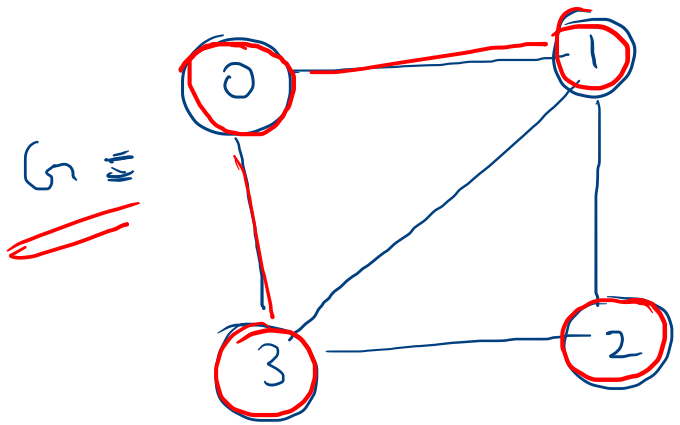
Shortest Path in Unweighted Graph



G ≡

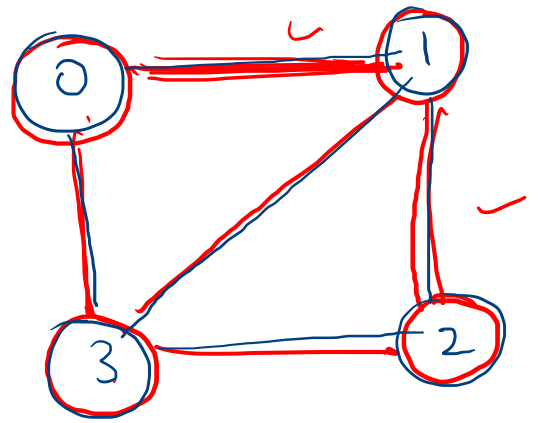
Source

s = 0

Shortest ?

⇒ what is the shortest path to delhi ?

Now source → 0



⇒

Print the shortest distance to every other vertex from source vertex

dist =

0	1	2	3
0	1	2	1

How, Solution

we are going to use BFS

Reco2,

It first traverse vertex

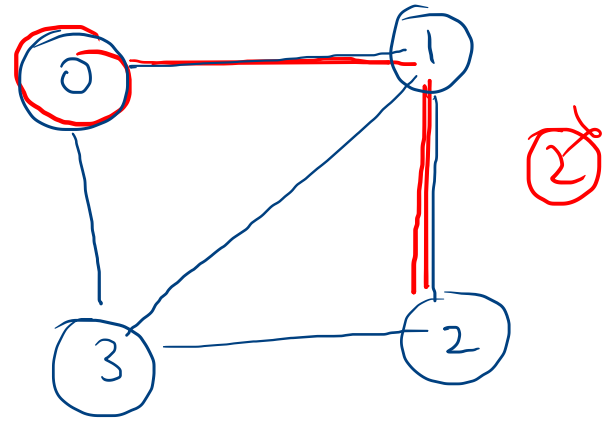
1 edge away

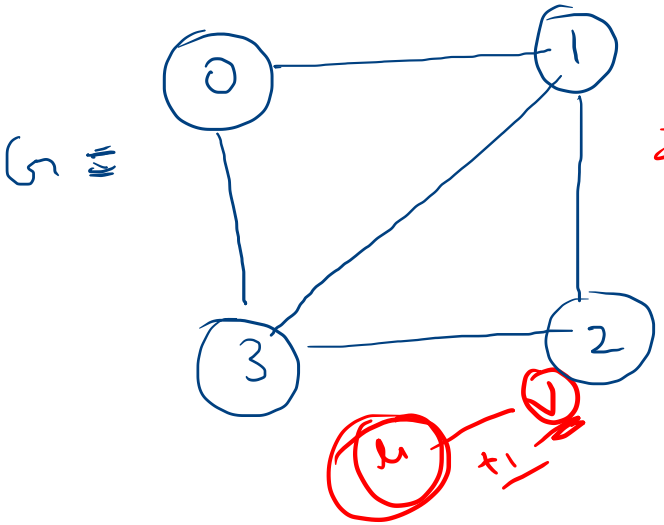
2 edge away

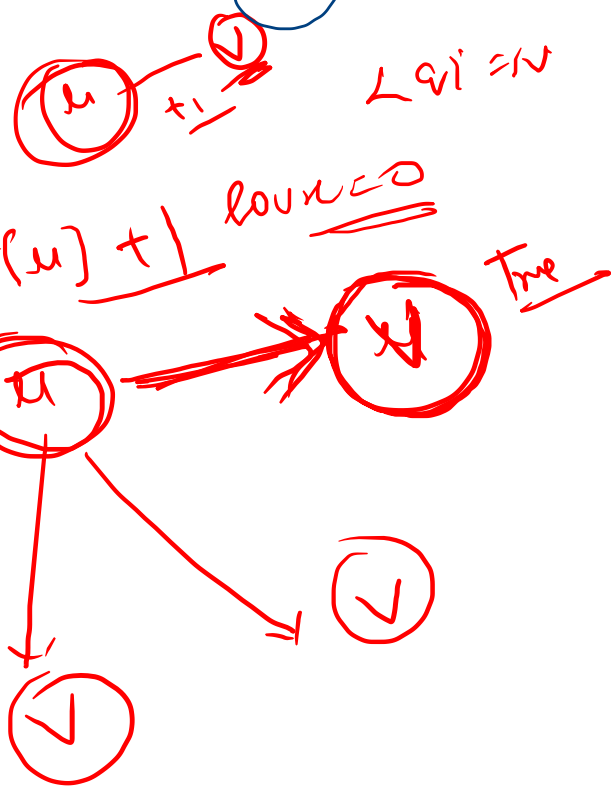
3 edge away

then

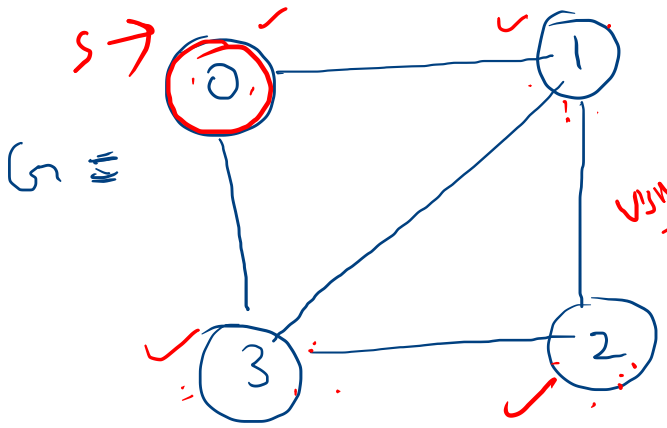
then







1. Initialise dist[V] = { ∞, ∞, ∞ . ∞ }
2. dist[s] = 0
3. Initialise, visited[V] = { F, F, F, F }
4. create, queue, q ;
5. q.push(s) ; visited[s] = T
6. while(! q.empty())
   {
   int u = q.pop()
   for(every adjacent v of u)
   {
   if(! visited[v])
   {
   ⇒ dist[v] = dist[u] + 1 ;
   visited[v] = T ; q.push(v);
   }
   }
7. Print dist[] ;



	0	1	2	3
visited	T	T	T	T

	0	1	2	3
dist[] =	0	1	2	1

q = ~~0~~ ~~1~~ ~~3~~ ~~2~~

u = 0   u = ~~0~~ ~~1~~ ~~3~~ ~~2~~

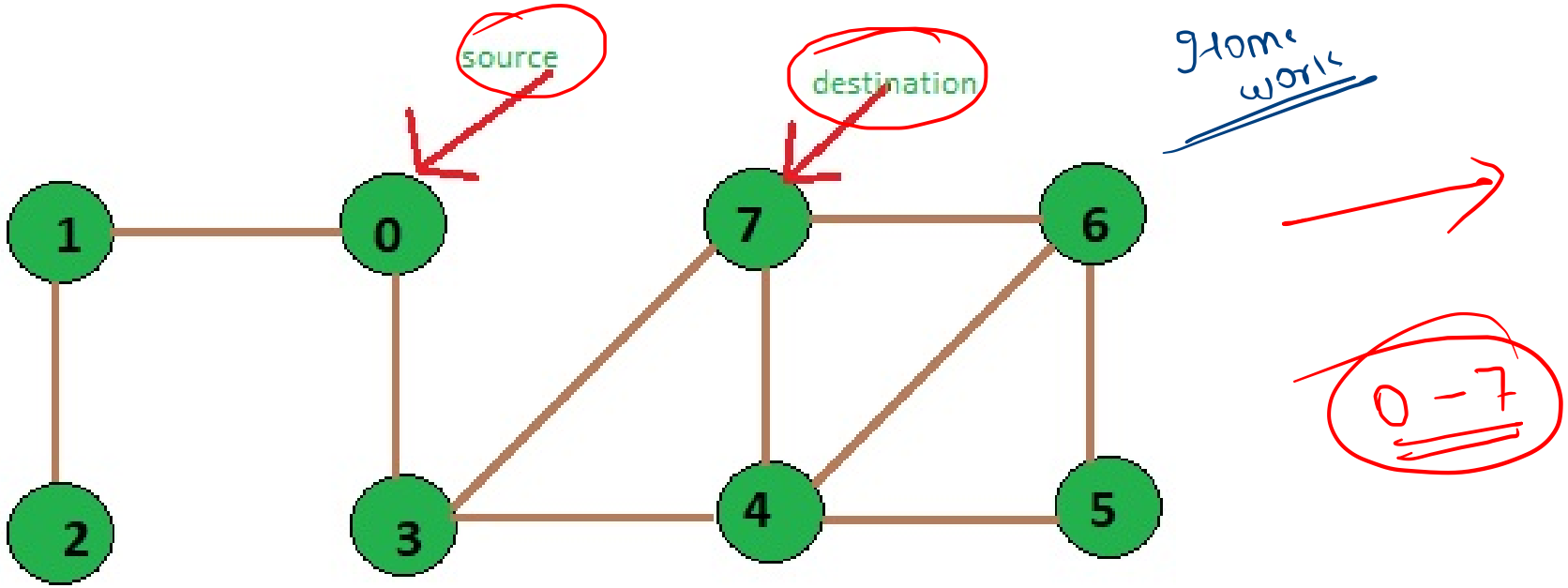
u =

distance 0 1 2 1

1. Initialise dist[v] = { ∞, ∞, ∞, ∞ }
2. dist[s] = 0
3. Initialise, visited[v] = { F, F, F, F }
4. create, queue, q ;
5. q.push(s) ; visited[s] = T
6. while(! q.empty())

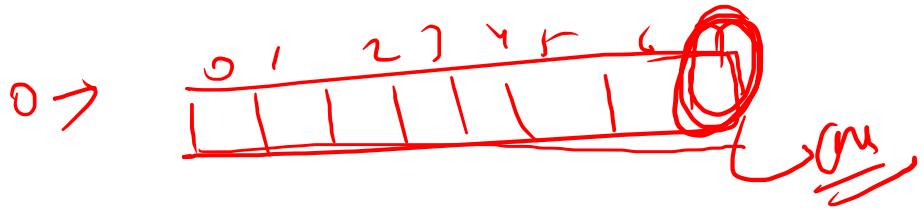
```
{
    int u = q.pop()
    for( every adjacent v of u )
    {
        if( ! visited[v] )
        {
            dist[v] = dist[u] + 1 ;
            visited[v] = T ; q.push(v);
        }
    }
```

7. Print dist[] ;



Home work

0 - 7

=> Do Dry Run of this Graph

=> click Pic and send it to my Telegram channel