Depth First Search (DFS)

```
DFS(G)
1  for each vertex u ∈ G.V
2      u.color = WHITE
3      u.π = NIL
4  time = 0
5  for each vertex u ∈ G.V
6      if u.color == WHITE
7          DFS-VISIT(G, u)
```

O(V) global

O(V) exclusive of the call to DFS-VISIT

```
DFS-VISIT(G, u)
1   time = time + 1
2   u.d = time
3   u.color = GRAY
4   for each v ∈ G.Adj[u]
5       if v.color == WHITE
6           v.π = u
7           DFS-VISIT(G, v)
8   u.color = BLACK
9   time = time + 1
10  u.f = time
```

E

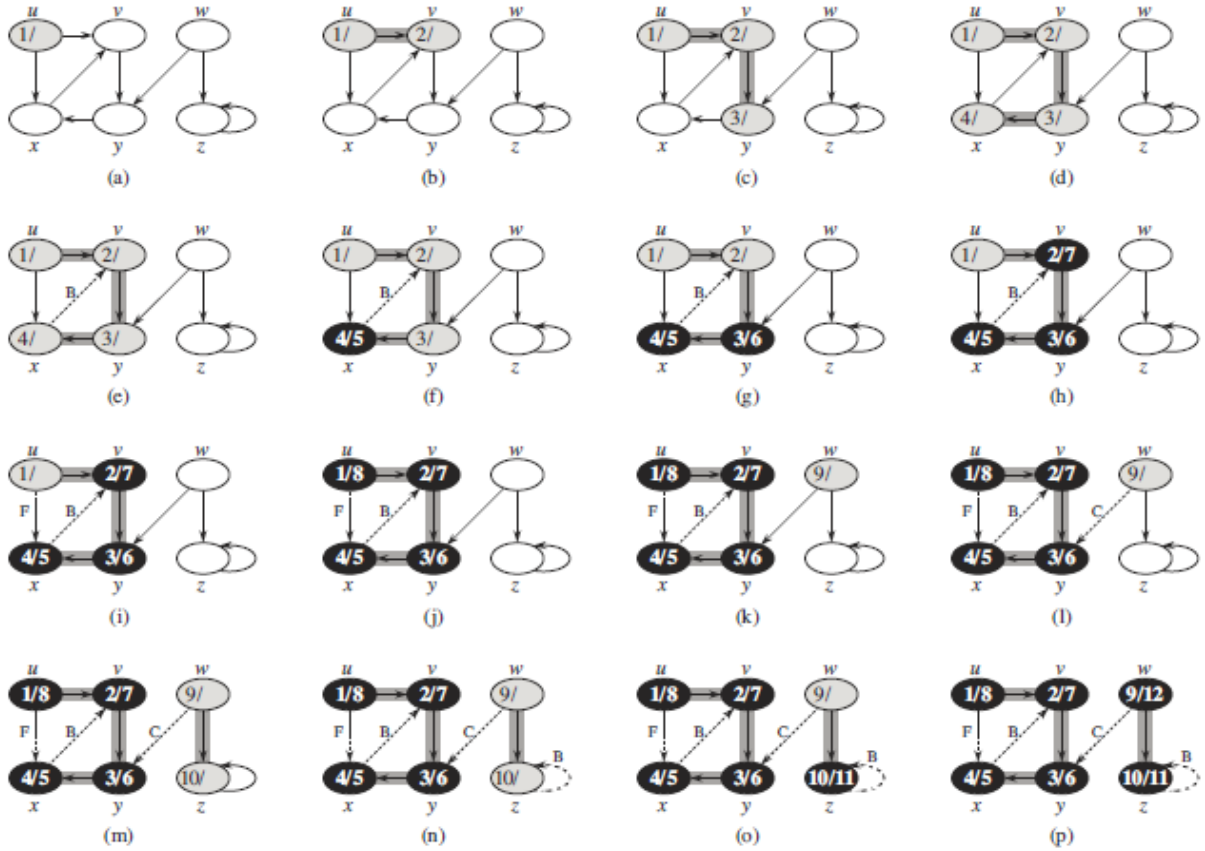
u.d = discovery time

u.f = finish time

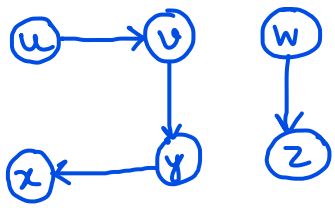
1 2 3 4 DFS-VISIT(G,u) → DFS-VISIT(G,v) → DFS-VISIT(G,y) → DFS-VISIT(G,x)

5 6 7 8

Time Complexity: V+E



DFS Forest:



Classification of edges in DFS:

T ***Tree edges*** are edges in the depth-first forest G_π. Edge (u, v) is a tree edge if v was first discovered by exploring edge (u, v).

B ***Back edges*** are those edges (u, v) connecting a vertex u to an ancestor v in a depth-first tree. We consider self-loops, which may occur in directed graphs, to be back edges.

F ***Forward edges*** are those nontree edges (u, v) connecting a vertex u to a descendant v in a depth-first tree.

C ***Cross edges*** are all other edges. They can go between vertices in the same depth-first tree, as long as one vertex is not an ancestor of the other, or they can go between vertices in different depth-first trees.