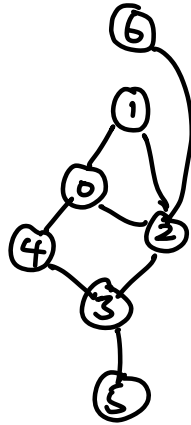


Graph Representation

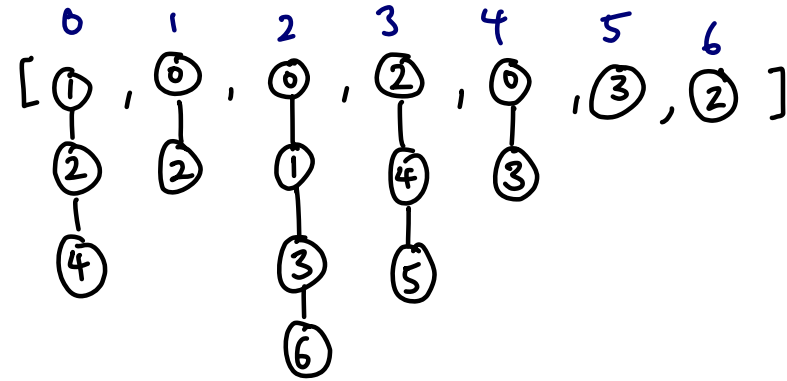
Adjacency Matrix

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



$O(V^2)$ space

Adjacency List



$O(V + E)$ space

Access / Find Element

find if 0 — 2 connected

Matrix

$$[0][2] = 1$$

Connected

List

$$[0] = [1, 2, 4]$$

traverse through list to check if 2 is i

Iterating All Edges

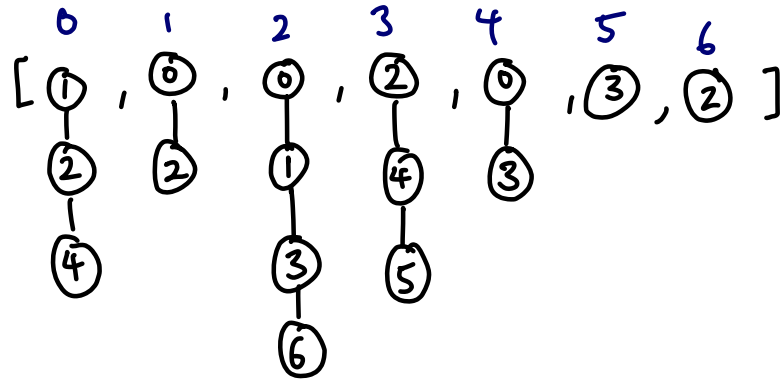
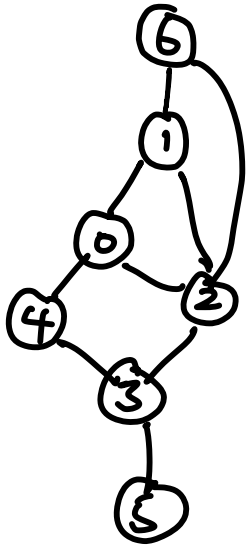
	0	1	2	3	4	5	6
0	0	1	1	0	1	0	0
1	1	0	1	0	0	0	0
2	1	1	0	1	0	0	1
3	0	0	1	0	1	1	0
4	1	0	0	1	0	0	0
5	0	0	0	1	0	0	0
6	0	0	1	0	0	0	0

Matrix

iterate all side.

Edge

iterate all of $O(E)$



iteration 1

current

0

Queue

1-2-4

added based on
connected vertex
of current
check if completed
before adding

Completed

add current after
it is completed.

iteration 2

current

1

Queue

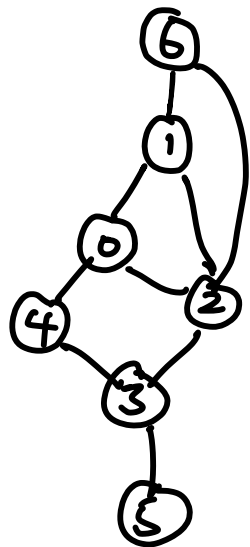
2-4-6

Completed

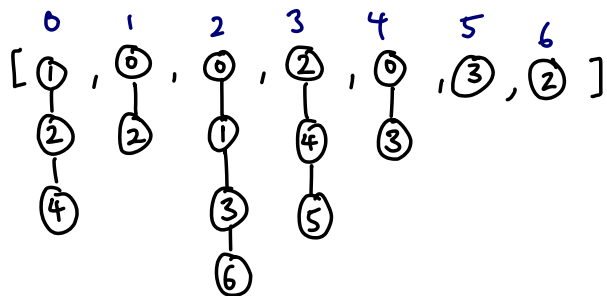
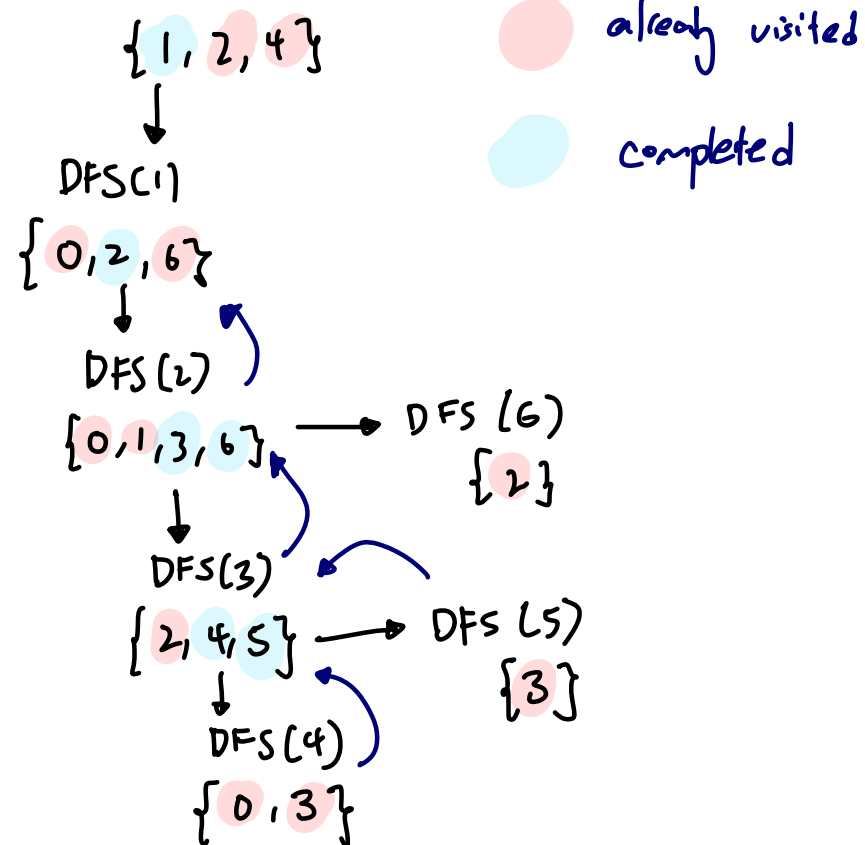
0

.... repeat

DFS



Adj List

Start \rightarrow DFS (0)

update visited to ensure not repeated

	0	1	2	3	4	5	6
visited	T	T	T	T	T	T	T