The SCC Algorithm : A shorter Proof

The subject is the Strongly Connected Components algorithm of Kosaraju and Sharir [1978] that appears in section 22.5 of [CLRS], and its proof of correctness. For the sake of completeness, we repeat the algorithm below (slightly rephrased).

STRONGLY-CONNECTED-COMPONENTS (G)

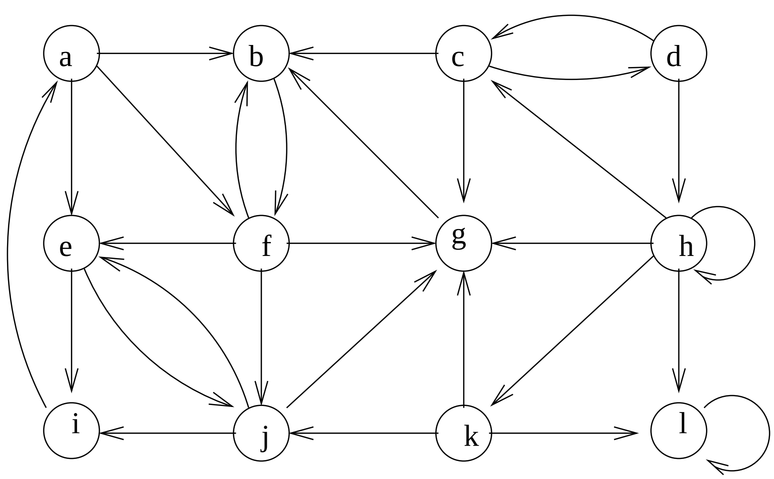
1. initialize stack S to empty, and call DFS(G) with the following modification:
   push vertices onto stack S in the order they finish their DFS-VISIT calls.
   That is, at the end of the procedure DFS-VISIT(u) add the statement
   $PUSH(u, S)$ (there is no need to compute $d[u]$ and $f[u]$ values explicitly).
2. construct the adjacency-list structure of G^T from that of G.
3. call DFS(G^T) with the following modification: initiate DFS-roots in
   stack-S-order, ie, in the main DFS algorithm instead of
   "**for** *each* $u \in V[G^T]$ **do if** $color[u] = white$ **then** DFS-VISIT(u)"
   perform the following:

   while $S \neq \emptyset$ **do**
   $u \leftarrow POP(S)$
   if $color[u] = white$ **then** DFS-VISIT(u)
   end {*while*}
4. each DFS-tree of step 3 (plus all edges between its vertices) forms an SCC.

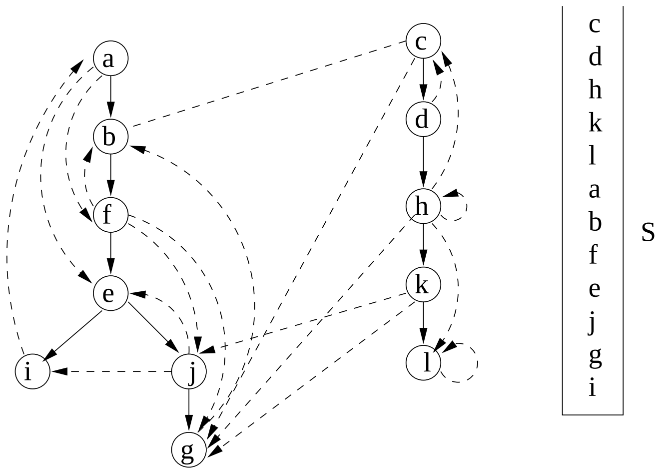
end

Example: Here is an example run of the algorithm:

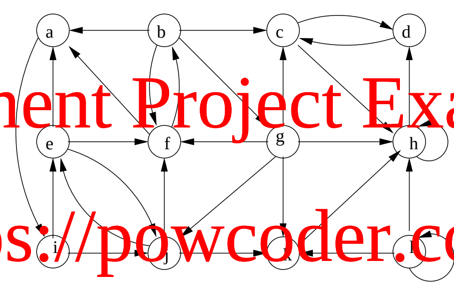
(a) Graph G:



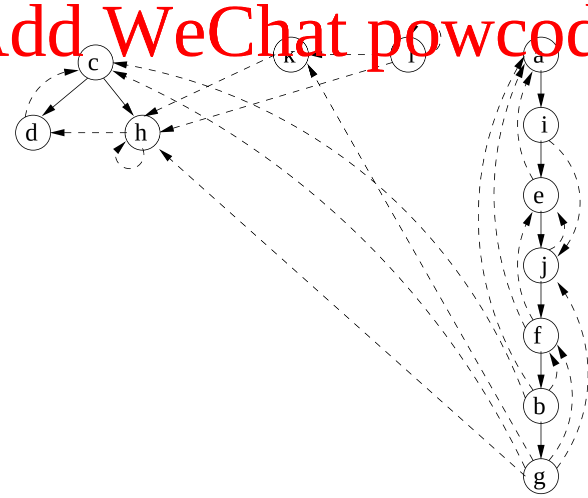
(b) Step 1: *DFS*(*G*)



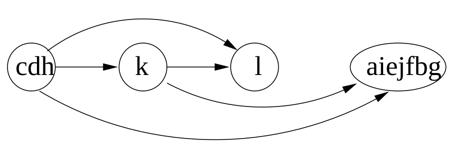
(c) Step 2: G^T



(d) Step 3: *DFS*(G^T)



(e) SCC-Component dag of *G*:



Proof of Correctness:

The correctness proof given in the book takes 4 pages. We give a shorter proof by using Theorem 23.6 (Parenthesis theorem), its Corollary 23.7, and Theorem 23.8 (White-path theorem). As we proceed, we will jump back and forth between the DFS(G) of step 1 and DFS(G^T) of step 3. The reader should keep the distinction in mind.

The fact that the algorithm takes $\Theta(V+E)$ time is obvious. Let us consider its correctness. First a few notations:

- Let $u \underset{G}{\rightsquigarrow} v$ mean vertex v is reachable from u in G, ie, there is a directed path in digraph G from u to v.
- Similarly, let $u \underset{G}{\leftrightsquigarrow} v$ denote $(u \underset{G}{\rightsquigarrow} v)\&(v \underset{G}{\rightsquigarrow} u)$, ie, vertices u and v are mutually reachable from each other in G.
- The $\underset{G}{\leftrightsquigarrow}$ relationship is reflexive, symmetric, and transitive. Hence for each pair of vertices u and v, $u \underset{G}{\leftrightsquigarrow} v$ iff $u \underset{G^T}{\leftrightsquigarrow} v$ iff u and v are in the same SCC, and the latter implies that no directed path between u and v ever leaves the vertices of their SCC.
- In what follows, for each vertex u, let $d[u]$ and $f[u]$ *always* denote the start and finish times of DFS-VISIT(u) with respect to the first DFS in step 1 of the algorithm (even while we refer to some events in the second DFS in step 3).

Consider an arbitrary SCC w.r.t. an arbitrary DFS. Let x be the first vertex of the SCC visited by the DFS. By the white-path theorem, since at the time x is visited, all vertices of the SCC are reachable from x, they will become DFS-descendents of x. Therefore, every vertex of any SCC is in a single DFS-tree. Thus, the vertices of each DFS-tree is the union of one or more SCC's.

What remains to prove is that each DFS-tree of step 3 consists of a single SCC, ie, every pair of vertices within a DFS-tree of step 3 are mutually reachable.

Let T be a DFS-tree of DFS(G^T) in step 3. We want to show that every pair of vertices in T are mutually reachable in G (and equivalently in G^T). Let vertex x denote the root of T. Because of the transitivity and symmetry of $\leftrightsquigarrow$, it suffices to show that for every vertex $v \in T$, $v \underset{G}{\leftrightsquigarrow} x$. Since x is the root of T, we conclude:

$$x \underset{G^T}{\rightsquigarrow} v \qquad \forall v \in T \tag{1}$$

Also, since x is the root of T, the stack-S-order selection during the DFS(G^T) implies that:

$$f[x] \geq f[v] \qquad \forall v \in T \tag{2}$$

That is $f[x]$ is maximum among vertices of T. (Once again, $f[.]$ is w.r.t. the DFS(G) of step 1, not step 3!)

Let vertex $y \in T$ have the minimum $d[.]$ value among the vertices in T. That is:

$$d[y] \leq d[v] \qquad \forall v \in T \tag{3}$$

We know that y is a descendent of x in T. Let P denote the path of tree edges in T (and in G^T) from x down to y. Existence of path P in G^T implies that $x \underset{G^T}{\rightsquigarrow} y$. The latter is equivalent to $y \underset{G}{\rightsquigarrow} x$. Because of eq. (3), during DFS(G) of step 1 we know that at time $d[y]$ every vertex on the path P is white. Therefore, by the white-path theorem, x is a DFS-descendent of y in

DFS(G) of step 1. Hence, because of the parenthesis theorem and its corollary, we conclude that $f[y] \geq f[x]$. This and eq. (2) imply that $f[x] = f[y]$, and hence $x = y$. From the latter and eqs. (2) and (3) we conclude:

$$d[x] \leq d[v] < f[v] \leq f[x] \qquad \forall v \in T \tag{4}$$

Because of the parenthesis theorem and eq. (4), we conclude that for every vertex $v \in T$, v is a DFS-descendent of x in DFS(G) of step 1. This implies that

$$x \underset{G}{\rightsquigarrow} v \qquad \forall v \in T \tag{5}$$

From eqs. (1) and (5) we conclude

$$x \underset{G}{\leftrightsquigarrow} v \qquad \forall v \in T \tag{6}$$

That is, all vertices of T are in the same SCC. □