

Chapter 3. Decompositions of graphs

$$G = (V, E)$$

nodes V , edges E if directed, $e = (x, y)$ x to y

$$a_{ij} = \begin{cases} 0 & v_i \text{ to } v_j \\ 1 & \text{otherwise} \end{cases} \quad \text{표현}$$

Depth-First search

Figure 3.5 Depth-first search.

```

procedure dfs( $G$ )
  for all  $v \in V$ :
    visited( $v$ ) = false
  for all  $v \in V$ :
    if not visited( $v$ ): explore( $v$ )
  
```

Figure 3.3 Finding all nodes reachable from a particular node.

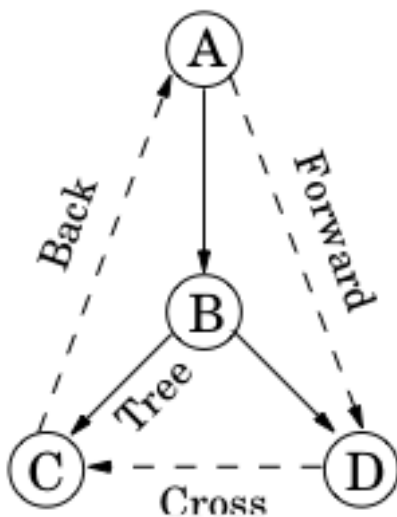
```

procedure explore( $G, v$ )
  Input:  $G = (V, E)$  is a graph;  $v \in V$ 
  Output: visited( $u$ ) is set to true for all nodes  $u$  reachable from  $v$ 

  visited( $v$ ) = true
  previsit( $v$ )
  for each edge  $(v, u) \in E$ :
    if not visited( $u$ ): explore( $u$ )
  postvisit( $v$ )
  
```

1. Run depth-first search on G^R .
2. Run the undirected connected components algorithm (from Section 3.2.3) on G , and during the depth-first search, process the vertices in decreasing order of their post numbers from step 1.

DFS tree



$pre(u) < pre(v) < post(v) < post(u)$
 forward : nonchild 밑으로 $[[]]$
 back : 위로 $[[]]$
 cross : 체크한 곳으로 $[][]$