

Strongly Connected Components

Defined in section 6-1. Used in variety of applications, including:

- Web Crawling and Search Engines: To identify groups of web pages that are mutually reachable
- Recommendation Systems: To identify closely connected groups of users or items
- Network Analysis (Social Networks, Communication Networks): To identify tightly-knit groups of users who interact frequently

The algorithm uses the transpose of G , that is $G^T = (V, E^T)$, where $E^T = \{(u, v) : (v, u) \in E\}$
(reversed edges in E)

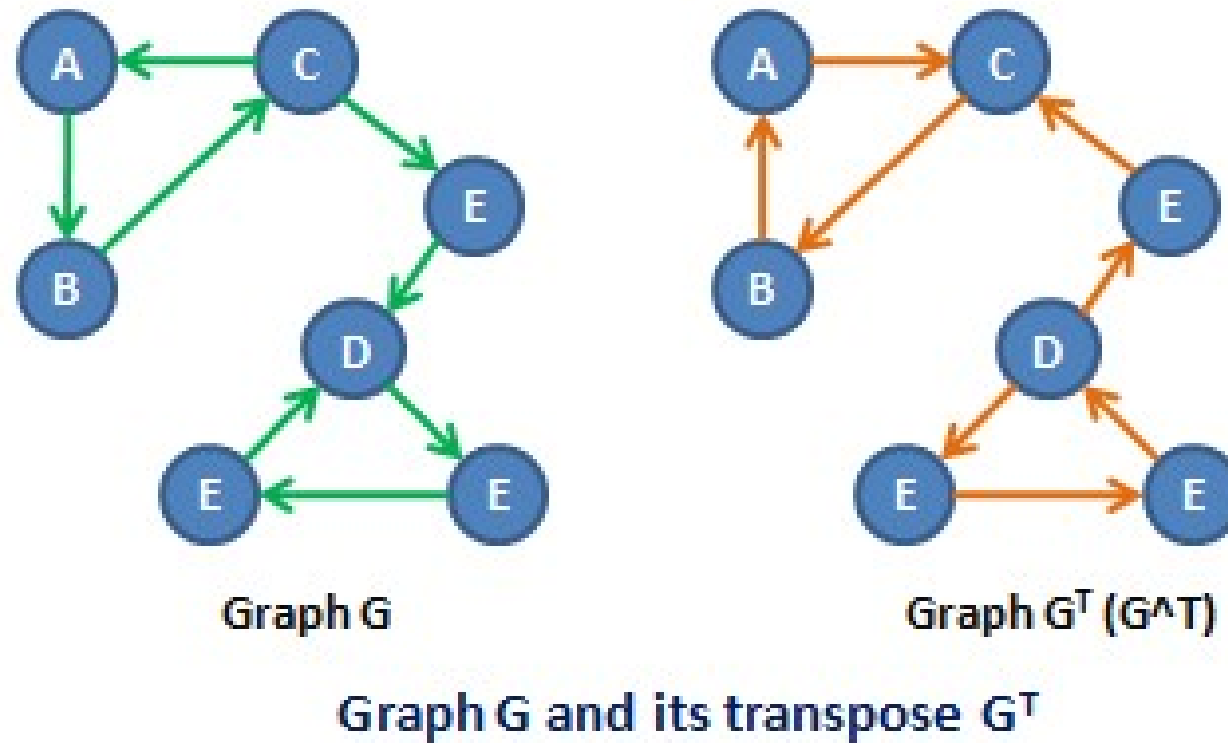


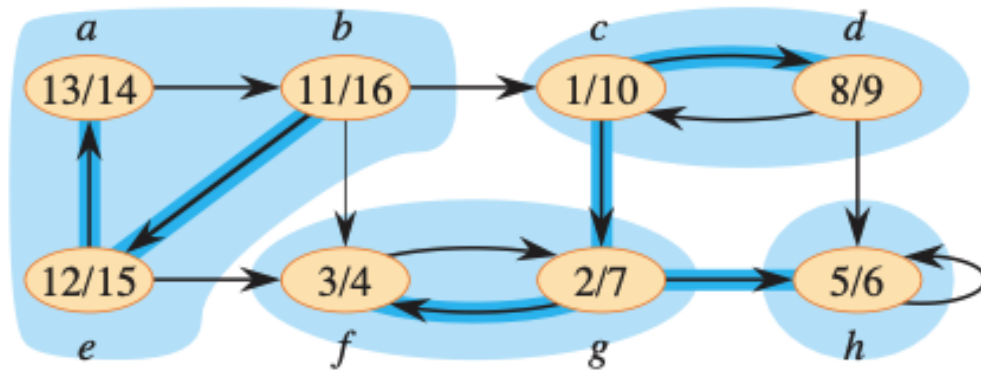
Figure 1: Graph Transpose (Wikipedia)

STRONGLY-CONNECTED-COMPONENTS (G)

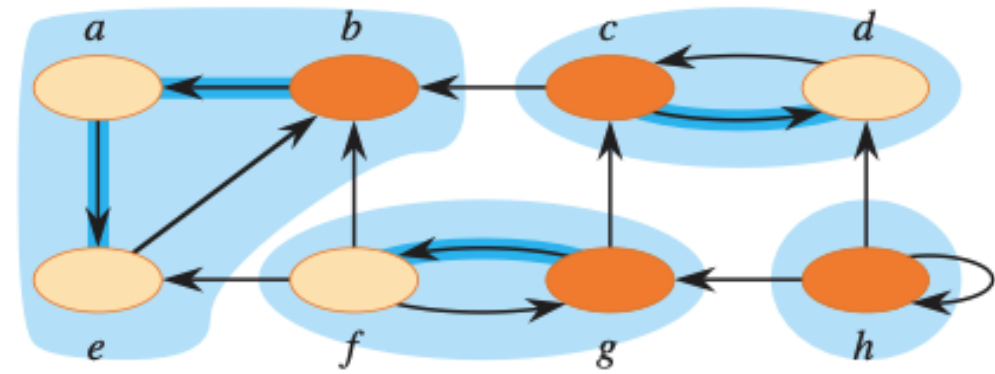
- 1 call DFS(G) to compute finish times $u.f$ for each vertex u
- 2 create G^T
- 3 call DFS(G^T), but in the main loop of DFS, consider the vertices in order of decreasing $u.f$ (as computed in line 1)
- 4 output the vertices of each tree in the depth-first forest formed in line 3 as a separate strongly connected component

Running time: $\Theta(V + E)$

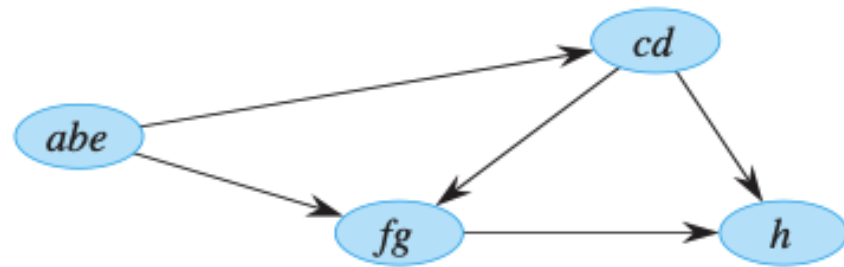
Example



(a)



(b)



(c)