

# DFS on directed graph & Kosaraju's algorithm

Asked 6 years, 8 months ago    Active 6 years, 8 months ago    Viewed 2k times

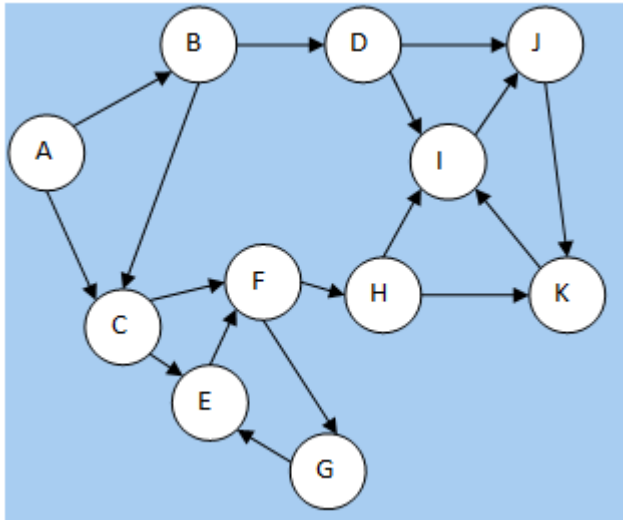
4

1

I'm having trouble to understand Kosaraju's algorithm for finding the strongly connected components of a directed graph. Here's what I have on my notebook (I'm a student :D):

1. Start from an arbitrary vertex (label it with #1) and perform a DFS. When you can't go any further, label the last visited vertex with #2, and start another DFS (skipping vertices already labeled), and so on.
2. Transpose the graph.
3. Do DFS starting from each vertex in reverse order, those vertices which end visited after each DFS belong to the same SCC.

I have this example:



And after the first step starting from E, the labels are:

1. E
2. G
3. K
4. J
5. I
6. H
7. F
8. C
9. D
10. B
11. A

So here comes the thing: Is there a difference for DFS in directed/undirected graphs? I did a mental test of the first step on my mind ignoring the arrows (just like it was undirected) and only got correct #1 for E (of course) and #2 for G, but #3 fell onto J, not K. So I thought maybe I should respect the arrows, and did a DFS considering that, but after the first pass starting from E, I can't go anywhere from G (which is #2), so I'm stuck there.

Is there anything about DFS on directed graphs that I'm not aware of? I've been taught DFS only on undirected graphs!

[algorithm](#) [graph](#) [depth-first-search](#) [directed-graph](#) [Edit tags](#)

edited Feb 5 '14 at 1:30  
 [Bernhard Barker](#)  
49.5k 13 78 118

asked Feb 4 '14 at 23:26  
 [user3272994](#)  
43 3

## 2 Answers

Active	Oldest	Votes
--------	--------	-------

1

Your second step is incomplete. See [Wikipedia](#):

Kosaraju's algorithm works as follows:

- Let G be a directed graph and S be an empty stack.
- While S does not contain all vertices:
  - Choose an arbitrary vertex v not in S. Perform a depth-first search starting at v. **Each time that depth-first search finishes expanding a vertex u, push u onto S.**
- Reverse the directions of all arcs to obtain the transpose graph.
- While S is nonempty:
  - Pop the top vertex v from S. Perform a depth-first search starting at v in the transpose graph. The set of visited vertices will give the strongly connected component containing v; record this and remove all these vertices from the graph G and the stack S. Equivalently, breadth-first search (BFS) can be used instead of depth-first search.

So you shouldn't only do something with the last vertex and first vertices, but with *each* vertex in the DFS.

Also note that you should be [backtracking](#) - when you can't go further, you go to the previous vertex and continue from there.

And no, you can't treat it as an undirected graph - the direction of the edges matter significantly.

So, starting from E, you'd, for example, go F, then G, then back to F, then H, then K, then I, then J, then back to I, K, H, F, and finally E, having pushed all visited vertices onto the stack.


answered Feb 5 '14 at 1:37  
 [Bernhard Barker](#)  
49.5k 13 78 118

- Oh!, I had completely misunderstood that! Now I'm OK. Also, the picture I uploaded has some missing edges, but nevermind. With that graph, and starting from E, I would do: E, F, G, back to F, H, I, J, K, back to J, back to I, back to H, back to F, but then what? how do I go to C? I have no edges to C from E or F... – [user3272994](#) Feb 5 '14 at 23:00
- Then you go back to E and, since that's the first vertex, and it doesn't have unexplored neighbours, we're done, so do another DFS from another arbitrary vertex (skipping all vertices already visited). – [Bernhard Barker](#) Feb 5 '14 at 23:19


0

This post is hidden. It was [deleted](#) 6 years ago by [animuson](#).

Oh!, I had completely misunderstood that! Now I'm OK. Also, the picture I uploaded has some missing edges, but nevermind. With that graph, and starting from E, I would do: E, F, G, back to F, H, I, J, K, back to J, back to I, back to H, back to F, but then what? how do I go to C? I have no edges to C

 [user3272994](#)

43 3

 This isn't really an answer to the question you asked, so it might be a good idea to edit your question to include this information. – [templatetypedef](#) Feb 5 '14 at 22:51

comments disabled on deleted / locked posts / reviews