Topological Sort

Consider the following directed graph.

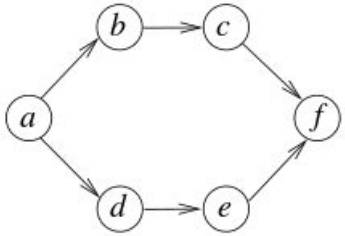
The number of different topological orderings of the vertices of the graph is

(A) 1

(B) 2

(C) 4

(D) 6



Correct ans: D)

Explanation:

- a-b-c-d-e-f
- a-d-e-b-c-f
- a-b-d-c-e-f
- a-d-b-c-e-f
- a-b-d-e-c-f
- a-d-b-e-c-f

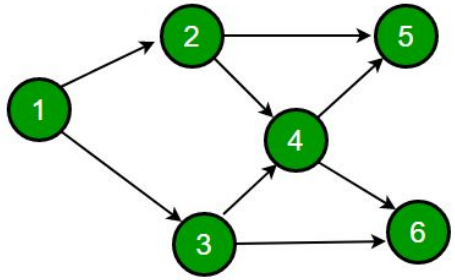
Consider the DAG with Consider V = {1, 2, 3, 4, 5, 6}, shown below. Which of the following is NOT a topological ordering?

(A) 1 2 3 4 5 6

(B) 1 3 2 4 5 6

(C) 1 3 2 4 6 5

(D) 3 2 4 1 6 5



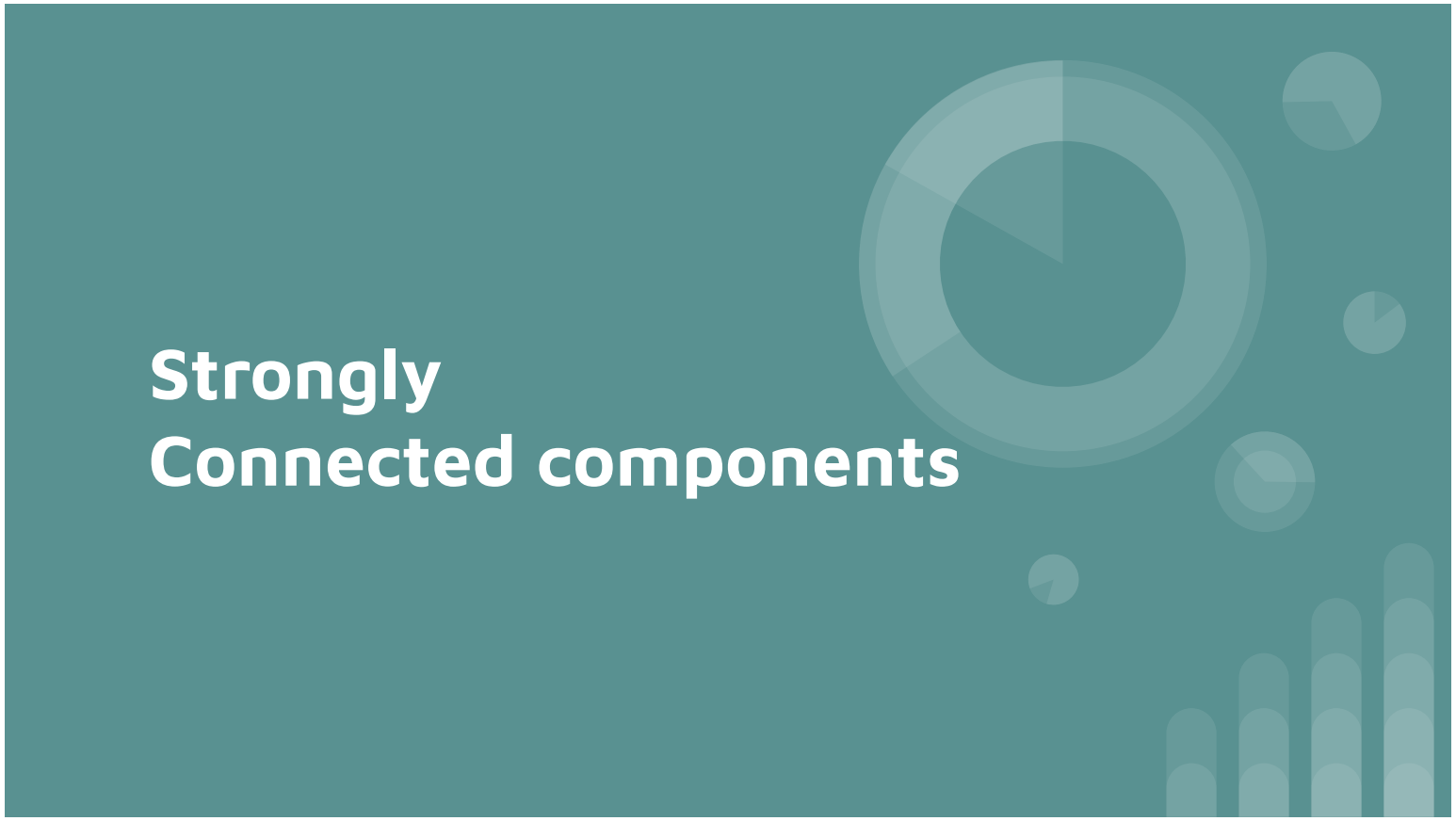
Correct ans: D)

Explanation:

In option D, 1 appears after 2 and 3 which is not possible in Topological Sorting.

In the given DAG it is directly visible that there is an outgoing edge from vertex 1 to vertex 2 and 3 hence 2 and 3 cannot come before vertex 1 so clearly option D is incorrect topological sort.

Strongly Connected components



Q1. There are cities and flight connections. Your task is to check if you can travel from any city to any other city using the available flights. In case not, name any pair of such cities.

Expected Time Complexity : O(n + m)

Example 1:
n = 4, m = 4,
Edges: [1,2], [2,3], [3,4], [4,1]

Answer: Yes

Example 2:
n = 4, m = 5,
Edges: [1, 2], [2, 3], [3, 1], [1, 4], [3, 4]

Answer: No, 4 2

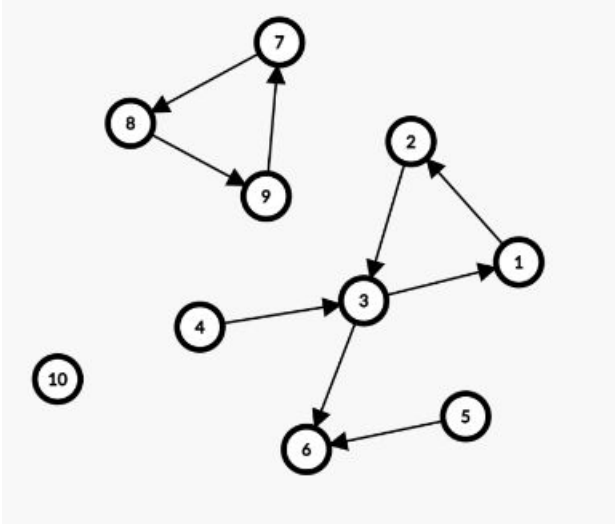
Q2. There are cities and flight connections. Your task is to determine the minimum number flights you need to add such that all nodes should be either directly or indirectly connected to city

Expected Time Complexity : O(n + m)

Example:
n = 10, m = 9, src = 10
Edges:
[1 2], [2 3], [3 1], [4 3], [5 6], [3 6], [7 8], [8 9], [9 7]

Answer: 3

Explanation: one possible answer is,
connect [10, 8], [9, 4], [3, 5]



Questions for practice

- What is a strongly connected component in a directed graph? How is it different from a weakly connected component?
- Explain Tarjan's algorithm for finding strongly connected components in a directed graph.
- How does Kosaraju's algorithm work for finding strongly connected components in a directed graph?
- Can you provide an example of a graph and demonstrate how to find its strongly connected components using Tarjan's algorithm?
- Discuss the time complexity of Tarjan's algorithm and Kosaraju's algorithm for finding strongly connected components.

Questions for practice

- Is it possible to have a directed graph with no strongly connected components? Justify your answer.
- How can the concept of strongly connected components be useful in real-world applications, such as social networks or web page ranking?
- Are strongly connected components unique in a given directed graph, or can there be multiple distinct SCCs with the same set of vertices?