Homework 5: DFS and SCC

Nicholai L'Esperance

12/6/2021

1 Given the following graph:

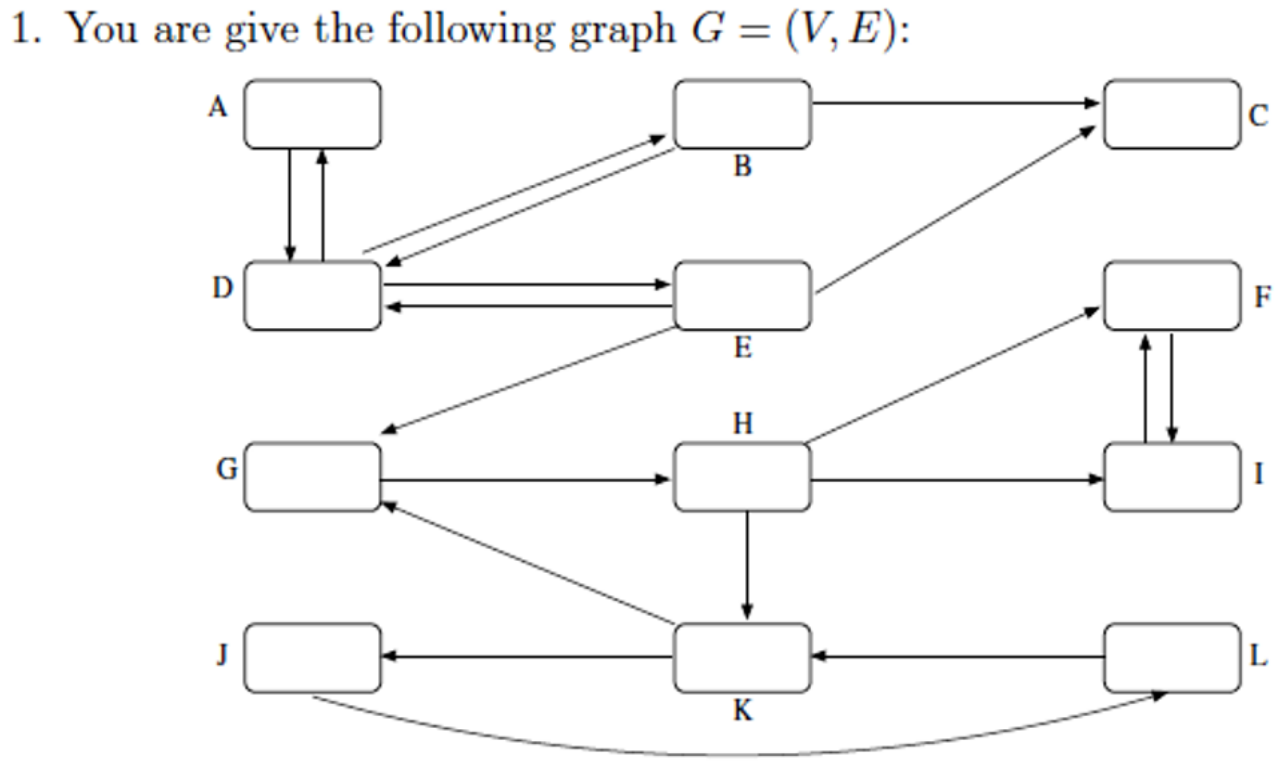

Figure 1: Original graph

a) Perform a DFS on the nodes in increasing order (A...L) and provide discovery (u.d) and finish (u.f) for every vertex.

Node	Start	Finish
A	0	23
B	2	5
C	3	4
D	1	22
E	6	21
F	9	12
G	7	20
H	8	19
I	10	11
J	14	17
K	13	18
L	15	16

i	j	Edge Type
A	*D*	*Tree*
B	*C*	*Tree*
B	*D*	*Back*
D	*A*	*Back*
D	*B*	*Tree*
D	*E*	*Tree*
E	*C*	*Cross*
E	*D*	*Back*
E	*G*	*Tree*
F	*I*	*Tree*
G	*H*	*Tree*
H	*F*	*Tree*
H	*I*	*Forward*
H	*K*	*Tree*
I	*F*	*Back*
J	*L*	*Tree*
K	*G*	*Back*
K	*J*	*Tree*
L	*K*	*Back*

The results of the edges are shown in the image below, with tree edges in green, and all the black edges are labeled forward, cross, or back.

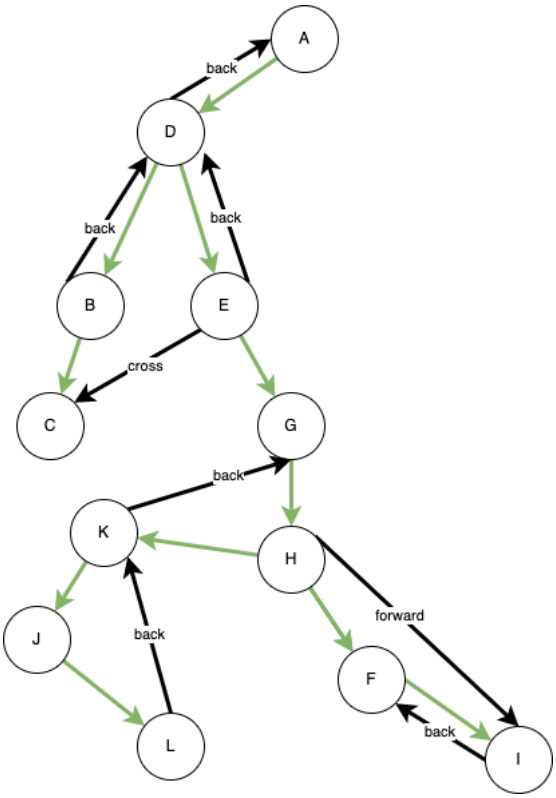

Figure 2: Tree diagram of DFS for nodes in increasing alphabetical order.

B) Perform a DFS on the nodes in decreasing order (L...A) and provide discovery (u.d) and finish (u.f) for every vertex.

Node	Start	Finish
A	20	21
B	16	19
C	17	18
D	15	22
E	14	23
F	7	8
G	4	11
H	5	10
I	6	9
J	2	3
K	1	12
L	0	13

i	j	Edge Type
L	*K*	*Tree*
K	*J*	*Tree*
J	*L*	*Back*
K	*G*	*Tree*
G	*H*	*Tree*
H	*K*	*Back*
H	*I*	*Tree*
I	*F*	*Tree*
F	*I*	*Back*
H	*F*	*Forward*
E	*G*	*Cross*
E	*D*	*Tree*
D	*E*	*Back*
B	*A*	*Back*
B	*C*	*Tree*
E	*C*	*Forward*
D	*A*	*Tree*
A	*D*	*Back*
D	*B*	*Tree*

The results of the edges are shown in the image below, with tree edges in green, and all the black edges are labeled forward, cross, or back.

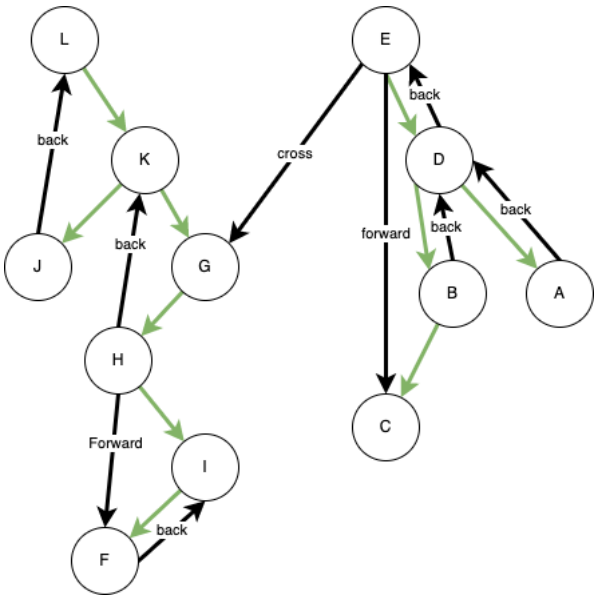

Figure 3: Tree diagram of DFS for nodes in decreasing alphabetical order.

C) Use the results from part a and the DFS algorithm to discover strongly connected components (SCCs). Give the discovery and finish times for the second run.

For ease, the first thing I did was draw out the graph in reverse. This can be used in both parts c) and d).

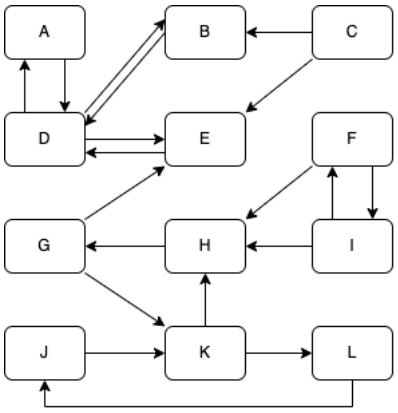

Figure 4: Transpose of original graph from figure 1.

To find the SCCs, we will run DFS on this graph, starting on nodes in part a) in decreasing order of the finish times. That order is:

$$a, d, e, g, h, k, j, l, f, i, b, c$$

Node	**Start**	**Finish**
A	**0**	7
B	2	3
C	**22**	23
D	1	6
E	4	5
F	**18**	21
G	**8**	17
H	10	11
I	19	20
J	13	14
K	9	16
L	12	15

In the table, I've highlighted every "jump" we had to make. This gives us the following strongly conencted components:

$$< \{A, B, D, E\},\ \{G, H, J, K, L\},\ \{F, I\},\ \{C\} >$$

The SCC graph is shown below, in figure 5.

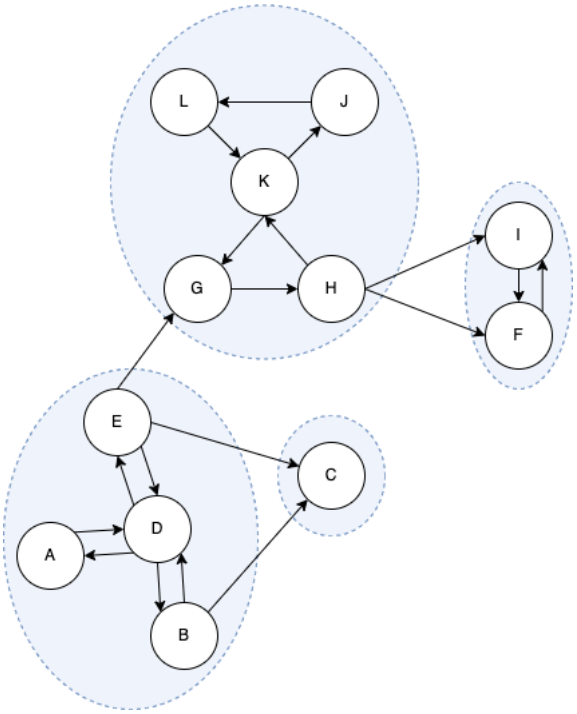

Figure 5: SCC graph

D) Use the results from part b and the DFS algorithm to discover strongly connected components (SCCs). Give the discovery and finish times for the second run.

To find the SCCs, we will run DFS on this graph, starting on nodes in part a) in decreasing order of the finish times. That order is:

$$e, d, a, b, c, l, k, g, h, i, f, j$$

Node	Start	Finish
A	4	5
B	2	3
C	**8**	9
D	1	6
E	**0**	7
F	21	22
G	14	15
H	13	16
I	**20**	23
J	11	18
K	12	17
L	**10**	19

In the table, I've highlighted every "jump" we had to make. This gives us the following strongly conencted components. **As expected, this matches our results from part c. The SCC graph is thus the same, so please refer to figure 5.**

$$< \{A, B, D, E\},\ \{G, H, J, K, L\},\ \{F, I\},\ \{C\} >$$