

Algorithmus DFS-SZK (* zerlegt den Digraphen $G = (V, E)$ in starke ZK *)

Eingabe: Digraph $G = (V, E)$ durch Inzidenzlisten L .

procedure suche (v : Knoten) ;

begin

markiere v BESUCHT;

tief[v] := dfnr[v] := zähler;

zähler := zähler + 1;

lege v auf den Keller K ; auf_Keller[v] := **true**;

for jeden Knoten w aus $L[v]$ **do**

if w ist nicht BESUCHT

then suche(w);

 tief[v] := min{tief[v], tief[w]}

else if dfnr[w] < dfnr[v] **and** auf_Keller[w]

then tief[v] := min{tief[v], dfnr[w]}

done;



if tief[v] = dfnr[v]

then repeat

 nimm obersten Knoten x vom Keller K und gib x aus;

 auf_Keller[x] := **false**

until $x = v$;

 schreibe „Ende einer starken ZK“;

fi

end

begin (* Hauptprogramm *)

 zähler := 1; K := empty_stack;

for alle $v \in V$ **do**

 markiere v als nicht BESUCHT;

 auf_Keller[v] := **false**

done;

while es gibt Knoten $r \in V$ mit r nicht BESUCHT **do**

suche(r);

end (* Hauptprogramm *)