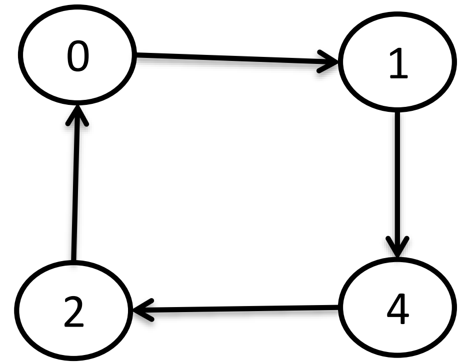
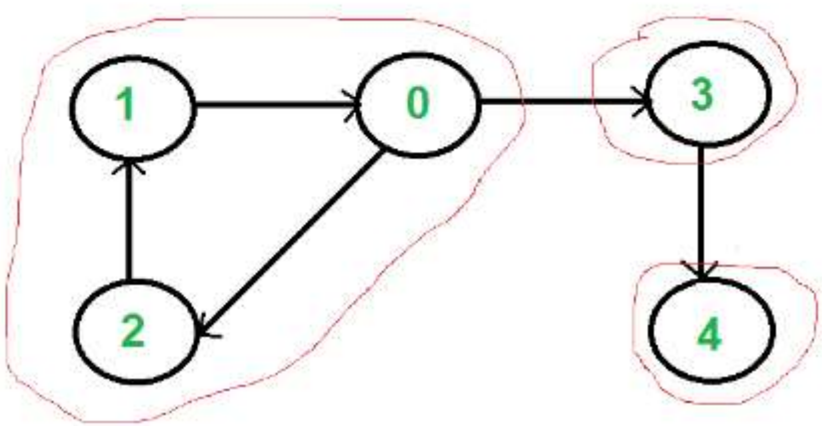
Strongly Connected Components

Strongly Connected Components

A directed graph is strongly connected if there is a path between all pairs of vertices.





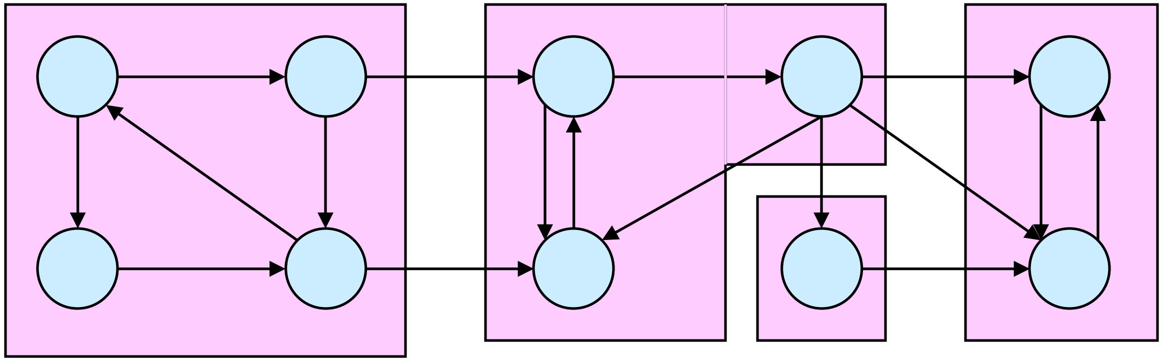
A strongly connected component (SCC) of a directed graph is a maximal strongly connected subgraph.

1-0-2
3
4

Strongly Connected Components

G is strongly connected if every pair (u, v) of vertices in G is reachable from one another.

A **strongly connected component** (*SCC*) of G is a maximal set of vertices $C \subseteq V$ such that for all $u, v \in C$, both $u \rightsquigarrow v$ and $v \rightsquigarrow u$ exist.

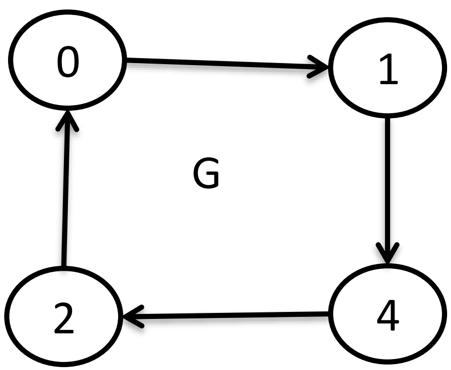


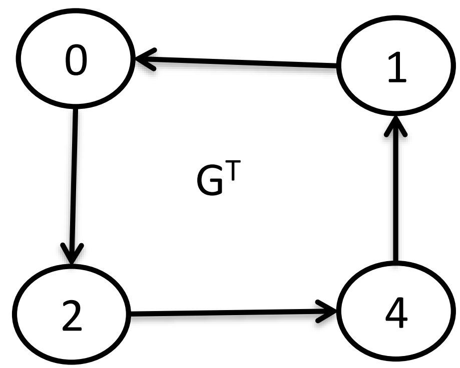
Kosaraju's Algorithm to determine SCCs

<u>SCC(G)</u>

1. call DFS(G) to compute finishing times $f[u]$ for all u
2. compute G^T
3. call DFS(G^T), but in the main loop, consider vertices in order of decreasing $f[u]$ (as computed in first DFS)
4. output the vertices in each tree of the depth-first forest formed in second DFS as a separate SCC

Time: $\Theta(V + E)$.





Kosaraju’s Algorithm to determine SCCs

1) Create an empty stack ‘S’
2) Do DFS traversal of a graph. In DFS traversal, after calling recursive DFS for adjacent vertices of a vertex, push the vertex to stack.
3) Reverse directions of all arcs to obtain the transpose graph.
4) One by one pop a vertex from S while S is not empty. Let the popped vertex be ‘v’. Take v as source and do DFS call on v. The DFS starting from v prints strongly connected component of v.

Example