Strongly Connected Components

**Input:** Directed graph $G = (V,E)$

$v, w \in V$

**Def** $v = w$ if $\exists$ path from $v$ to $w$ and $\exists$ path from $w$ to $v$

**Def** $R$ on a set $X$ is an equivalence relation

Reflexive: $vRv$

Symmetric: $vRw \Rightarrow wRv$

Transitive: $vRw \land wRx \Rightarrow vRx$

Claim $\equiv$ is an equivalence relation on $V$

**Def** The equivalence class are the strongly connected Components or strong Component
An Example:

Condensation
Alg: DFS(G)
1) \(\forall u \in V\), color\((u)\) \(\leftarrow\) white \(\ \); time \(\leftarrow 0\)
2) \(\forall u \in V\) if color\((u)\) = white then DFS-visit\((u)\)
   what order?

DFS-visit\((u)\)
1) \(\text{color\((u)\)} \leftarrow \text{gray} \); pop-time\((u)\) \(\leftarrow\) time ++
- 2) \(\forall v \in \text{Adj}(u)\)
   if color\((v)\) = white then DFS-Visit\((v)\)
3) \(\text{color\((u)\)} \leftarrow \text{black} \); pop-time\((u)\) \(\leftarrow\) time ++

note: dfs\((u)\) = push-time\((u)\)
Conceptually Simplify DFS

Add a super vertex R
1) Add new vertex R
2) Add new edges from R to all old vertices
3) Run DFS(R)

Fix the DFS

Def: For each SC the first vertex visited be SC is called the base node of SC.
1) In our case, DFS forest is a Tree rooted at $T$. $T$ is an ordered tree by push-time.

2) We can ignore forward edges! Why?

3) All cross edges go from right to left.

**Def** Let $T$ be our DFS Tree

$T_v = $ subtree rooted at $V_0$

**Lemma** If $T_v \cap T_w = \emptyset$ then

$pushtimes(T_v) < pushtimes(T_w)$ or $pushtimes(T_w) < pushtimes(T_v)$

**Note**: pushtimes are just preorder times for $T$. 
Lemma 2. If $b$ is the base for $X$, a SC, then
1) If $P$ is a path from $b$ to $v \in X \& w \in P$ then $w \in X$
2) If $v \in X$ then $v$ is a descendant of $b$ in $T$.

Proof
Claim 1. Suppose $w \in P$, a path from $b$ to $v$.
9) $bpw$ is a path from $b$ to $w$
10) The path from $w$ to $b$ will consist of
\begin{enumerate}
  \item $wpr$ in $P$.
  \item Path from $v$ to $b$ (by hypothesis).
\end{enumerate}
Proof of Claim 2

Will show: If $P$ is a path from $b$ to $v$ then $v(P) \leq T_b$

Proof by induction on $|P|$

$|P| = 1$ then done

Consider first edge $e = (x, y)$ of $P$ s.t. $y \notin T_b$

Then $e$ is either a cross edge or a back edge

In either case $\text{push}(y) < \text{push}(b)$

A contra!
Finding the Base Nodes

Note: SC are just the forest of T obtained by removing the edge into each base node.

Idea: Use something like lowpoint from Biconnected Components

Prob: Tree, Back, Forward edge are OK. Cross edges are a problem.

They can give wrong answer


diagram
Solution: Determine when a cross edge leaves the SC.

Remove each SC when base is popped.

Main Tool:

\[ u \in X \subseteq V \quad \text{X is SC,} \]

\[ \text{lowlink}(u) = \min \{ \text{push}(v) \mid \exists \text{Path of form } T^*[c,B]^+ \text{ & } v \in X \} \]

Claim: \( \text{lowlink}(u) \Rightarrow \text{push}(u) \) iff \( u \) is a base node.
Component(G)

1) Init(S: stack), time = 0 \forall v \text{ color}(v) \in \text{white}
2) If \text{color}(u) = \text{white} then Strong(u)

Strong(u)

1) \text{color}(u) \rightarrow \text{gray} \; ; \; \text{dfs}(u) \leftarrow \text{lowlink}(u) \leftarrow \text{time} + t \; ; \; \text{push}(u, S)

3) \forall v \in \text{Adj}(u)
   \text{if} \; \text{color}(v) = \text{white} \text{ then }
   \text{strong}(v) \; ; \; \text{ll}(u) = \min \{\text{ll}(u), \text{ll}(v)\}
   \text{else if} \; \lbrack v \in S \rbrack
   \text{then} \; \text{ll}(u) \leftarrow \min \{\text{ll}(u), \text{dfs}(v)\}

3) \text{if} \; \text{ll}(u) = \text{dfs}(u) \; \text{[u is a base vertex]}
   \text{then while} \lbrack \text{top}(s) \neq u \rbrack \; \text{pop}(s) \; ; \; \text{pop}(s)

end Strong