Design and Analysis of Algorithms I

Graph Primitives

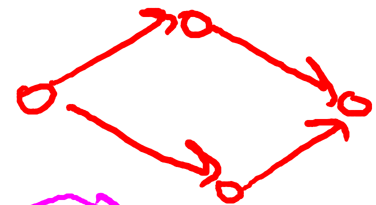
An O(m+n) Algorithm for Computing Strong Components

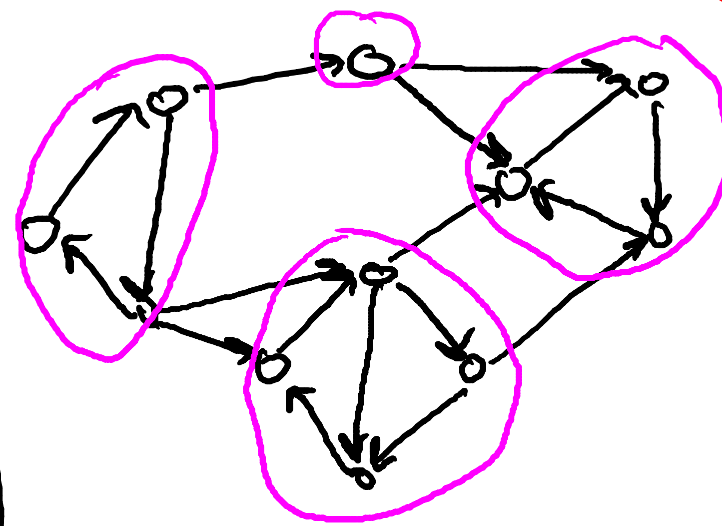
Strongly Connected Components

Formal Definition: the strongly connected components (SCCs) of a directed graph G are the equivalence classes of the relation

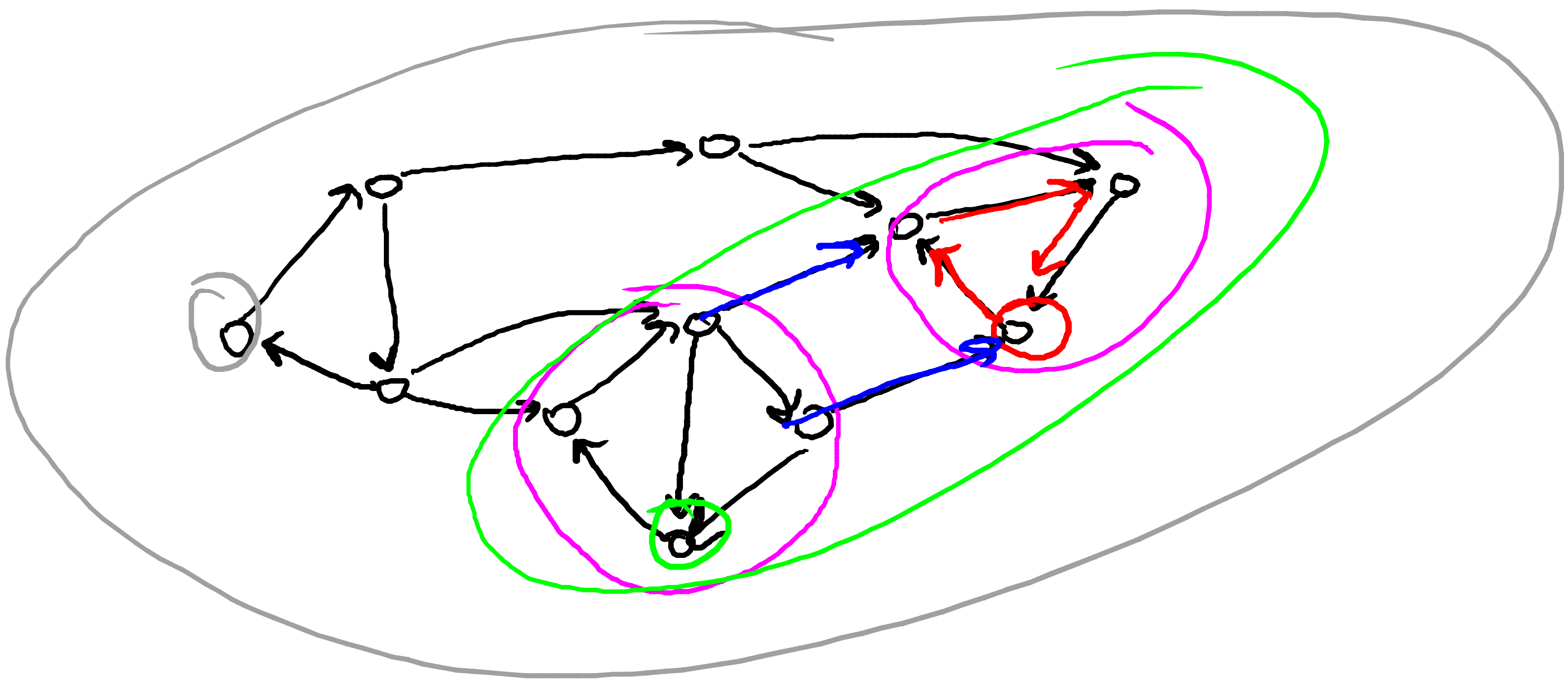
u~v ⟺ ∃ path u ⟶ v and a path v ⟶ u in G.

You check: ~ is an equivalence relation.





Why Depth-First Search?



Tim Roughgarden

Kosaraju's Two-Pass Algorithm

Theorem: Can compute the SCCs in $O(m+n)$ time.

Algorithm: (given directed graph G)

1. Let G^{rev} = G with all arcs reversed
2. run DFS-Loop on G^{rev} ← goal: compute "magical ordering" of nodes
   Let $f(v)$ = "finishing time" of each $v \in V$.
3. run DFS-Loop on G, ← goal: discover the SCCs one-by-one
   processing nodes in decreasing order of finishing times

[SCCs = nodes with the same "leader"]

DFS-Loop

DFS-Loop (graph G)

global variable t = 0 (for finishing times in 1st pass)
[# of nodes processed so far]

global variable s = NULL (for leaders in 2nd pass)
[current source vertex]

Assume nodes labelled 1 to n.

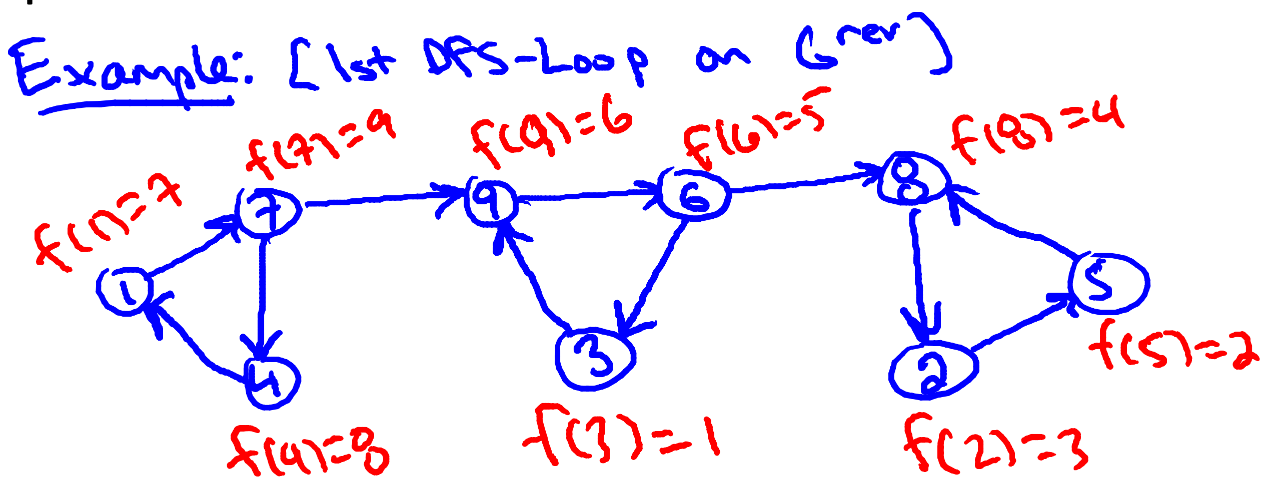
For i := n downto 1
- if i not yet explored
  - s := i
  - DFS (G, i)

DFS (graph G, node i) (for rest of DFS-Loop)
- mark i as explored
- set leader(i) := node s
- for each arc (i, j) ∈ G:
  - if j not yet explored:
    - DFS (G, j)
- t++
- set f(i) := t (i's finishing time)

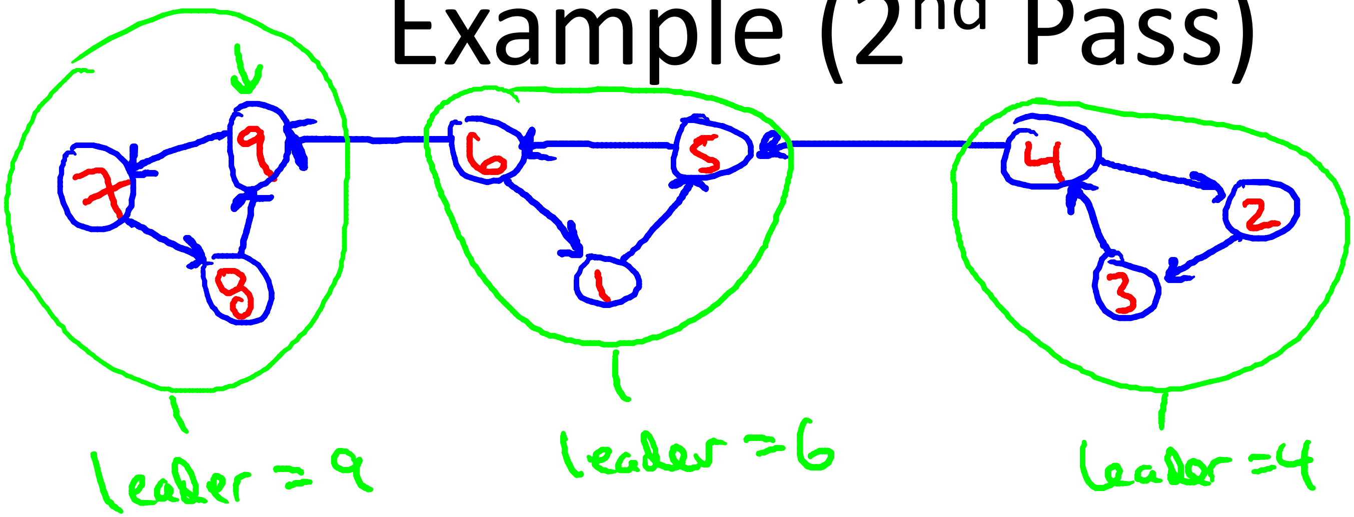
Tim Roughgarden

Only one of the following is a possible set of finishing times for the nodes 1,2,3,...,9, respectively, when the DFS-Loop subroutine is executed on the graph below. Which is it?

- ○ 9,8,7,6,5,4,3,2,1
- ○ 1,7,4,9,6,3,8,2,5
- ○ 1,7,9,6,8,2,5,3,4
- ○ 7,3,1,8,2,5,9,4,6



Example (2nd Pass)



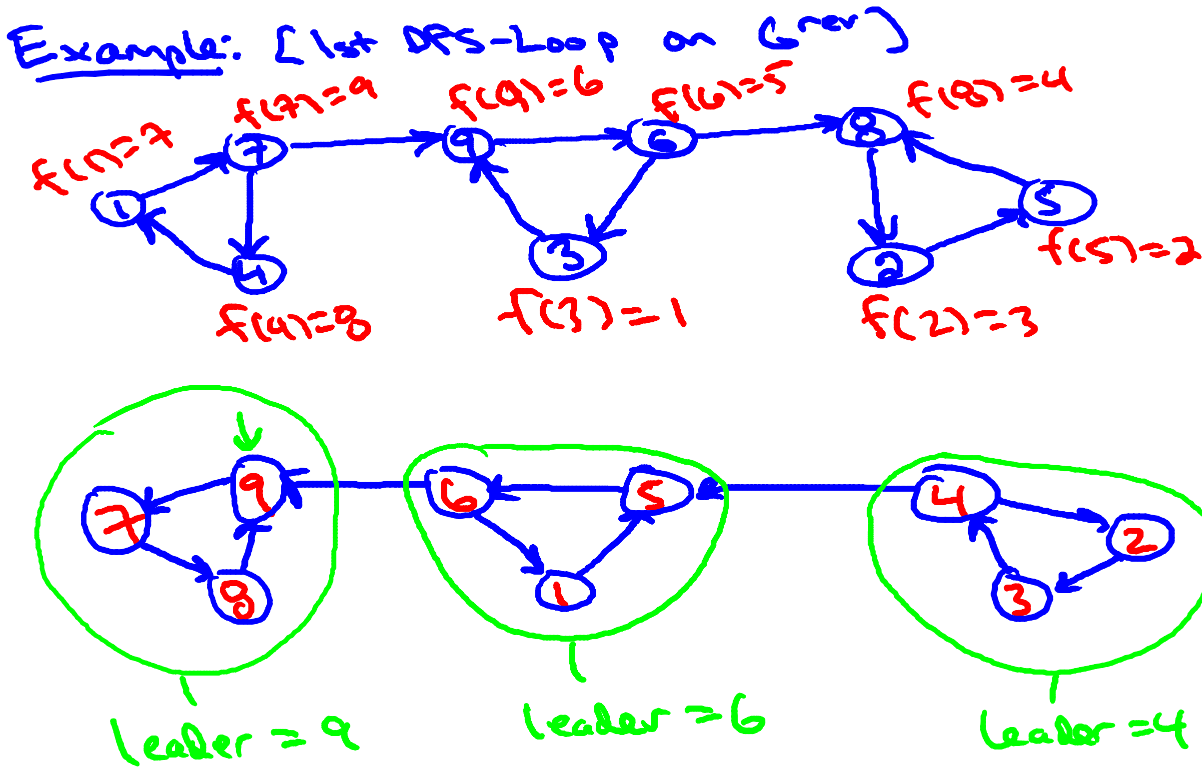
Running Time: $2 * DFS = O(m+n)$.

Design and Analysis of Algorithms I

Graph Primitives

Correctness of Kosaraju's Algorithm

Example Recap



Observation

Tim Roughgarden

What how are the SCC of the original graph G and its reversal G^{rev} related?

- In general, they are unrelated.
- Every SCC of G is contained in an SCC of G^{rev}, but the converse need not hold.
- Every SCC of G^{rev} is contained in an SCC of G, but the converse need not hold.
- They are exactly the same.

Key Lemma

Tim Roughgarden

Correctness Intuition

Tim Roughgarden

Proof of Key Lemma

Tim Roughgarden