Design and Analysis of Algorithms

Graphs 3: Depth-First Search

Daniel Shannon

April 21st, 2022

3.4.2

Claim: A directed graph has a cycle *if and only if* its DFS reveals a back edge.
Proof:
Consider a directed graph with two nodes, A and B ($A \rightarrow B \rightarrow A$...), in a cycle. **Base Case:** *If we execute a DFS from node $N = 0$ (node A), we will explore node B. Since the only other option is A, which has been marked as explored, we will create a back edge to include the cycle.* If we start exploring at $N = 1$ (node B), A will be marked as explored, and by the same logic we will create a back edge to include the cycle from $B \Rightarrow A$. Here we can see the cyclic nature of back edges from $A \rightarrow B$ and $B \rightarrow A$ regardless of the starting node.

3.4.4

Claim: A directed graph has a cycle *if and only if* its DFS reveals a back edge. If a directed graph has a cycle, that meas the DFS tree will have a back edge.
Proof:
Consider the same graph $A \rightarrow B \rightarrow A$.... **Base Case:** A graph was explored from $N = 0$ (node A) to generate a DFS tree that has the structure $A \rightarrow B...A$ with a back edge from $B \rightarrow A$. This is cyclic in nature. Now consider a DFS tree generated from $N = 1$ (node B) with the structure $B \rightarrow A$ and a back edge to from $B \rightarrow A$. We can see that a DFS tree generated from a cyclic graph is reproducible regardless of the starting node.

3.4.6

Find the sources, sinks, and all possible linearizations.

1. **Sources** B
2. **Sinks** E,F
3. **Linearizations**
   $B \rightarrow A \rightarrow D \rightarrow C \rightarrow E$
   $B \rightarrow D \rightarrow A \rightarrow C \rightarrow E$

$B \to A \to D \to C \to F$
$B \to D \to A \to C \to F$

3.4.8

3.4.10

> If the explore procedure is called on a node in a sink SCC, then the set of nodes visited is exactly that SCC. Why?

A sink SCC means that there are no outgoing edges from that SCC node. So if there are multiple SCCs then you can *enter* an sink SCC with explore, but you cannot leave.

3.4.12

> If C and C' are SCCs, and there is an edge from a node in C to a node in C', then the highest post number in C is higher than the highest post number in C'. Why?

Explore increments the post count after the recursive call. Recall that the time it takes for the inner recursive call to complete will always be within the parent recursive call. Since explore explores all nodes in an C, and explore will eventually lead to C', we know that the post counter for C' will be encapsulated within C since we are still in the recursive call for C when exploring C'.