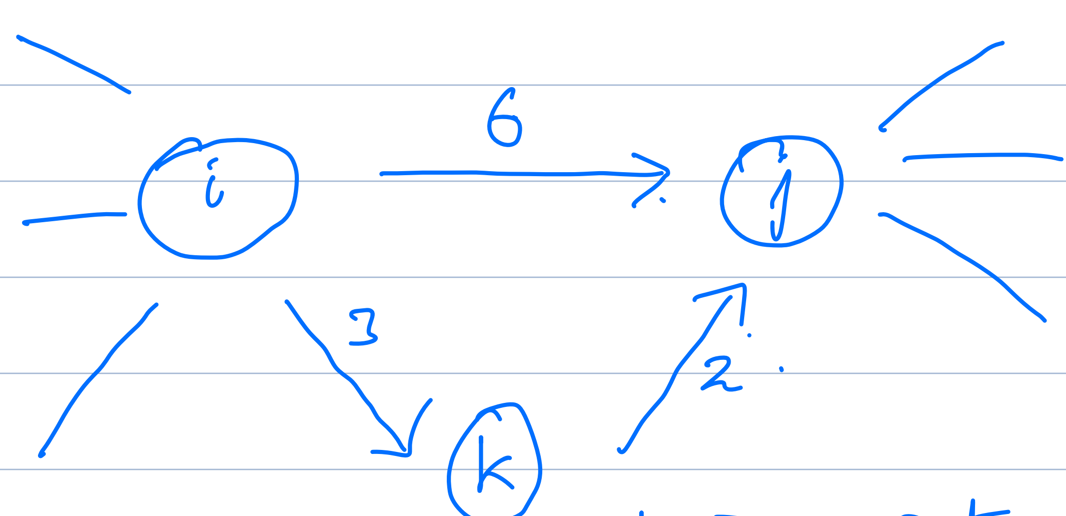
Today's Agenda:-

1. Floyd Warshall
2. Graph Coloring.
3. Bi-partite graph
4. Construct Roads.

Floyd Warshall

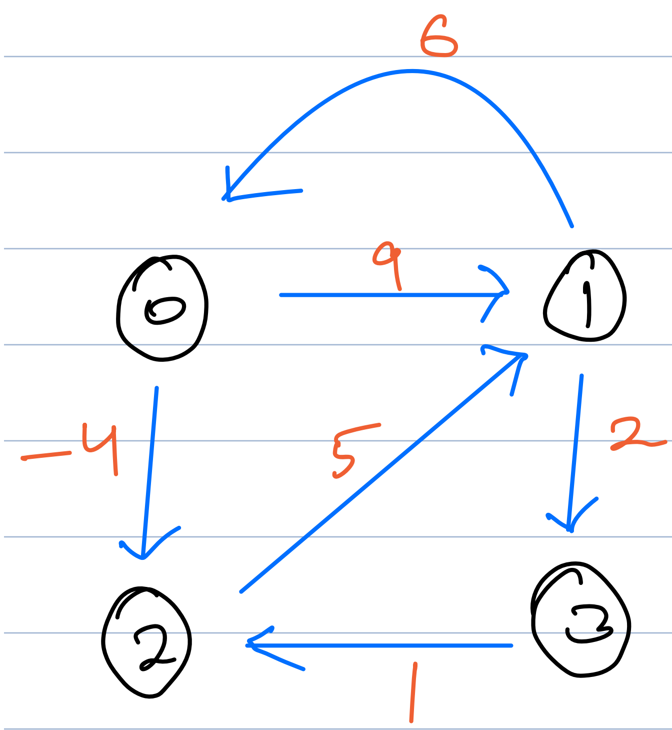
Find shortest distance from every node to every other node.

(All pairs shortest path)



↳ intermediate node.

idea – Consider every node as an intermediate node & try to relax the edges with smaller edge wts.



Adjacency Matrix

	0	1	2	3
0	0	9	-4	∞
1	6	0	∞	2
2	∞	5	0	∞
3	∞	∞	1	0

$\infty \rightarrow$ there is no edge b/w i & j.

(1) Consider Node 0 as I.N.

$d[i][j]$

$= \min(d[i][0] + d[0][j], d[i][j])$

	0	1	2	3
0	0	9	-4	∞
1	6	0	2	2
2	∞	5	0	∞
3	∞	∞	1	0

$d[2][0] = \min(d[2][0] + d[0][0], d[2][0])$

$d[1][1] = \min(d[1][0] + d[0][1], d[1][1]).$

$d[2][3] = \min(d[2][0] + d[0][3], d[2][2]).$

$d[3][1] = \min(d[3][0] + d[0][1], d[3][1]).$

② Take 1 as Intermediary.

	0	1	2	3
0	0	9	-4	11
1	6	0	2	2
2	11	5	0	7
3	∞	∞	1	0

$$d[1][2] = \min\left(d[1][1] + d[1][2],\ d[1][2]\right)$$

$$d[0][0] = \min\left(d[0][1] + d[1][0],\ d[0][0]\right)$$

③ Take 2 as intermediary.

	0	1	2	3
0	0	1	-4	3
1	6	0	2	2
2	11	5	0	7
3	12	6	1	0

④ Take 3 as intermediary.

	0	1	2	3
0	0	1	-4	3
1	6	0	2	2
2	11	5	0	7
3	12	6	1	0

Code

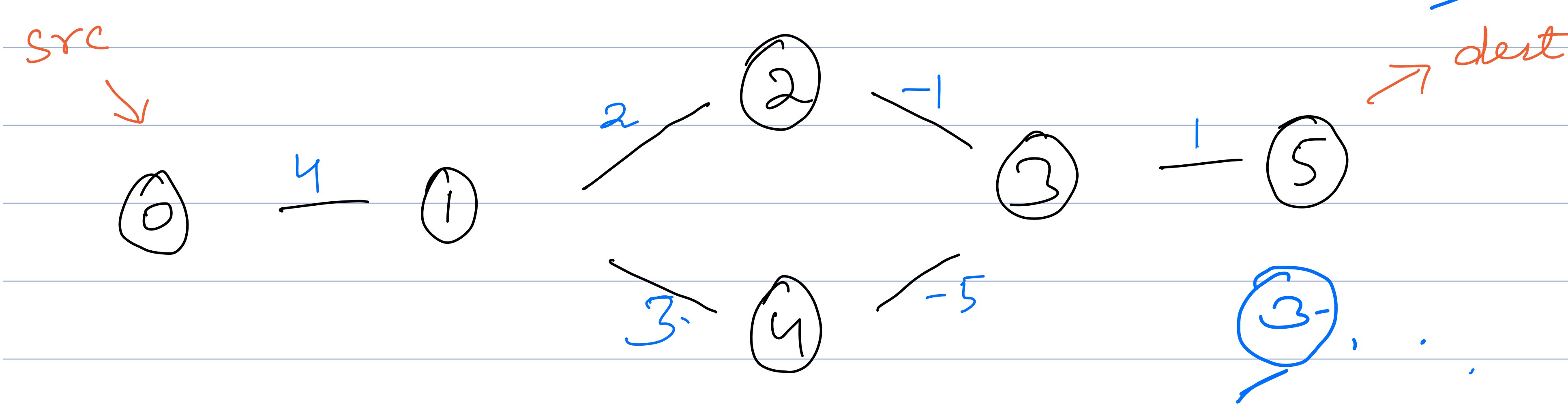
Intermediary Node.

```
for (k=0; k<N; k++) {
    for (i=0; i<N; i++) {
        for (j=0; j<N; j++) {
            if (d[i][k] + d[k][j]
                        < d[i][j]) {
                d[i][j] = d[i][k] + d[k][j];
            }
        }
    }
}
```

T.C → $O(N^3)$.
S.C → $O(1)$.

Qr Is shortest distance b/w 2 nodes always possible ?

For -ve weight cycle Floyd Warshall will give an incorrect answer.



0 - 1 - 4 - 3 - 5

0 - 1 - 4 - 3 - 2 - 1 - 4 - 3 - 5

2

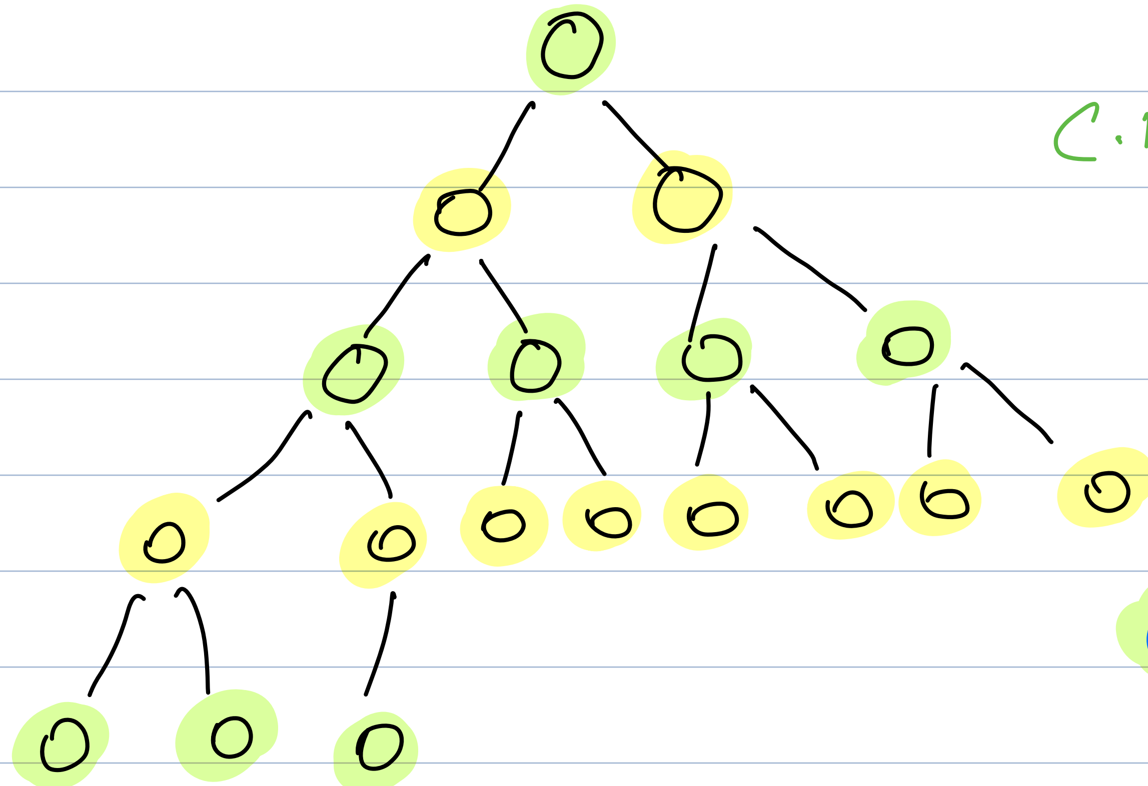
Graph Coloring

Francis Guthrie (1852)

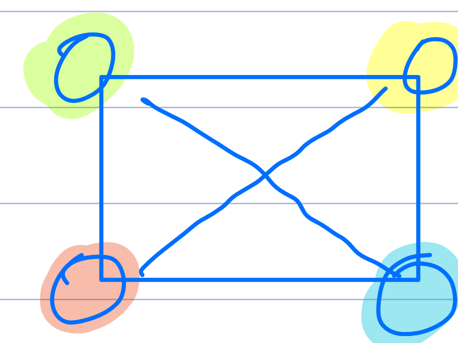
I do not need more than 4 colours to color any map.

Min. no of colors required to color all the nodes such that no two adjacent nodes share the same color. → **Chromatic Number**

① Tree

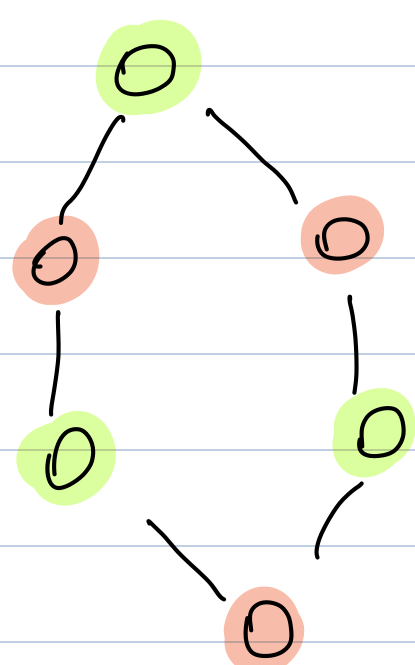


C.N = 2.

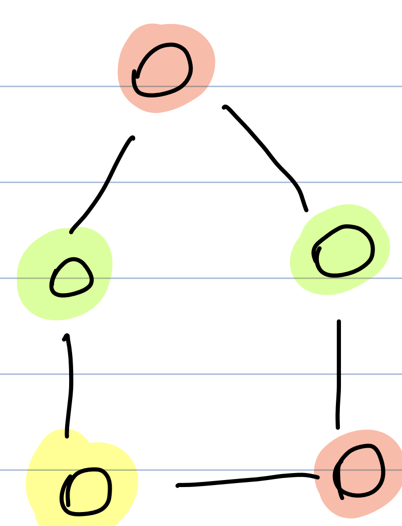


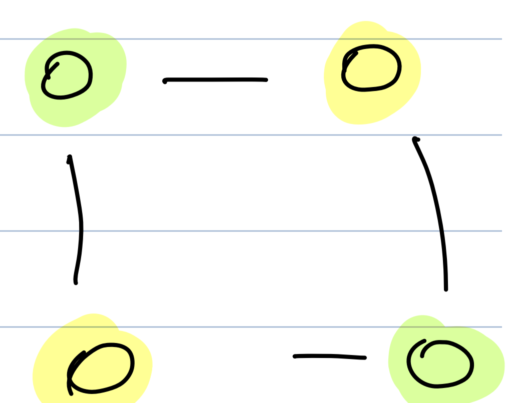
② Cycle Graph (whole graph is cycle)

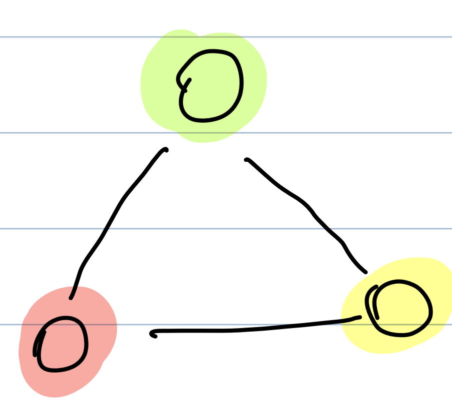
CN = 2



CN = 3.







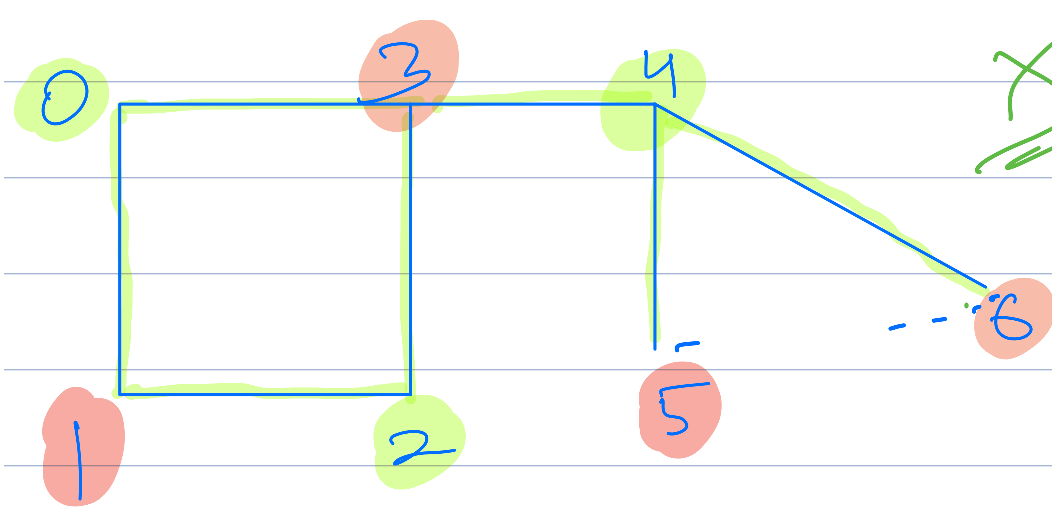
N is no. of nodes.

CN. of a cyclic graph is $2 + (N \% 2)$.

Bi-partite Graph

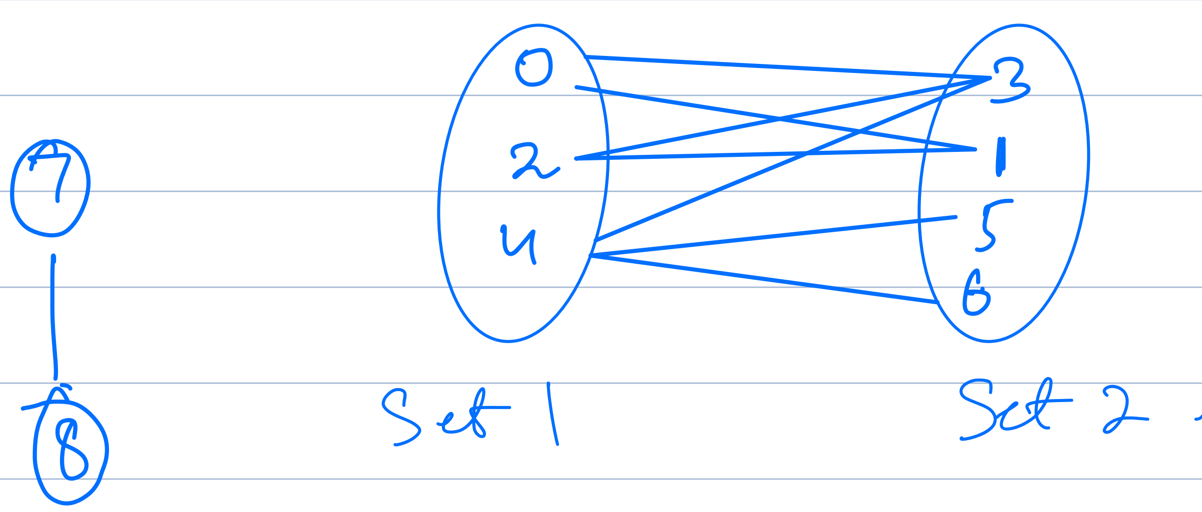
- Any graph with C.N = 2.

→ A graph is called bipartite if we can divide all the nodes in 2 sets, such that all the edges are across the sets.



col:

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



0 → Green.
1 → Red.

$3 \neq 4$.

8:25.

Code:

col[N] ; ∀i col[i] = -1,

```
for (i=0; i<N; i++) {
        if (col[i] == -1) {
              col[i] = 0;
          if (dfs(graph, i) == false){
              return false;
          }
        }
}
return true
```

```
boolean dfs( graph, src) {
     for (int nbr : graph[src]){
        if (col[nbr] == col[src])
            { return false;}
        else if ( col[nbr] == -1) {
            col[nbr] = 1 - col[src];
                            // opposite colour
                               of src.
            if (dfs(graph, nbr)
                   == false) {
              return false;
            }
        }
     }
  return true;
}
```

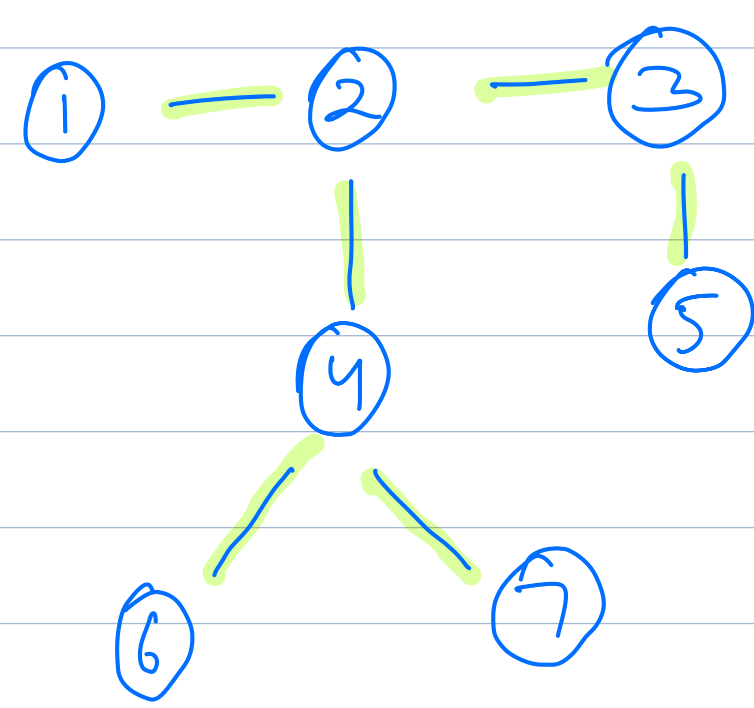
T.C → O(N+E)

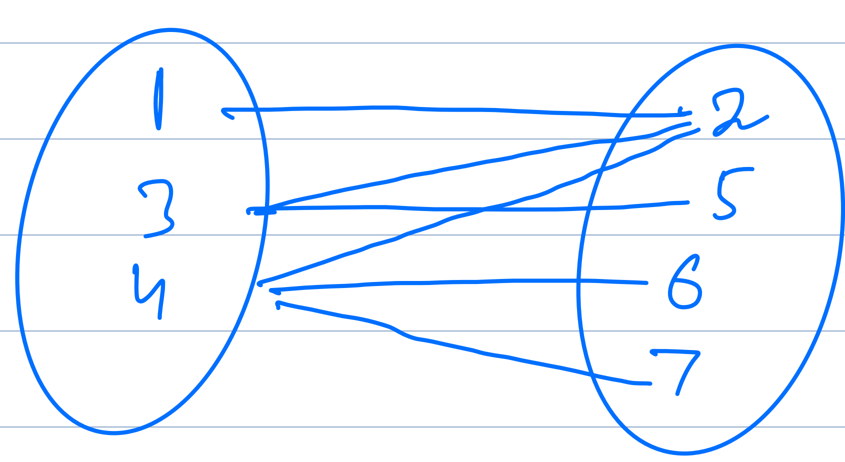
S.C → O(N+E)

↑ Recursive stack.

Q:- A country consists of N cities connected by (N-1) roads. King of that country wants to construct max. roads such that cities can be divided into sets & there is no road b/w cities in the same set. Find max. no. of new roads that can be created?

Note:- All cities can be visited from one city.





Set 1.   Set 2.

Total Roads needed
$= 3 * 4$
$= 12.$

ans $= 12 - 6$
$= 6.$

Using concept of bipartite graph, identify the nodes in each set.

a ; b

Answer is $(a * b - (N-1))$