Depth-First Search

- BFS: first discovered -> first explored
  - implemented with a queue
- DFS: last discovered -> first explored
  - implemented with a stack

Graph Search Recap

Algorithm keeps track of:

- `colour[v]`: white, grey, or black:
  - white: undiscovered
  - grey: discovered but not completely explored
  - black: explored
- `p[v]`: u iff v was discovered while exploring u
- `d[v]`: **time** when v was discovered (different from `d[v]` in BFS!)
- `f[v]`: time when exploration of v was finished

Depth First Search

Algorithm

```
def DFS(G):
    for each v in V:
        colour[v] = "white"
        p[v] = NIL
        d[v] = infinity
        f[v] = infinity
    global time = 0
    for each v in V:
```

```
        "..."
    "..."

def DFS_Explore(G, u):
    color[u] = "grey"
    time += 1
    d[u] = time
    for each edge (u, v) in E:
        if colour[v] = "white":
            p[v] = u
            DFS-Explore(G, v)
    colour[u] = "black"
    time += 1
    f[u] = time
```

Discovery forest

- Parenthesization: can tell if v is a descendent of u in the discovery forest if $d[u] < d[v] < f[v] < f[u]$
- In general:
  - cannot have $d[u] < d[v] < f[u] < f[v]$
  - if $(u, v) \in E$, then $d[v] < f[u]$
- forests are determined by starting vertex

Types of edges:

- (u, v) is a *tree edge* if $u = p[v]$
- (u, v) is a *forward edge* if u is an ancestor of v
  - $d[u] < d[v] < f[v] < f[d]$
- (u, v) is a *back edge* if u is a descendant of v
  - $d[v] < d[u] < f[u] < f[v]$

- (u, v) is a *cross edge* if u is neither an ancestor nor a descendent of v (aka if u and v are in different discovery trees)
  - $f[v] < d[u]$
  - cannot have $f[u] < d[v]$ because there is an edge from u to v, so if u is explored before v then u will discover v

White Path Theorem

For all graphs $G = (V, E)$ and all depth first searches on G, v becomes a descendant of u if and only if when u is discovered (at time $d[u]$) there is a path from u to v consisting entirely of white nodes.

Proof.

($\Leftarrow$) Suppose at $d[u]$ there is a white path from u to v.

Suppose not all nodes in that white path become descendants of u. Let z be the closest node to u in that path that does not become a descendant of u. Let w be the node before z in that path, then w becomes a descenant of u (or $w = u$). Then:

1. $d[u] < d[z]$ since z is white when u is discovered
2. $d[z] < f[w]$ since z is discovered while exploring w, because it is white and $(w, z) \in E$
3. $f[w] \leq f[u]$ since w is a descenant of u

(1), (2), (3) together show us that $d[u] < d[z] < f[u]$. It is impossible that $d[u] < d[z] < f[u] < f[z]$, so $d[u] < d[z] < f[z] < f[u]$, so z becomes a descendant of u, so we have reached a contradiction.

Thus, every node in the white path from u to v becomes a descendant of u.

($\Rightarrow$)

Corollary - If G has a cycle, then any DFS on G has a back edge

($\Leftarrow$) Suppose G has a cycle C

Let u be the first node in C that the DFS discovers, let v be the node before u in C. By white path theorem, v becomes a descendant of u (not necessarily with C part of the discovery tree). Then (v, u) is a back edge.