By Anando Zaman

Exercise 4.2.31: Describe a linear time algo for computing the strong component containing a given vertex v. On the basis of that algorithm, describe a simple $O(V^2)$ algorithm for computing the strong components of a digraph.

Solution:

- Recall a strongly connected component has a path from $u \rightarrow v$ and from $v \rightarrow u$

- We cab use Kosaraju-Shariars algorithm which uses DFS twice to compute the strongly connected components. The running time of DFS is linear at $O(V + E)$ using adjacency list graph. V is the number of vertices and E is the number of edges in the graph.

- Since this algorithm uses DFS twice, it is linear-time complexity.

The steps to do this include:

1. Run DFS on reverse Graph to compute reverse post order that will be used in 2nd DFS to compute the SCC(Strongly connected component).

2. Run DFS on original graph, considering vertices in order given by reverse post order DFS in step/phase 1.

The question also asks to compute $O(V^2)$ complexity solution. To do this, we can use an adjacency matrix instead of adjacency list as finding adjacenct nodes take far longer and more memory. We use the same algorithm as shown above with 2 instances of DFS on the adjacency matrix. This will result in$O(V^2)$

1