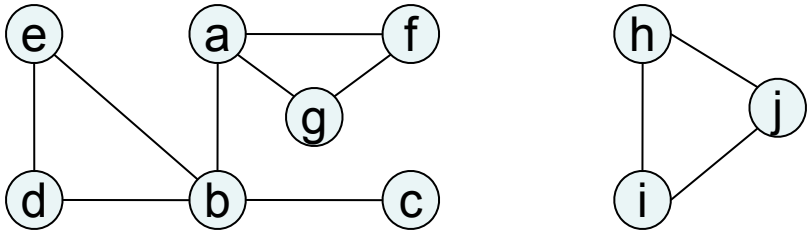
Lecture 5: Undirected connectivity

An undirected graph is *connected* if there is a path between any pair of nodes.



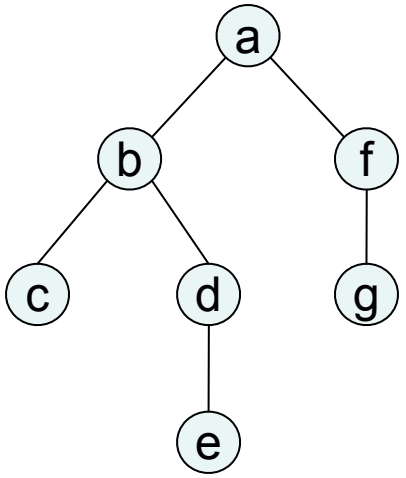
This graph has 2 *connected components*.

explore(G,v) returns the connected component containing v.
To explore the rest of the graph, restart explore() elsewhere.

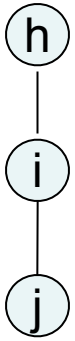
DFS decomposes an undirected graph into its connected components!

```
procedure dfs(G)
for all v in V:
   visited[v] = false
for all v in V:
   if not visited[v]:
        explore(G,v)
```

explore(G,a)



explore(G,h)



Running time analysis

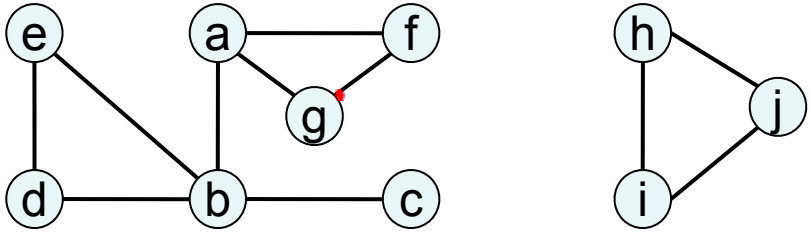
```
procedure explore(G,v)
visited[v] = true
for each edge (v,u) in E:
   if not visited[u]:
        explore(G,u)
```

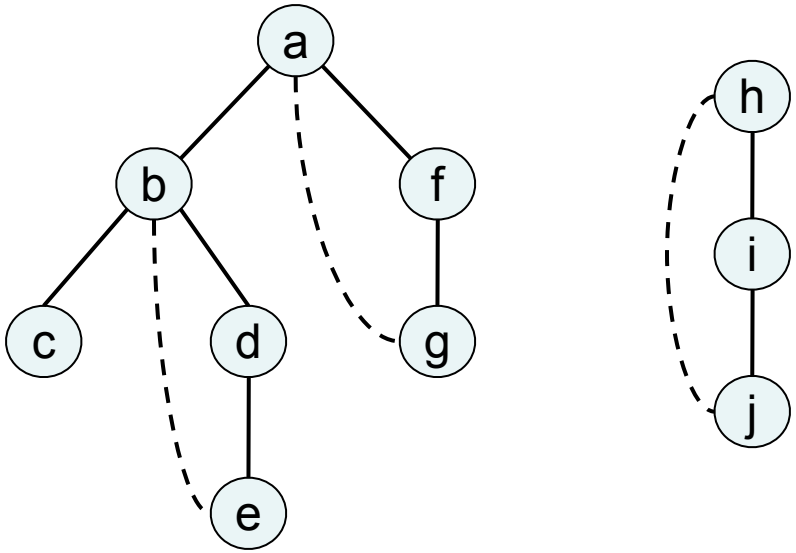
```
procedure dfs(G)
for all v in V:
   visited[v] = false
for all v in V:
   if not visited[v]:
        explore(G,v)
```

How long does dfs(G) take?

explore(G,v) is called exactly once for each node v.

DFS search forest





Terminology:
DFS search forest consisting of two *DFS search trees*

tree edge: traversed by DFS

back edge: not traversed (led to a node already visited)