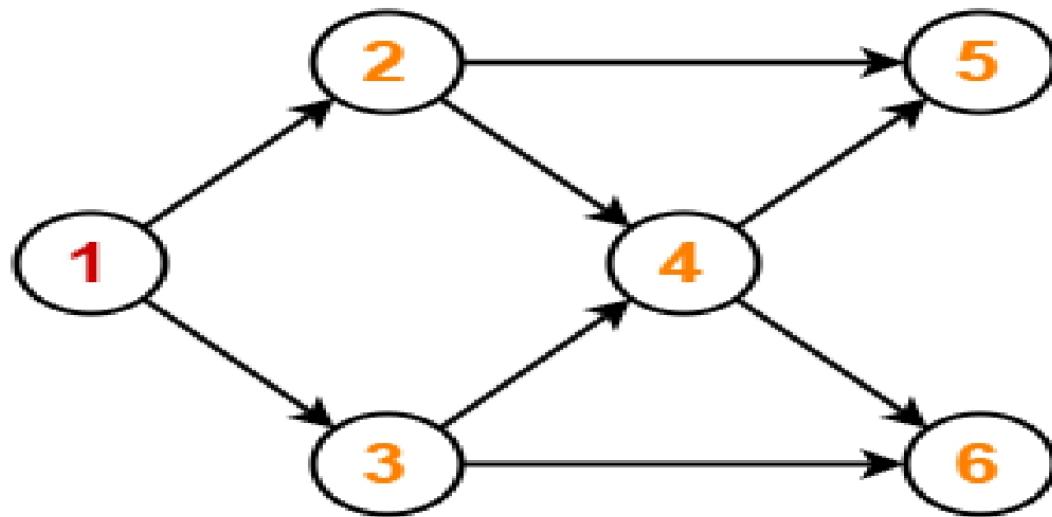


Topological Sorting

Topological Sorting

- Topological sorting for Directed Acyclic Graph (DAG) is a linear ordering of vertices such that for every directed edge (u,v) , vertex u comes before v in the ordering.
- A topological sort of a graph is an ordering of its vertices along a horizontal line so that all directed edges go from left to right.
- If the graph contains a cycle, then no linear ordering is possible.
- Topological Sorting for a graph is not possible if the graph is not a DAG.

Topological Sorting



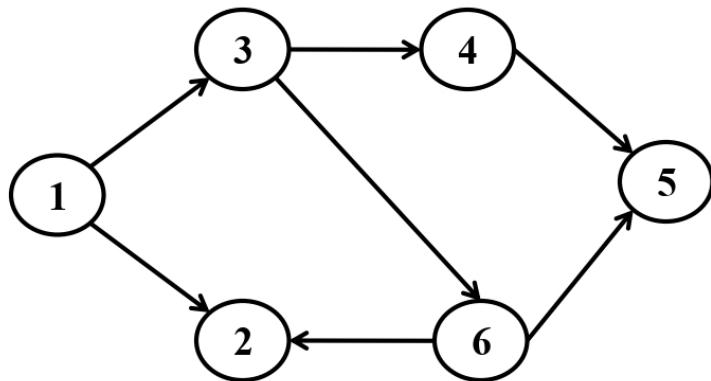
Topological Sort Example

One possible Topological Sort=[1,2,3,4,5,6]

Topological Sorting Algorithm

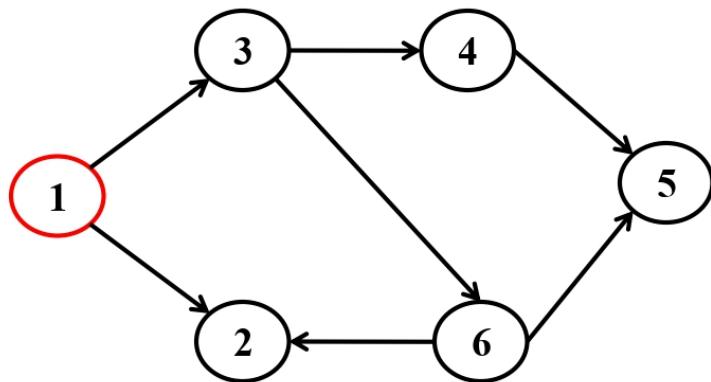
1. Identify a node with no incoming edges($\text{indegree}=0$)
2. Add this node to the ordering.
3. Remove this node and all its outgoing edges from the graph.
4. Repeat step 1 to 3 until the graph becomes empty

Write the topological sorting for the DAG given below



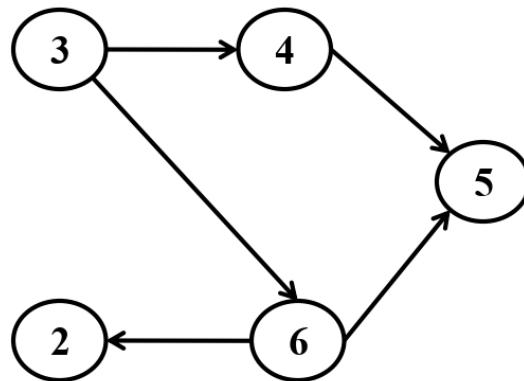
Topological Ordering:

Write the topological sorting for the DAG given below



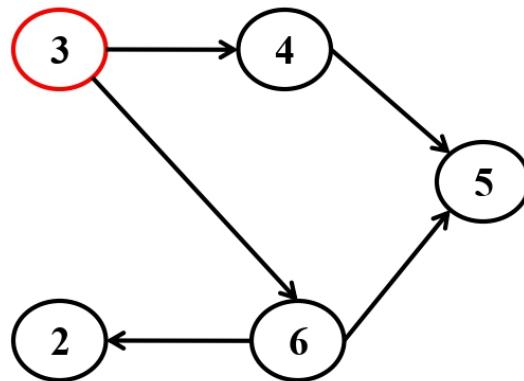
Topological Ordering: 1

Write the topological sorting for the DAG given below



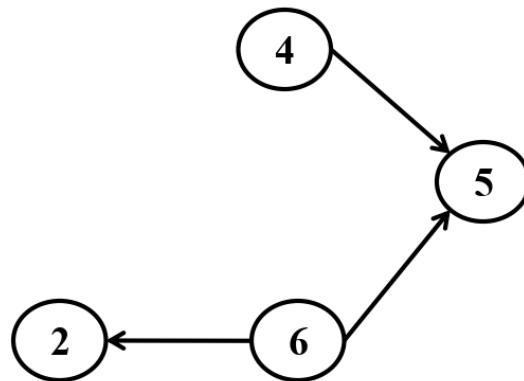
Topological Ordering: 1

Write the topological sorting for the DAG given below



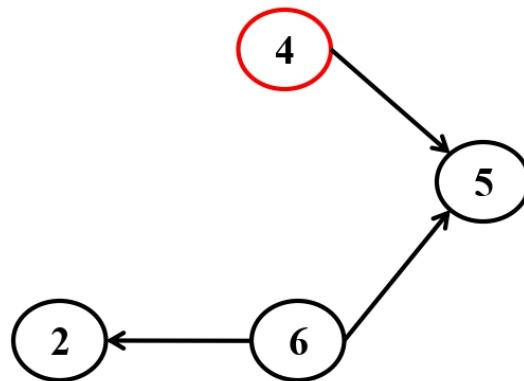
Topological Ordering: 1, 3

Write the topological sorting for the DAG given below



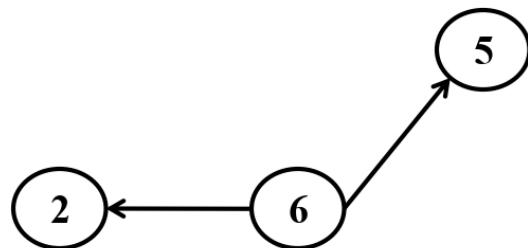
Topological Ordering:1,3

Write the topological sorting for the DAG given below



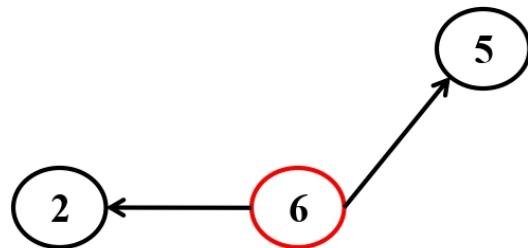
Topological Ordering: 1, 3, 4

Write the topological sorting for the DAG given below



Topological Ordering:1,3,4

Write the topological sorting for the DAG given below



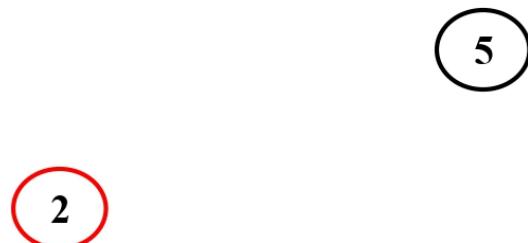
Topological Ordering: 1, 3, 4, 2

Write the topological sorting for the DAG given below



Topological Ordering:1,3,4,6

Write the topological sorting for the DAG given below



Topological Ordering:1,3,4,6,2

Write the topological sorting for the DAG given below



A single node labeled 5, enclosed in a thin black circle.

Topological Ordering: 1, 3, 4, 6, 2

Write the topological sorting for the DAG given below



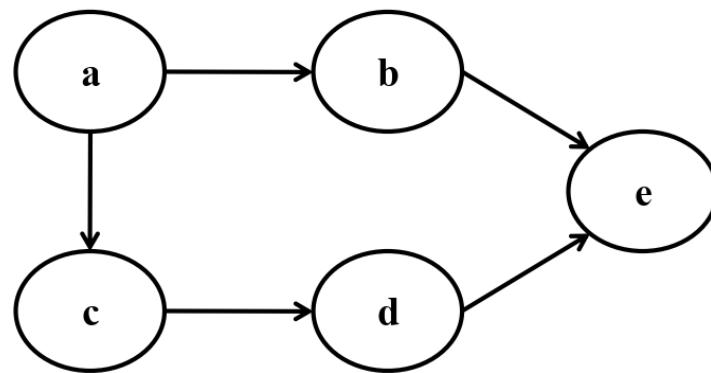
5

Topological Ordering: 1, 3, 4, 6, 2, 5

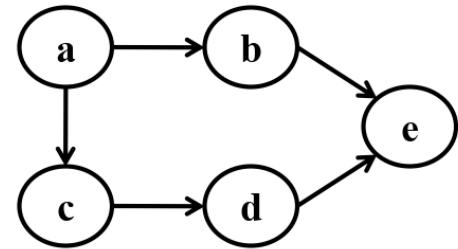
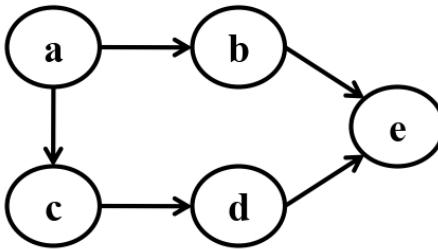
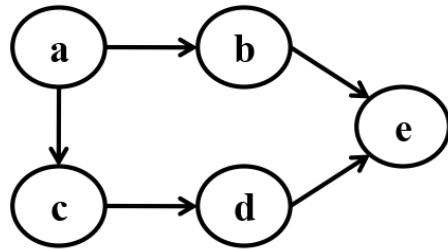
Write the topological sorting for the DAG given below

Topological Ordering:1,3,4,6,2,5

Find the possible topological orderings for the following graph

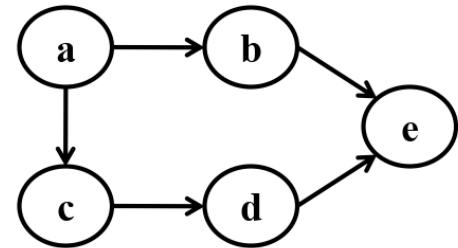
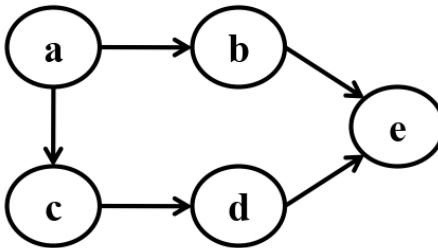
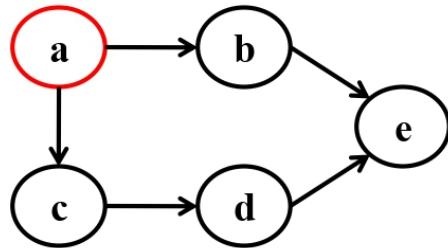


Find the possible topological orderings for the following graph



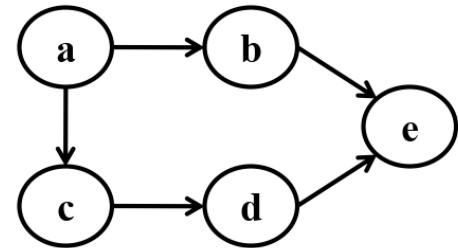
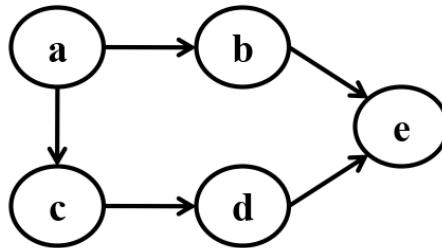
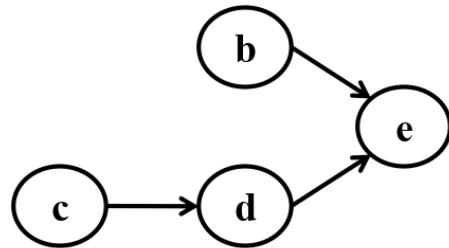
Topological Ordering-1: Topological Ordering-2: Topological Ordering-3:

Find the possible topological orderings for the following graph



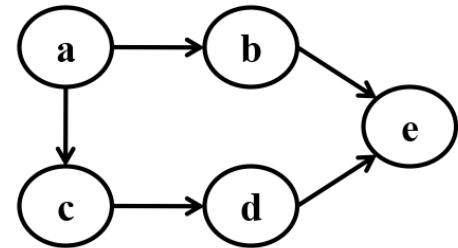
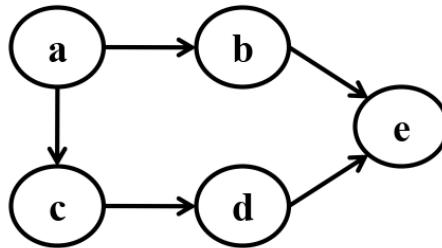
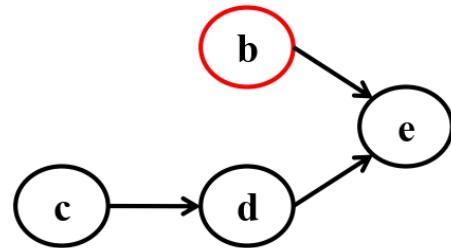
Topological Ordering-1: Topological Ordering-2: Topological Ordering-3:
a

Find the possible topological orderings for the following graph



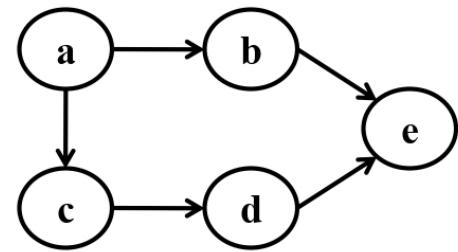
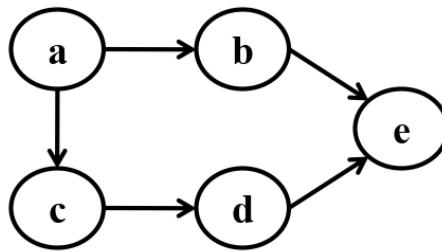
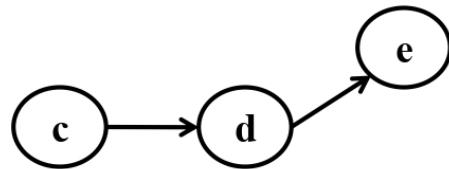
Topological Ordering-1: Topological Ordering-2: Topological Ordering-3:
a

Find the possible topological orderings for the following graph



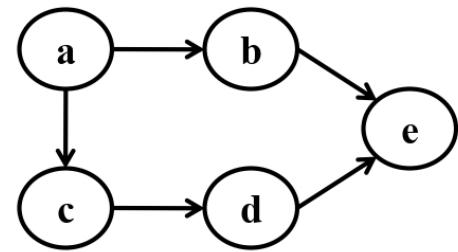
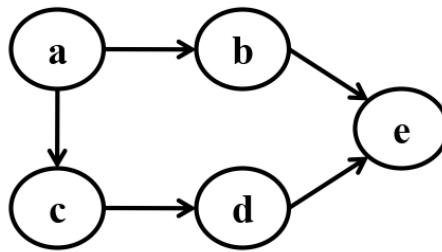
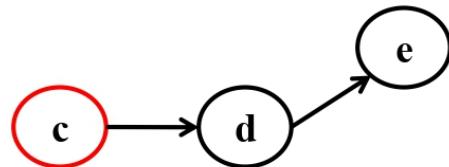
Topological Ordering-1: a,b
Topological Ordering-2: a,c,d,e
Topological Ordering-3: a,c,b,d,e

Find the possible topological orderings for the following graph



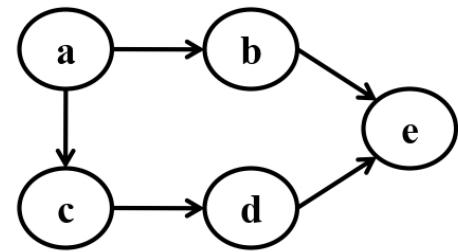
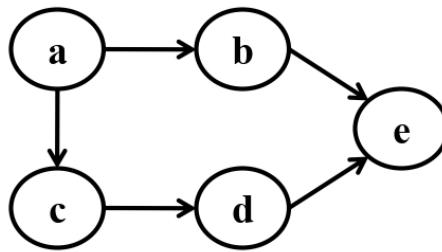
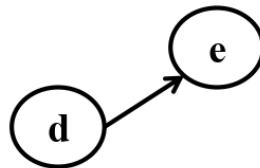
Topological Ordering-1: a,b
Topological Ordering-2: a,b,c,d,e
Topological Ordering-3: a,b,c,d,e

Find the possible topological orderings for the following graph



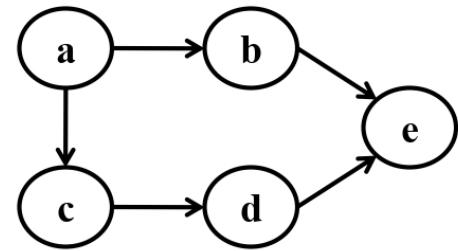
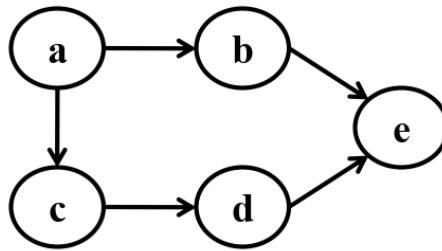
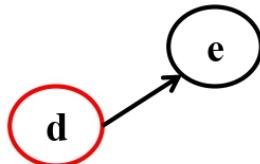
Topological Ordering-1: Topological Ordering-2: Topological Ordering-3:
a,b,c

Find the possible topological orderings for the following graph



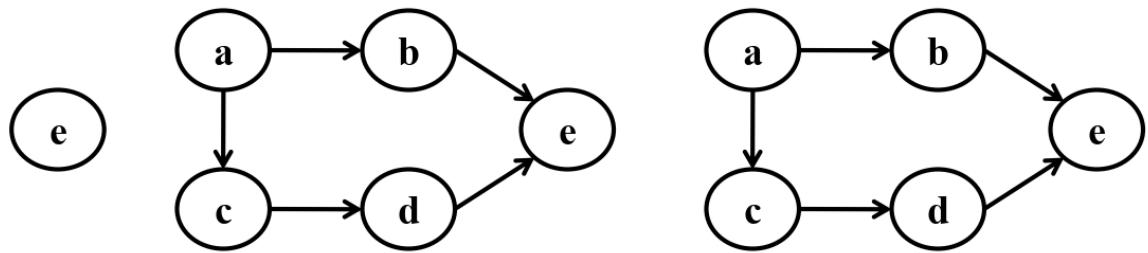
Topological Ordering-1: a,b,c
Topological Ordering-2: a,c,b
Topological Ordering-3: c,a,b

Find the possible topological orderings for the following graph



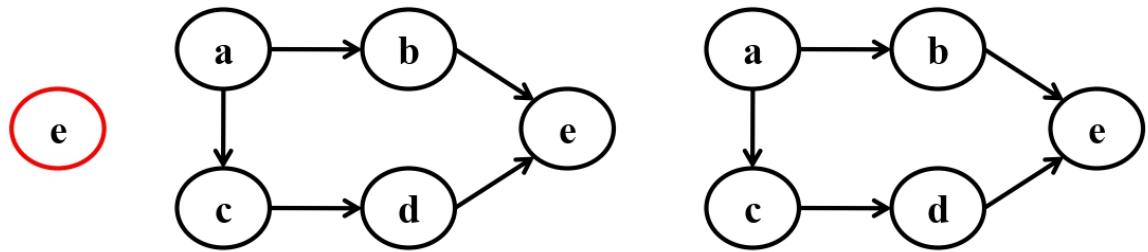
Topological Ordering-1: a,b,c,d
Topological Ordering-2: a,b,c,d,e
Topological Ordering-3: a,b,c,d,e

Find the possible topological orderings for the following graph



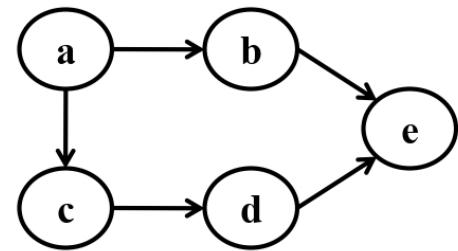
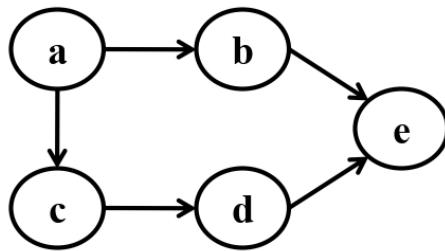
Topological Ordering-1: a,b,c,d
Topological Ordering-2: a,c,b,d
Topological Ordering-3: a,c,d,b

Find the possible topological orderings for the following graph



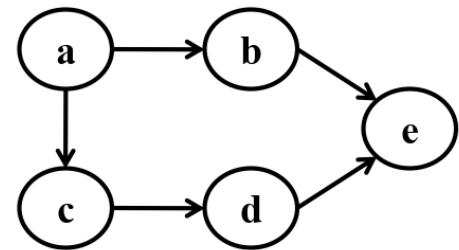
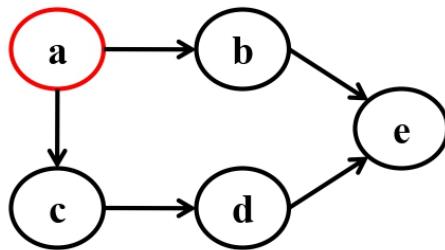
Topological Ordering-1: a,b,c,d,e

Find the possible topological orderings for the following graph



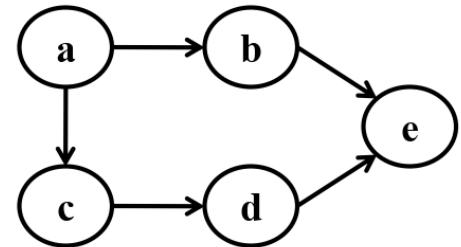
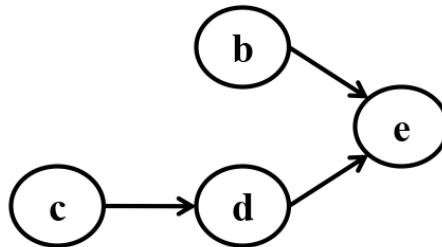
Topological Ordering-1: Topological Ordering-2: Topological Ordering-3:
a,b,c,d,e

Find the possible topological orderings for the following graph



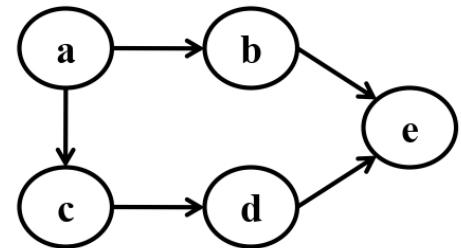
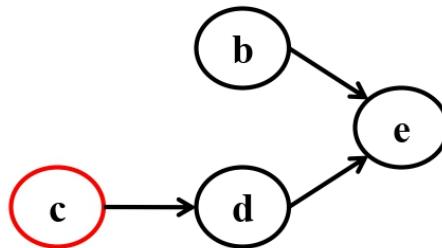
Topological Ordering-1: Topological Ordering-2: Topological Ordering-3:
a,b,c,d,e a, a,

Find the possible topological orderings for the following graph



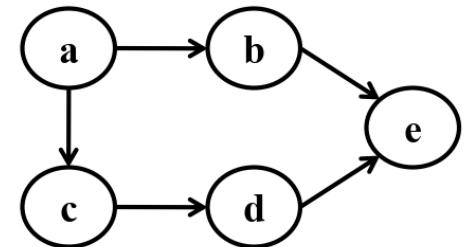
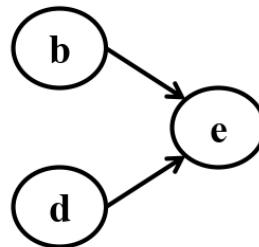
Topological Ordering-1: a,b,c,d,e
Topological Ordering-2: a,
Topological Ordering-3:

Find the possible topological orderings for the following graph



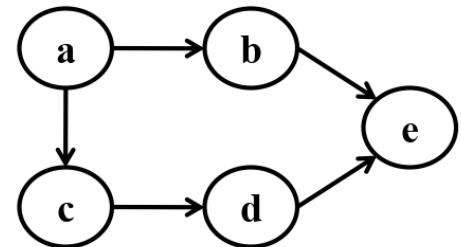
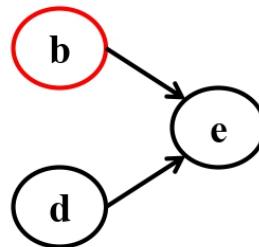
Topological Ordering-1: a,b,c,d,e
Topological Ordering-2: a,c
Topological Ordering-3:

Find the possible topological orderings for the following graph



Topological Ordering-1: Topological Ordering-2: Topological Ordering-3:
a,b,c,d,e a,c

Find the possible topological orderings for the following graph

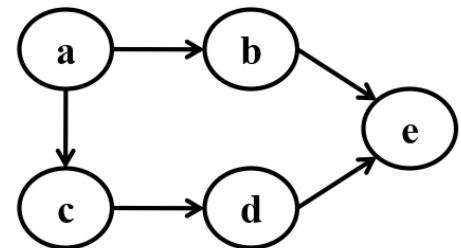
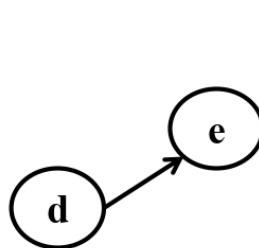


Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b

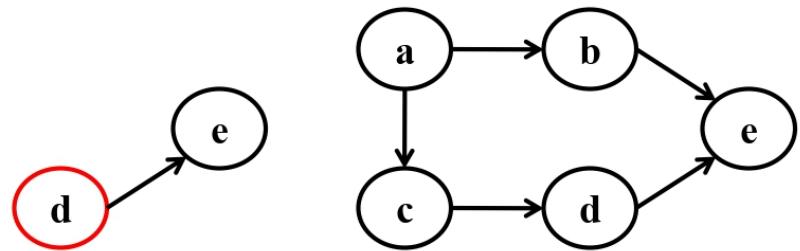
Topological Ordering-3:

Find the possible topological orderings for the following graph



Topological Ordering-1: Topological Ordering-2: Topological Ordering-3:
a,b,c,d,e a,c,b

Find the possible topological orderings for the following graph



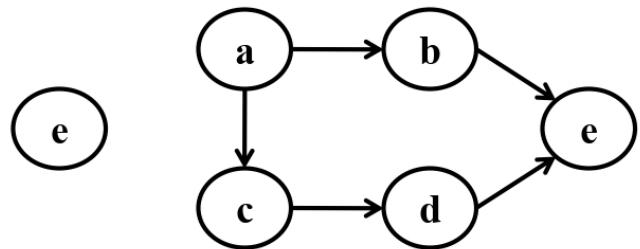
Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b,d

Topological Ordering-3:

a,c,d,b,e

Find the possible topological orderings for the following graph



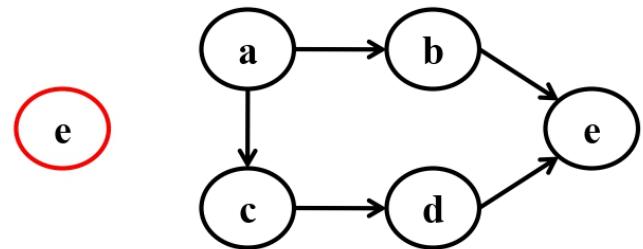
Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b,d

Topological Ordering-3:

a,c,d,b,e

Find the possible topological orderings for the following graph



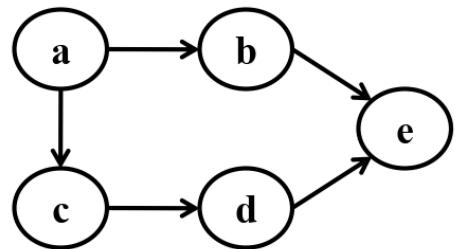
Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b,d,e

Topological Ordering-3:

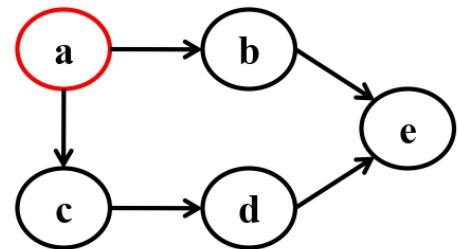
a,c,d,b,e

Find the possible topological orderings for the following graph



Topological Ordering-1: a,b,c,d,e Topological Ordering-2: a,c,b,d,e Topological Ordering-3:

Find the possible topological orderings for the following graph

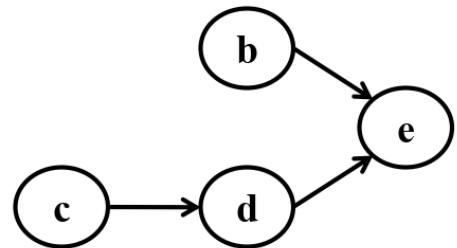


Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b,d,e

Topological Ordering-3:
a,

Find the possible topological orderings for the following graph

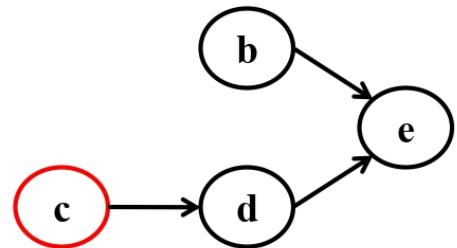


Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b,d,e

Topological Ordering-3:
a,

Find the possible topological orderings for the following graph

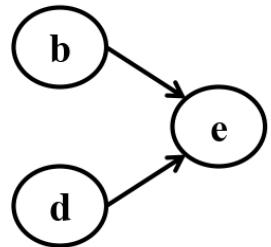


Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b,d,e

Topological Ordering-3:
a,c

Find the possible topological orderings for the following graph

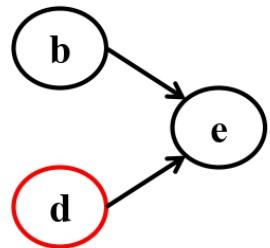


Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b,d,e

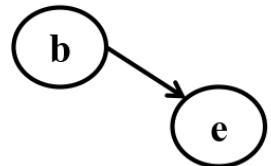
Topological Ordering-3:
a,c

Find the possible topological orderings for the following graph



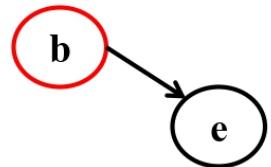
- Topological Ordering-1: a,b,c,d,e Topological Ordering-2: a,c,b,d,e Topological Ordering-3: a,c,d

Find the possible topological orderings for the following graph



- Topological Ordering-1: a,b,c,d,e Topological Ordering-2: a,c,b,d,e Topological Ordering-3: a,c,d

Find the possible topological orderings for the following graph



Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b,d,e

Topological Ordering-3:
a,c,d,b

Find the possible topological orderings for the following graph



Topological Ordering-1: a,b,c,d,e Topological Ordering-2: a,c,b,d,e Topological Ordering-3: a,c,d,b

Find the possible topological orderings for the following graph



Topological Ordering-1: a,b,c,d,e Topological Ordering-2: a,c,b,d,e Topological Ordering-3: a,c,d,b,e

Find the possible topological orderings for the following graph

Topological Ordering-1:
a,b,c,d,e

Topological Ordering-2:
a,c,b,d,e

Topological Ordering-3:
a,c,d,b,e

Topological Sorting Algorithm Complexity

- Time to determine the indegree for each node = **$O(E)$** time. This involves looking at each directed edge in the graph once.
- Time to determine the nodes with no incoming edges = **$O(V)$**
- So Step 1 complexity = **$O(E + V)$**
- Add nodes until we run out of nodes with no incoming edges. This loop could run once for every node— **$O(V)$** times
- All together, the time complexity is **$O(V+E)$**

Topological Sorting Applications

- Scheduling jobs from the given dependencies among jobs
- Instruction Scheduling
- Determining the order of compilation tasks to perform in makefiles
- Data Serialization