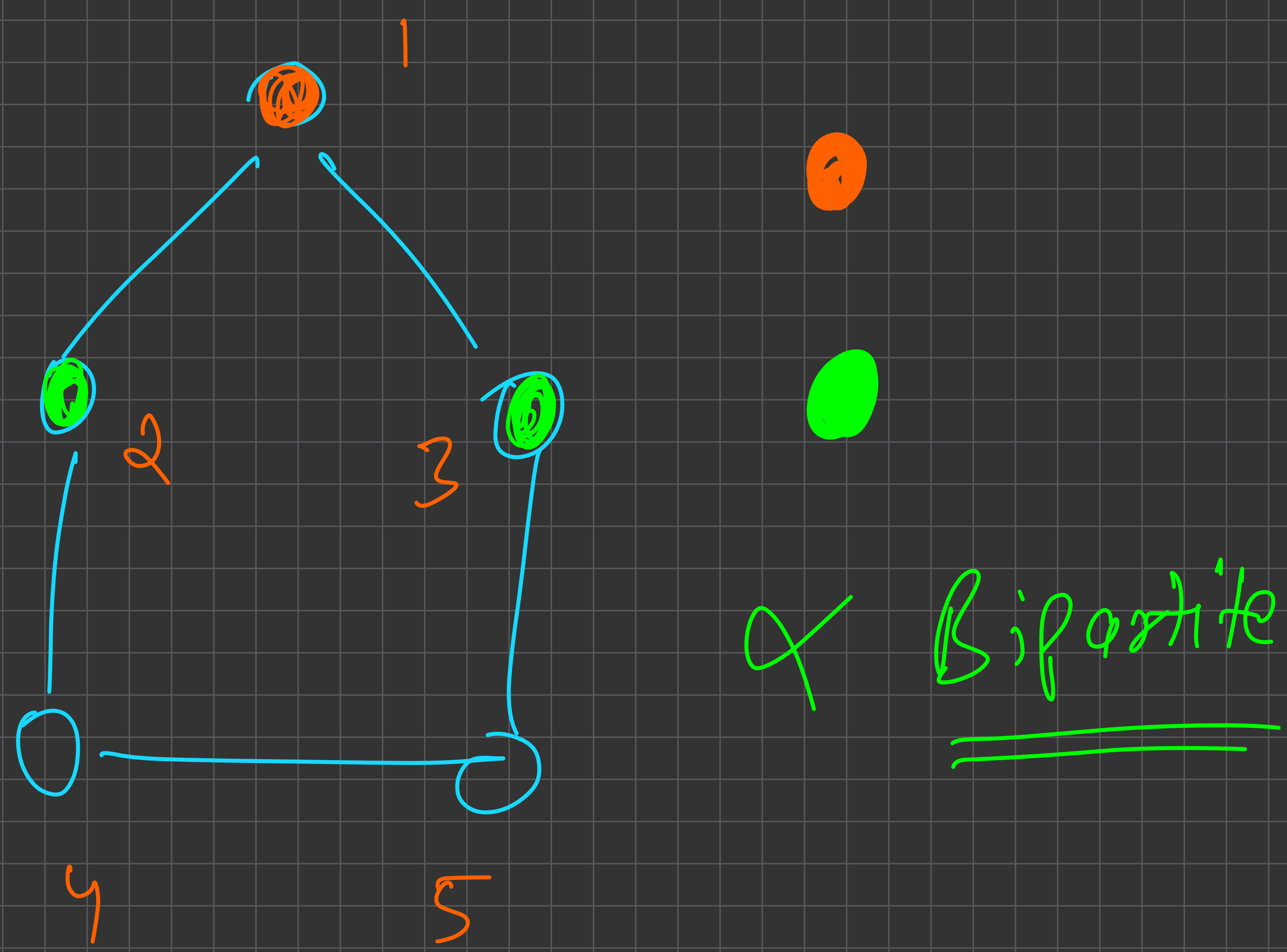
Bi-partite Graphs

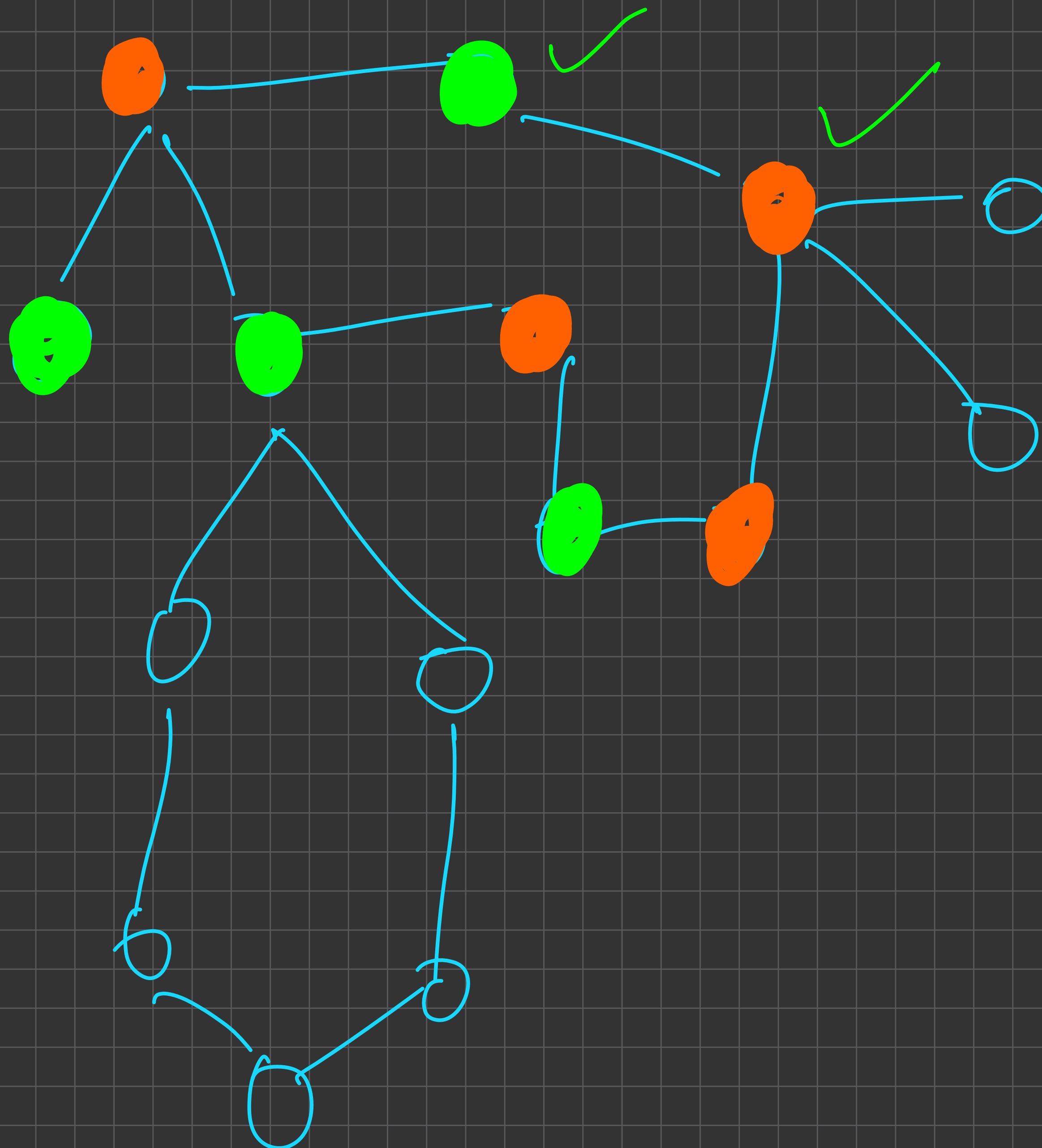
- Definition → 2 neighbours don't have same colour
- Algorithm
- Odd Length Cycle
- Tree Property
- Problem: [Link]

A graph can be coloured in a bipartite manner if and only if there is no odd length cycle in the graph



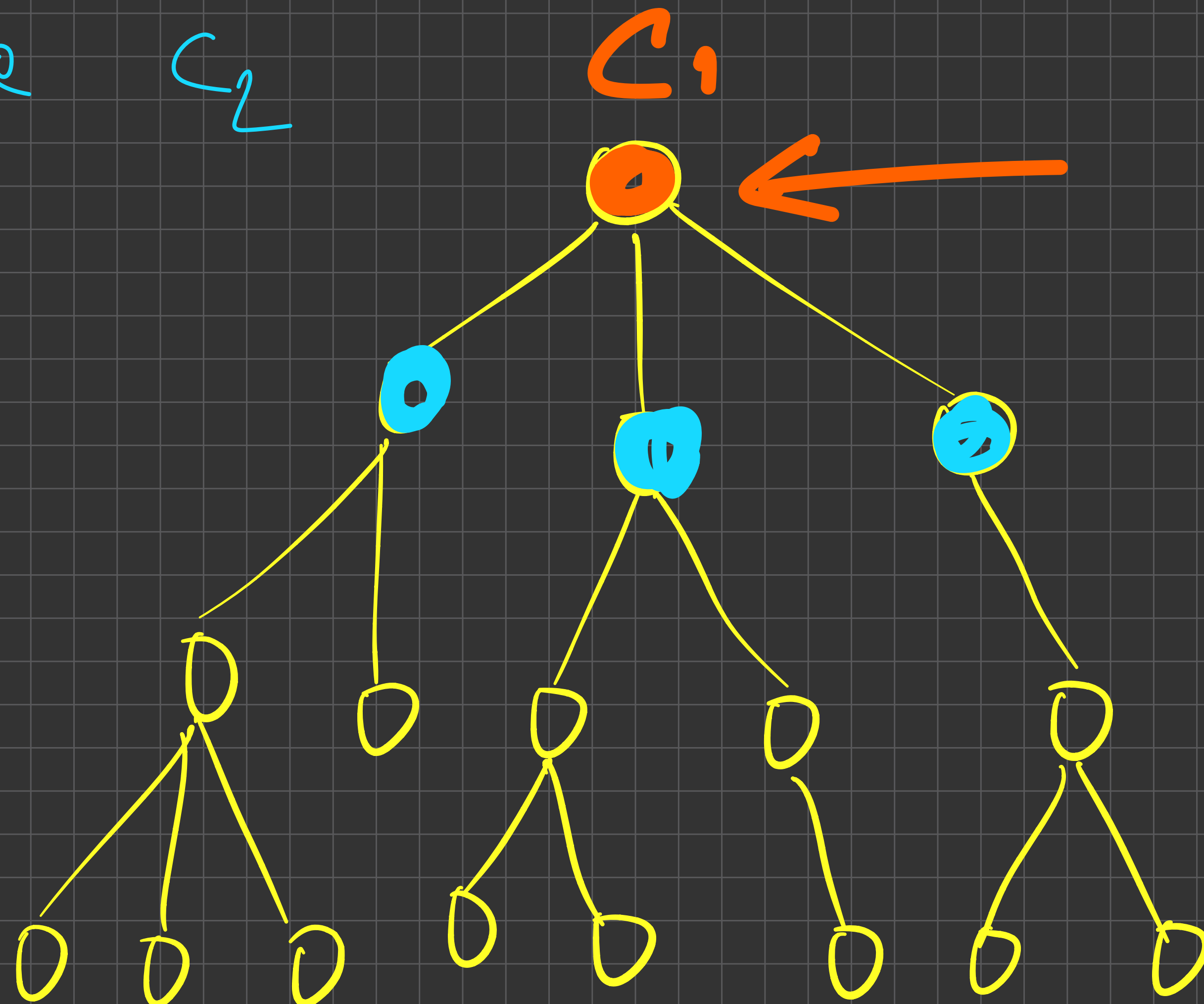
2 neighbours shouldn't have the same colour


1
2
3
4
5
✗ Bipartite



if level of node = odd → C_1

else C_2



```
bool: dfs ( int curr, edges, visited, color)
  if ((color[curr]) = 0) → color[curr] = 1
  for ( int child : edges[curr] )
      if ( color[child] == 0 )
          color[child] = color[parent] ^ 1
          dfs (child)
      else if ( color[child] == color[curr])
          return false
```

Dijkstra (most important)

- Single Source Shortest Path Algorithm - Idea + Visualization
- Non-negative edge weights
- Proof/Intuition:
    - On every iteration the marked vertex is the one that can never have a better distance later on.
- Code
- Retrieving the shortest path?
- Problems
    - Google Interview Problem

undirected + directed weighted

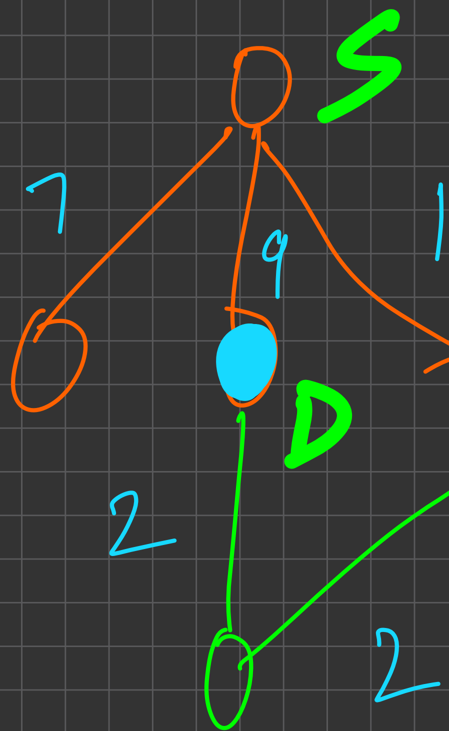
Q

A

SSSP

Dijkstra Tree

= 2



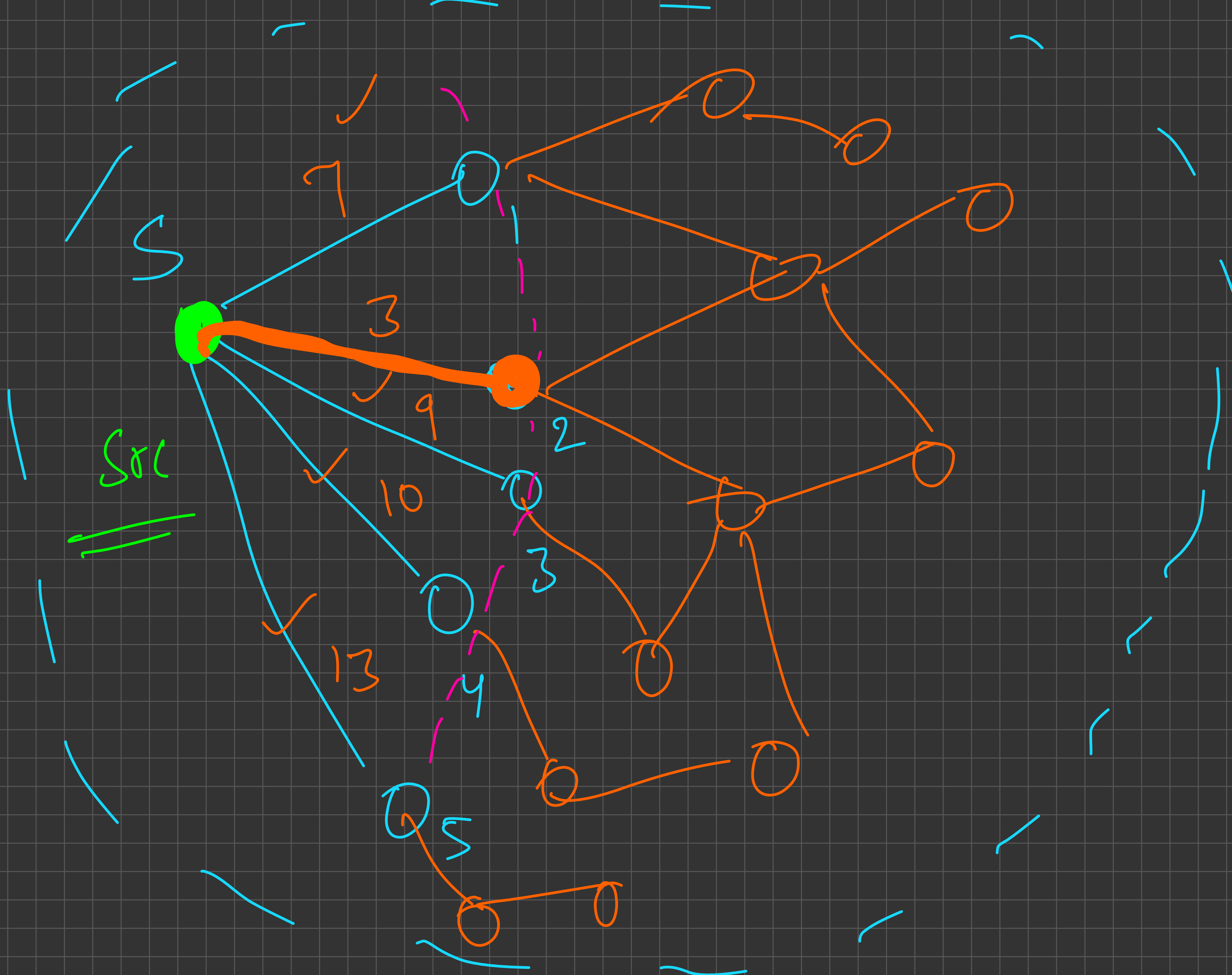
Dijkstra

undirected directed

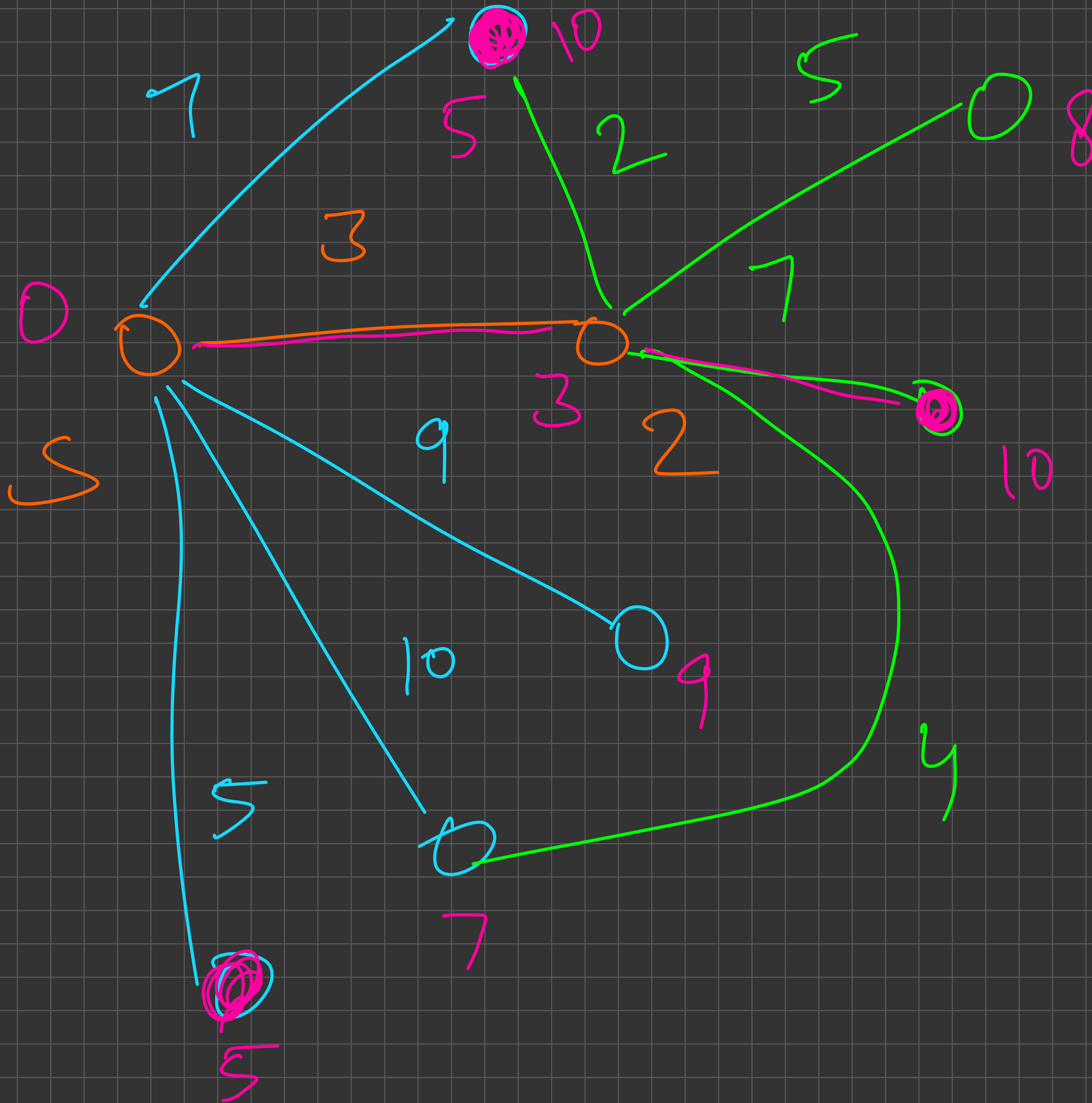
non negative edge weights

$O(n+m)$

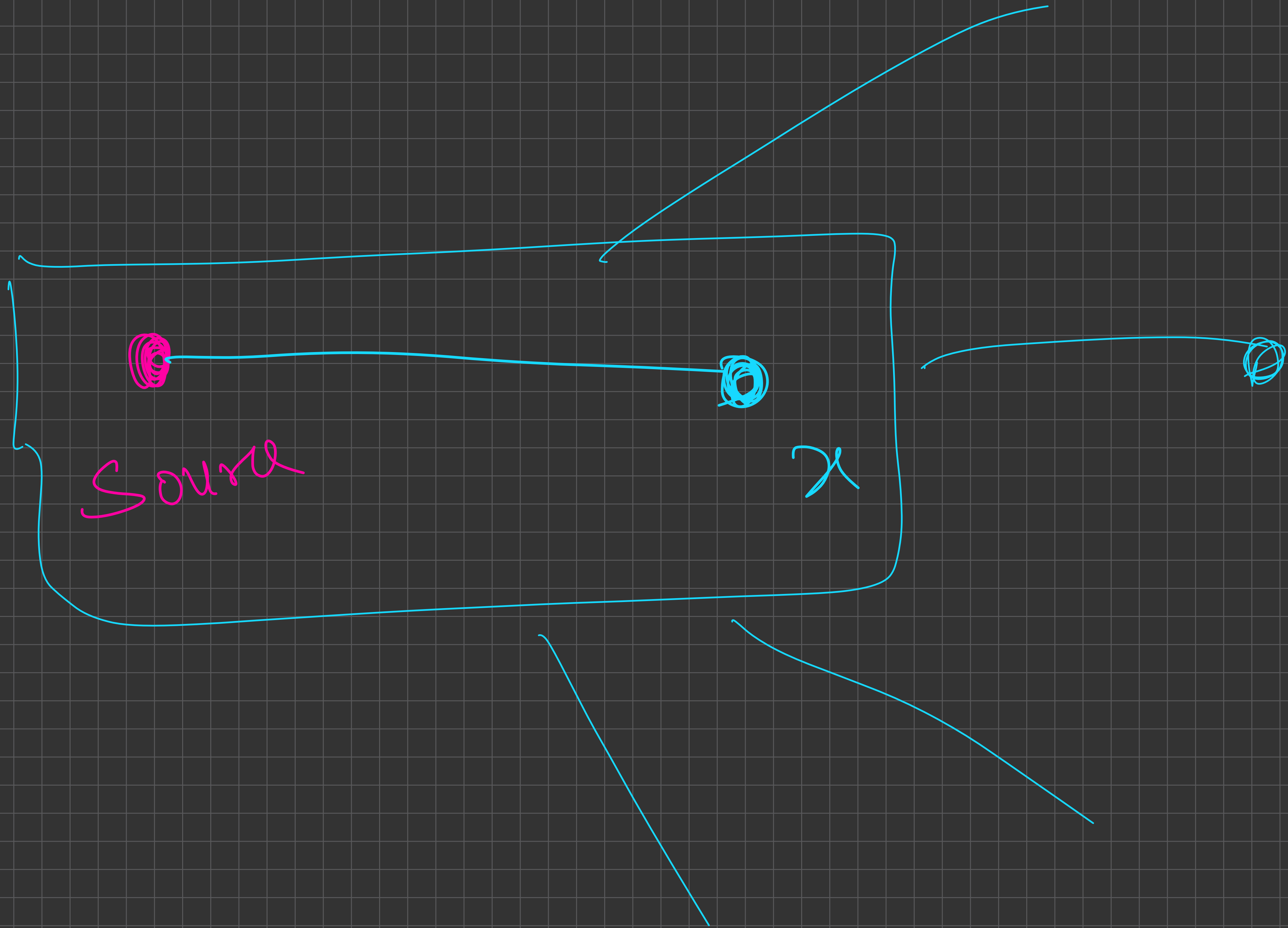
$O((n+m)(\log n))$

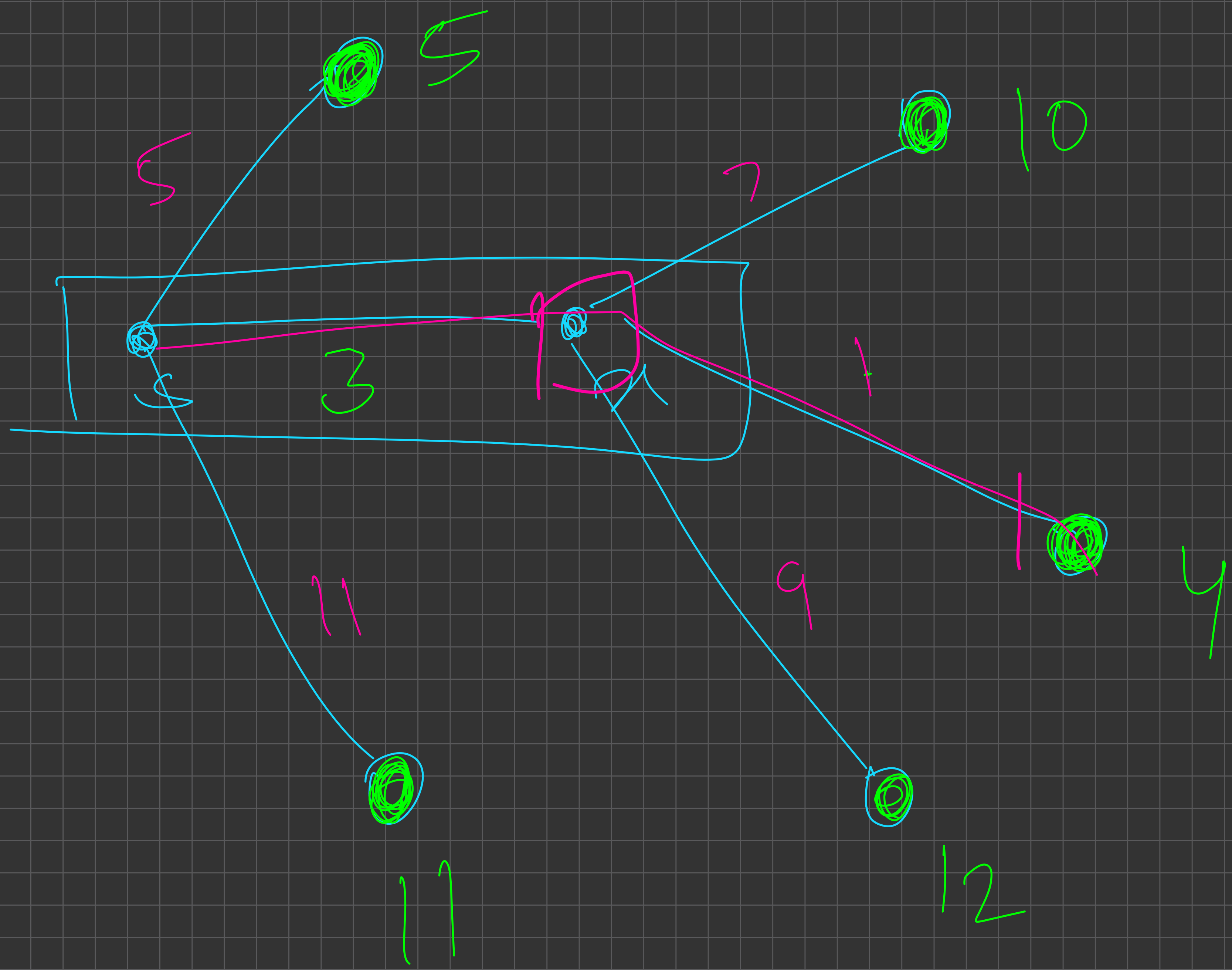


S
Src
7
3
9
2
10
3
13
4
5



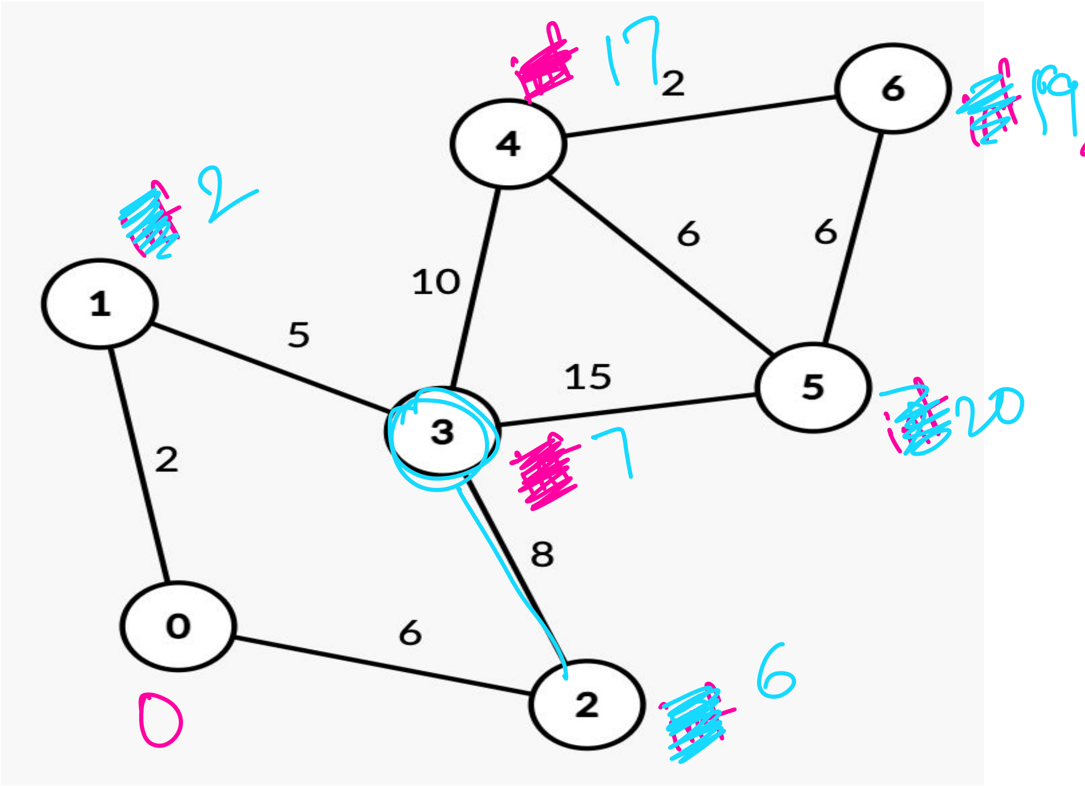
7
10
5
2
5
8
3
7
0
3
2
10
S
9
9
10
4
5
7
5


Source
x



5
5
10
7
3
1
4
9
11
11
12

Dijkstra Visualization



0	0	sat
1	2	sat
2	6	sat
3	7	sat
4	17	sat
5	20	sat
6	19	sat

n steps to saturating n nodes

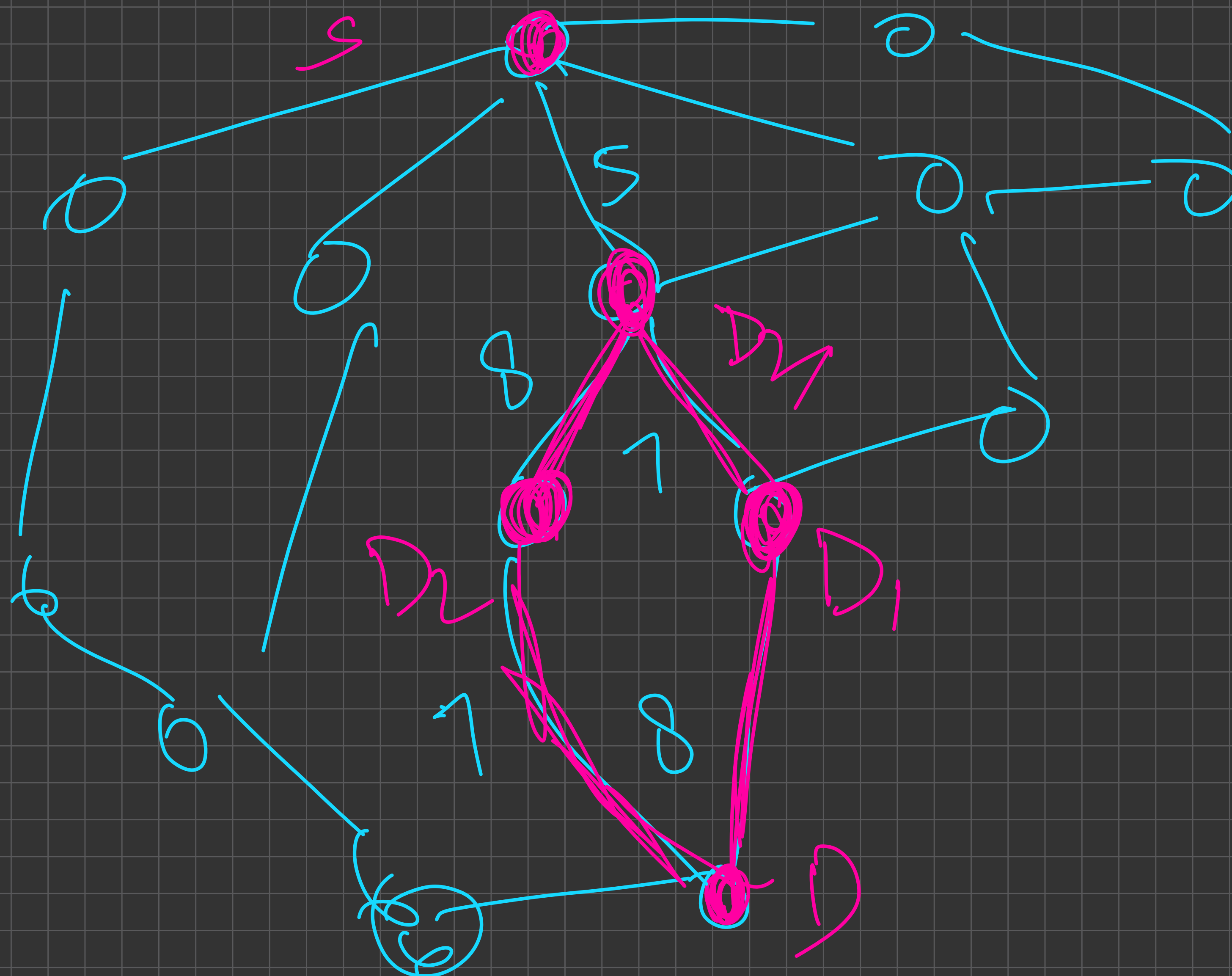
min dist answer → $O(\log n)$

iterate over the neighbours of the node

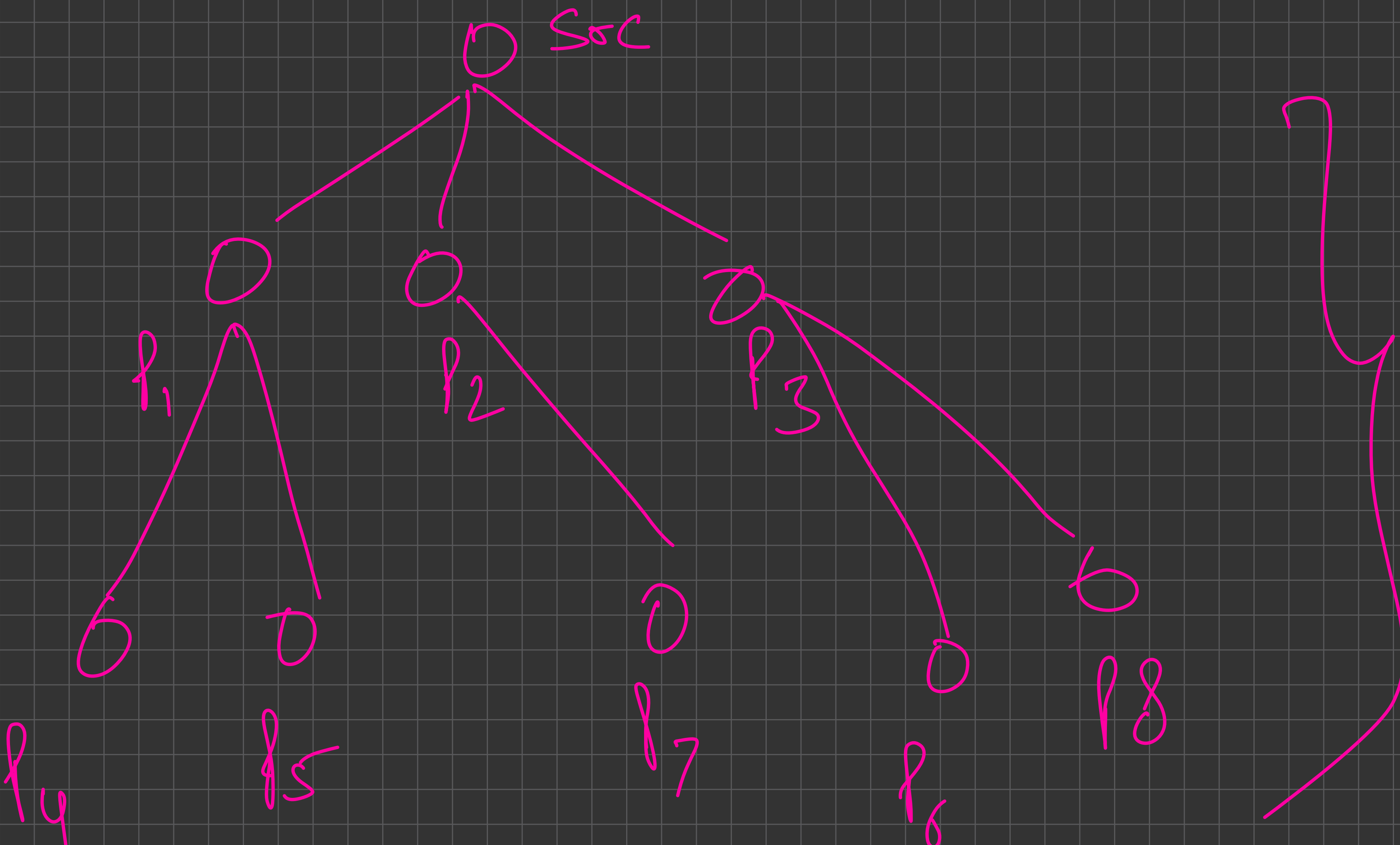
→ update the answer

$n \log n + m \cdot \log n$

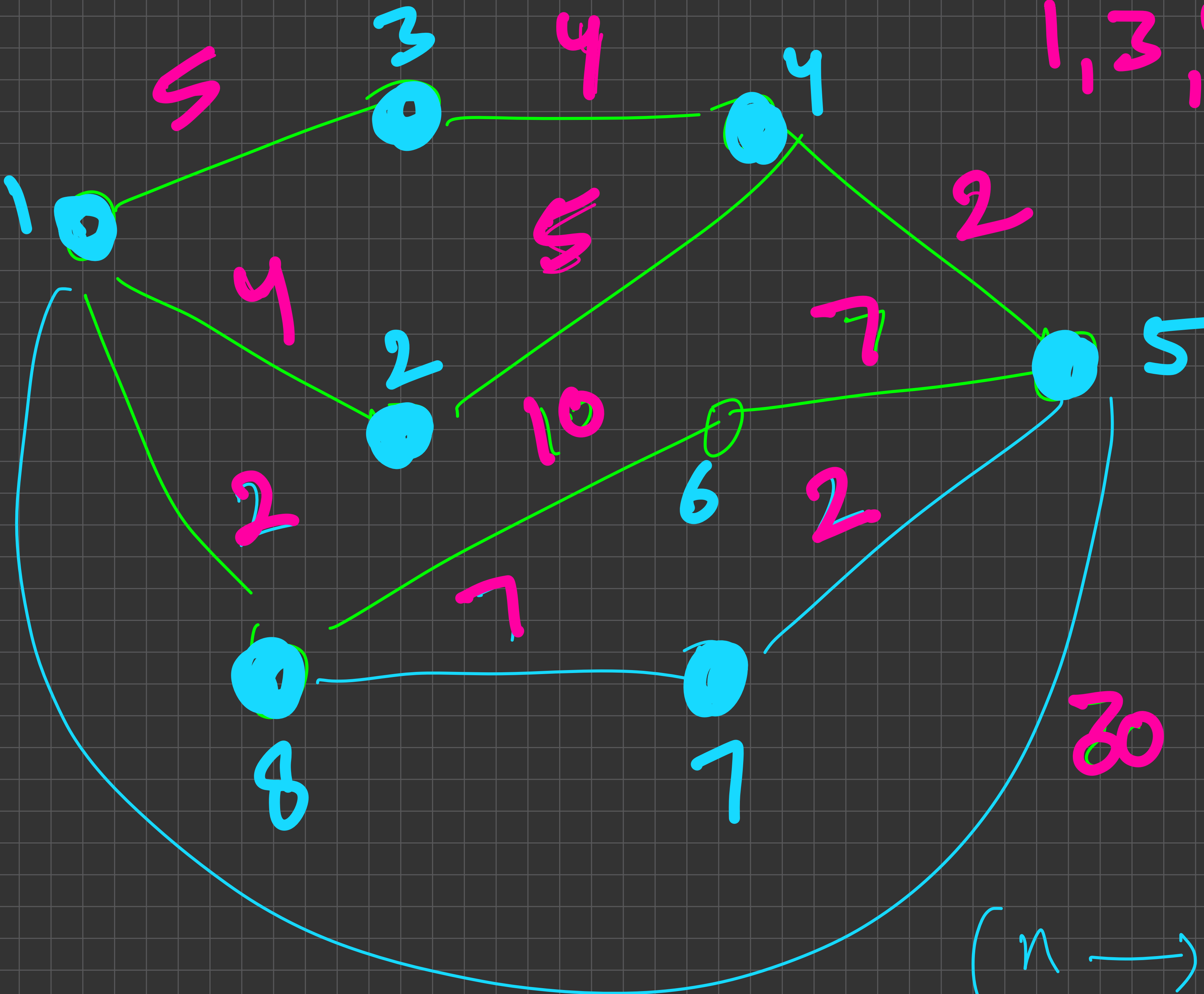
$O((n+m) \cdot \log n)$



S
5
8
7
D2
D1
D


Src
P1
P2
P3
P4
P5
P7
P6
P8

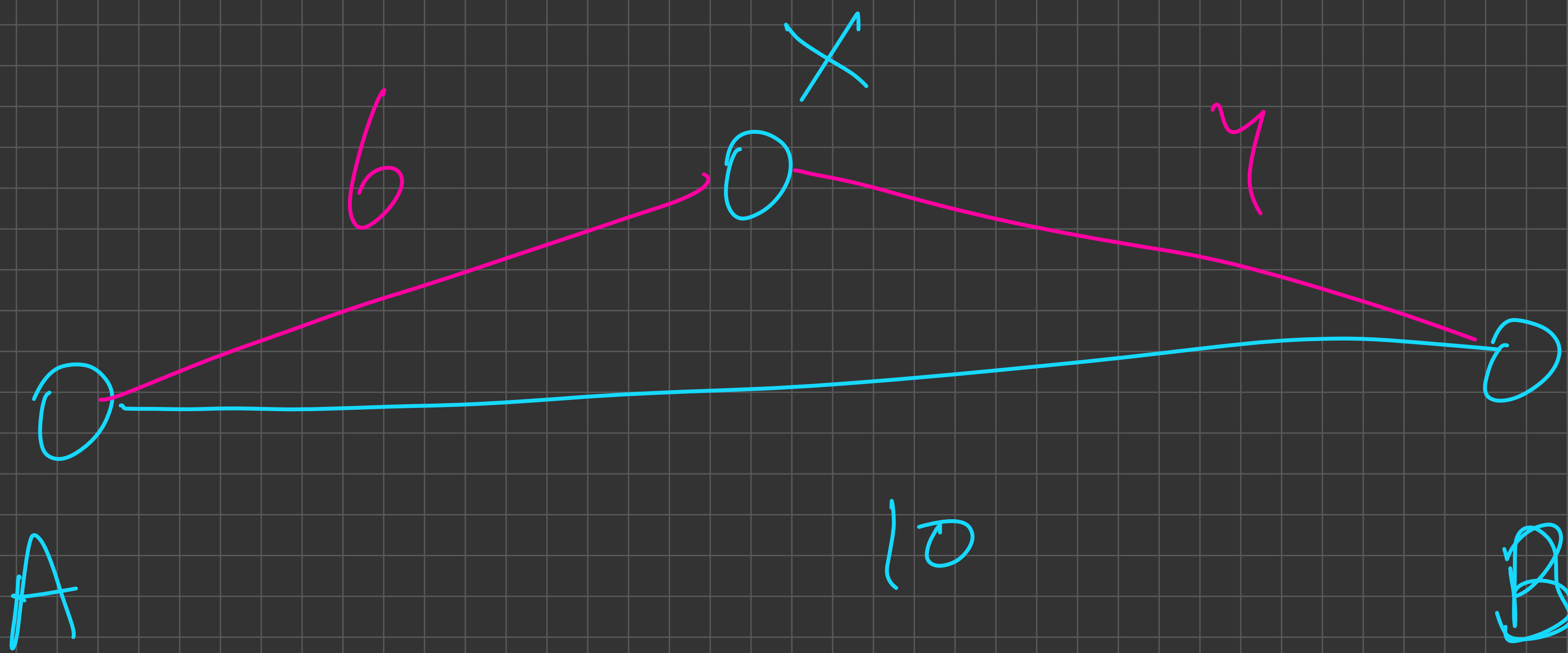
Given a graph with n ($n \le 10^5$) nodes and m ($m \le 10^5$) edges and 2 nodes A and B. Find out all the nodes which lie on any one of the shortest paths from A to B



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

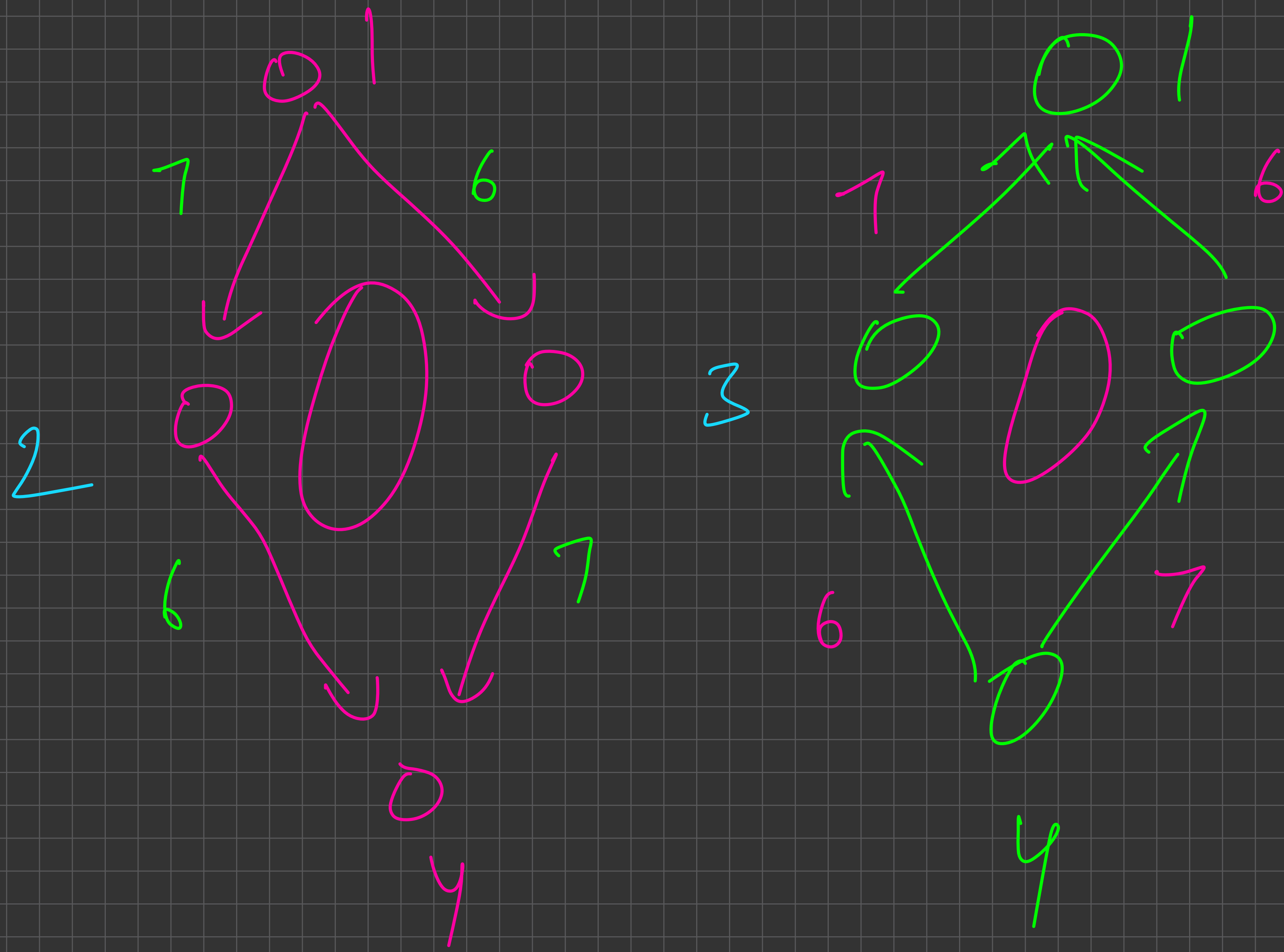
$(n \longrightarrow n)$

n^2

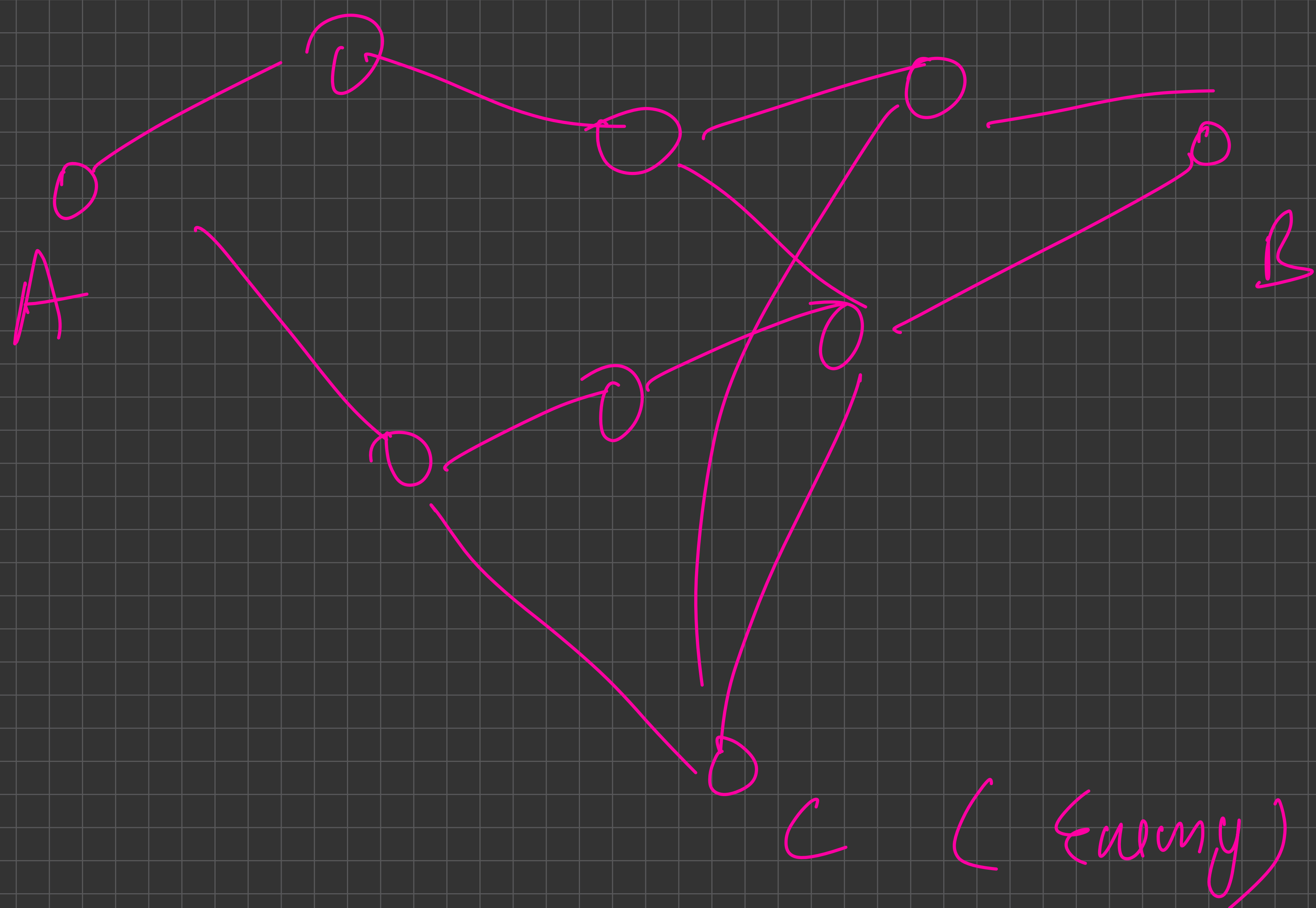


if $\text{dist}(A, x) + \text{dist}(B, x)$

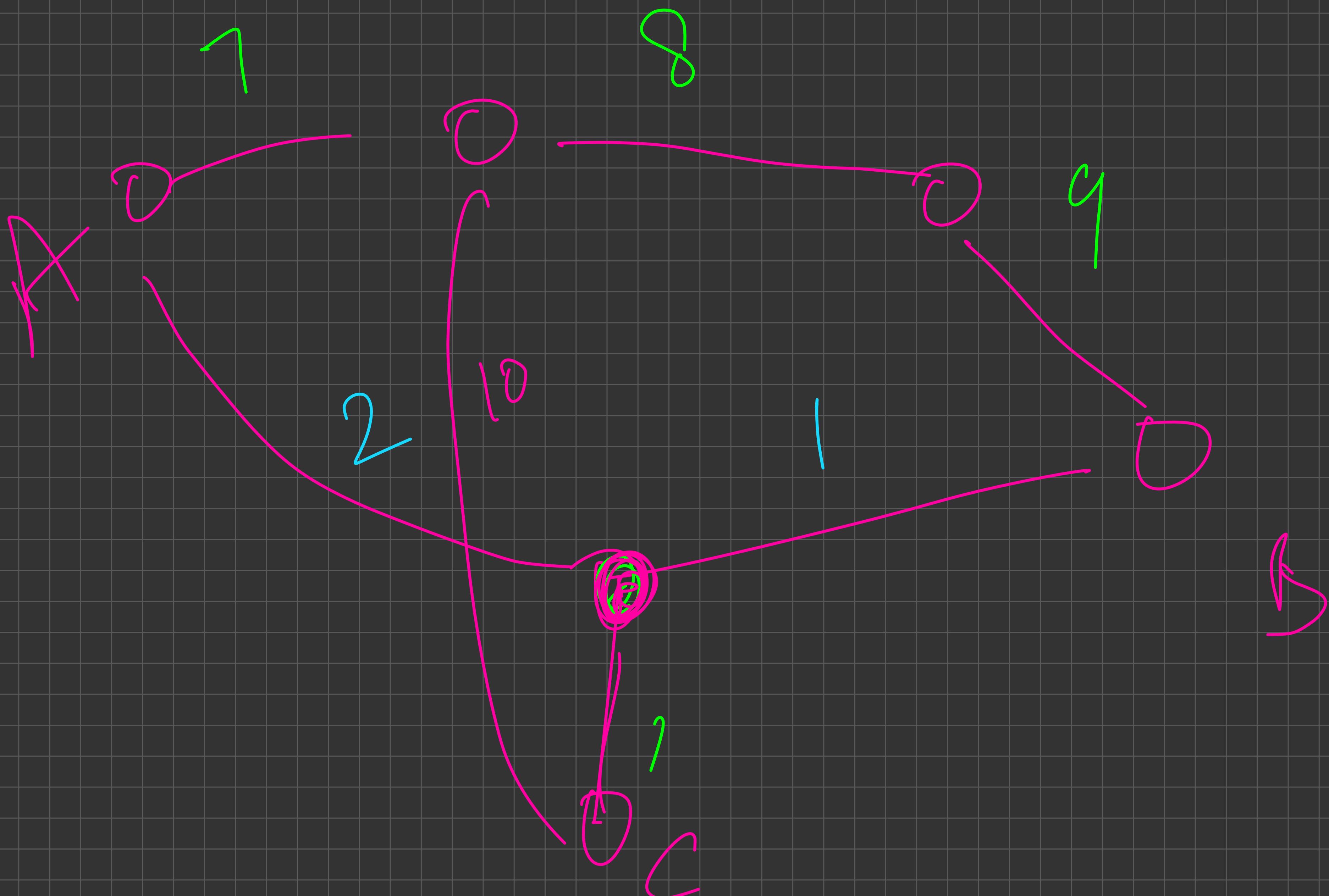
$$= \text{dist}(A, B)$$



1
7
6
2
6
7
4
3
1
7
6
6
7
4



A
B
C
(Enemy)


A
B
C
7
8
9
10
2
1
1

Run a Dijkstra from A

" " C " "

for all nodes x for which

$$\mathrm{dist}(A, x) \ngeq \mathrm{dist}(C, x)$$

remove x from the graph

Run a final dijkstra for

non removed nodes