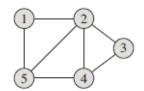
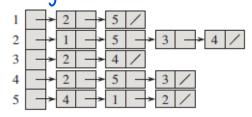
Graph Algorithms Adjacency list



Undirected graph



Adjacency matrix

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

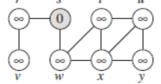
1 2	3
4-5	6

V	1 2 4 /
	2 5 /
1 3	3 - 6 - 5 /
T /T /T	4 2 /
1	5 4 /
4 ← 5 6 5	6 /
	

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

Directed graph

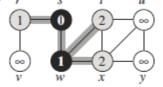
(Breadth First Search)

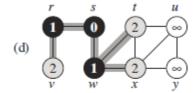


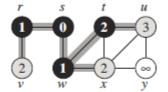
Q s

	r	S	t	u
(b)	(1)=	1	9	9
	ν	w	x	y

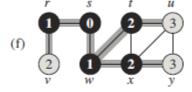
 $w \mid r$

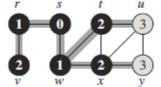




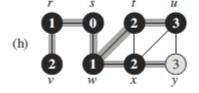


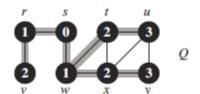
 $\begin{array}{c|cccc}
Q & x & v & u \\
\hline
& 2 & 2 & 3
\end{array}$

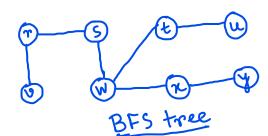




 $\begin{array}{c|cc}
Q & u & y \\
\hline
& 3 & 3
\end{array}$







```
BFS(G,s)
                for each vertex u \in G.V - \{s\}
O(N)
\begin{cases}
2 & u.color = W \\
3 & u.d = \infty \\
4 & u.\pi = NIL \\
5 & s.color = GRAY \\
6 & s.d = 0 \\
7 & s.\pi = NIL
\end{cases}
                      u.color = WHITE
            9 ENQUEUE(Q, s)
            10
                while Q \neq \emptyset
            11
                        u = \text{DEQUEUE}(Q)
            12
                        for each v \in G.Adj[u]
            13
                              if v.color == WHITE
            14
                                    v.color = GRAY
                                    v.d = u.d + 1
            15
            16
                                    v.\pi = u
            17
                                    ENQUEUE(Q, \nu)
            18
                        u.color = BLACK
```

ENQUEUE & DEQUEUE > O()

E > No. of edges present in G.

Total time spent in

scanning adjacency list

= O(E)

... O(V+E)