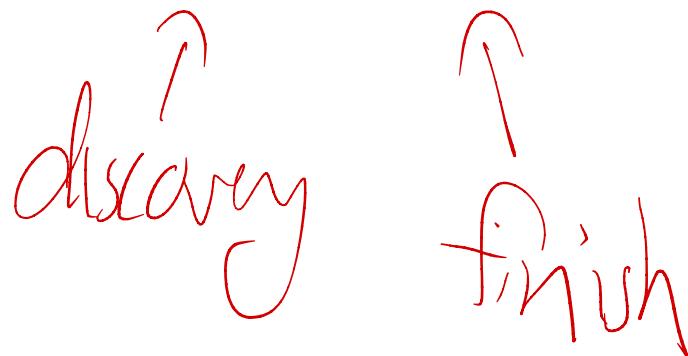


## Depth First Search

- More interesting than BFS
- Works for directed and undirected graphs. Example is for directed graphs.
- Time stamp nodes with discovery and finishing times.
- Lifetime: white,  $d(v)$ , grey,  $f(v)$ , black

discovery      finish



The image shows two handwritten words in red ink: 'discovery' on the left and 'finish' on the right. Above each word is a small red arrow pointing towards it, indicating the direction of the text. The handwriting is somewhat cursive and informal.

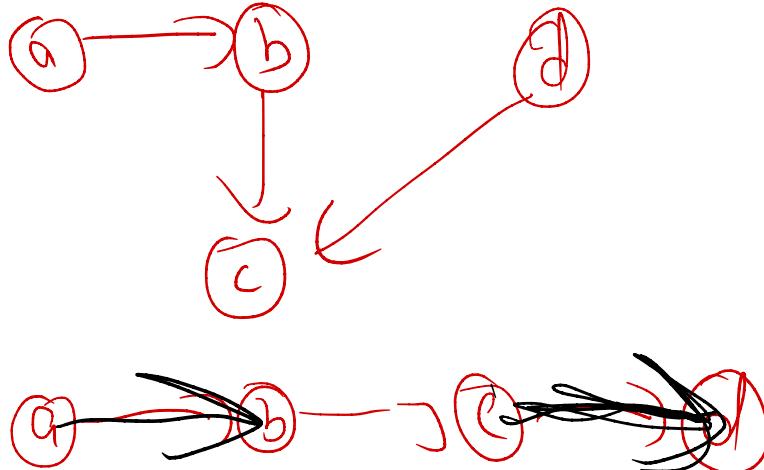
## Code

$DFS(G)$

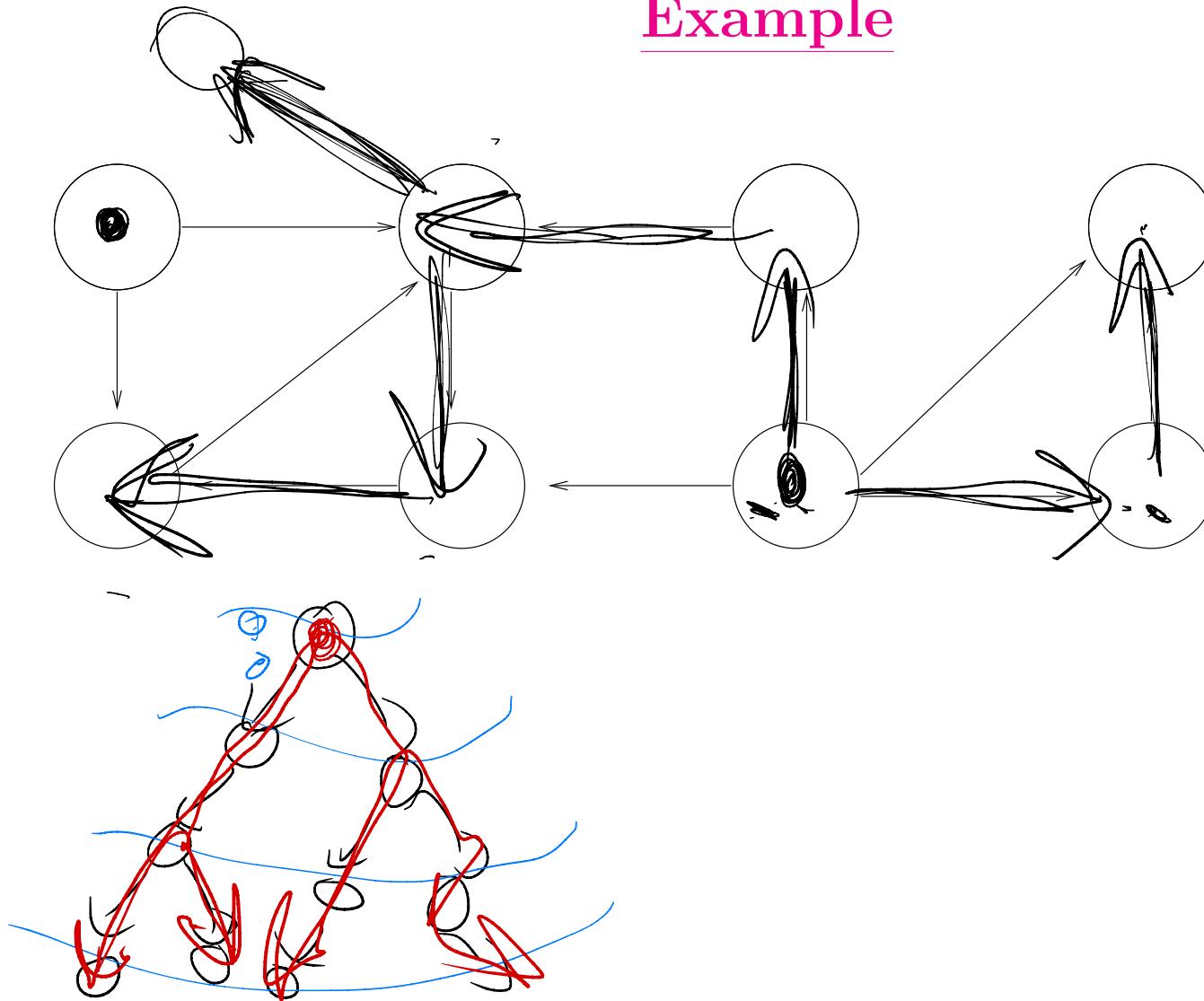
```
1  for each vertex  $u \in V[G]$ 
2      do  $color[u] \leftarrow \text{WHITE}$ 
3           $\pi[u] \leftarrow \text{NIL}$ 
4       $time \leftarrow 0$ 
5  for each vertex  $u \in V[G]$ 
6      do if  $color[u] = \text{WHITE}$ 
7          then DFS-VISIT( $u$ )
```

$n$  [ **DFS-Visit( $u$ )**

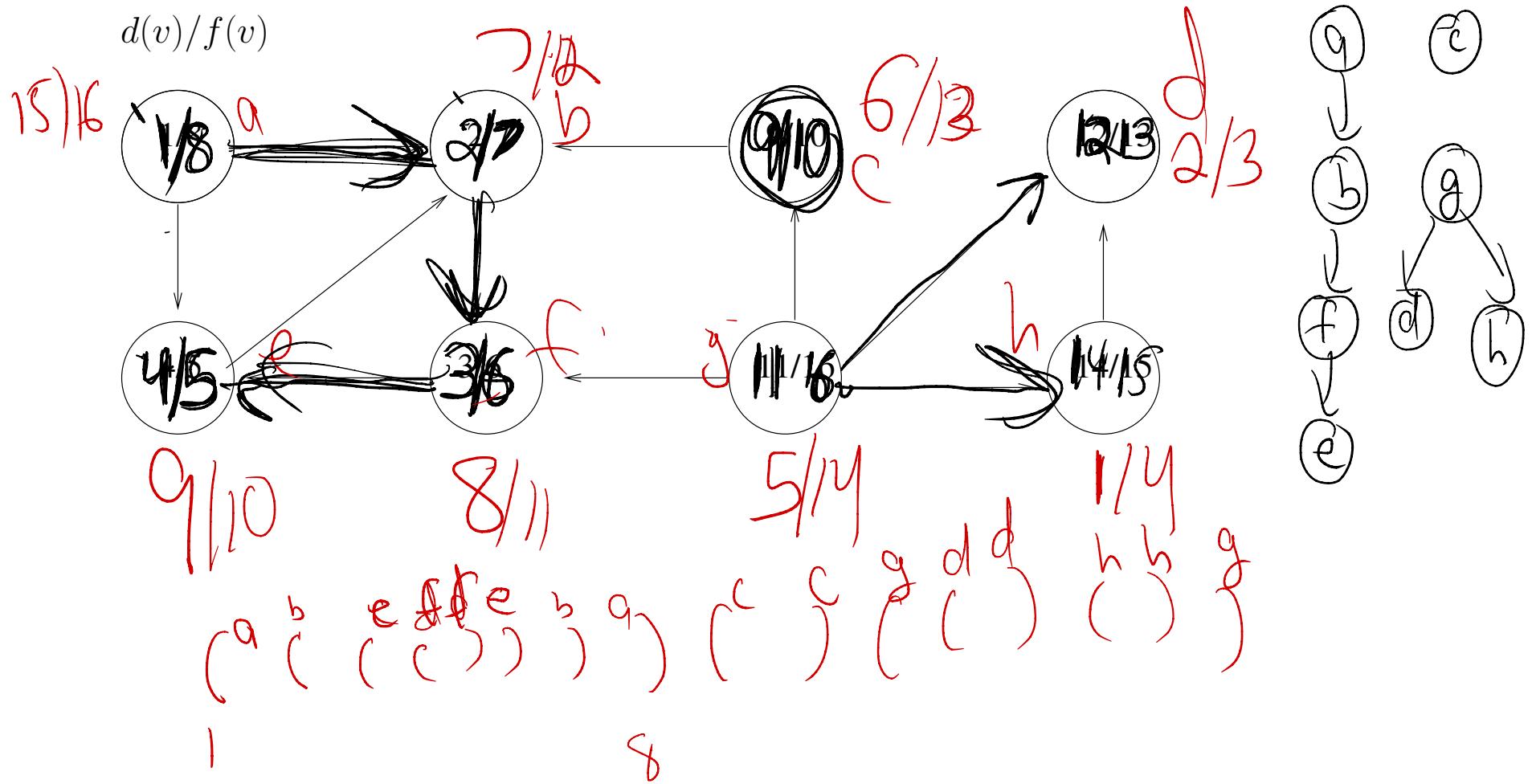
```
1   $color[u] \leftarrow \text{GRAY}$             $\triangleright$  White vertex  $u$  has just been discovered.
2   $time \leftarrow time + 1$ 
3   $d[u] \leftarrow time$ 
4  for each  $v \in Adj[u]$             $\triangleright$  Explore edge  $(u, v)$ .
5      do if  $color[v] = \text{WHITE}$ 
6          then  $\pi[v] \leftarrow u$ 
7              DFS-VISIT( $v$ )
8   $color[u] \leftarrow \text{BLACK}$           $\triangleright$  Blacken  $u$ ; it is finished.
9   $f[u] \leftarrow time \leftarrow time + 1$ 
```



## Example



## Labeled



# Structure

## Parenthesization

If we represent the discovery of vertex  $u$  with a left parenthesis “ $(u$ ” and represent its finishing by a right parenthesis “ $u)$ ”, then the history of discoveries and finishings makes a well-formed expression in the sense that the parentheses are properly nested.

**Parenthesis theorem** In any depth-first search of a (directed or undirected) graph  $G = (V, E)$ , for any two vertices  $u$  and  $v$ , exactly one of the following three conditions holds:

- the intervals  $[d[u], f[u]]$  and  $[d[v], f[v]]$  are entirely disjoint, and neither  $u$  nor  $v$  is a descendant of the other in the depth-first forest,
- the interval  $[d[u], f[u]]$  is contained entirely within the interval  $[d[v], f[v]]$ , and  $u$  is a descendant of  $v$  in a depth-first tree, or
- the interval  $[d[v], f[v]]$  is contained entirely within the interval  $[d[u], f[u]]$ , and  $v$  is a descendant of  $u$  in a depth-first tree.

## Nesting of descendants' intervals

Vertex  $v$  is a proper descendant of vertex  $u$  in the depth-first forest for a (directed or undirected) graph  $G$  if and only if  $d[u] < d[v] < f[v] < f[u]$ .

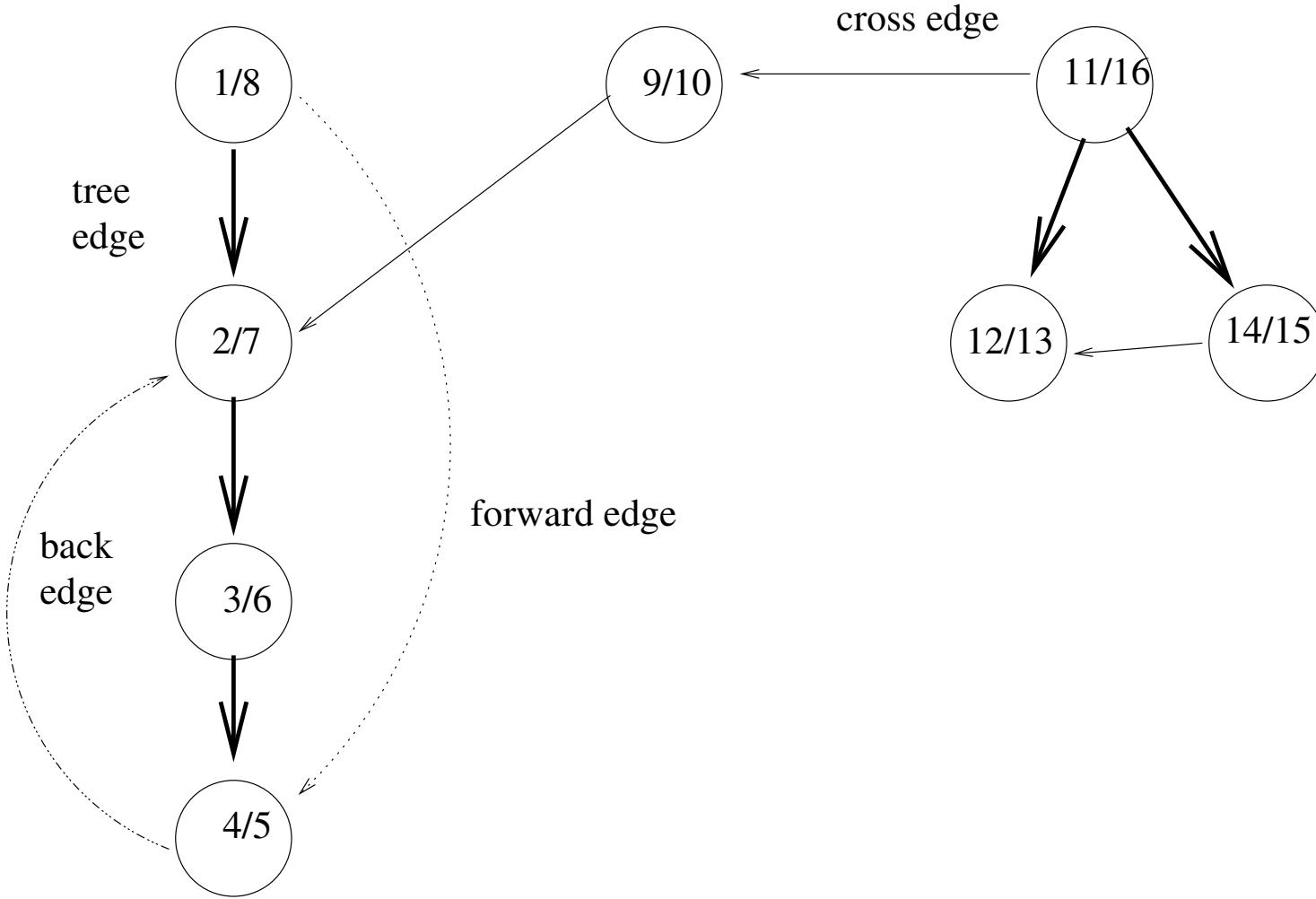
# More Structure

## White-path theorem

In a depth-first forest of a (directed or undirected) graph  $G = (V, E)$ , vertex  $v$  is a descendant of vertex  $u$  if and only if at the time  $d[u]$  that the search discovers  $u$ , vertex  $v$  can be reached from  $u$  along a path consisting entirely of white vertices

## Edge classification

1. **Tree edges** are edges in the depth-first forest  $G_\pi$ . Edge  $(u, v)$  is a tree edge if  $v$  was first discovered by exploring edge  $(u, v)$ .
2. **Back edges** are those edges  $(u, v)$  connecting a vertex  $u$  to an ancestor  $v$  in a depth-first tree. Self-loops, which may occur in directed graphs, are considered to be back edges.
3. **Forward edges** are those nontree edges  $(u, v)$  connecting a vertex  $u$  to a descendant  $v$  in a depth-first tree.
4. **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.



## Running Time:

```
1 for each  $u \in V$ 
2   do for each  $v \in \text{Adj}(v)$ 
3     do Something  $O(1)$  time
```

Each edge and vertex is processed once:

$$O(E + V) = O(E)$$

