Directed Graphs (Digraphs)

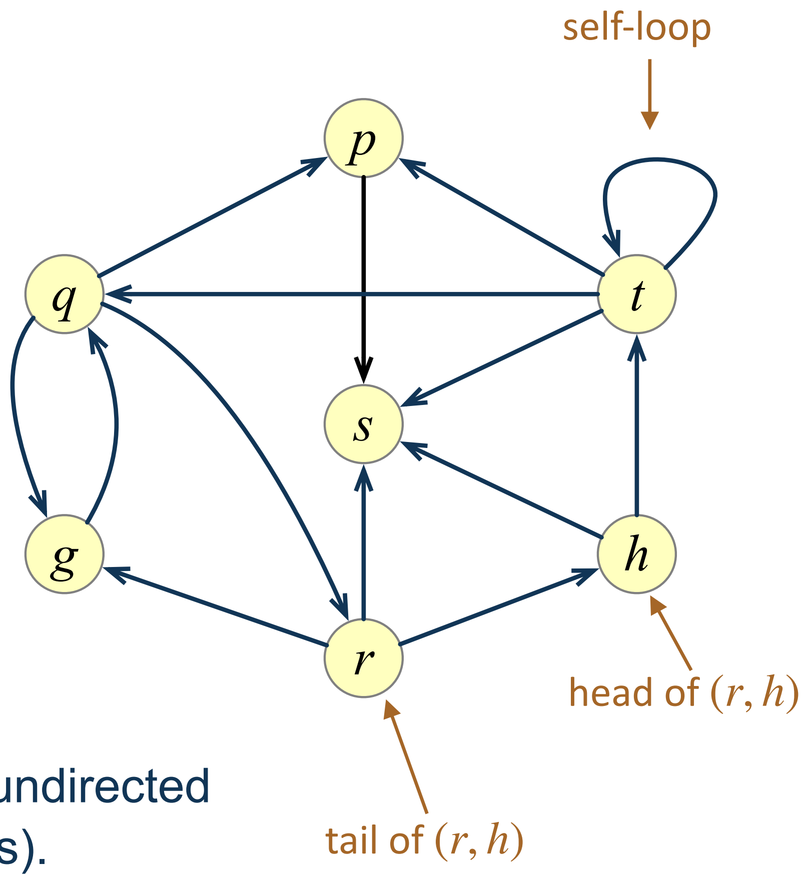
- Digraphs
- Connectivity, strongly connected components
- Transitive closure
- Acyclic digraphs
- Topological sorting
- Other facts about digraphs

Directed graphs

Formal definition: A digraph is a pair $G = (V, E)$, where V is a finite set of elements called vertices (or nodes) and $E \subseteq V \times V$ is a set of edges.

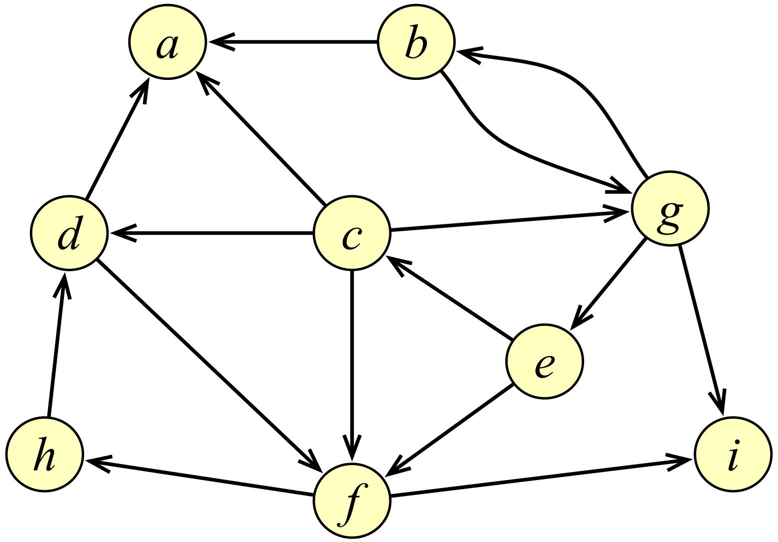
ordered pairs of vertices

- If $e = (u, v) \in E$ then we say that
  - u is the tail of e and in-neighbor of v
  - v is the head of e and out-neighbor of u
  - if $u = v$ then e is called self-loop
- Each vertex v has two degrees
  - in-degree of v is its number of in-neighbors
  - out-degree of v is its number of out-neighbors
- Subgraphs, paths, cycles, etc. are defined analogously to undirected graphs (paths and cycles need to follow the edge directions).



Directed graphs: strong connectivity

Question: Should we consider this digraph to be connected?



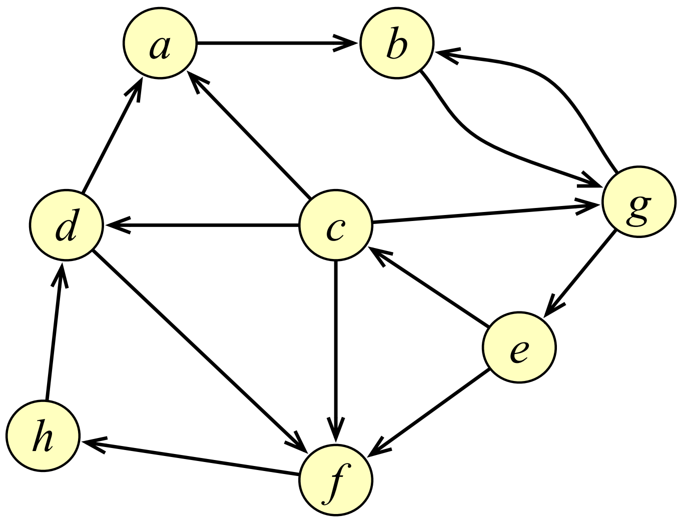
Not really: for example, there is no path from i to any other vertex.

Directed graphs: strong connectivity

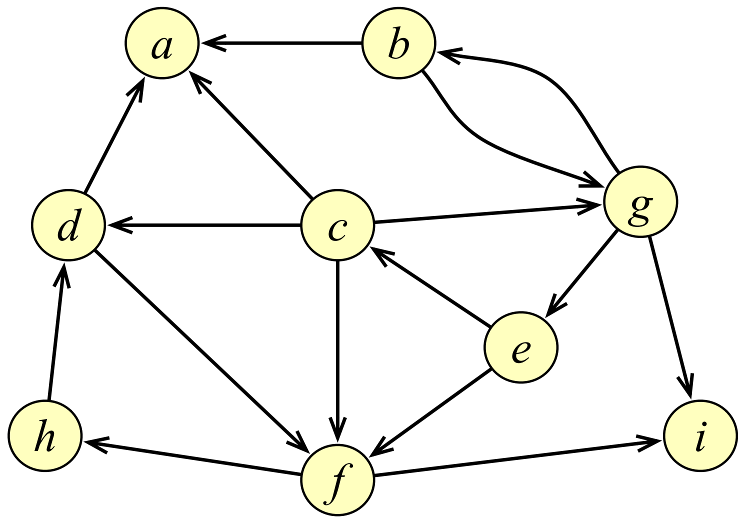
- Analog of connectivity for digraphs: *strong connectivity.*

A digraph $G = (V, E)$ is called *strongly connected* if there is a (directed) path in G from any vertex to any other vertex.

Example:



strongly connected

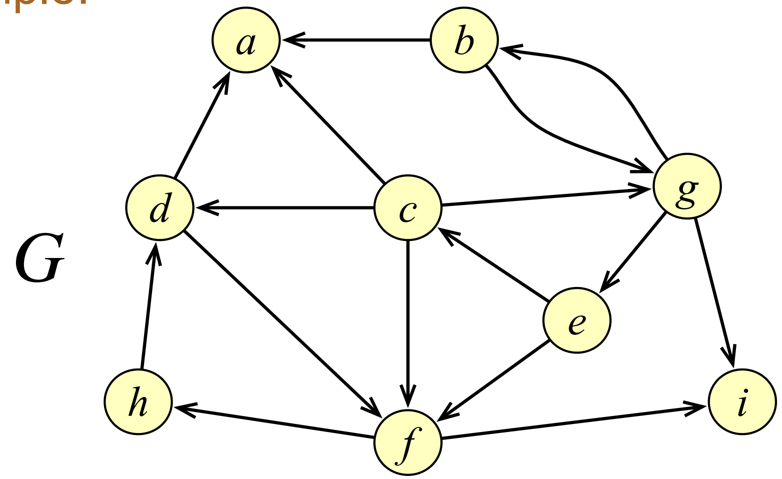


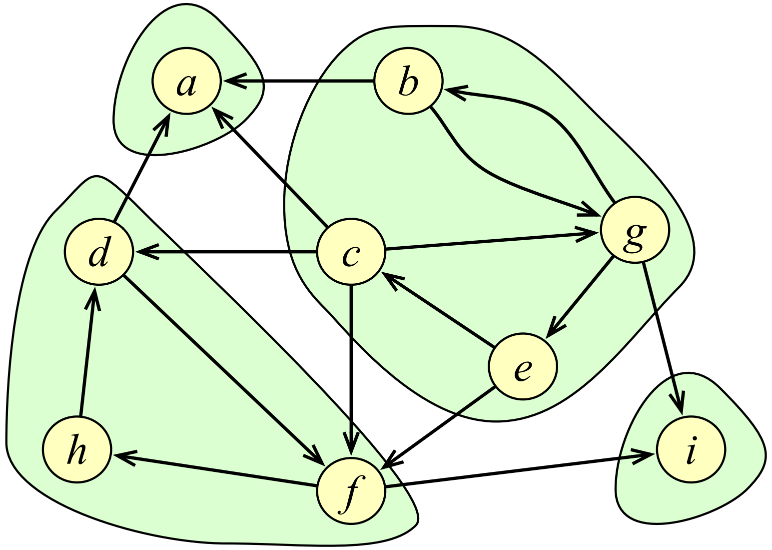
not strongly connected

Directed graphs: strong connectivity

- Let $G = (V, E)$ be a digraph. A graph H is called a strongly connected component of G if
  - H is a strongly connected subgraph of G, and
  - There is no other strongly connected subgraph of G that contains H.

Example:

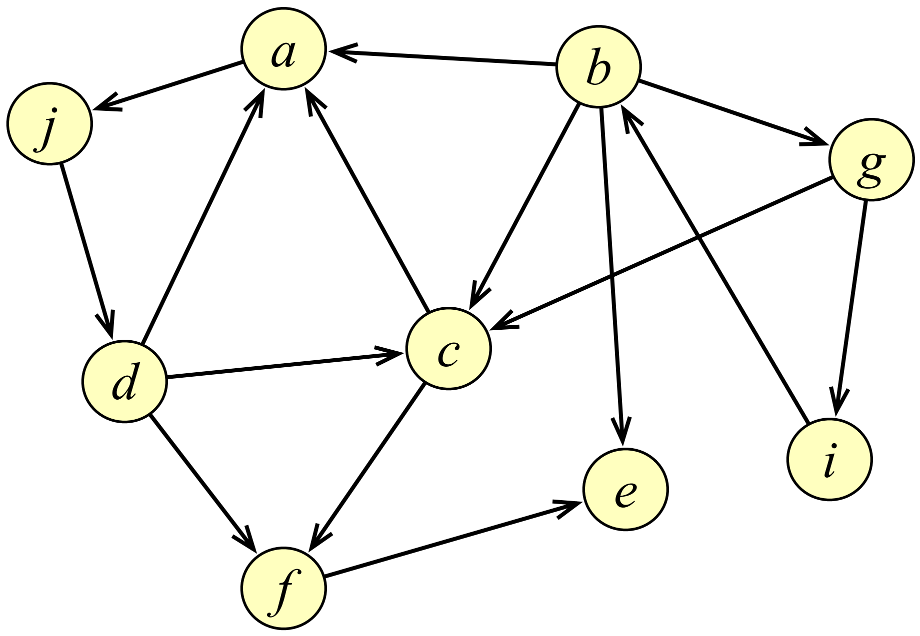




strongly connected components of G

Directed graphs: strong connectivity

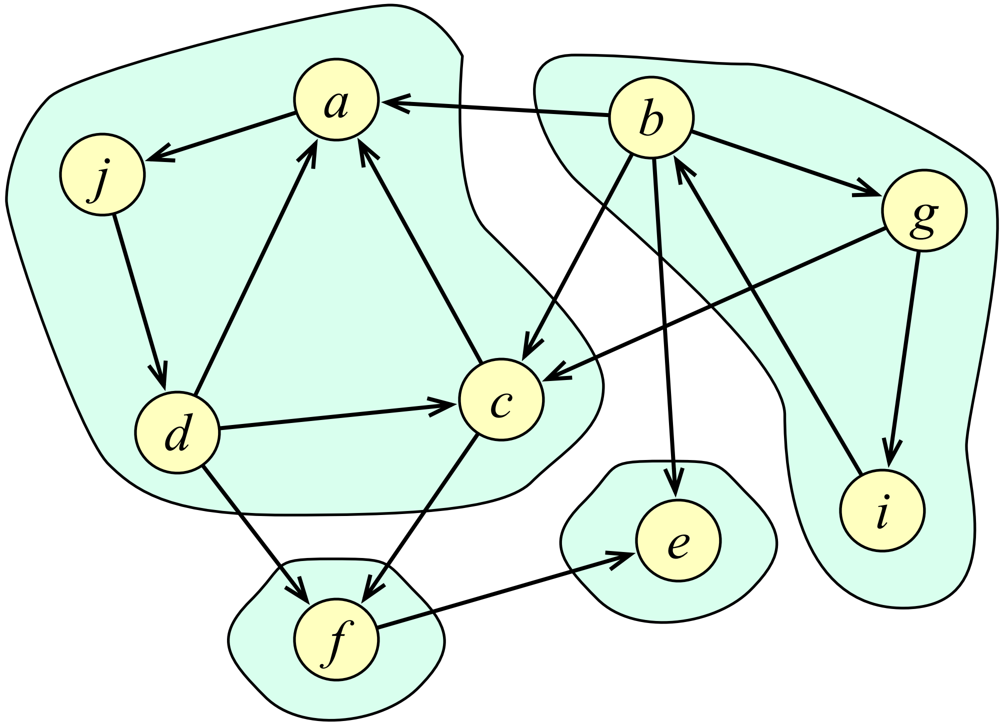
Zoom poll: How many strongly connected components this digraph has?



Directed graphs: strong connectivity

Zoom poll: How many strongly connected components this digraph has?

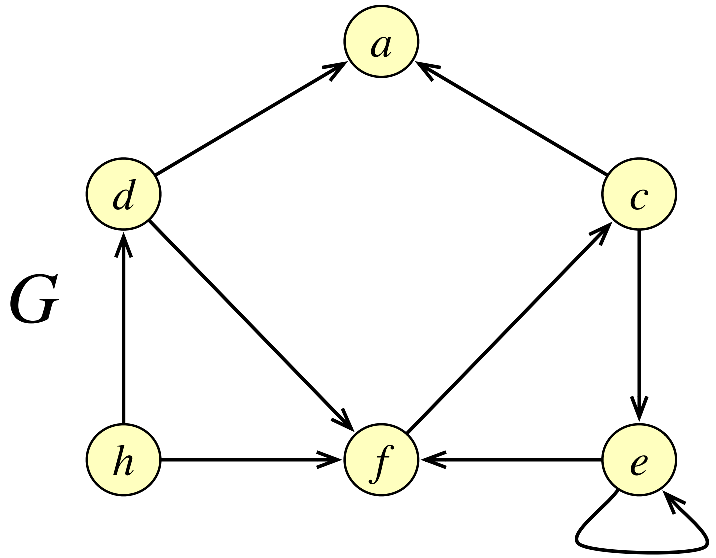
Answer: 4.

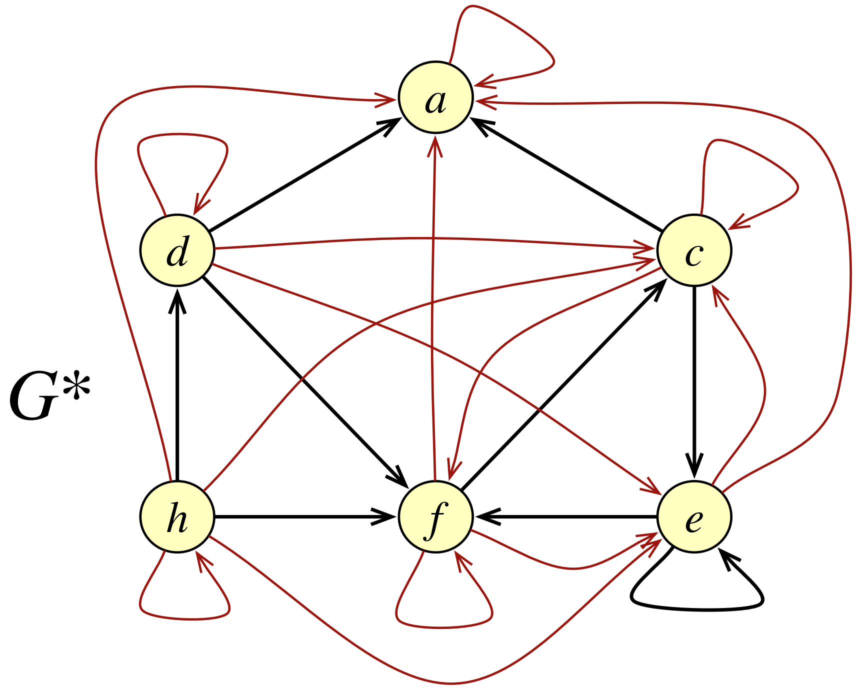


Directed graphs: transitive closure

- *A transitive closure* of a digraph $G = (V, E)$ is a digraph $G^* = (V, E^*)$ such that $(u, v) \in E^*$ if and only if there is a path from u to v in G.

Example:





Observation: G is strongly connnected if and only if G^* is a complete directed graph (that is, $E^* = V \times V$).

Traversing directed graphs

- Euler tours: Is there a simple characterization of Euler tours in digraphs?

Theorem: Let $G = (V, E)$ be a strongly connected digraph. Then G has an Euler tour if and only if $\text{indeg}(v) = \text{outdeg}(v)$ for each vertex $v \in V$.

Proof: Same as for undirected graphs.

- Hamiltonian cycles: Is there an analog of Dirac's theorem about hamiltonian cycles?

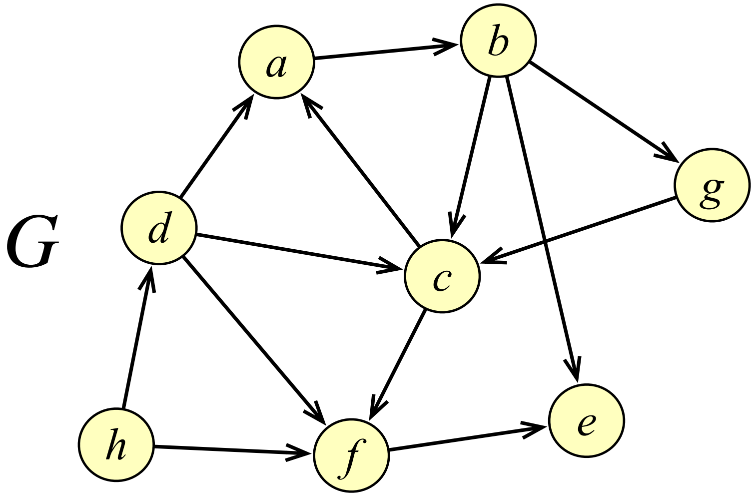
Theorem (Ghouila-Houri): Let $G = (V, E)$ be a digraph. If each vertex $v \in V$ satisfies $\text{indeg}(v) \geq n/2$ and $\text{outdeg}(v) \geq n/2$ then G has a hamiltonian cycle.

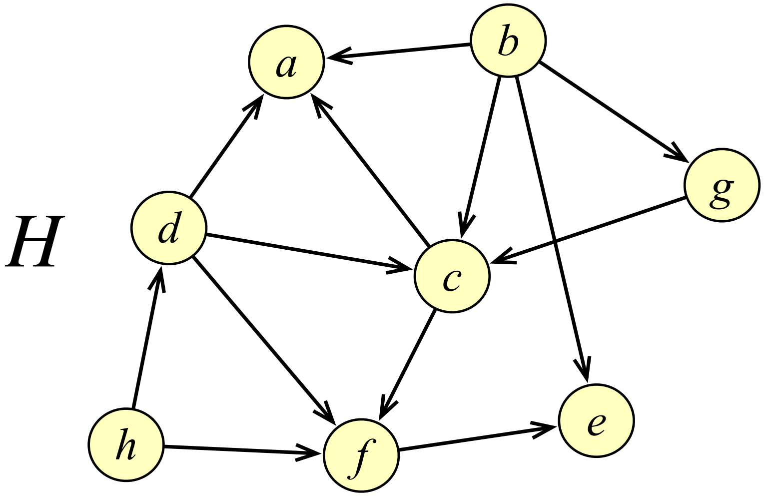
Proof: Harder… omitted.

DAGs - acyclic directed graphs

- A digraph $G = (V, E)$ is called a *DAG (acyclic directed graph)* if G does not contain any cycles.

Example: Which of these digraphs is a DAG?





G is not a DAG, because (a, b, c, a) is a cycle

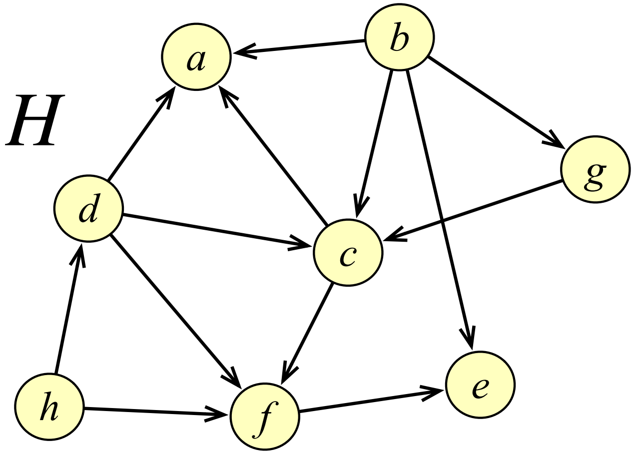
H is a DAG

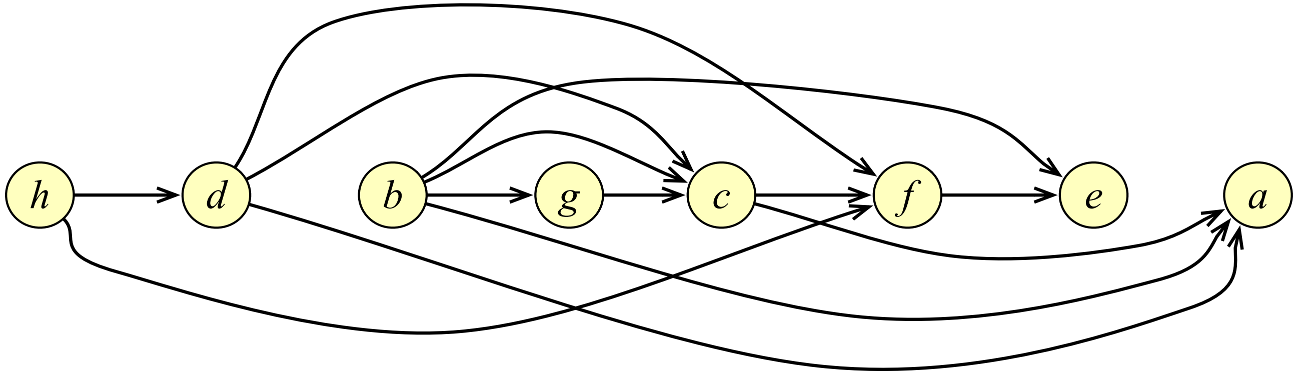
how to determine if a graph sia DAG?

DAGs - acyclic directed graphs

- A *topological ordering* (or sort) of a digraph $G = (V, E)$ is a total order $\prec$ on its vertex set V such that $u \prec v$ for each edge $(u, v) \in E$.

Example:





In ordering h, d, b, g, c, f, e, a all edges are forward

Another drawing of H showing that in ordering h, d, b, g, c, f, e, a all edges are forward

DAGs - acyclic directed graphs

Theorem: A digraph $G = (V, E)$ is a DAG if and only if it has a topological ordering.

Proof: ($\Leftarrow$) Trivial: if there is an ordering with all edges going forward, we cannot have cycles.

($\Rightarrow$) Let G be a DAG. We show how to construct a topological ordering of G.

```
Algorithm TopSort(G)
   G_1 ← G
   for i = 1,2,...,n
       v_i ← any vertex v in G_i
             with indeg(v) = 0
       G_{i+1} ← G_i − {v_i}
       print(v_i)
```

such v must exist, why?

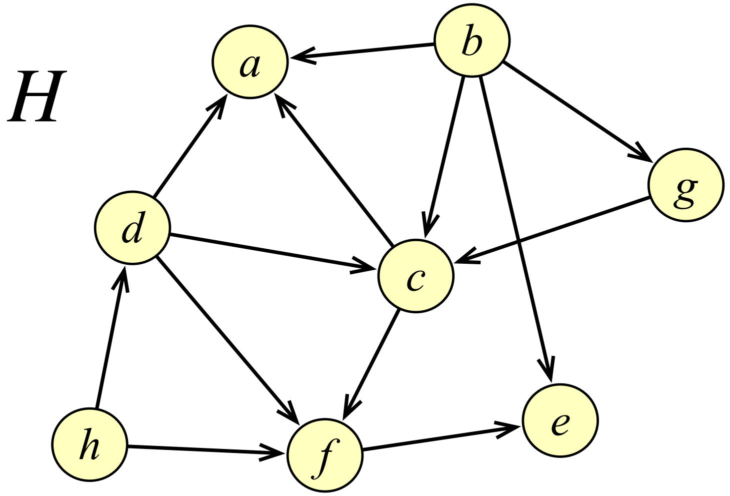
(Otherwise we could start at any vertex and follow edge backwards, and eventually we'll obtain a cycle)

Remove v from G_i. After removing v_i, G_{i+1} remains acyclic.

Sequence $v_1, v_2, \ldots, v_n$ is a topological sort, because, for each i, vertices $v_{i+1}, v_{i+2}, \ldots, v_n$ do not have edges to v_i. ■

DAGs - acyclic directed graphs

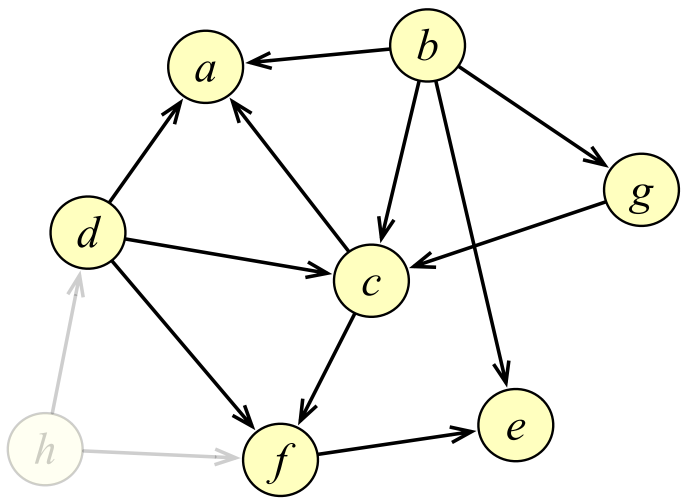
Example: Computing topological ordering.



Ordering:

DAGs - acyclic directed graphs

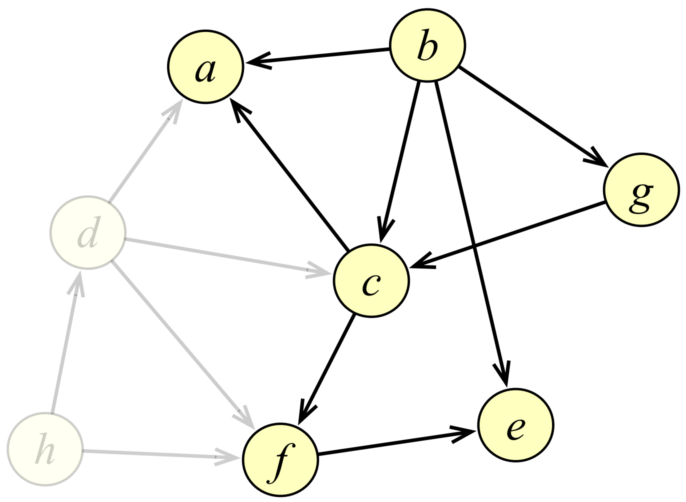
Example: Computing topological ordering.



Ordering: h

DAGs - acyclic directed graphs

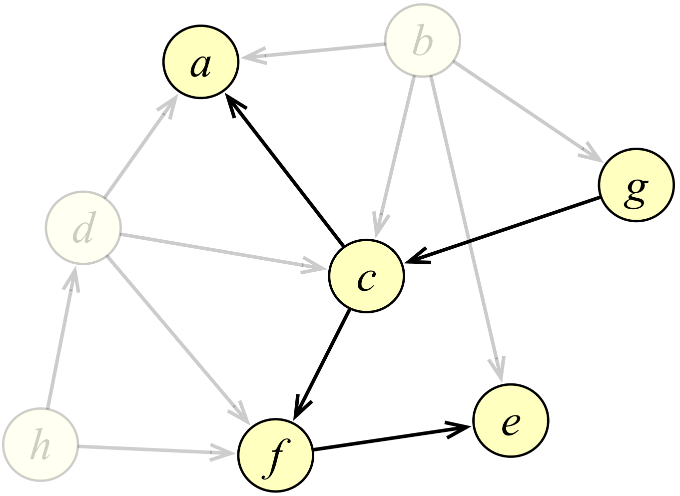
Example: Computing topological ordering.



Ordering: h , d

DAGs - acyclic directed graphs

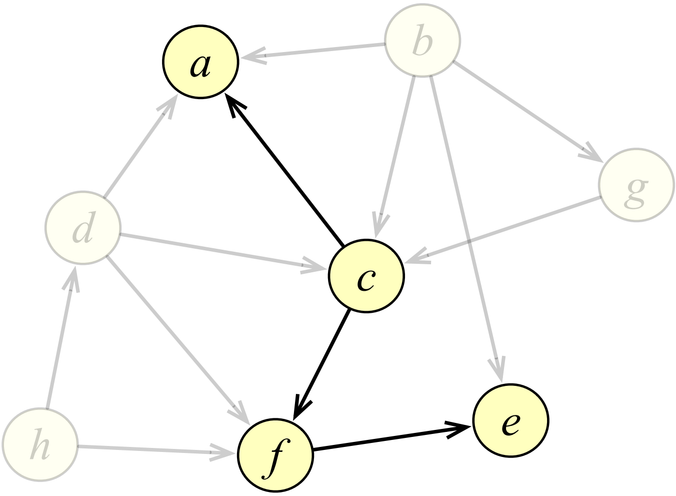
Example: Computing topological ordering.



Ordering: $h\,,d\,,b$

DAGs - acyclic directed graphs

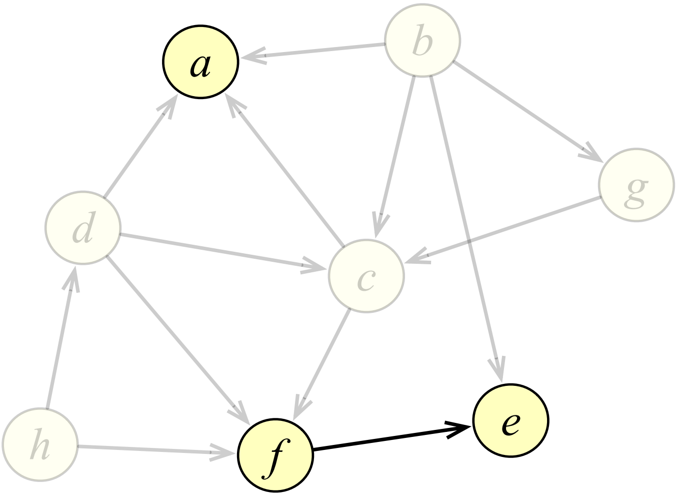
Example: Computing topological ordering.



Ordering: h, d, b, g

DAGs - acyclic directed graphs

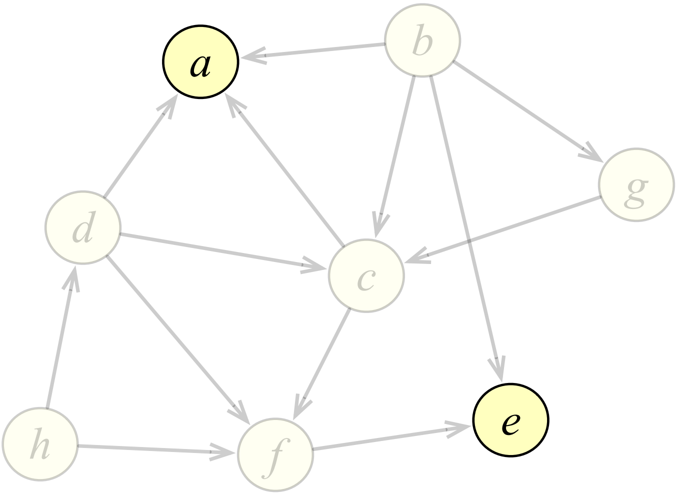
Example: Computing topological ordering.



Ordering: $h\,,d\,,b\,,g\,,c$

DAGs - acyclic directed graphs

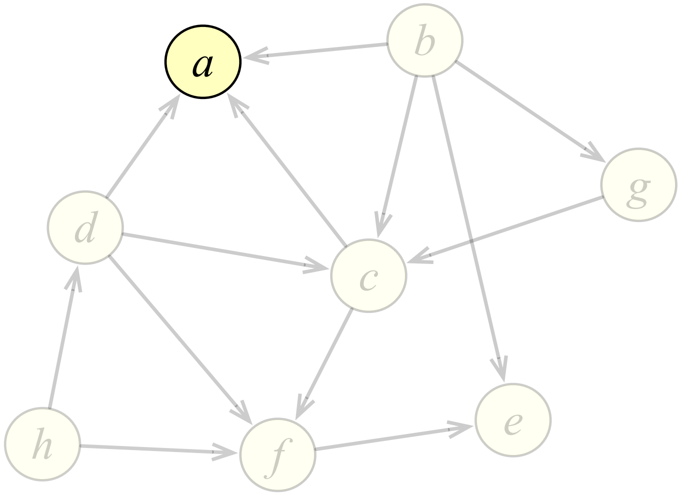
Example: Computing topological ordering.



Ordering: $h\,,d\,,b\,,g\,,c\,,f$

DAGs - acyclic directed graphs

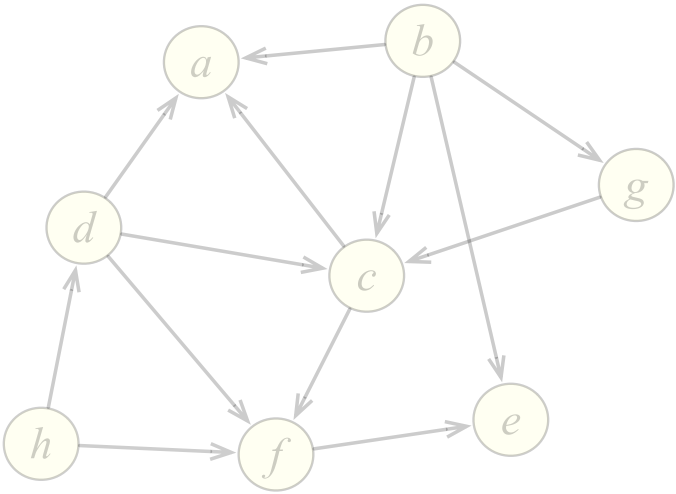
Example: Computing topological ordering.



Ordering: $h\,, d\,, b\,, g\,, c\,, f\,, e$

DAGs - acyclic directed graphs

Example: Computing topological ordering.



Ordering: h, d, b, g, c, f, e, a

DAGs - acyclic directed graphs

Zoom poll: For a digraph G, are the following two conditions equivalent :

- G is a DAG
- Each strongly connected component of G consists of one vertex

Answer: Yes.

Because

- each non-singleton strongly connected component contains a cycle, and
- each cycle is included in a non-singleton strongly connected component.