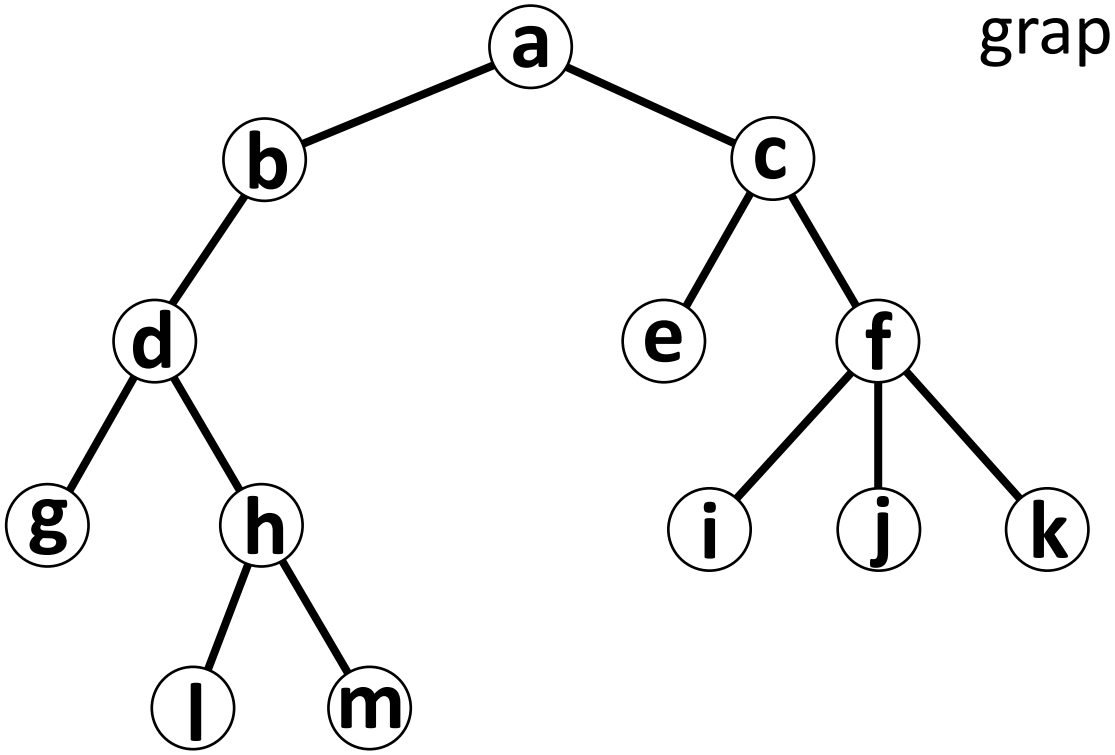
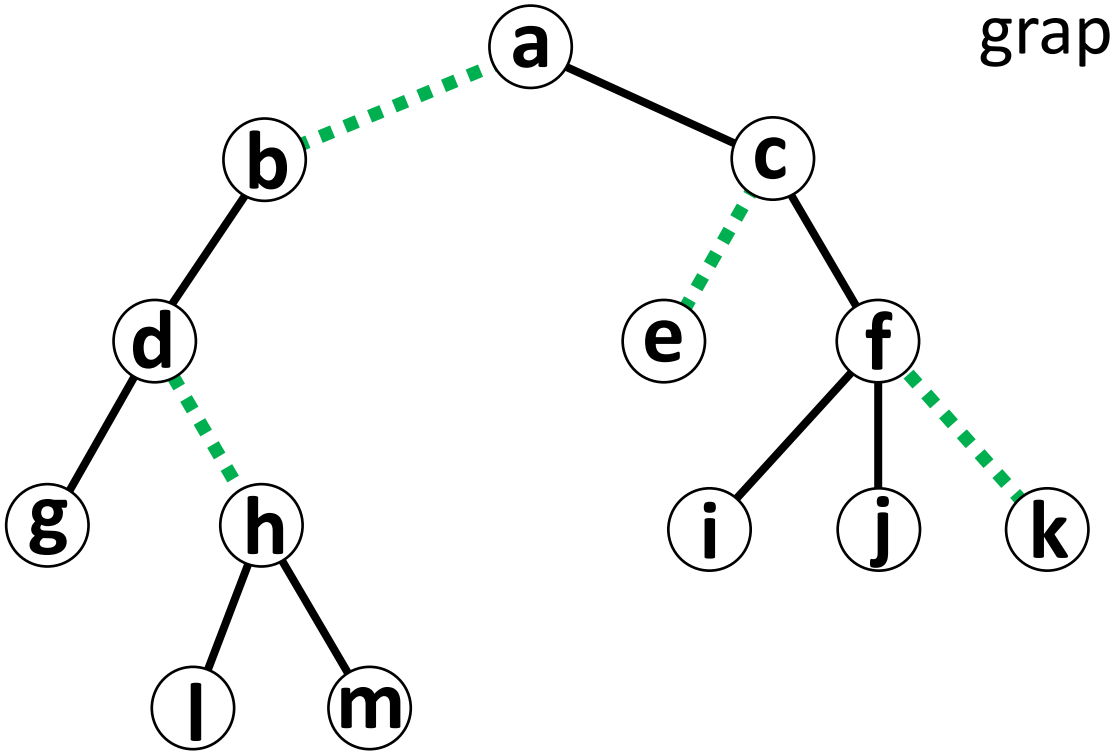
Flow Networks
CSCI 532

Maximum Matching



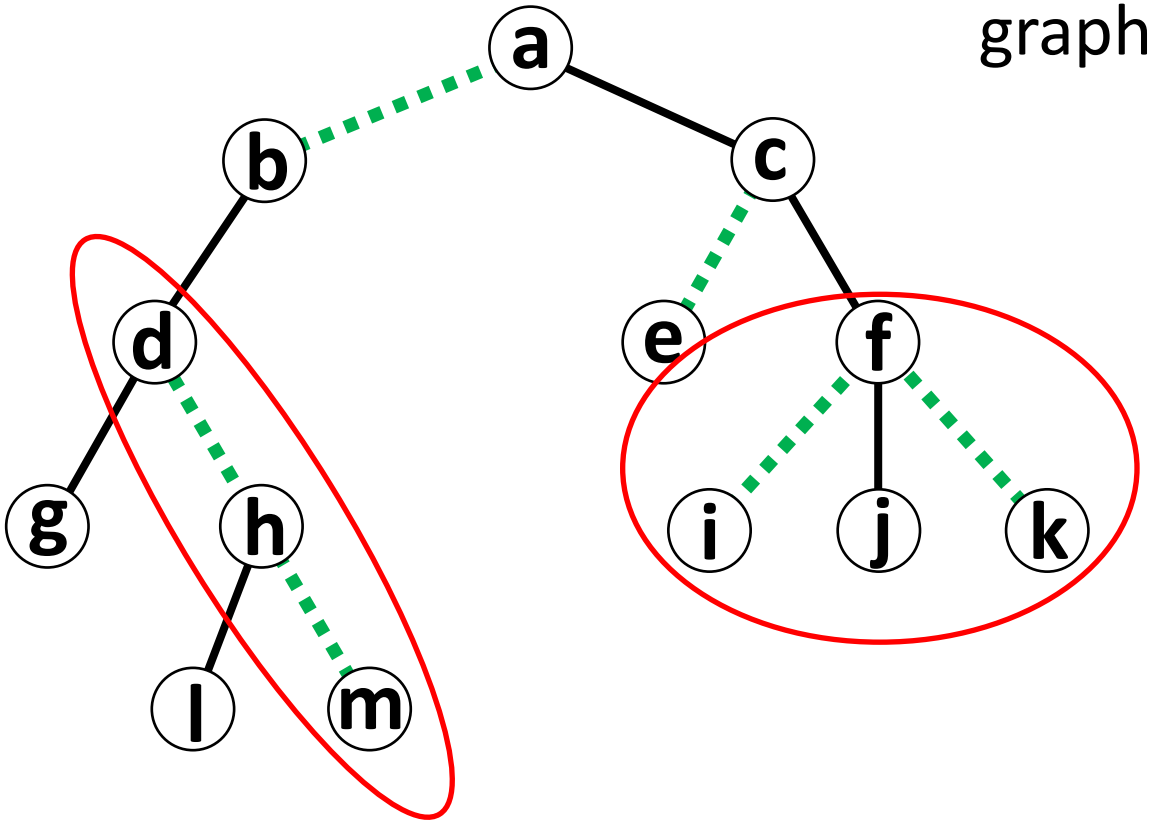
Matching: A subset of edges in a graph that do not share vertices.

Maximum Matching



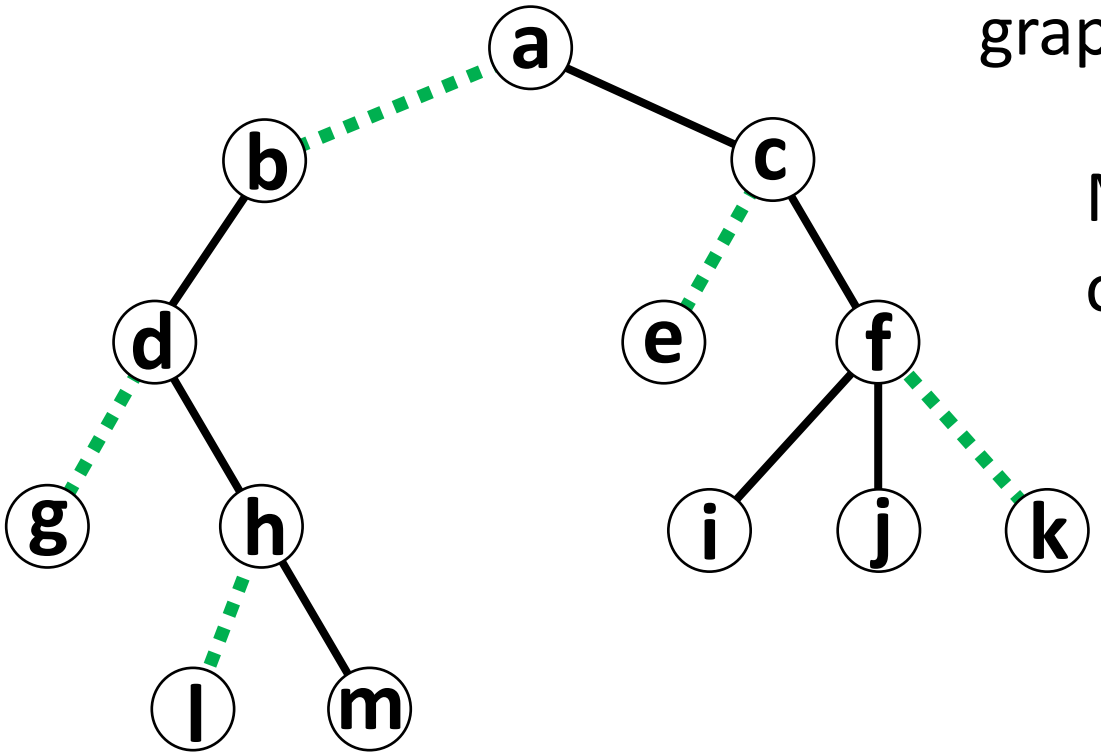
Matching: A subset of edges in a graph that do not share vertices.

Maximum Matching



Matching: A subset of edges in a graph that do not share vertices.

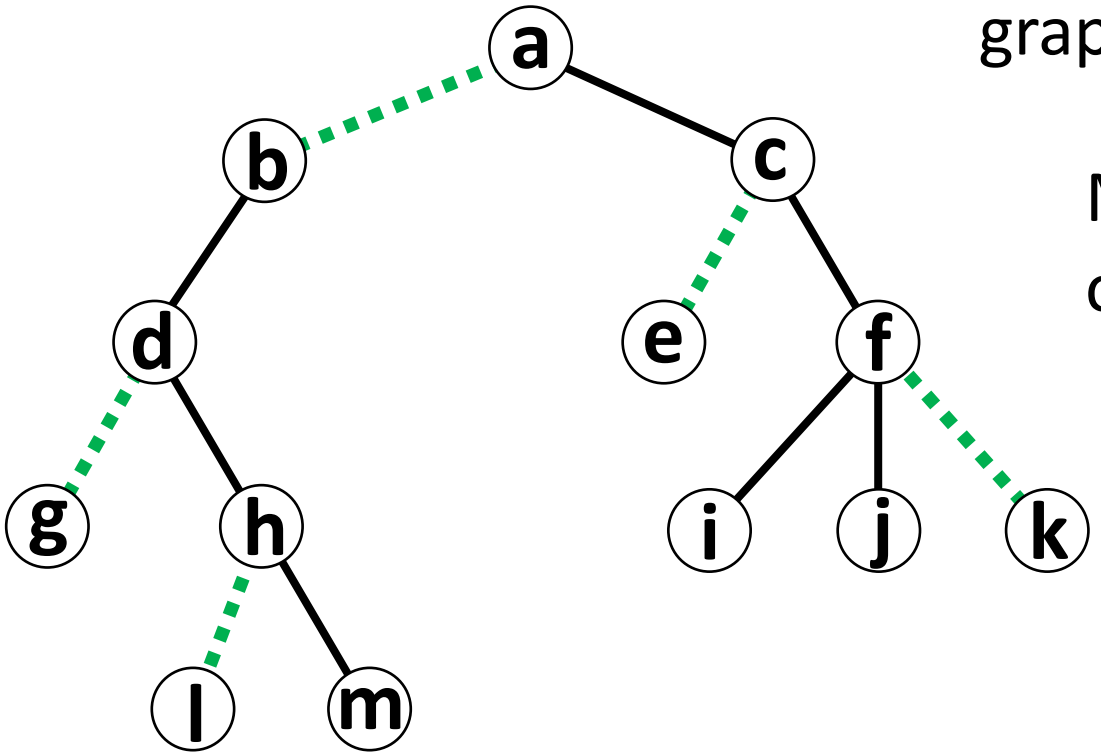
Maximum Matching



Matching: A subset of edges in a graph that do not share vertices.

Maximum Matching: The matching of largest size.

Maximum Matching

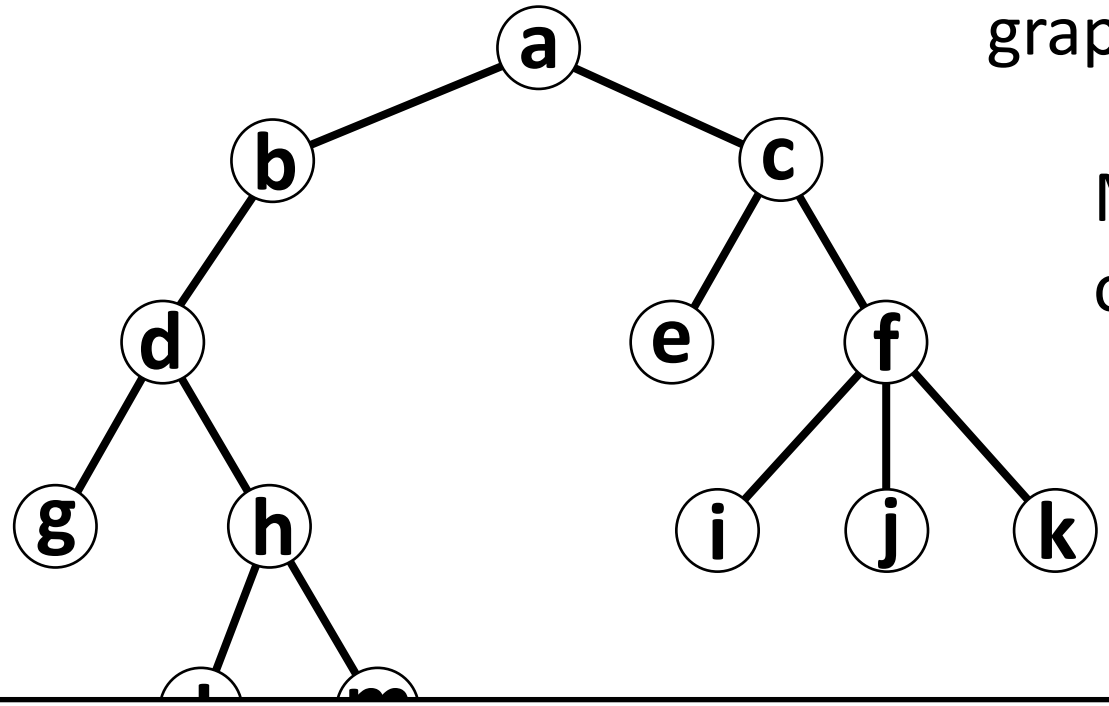


Matching: A subset of edges in a graph that do not share vertices.

Maximum Matching: The matching of largest size.

Maximum Matching = Maximum Independent Edge Set

Maximum Matching



Matching: A subset of edges in a graph that do not share vertices.

Maximum Matching: The matching of largest size.

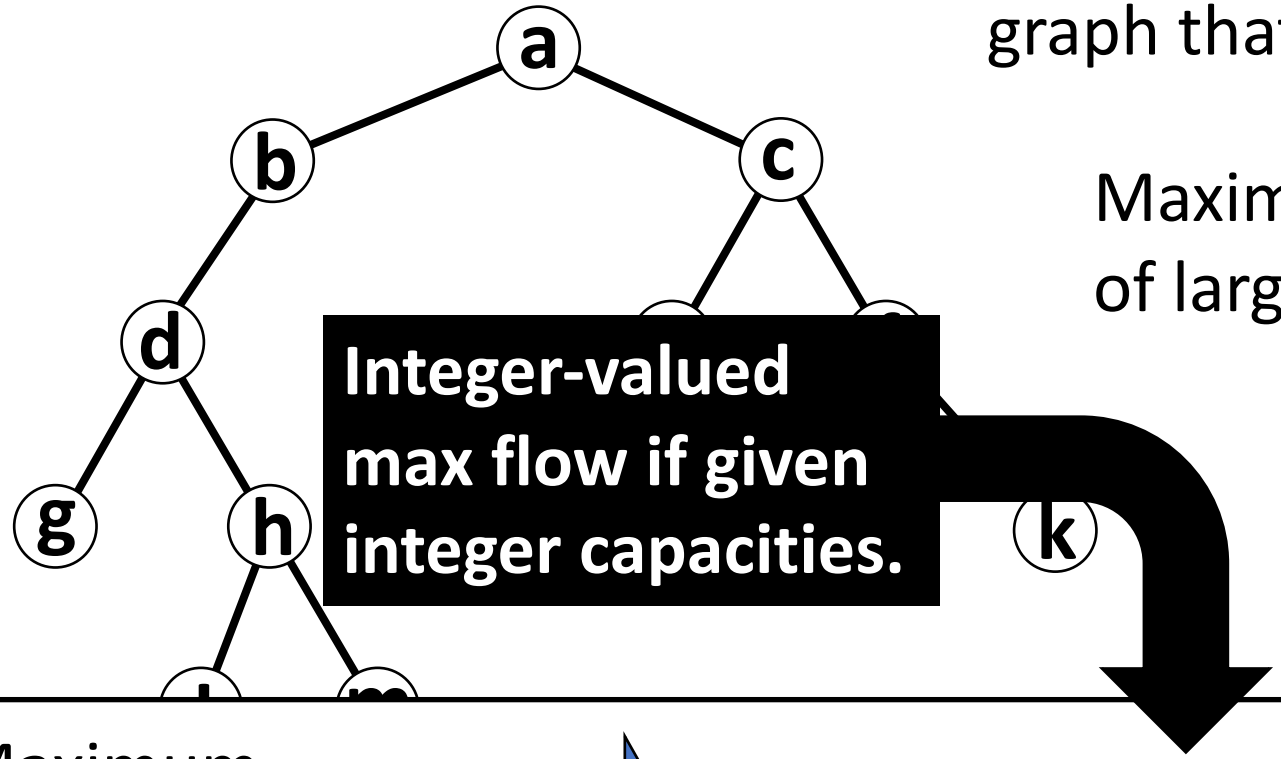
Maximum Matching = Maximum Independent Edge Set

Maximum Matching Input → **Max Flow Algorithm** → Maximum Matching

Maximum Matching

Matching: A subset of edges in a graph that do not share vertices.

Maximum Matching: The matching of largest size.

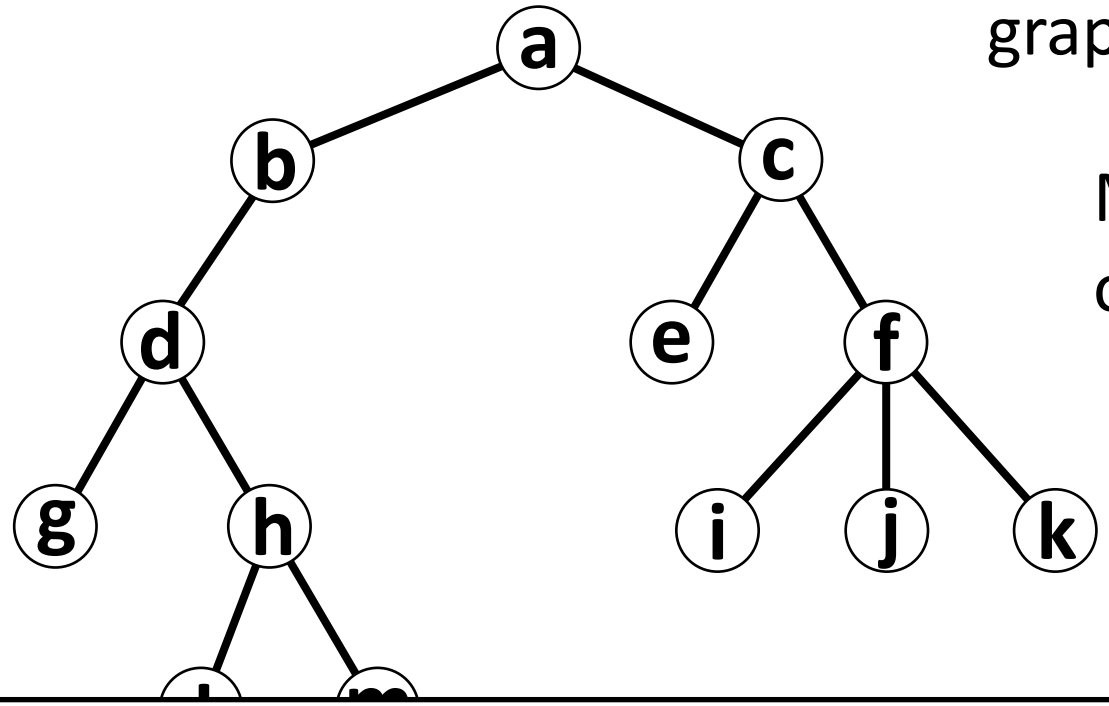


Integer-valued max flow if given integer capacities.

Maximum Matching = Maximum Independent Edge Set

Maximum Matching Input → **Max Flow Algorithm** → Maximum Matching

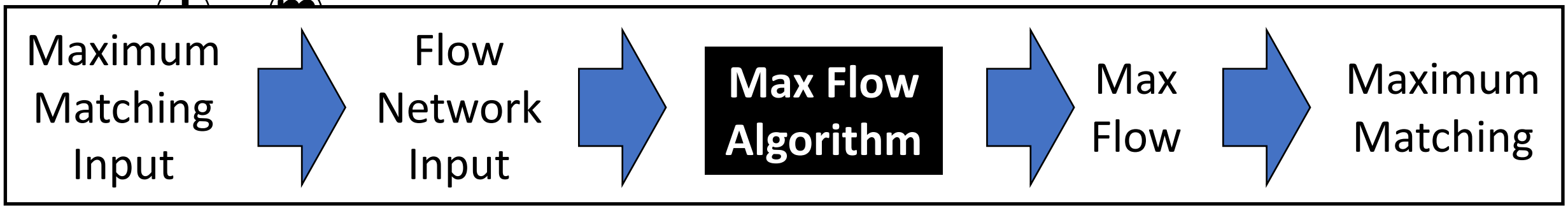
Maximum Matching



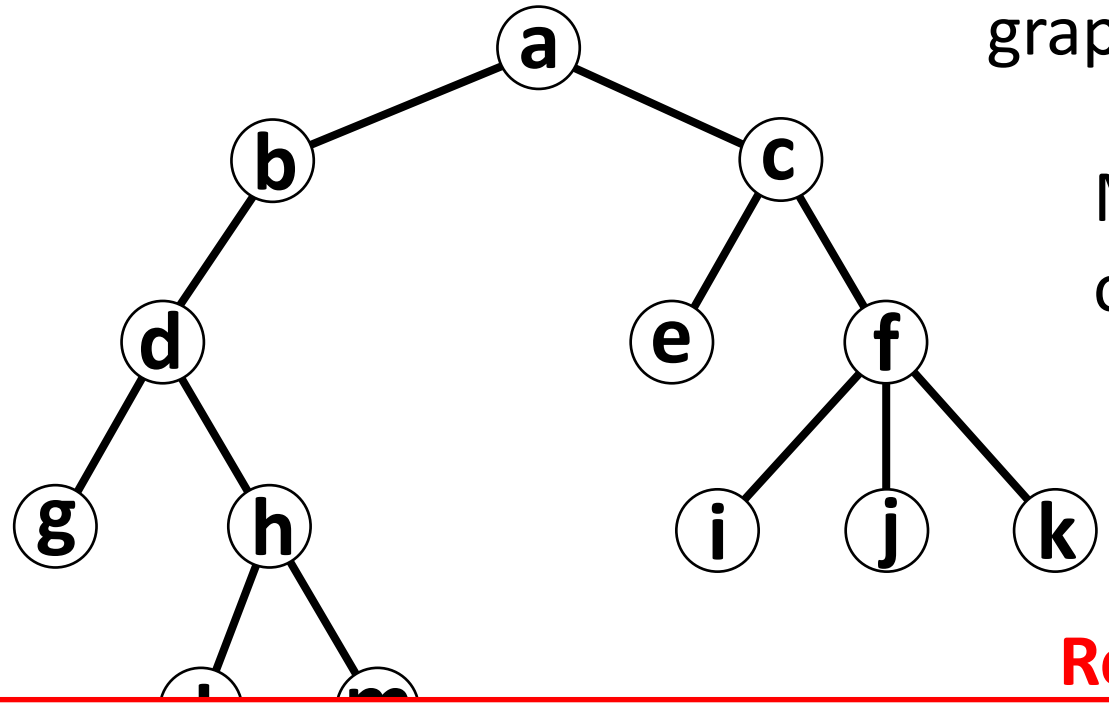
Matching: A subset of edges in a graph that do not share vertices.

Maximum Matching: The matching of largest size.

Maximum Matching = Maximum Independent Edge Set



Maximum Matching



Matching: A subset of edges in a graph that do not share vertices.

Maximum Matching: The matching of largest size.

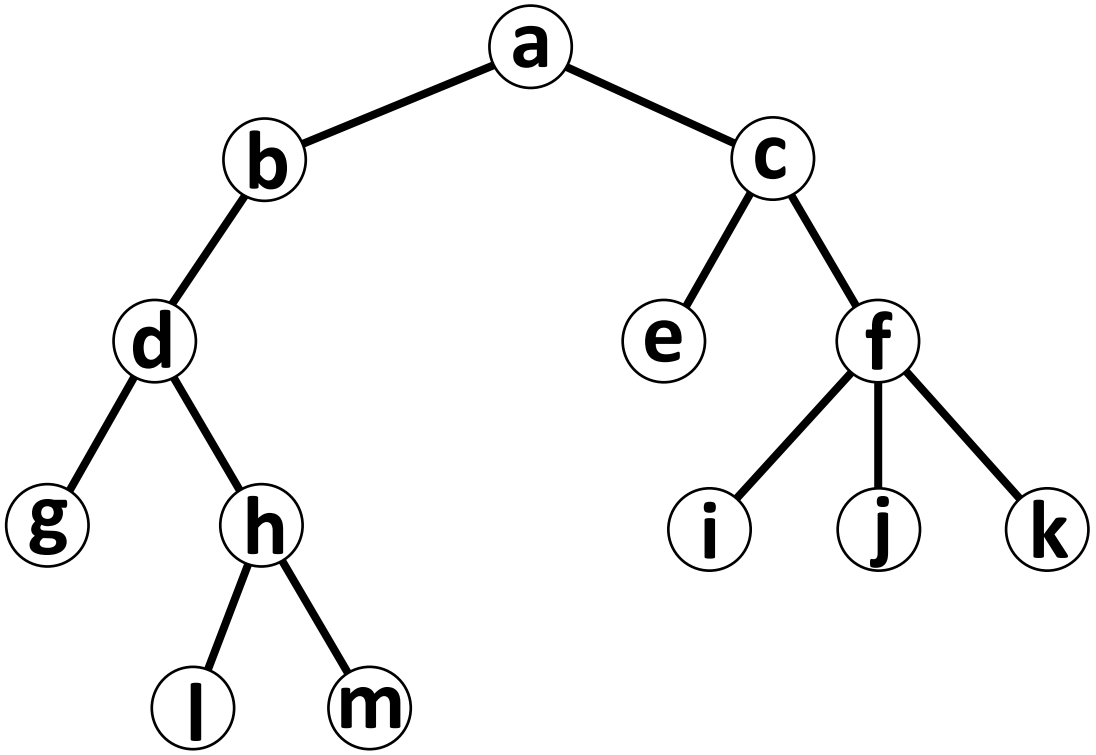
Maximum Matching = Maximum Independent Edge Set

Reduction!

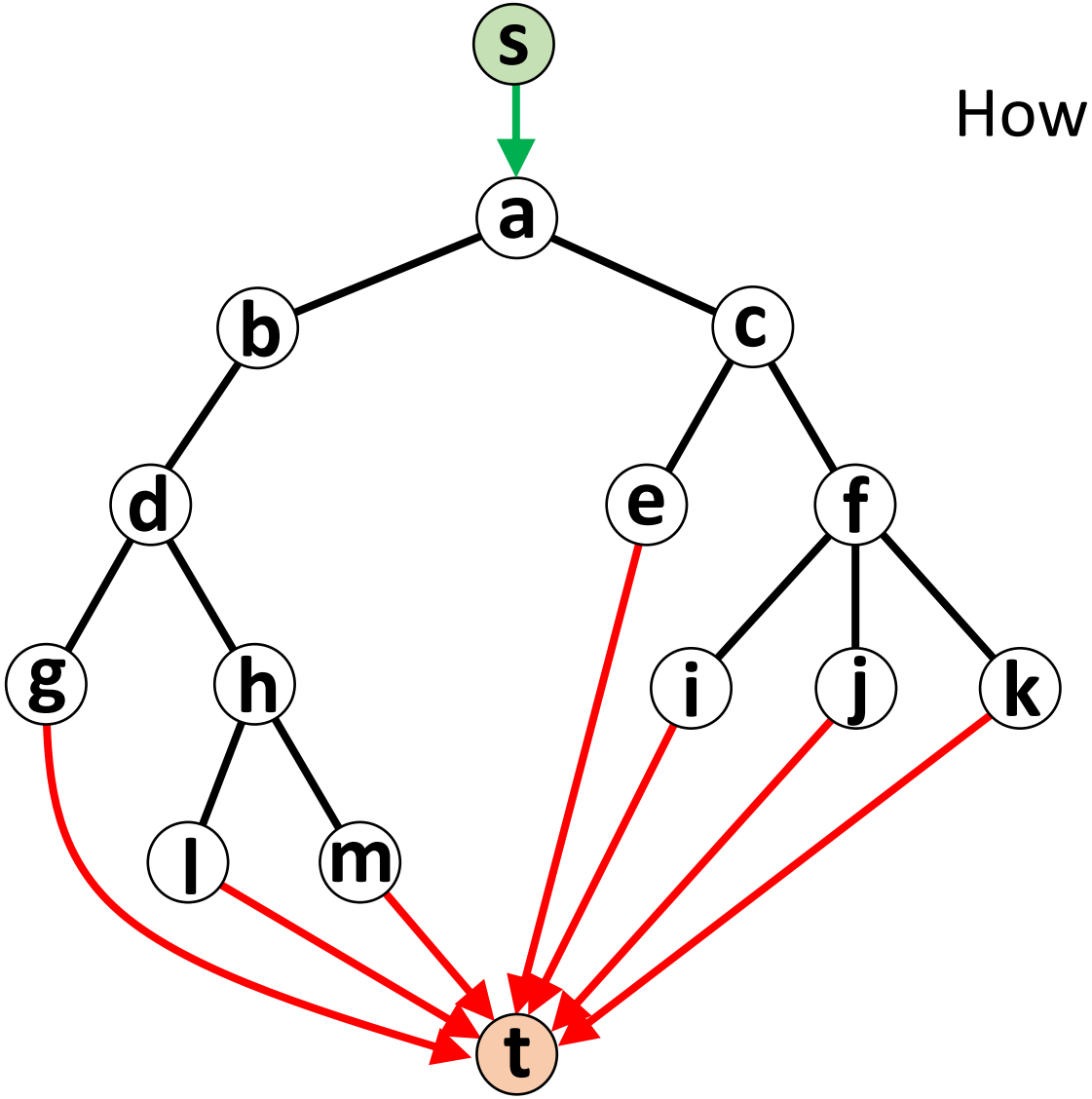
Maximum Matching Input → Flow Network Input → **Max Flow Algorithm** → Max Flow → Maximum Matching

Maximum Matching

How should we start?

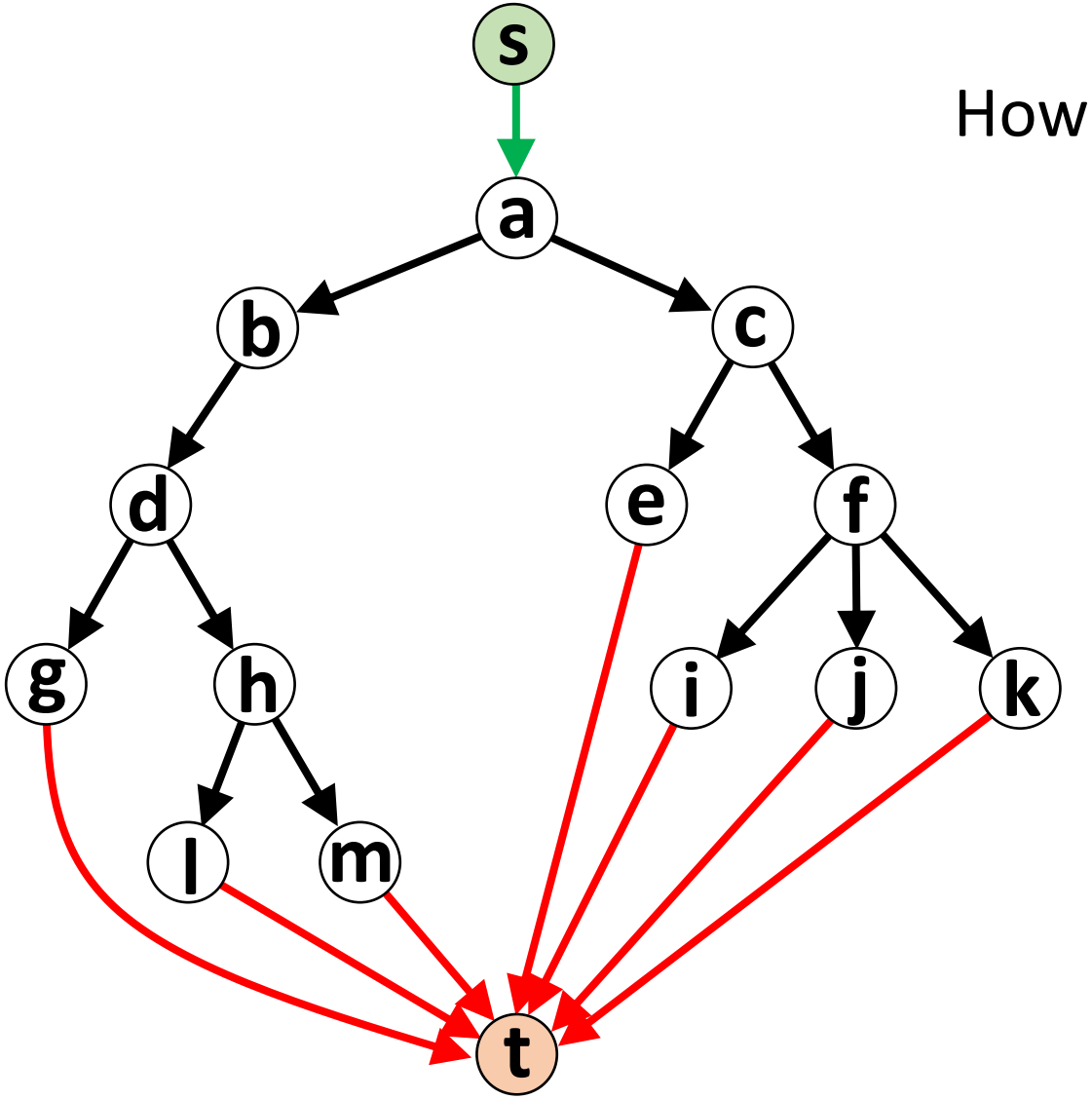


Maximum Matching



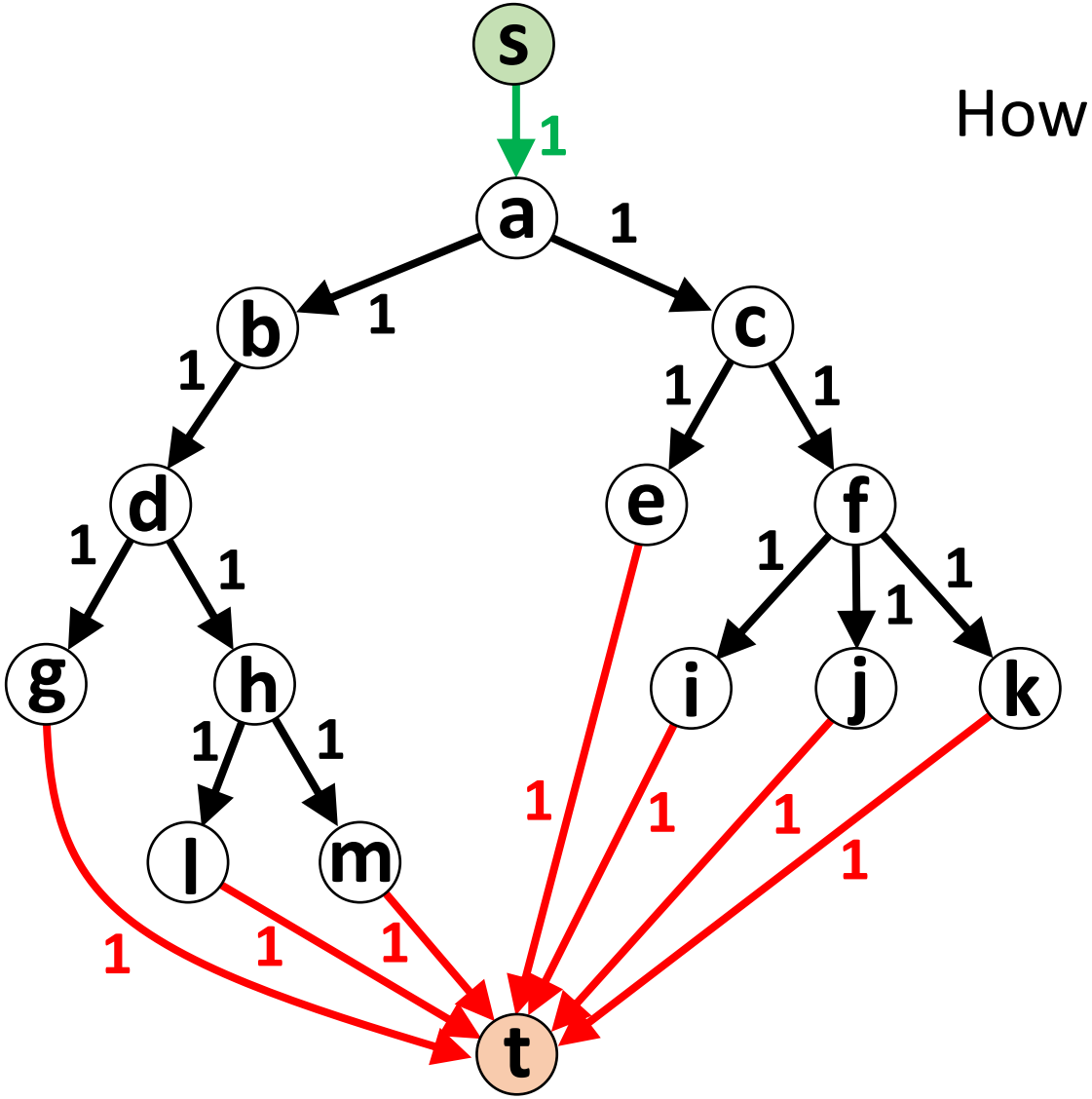
How should we start?

Maximum Matching



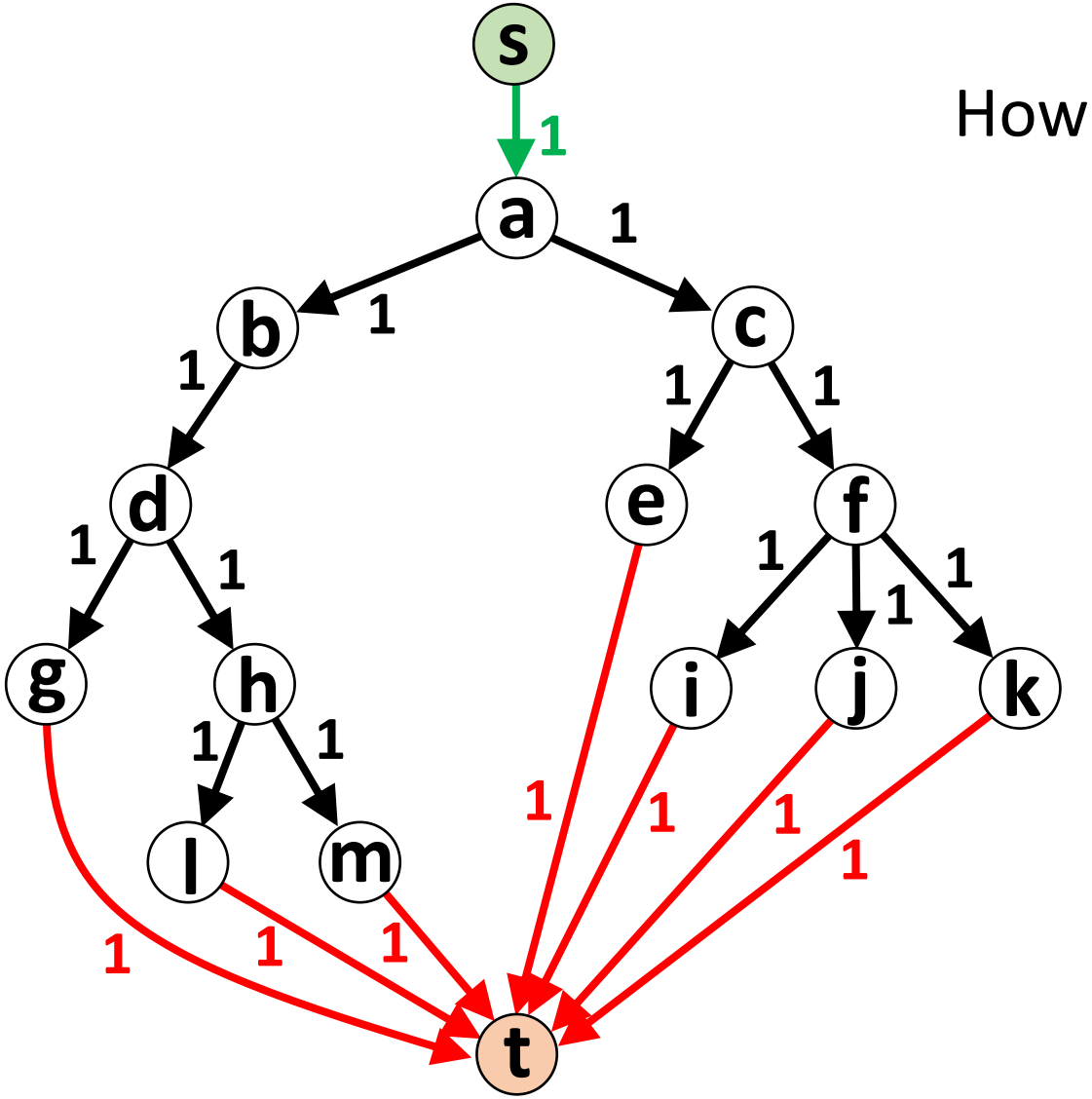
How should we start?

Maximum Matching



How should we start?

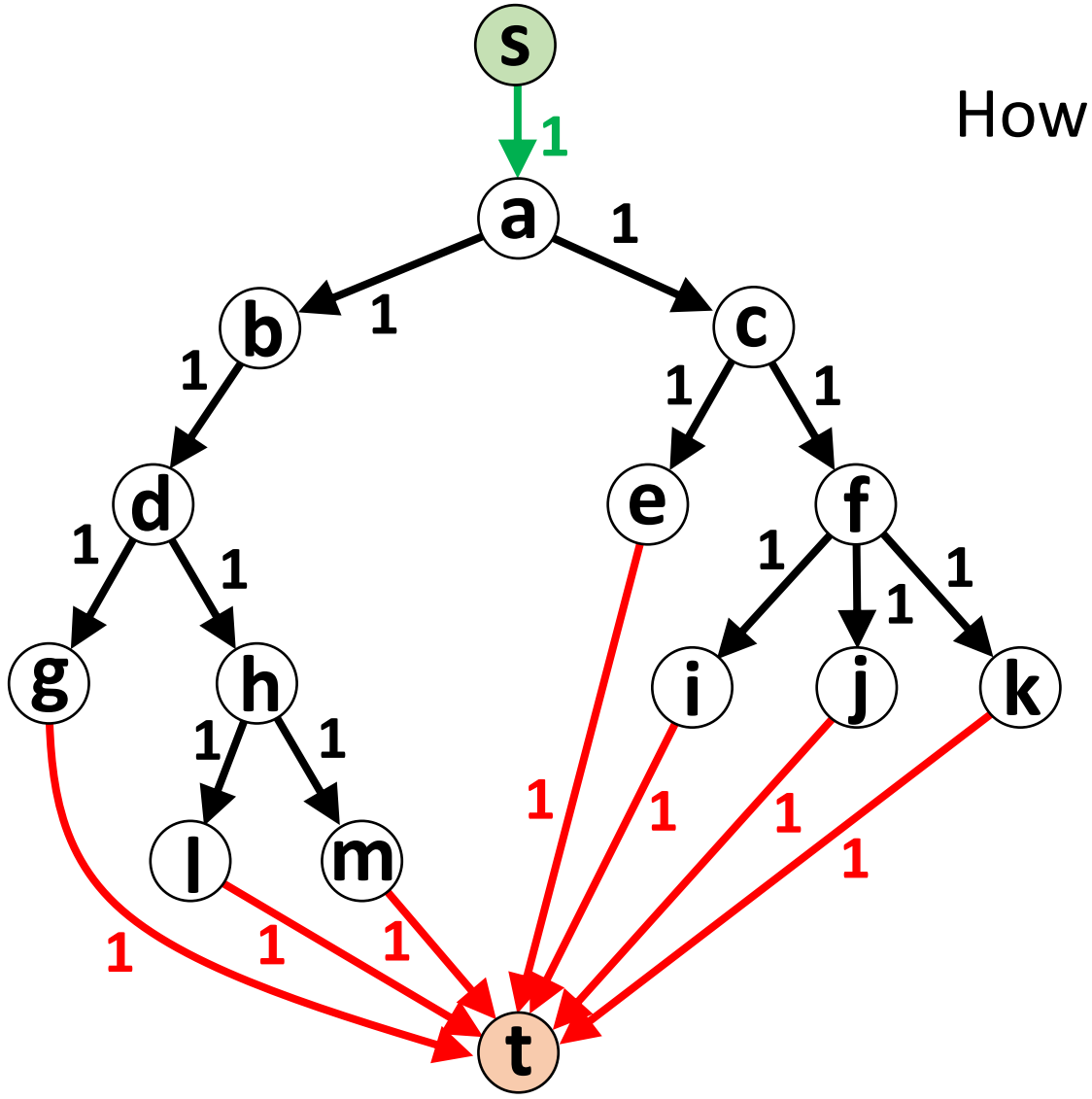
Maximum Matching



How should we start?

What do we think? Any successes? Failures?

Maximum Matching

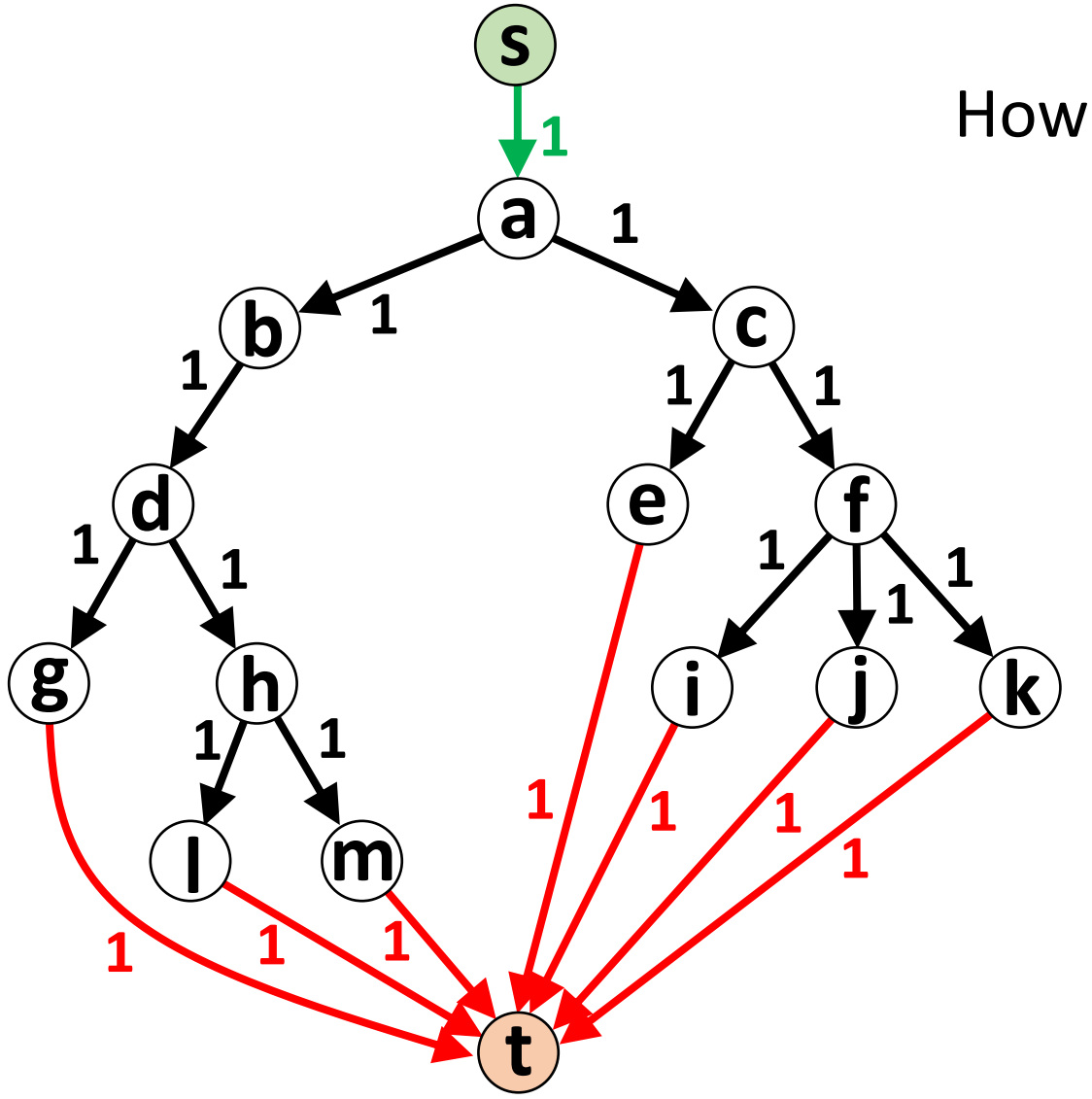


How should we start?

What do we think? Any successes? Failures?

+ It's a flow network.
- Conservation of flow means neighboring edges will host flow.

Maximum Matching



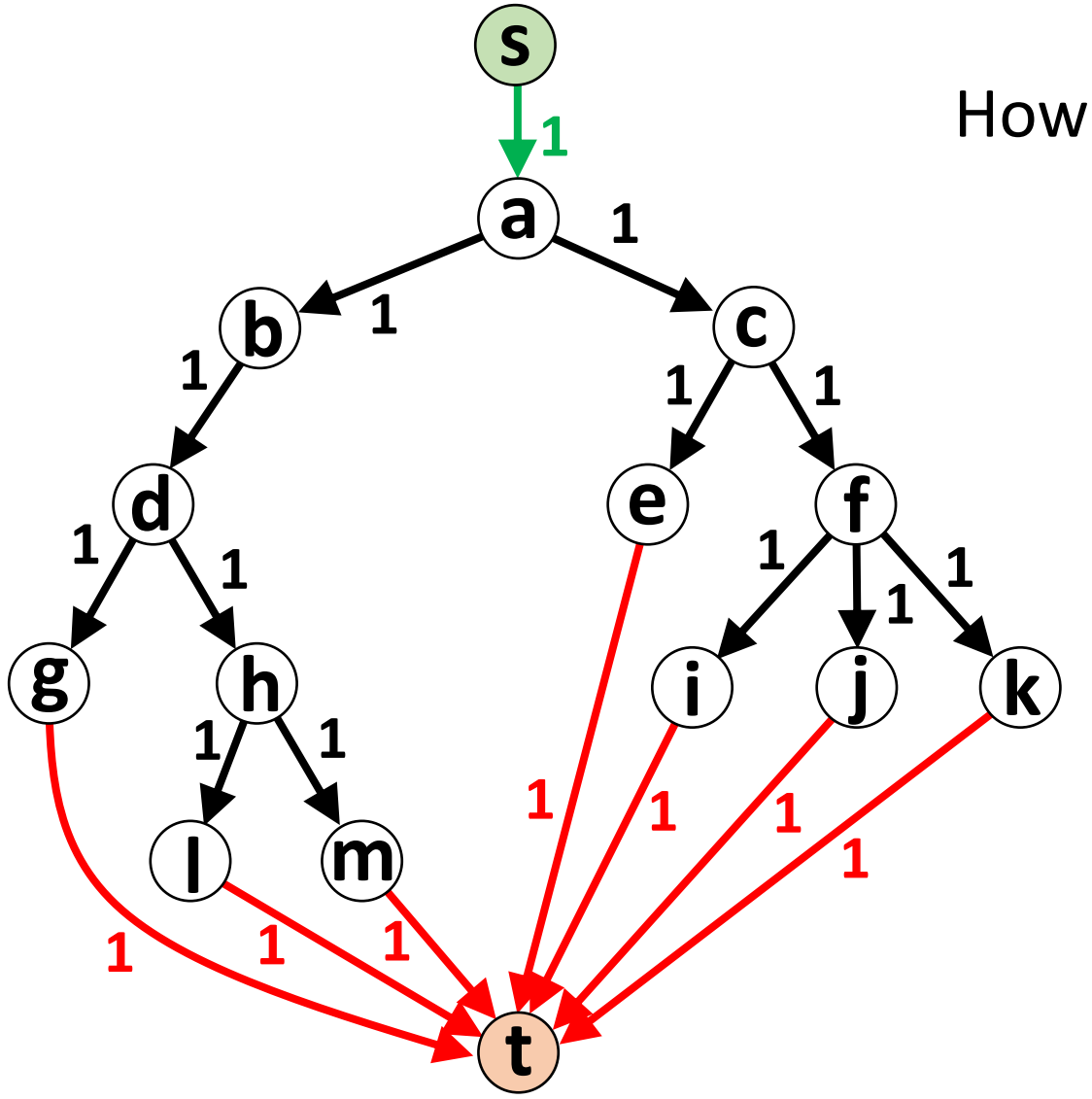
How should we start?

What do we think? Any successes? Failures?

+ It's a flow network.
- Conservation of flow means neighboring edges will host flow.

Lessons?

Maximum Matching



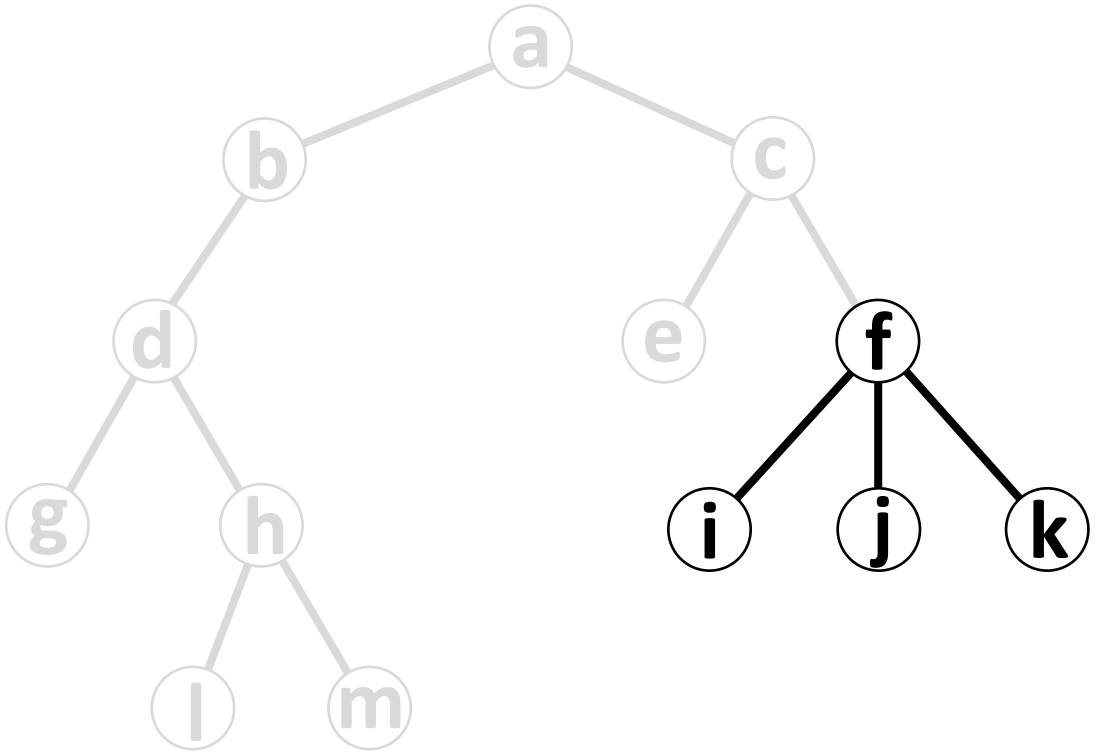
How should we start?

What do we think? Any successes? Failures?

+ It's a flow network.
- Conservation of flow means neighboring edges will host flow.

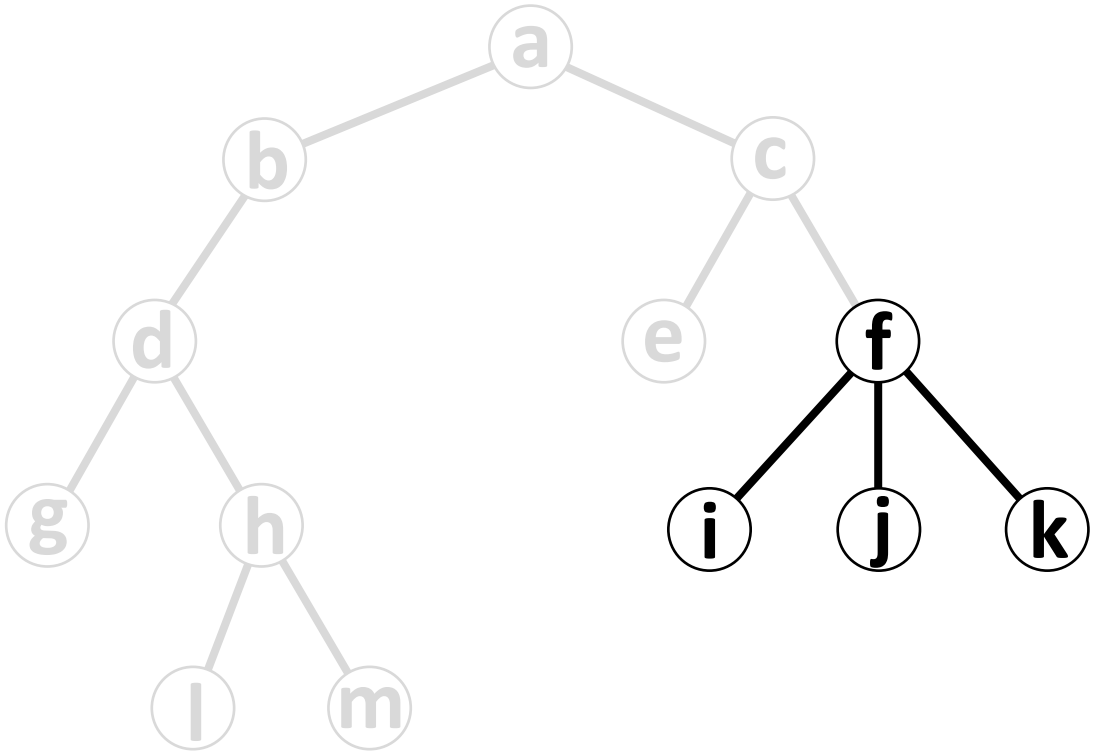
Lessons?
Need a way to select individual edges, not paths of edges.

Maximum Matching



How many edges can be selected here?

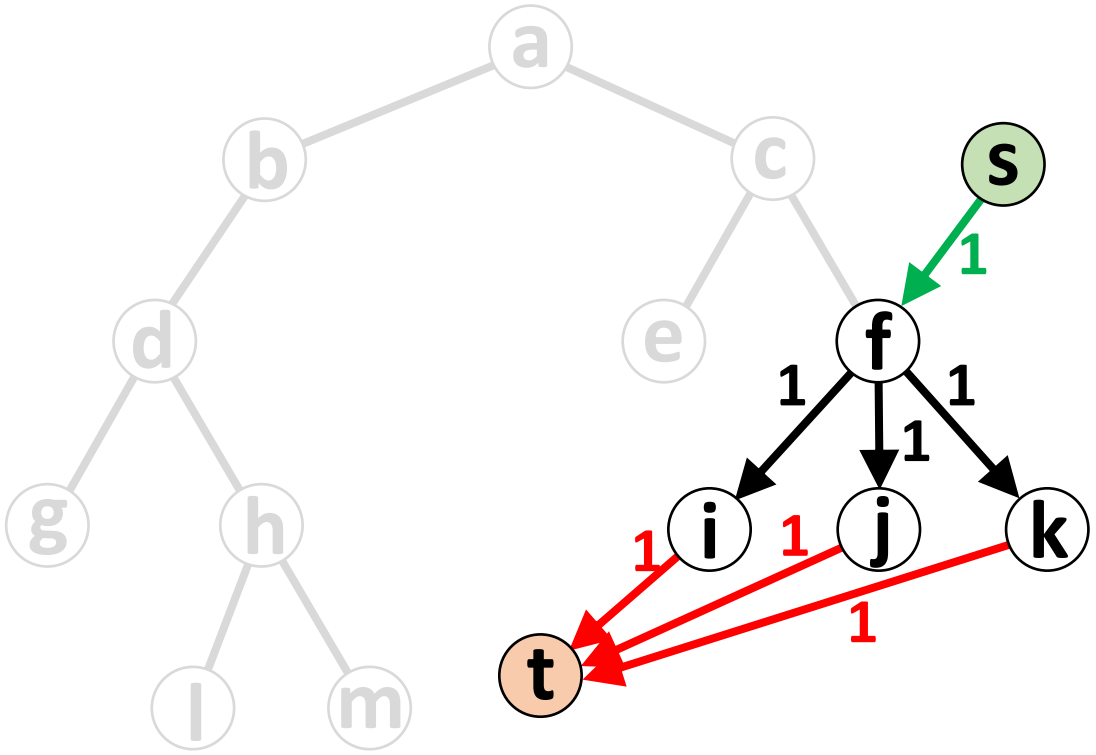
Maximum Matching



How many edges can be selected here?

How can we turn this into a flow network so that only one of the edges is selected?

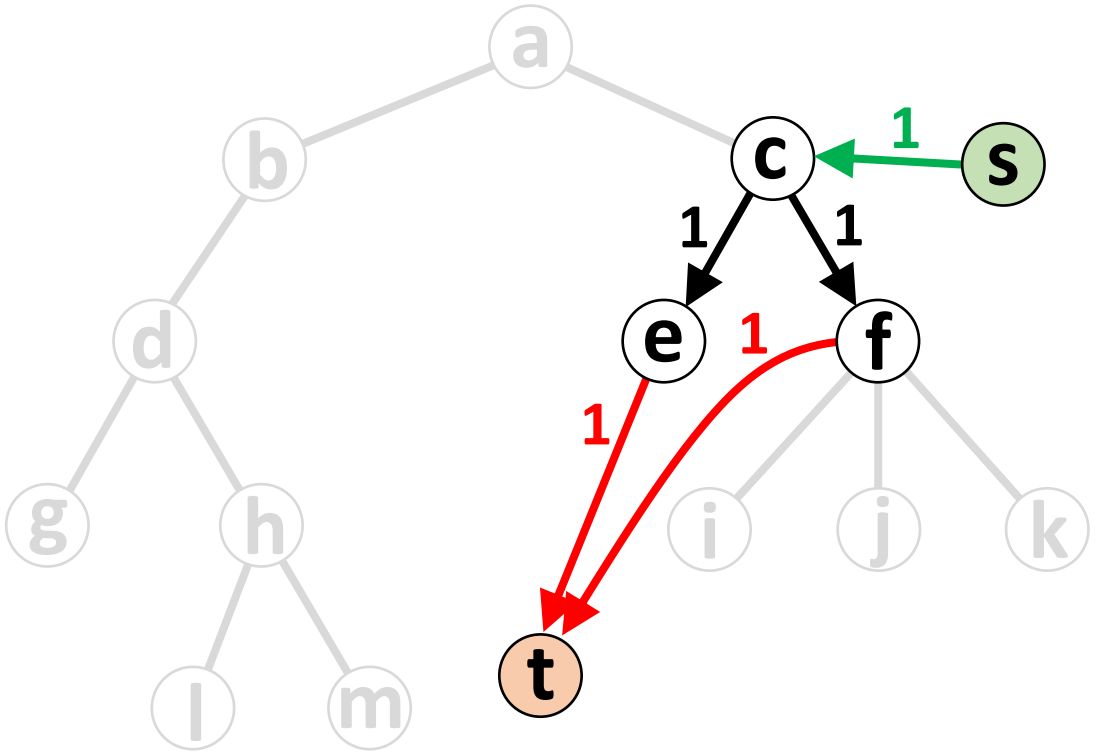
Maximum Matching



How many edges can be selected here?

How can we turn this into a flow network so that only one of the edges is selected?

Maximum Matching

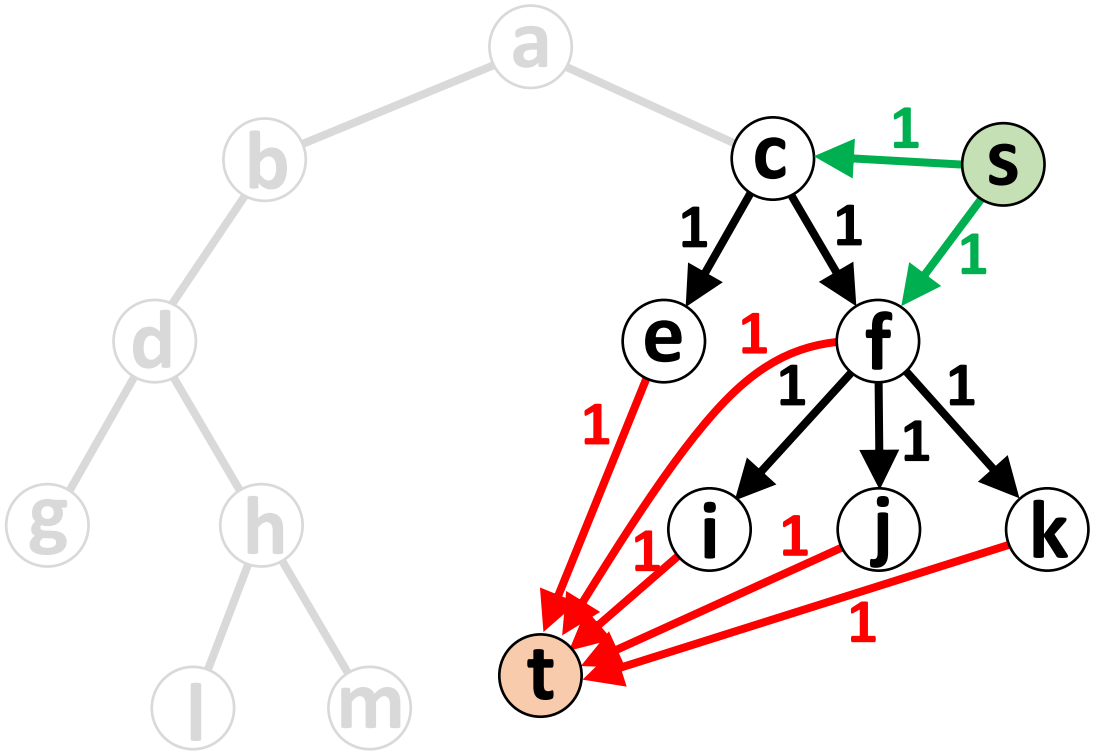


How many edges can be selected here?

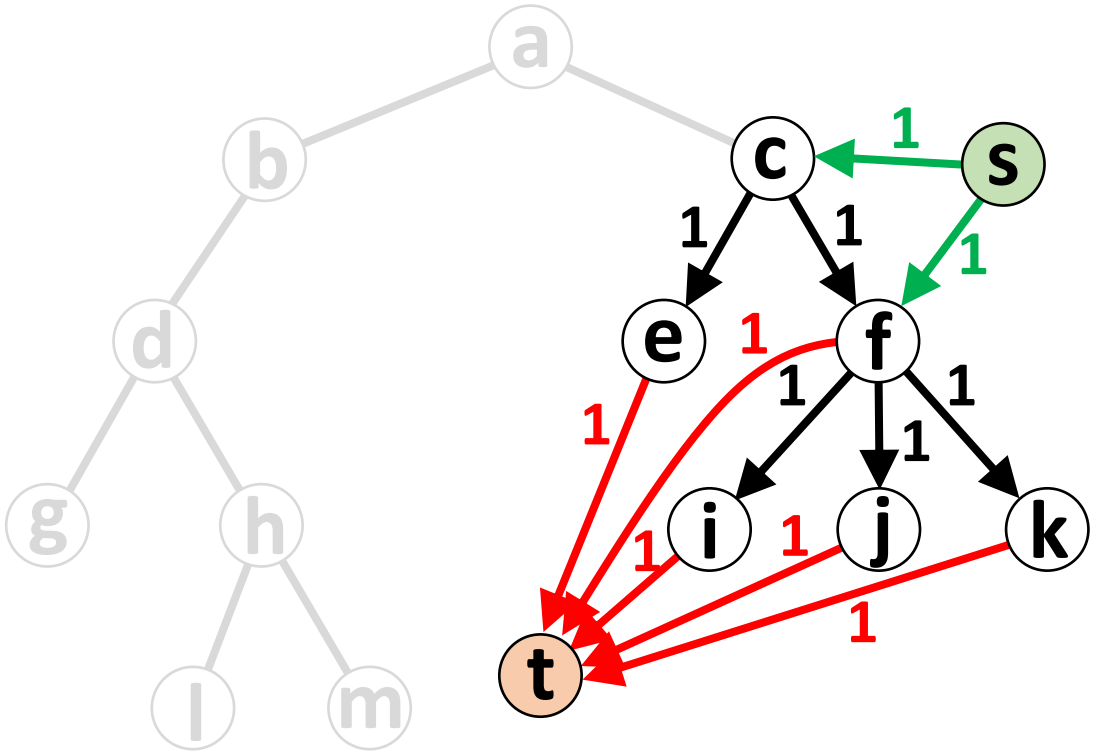
How can we turn this into a flow network so that only one of the edges is selected?

Maximum Matching

Any problems?



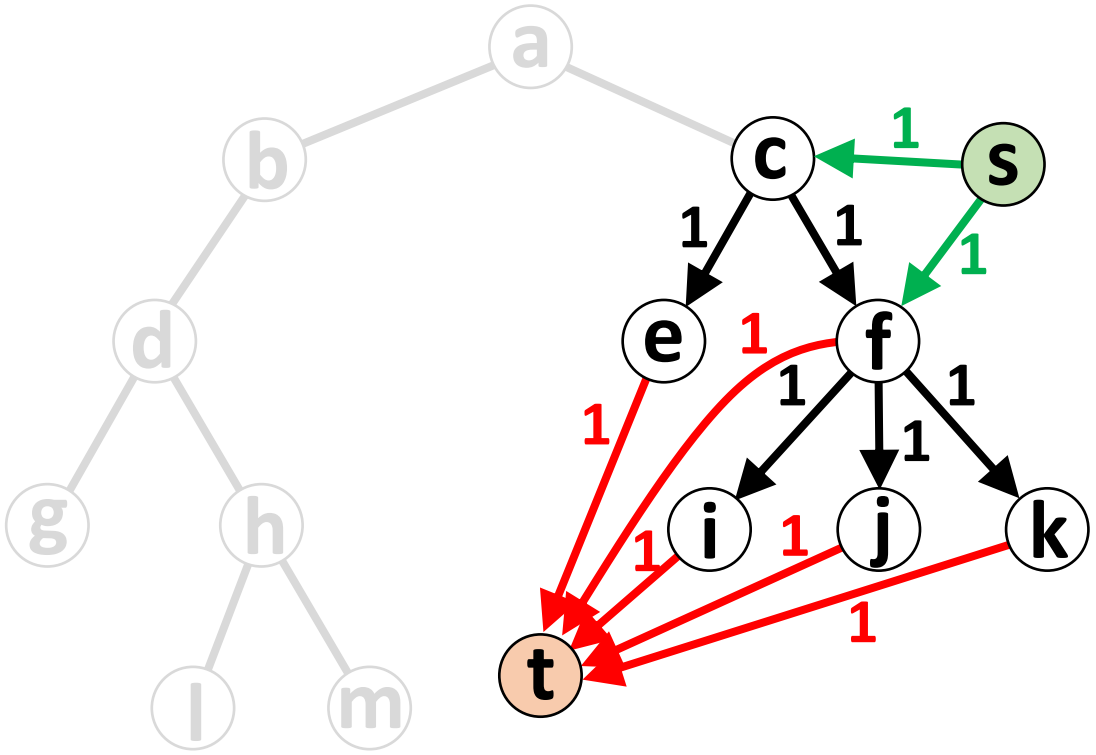
Maximum Matching



Any problems?

- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

Maximum Matching

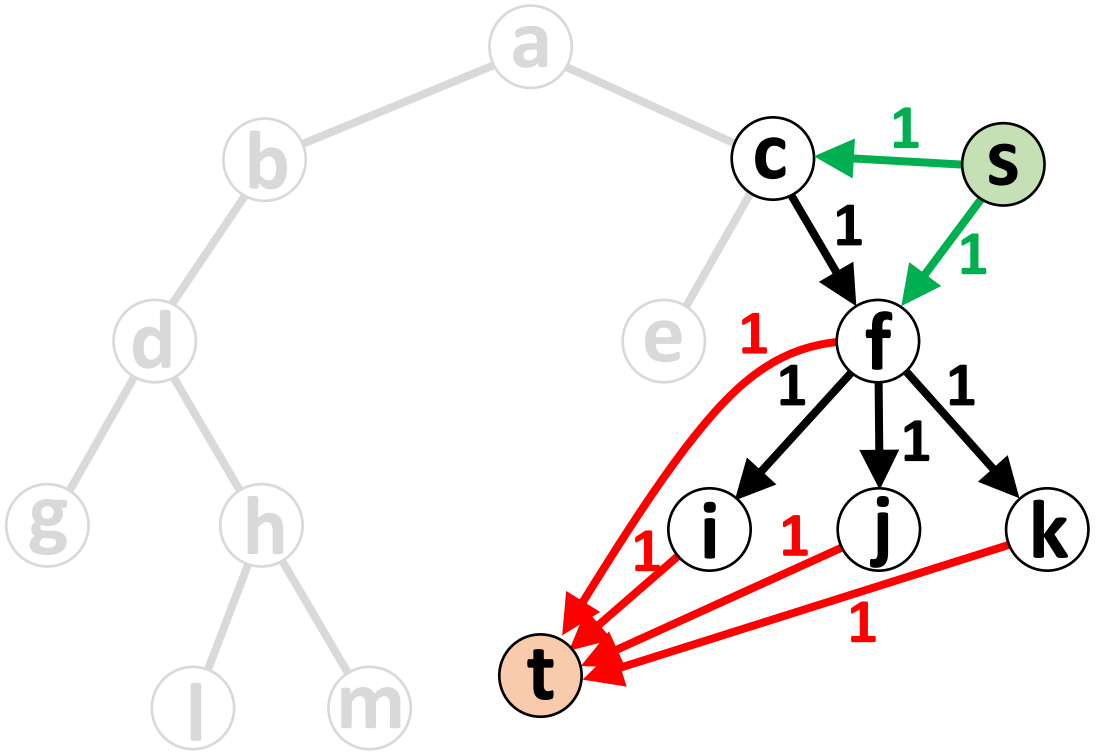


Any problems?

- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

How can we fix it?

Maximum Matching

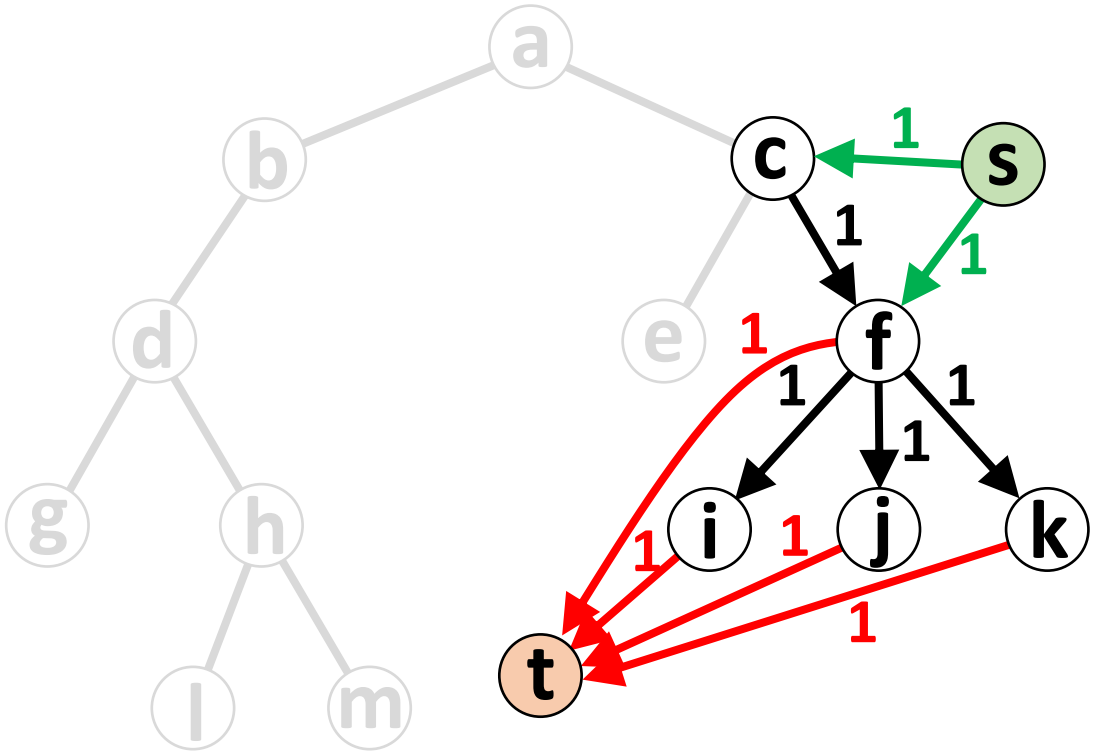


Any problems?

- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

How can we fix it?

Maximum Matching



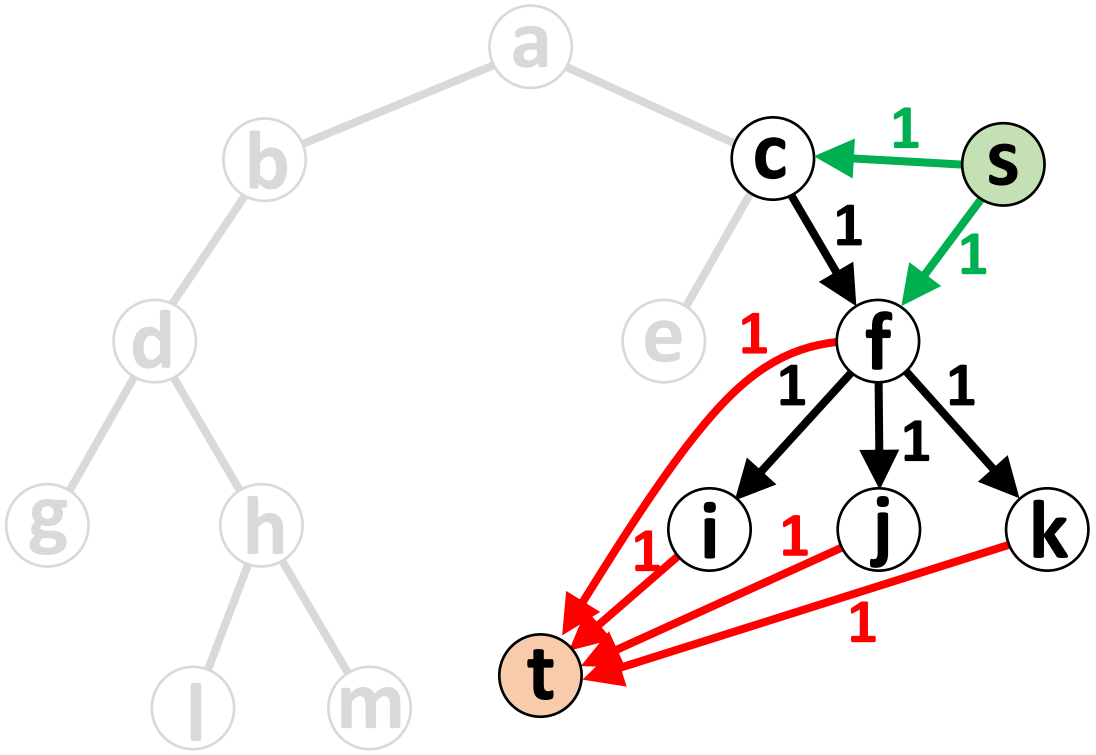
f can connect to at most 1 edge, so give it one unit of flow to ‘purchase’ an edge with.

Any problems?

- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

How can we fix it?

Maximum Matching



f can connect to at most 1 edge, so give it one unit of flow to ‘purchase’ an edge with.

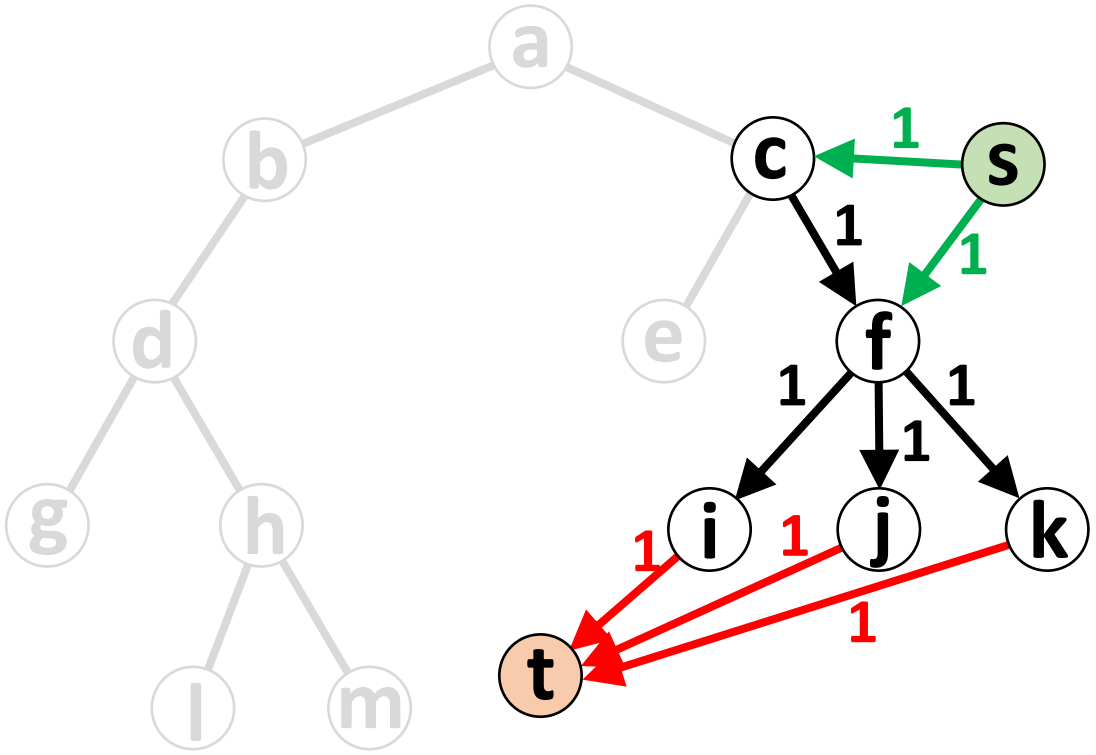
Any problems?

- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

How can we fix it?

- **f** cannot directly connect to both **s** and **t**.

Maximum Matching



f can connect to at most 1 edge, so give it one unit of flow to ‘purchase’ an edge with.

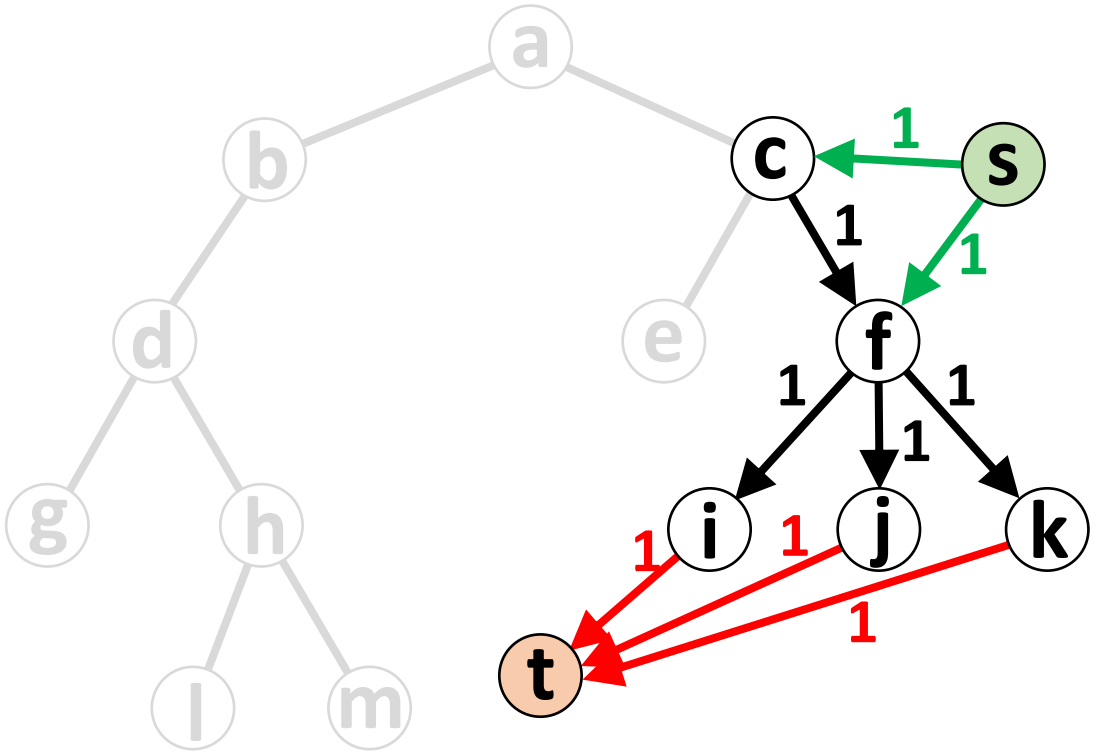
Any problems?

- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

How can we fix it?

- **f** cannot directly connect to both **s** and **t**.

Maximum Matching



f can connect to at most 1 edge, so give it one unit of flow to ‘purchase’ an edge with.

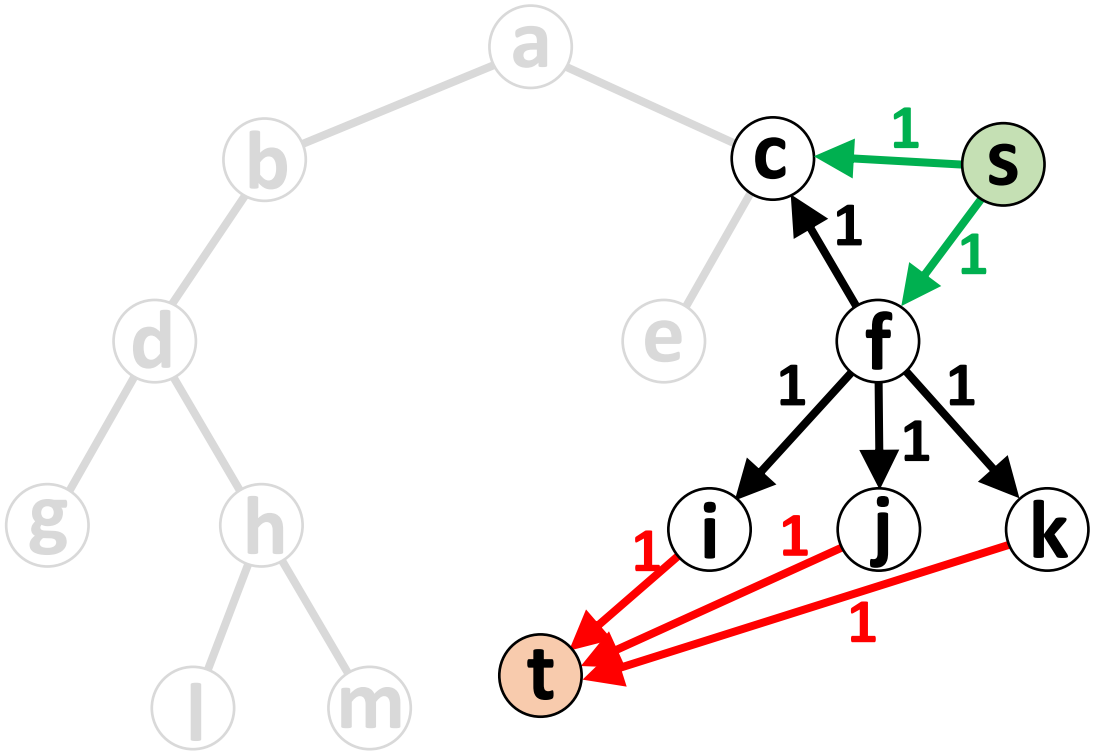
Any problems?

- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

How can we fix it?

- **f** cannot directly connect to both **s** and **t**.
- **f** cannot get flow other than from **s**.

Maximum Matching



f can connect to at most 1 edge, so give it one unit of flow to ‘purchase’ an edge with.

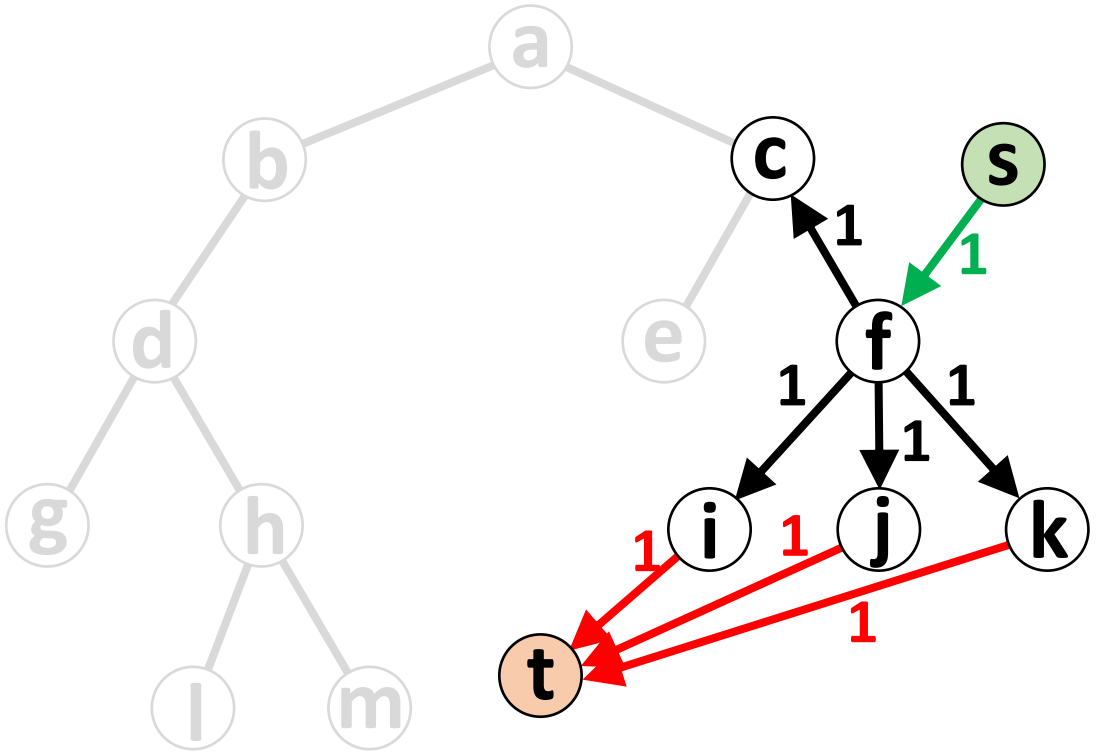
Any problems?

- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

How can we fix it?

- **f** cannot directly connect to both **s** and **t**.
- **f** cannot get flow other than from **s**.

Maximum Matching



f can connect to at most 1 edge, so give it one unit of flow to ‘purchase’ an edge with.

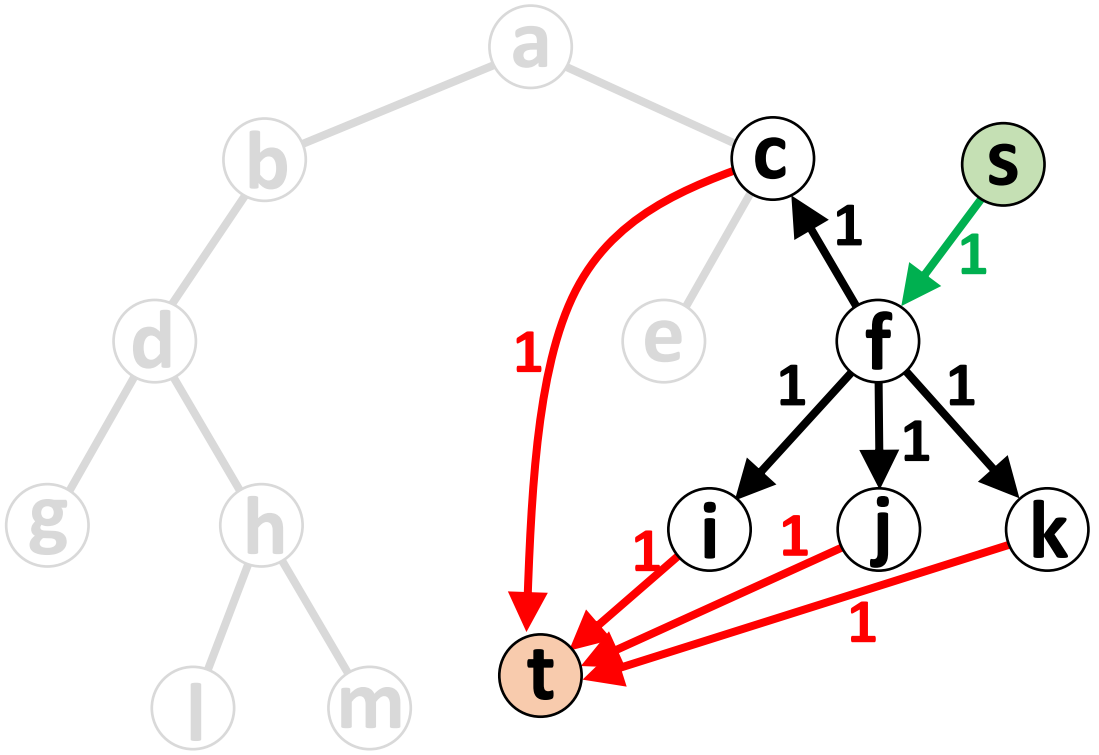
Any problems?

- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

How can we fix it?

- **f** cannot directly connect to both **s** and **t**.
- **f** cannot get flow other than from **s**.

Maximum Matching



f can connect to at most 1 edge, so give it one unit of flow to ‘purchase’ an edge with.

Any problems?

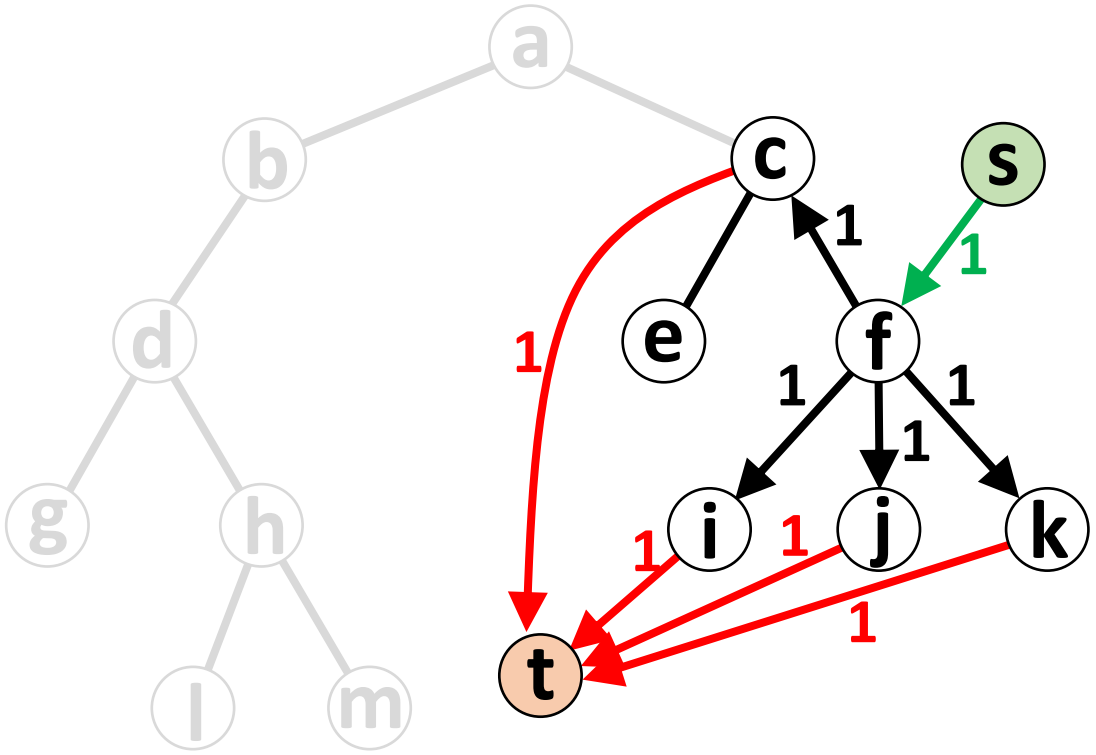
- We can deploy incompatible edges at the same time (with 1 unit of flow nonetheless!).
- We can push flow without deploying edges.

How can we fix it?

- **f** cannot directly connect to both **s** and **t**.
- **f** cannot get flow other than from **s**.

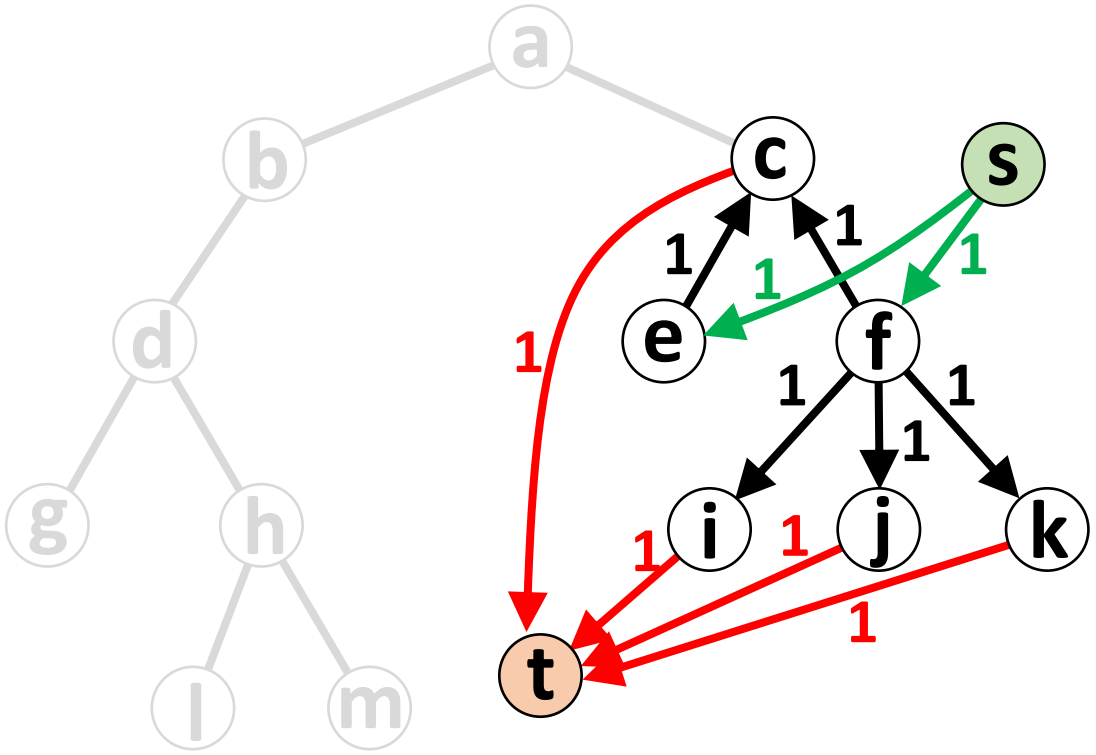
Maximum Matching

How can we incorporate **e**?



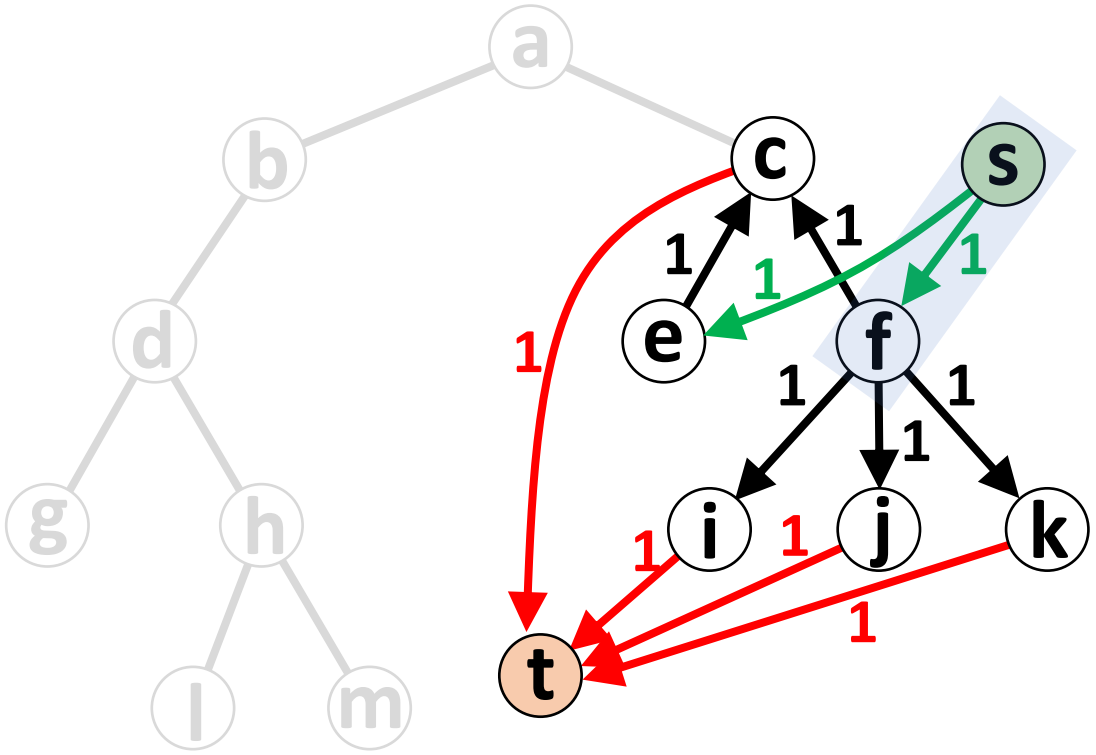
Maximum Matching

How can we incorporate **e**?



Maximum Matching

How can we incorporate **e**?



No way for f to connect to > 1 edge

+

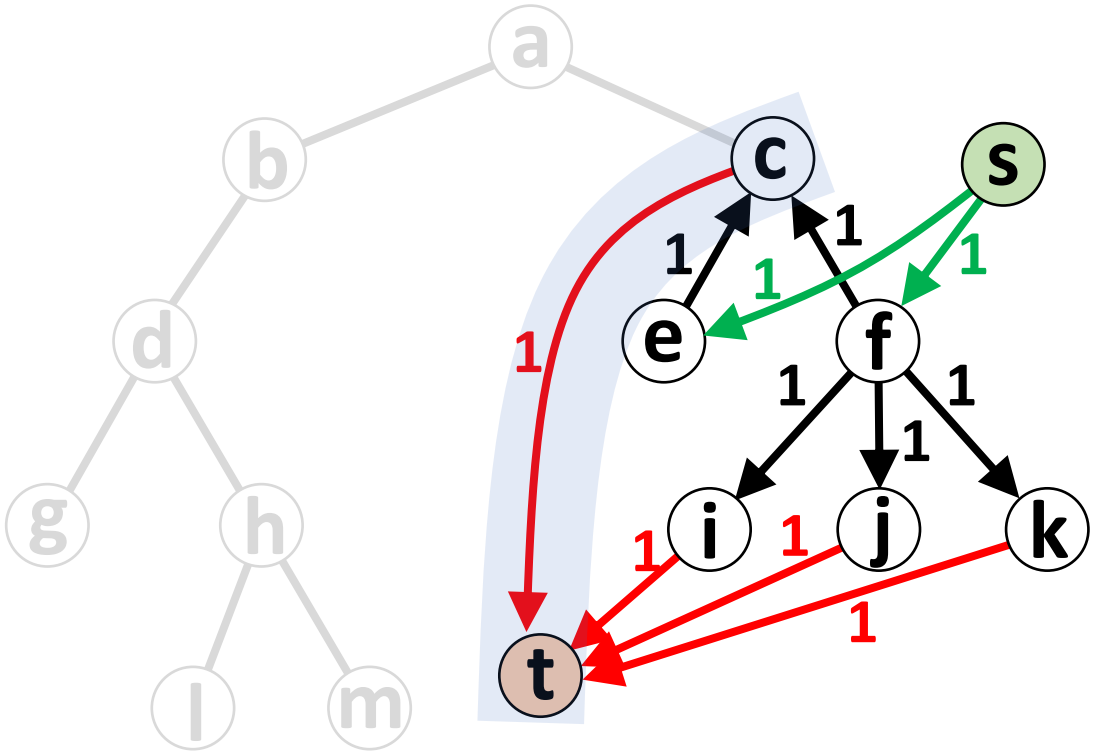
No way for c to connect to > 1 edge

=

(**e**,**c**) and (**f**,**c**) can't both deploy.

Maximum Matching

How can we incorporate **e**?



No way for f to connect to > 1 edge

+

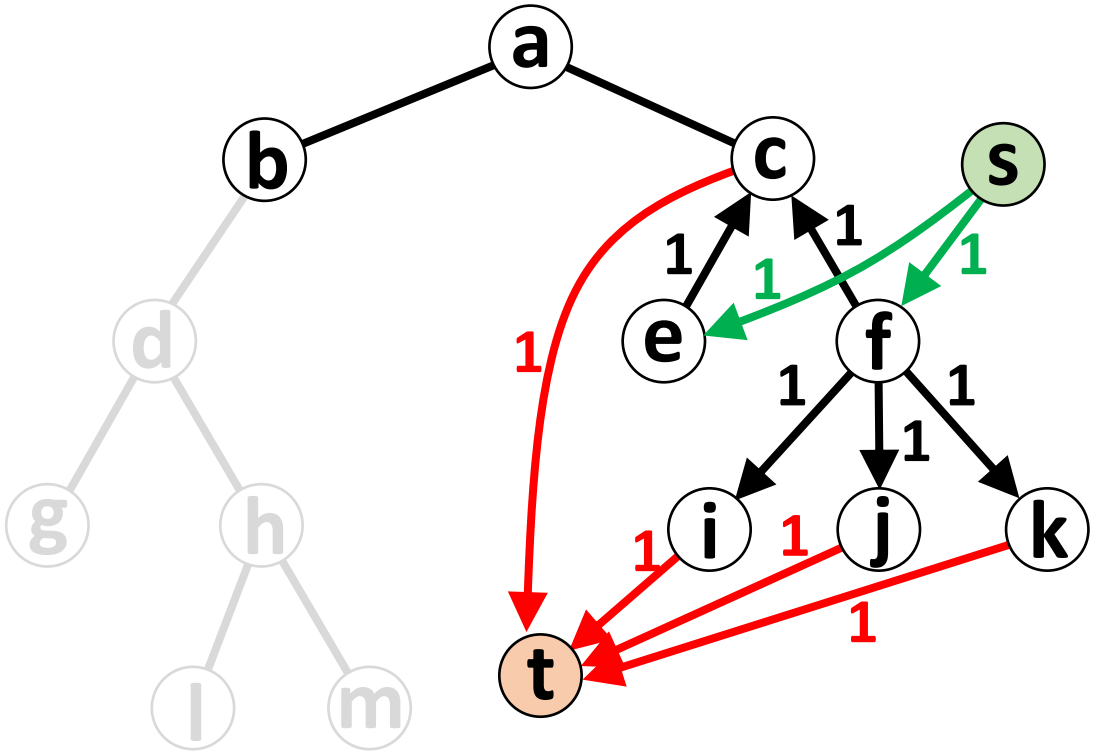
No way for c to connect to > 1 edge

=

(**e**,**c**) and (**f**,**c**) can't both deploy.

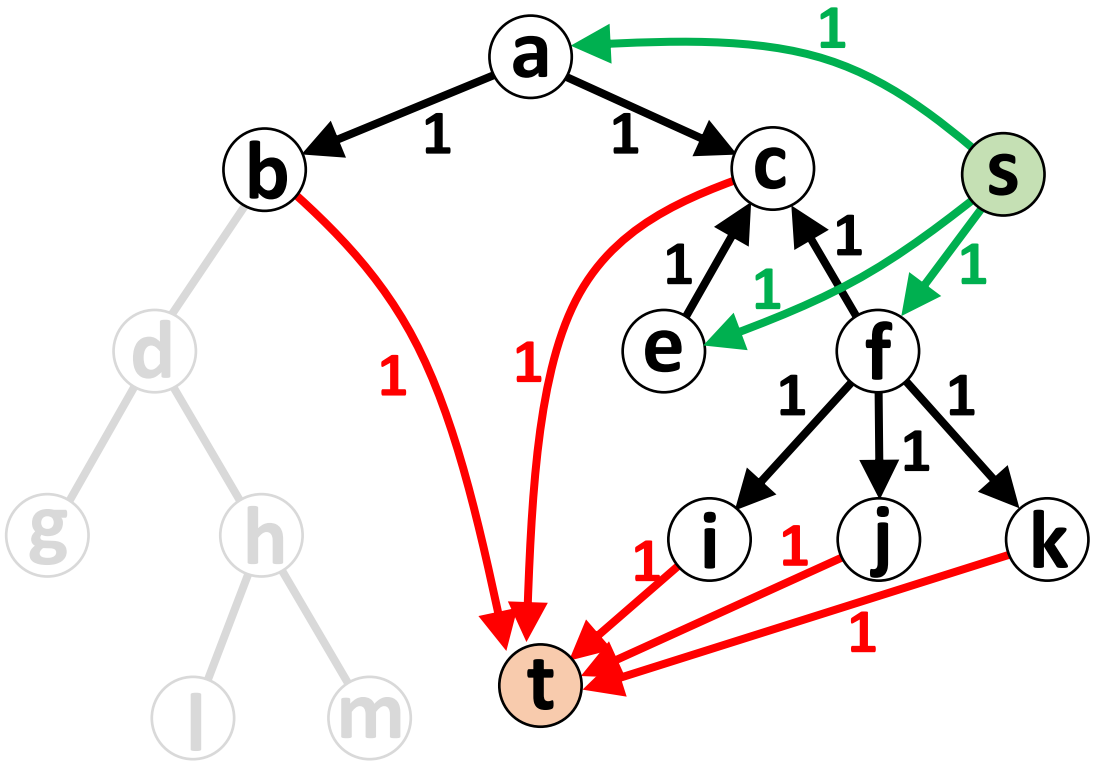
Maximum Matching

How can we incorporate **a**?



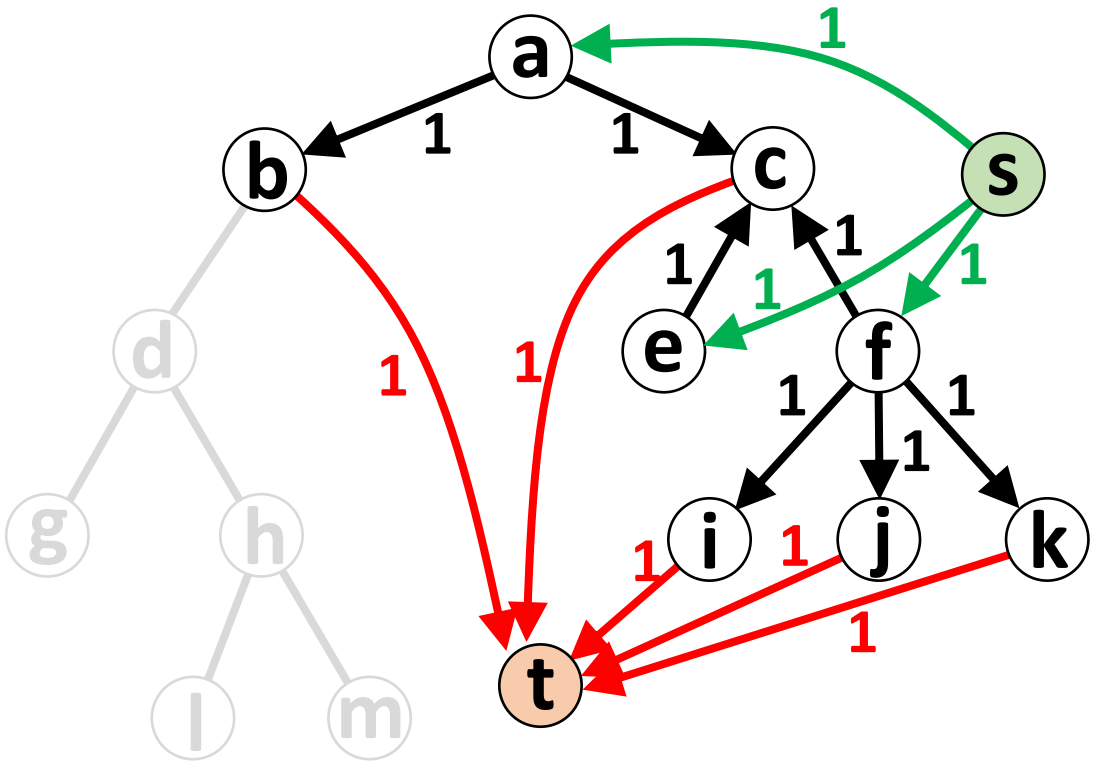
Maximum Matching

How can we incorporate **a**?

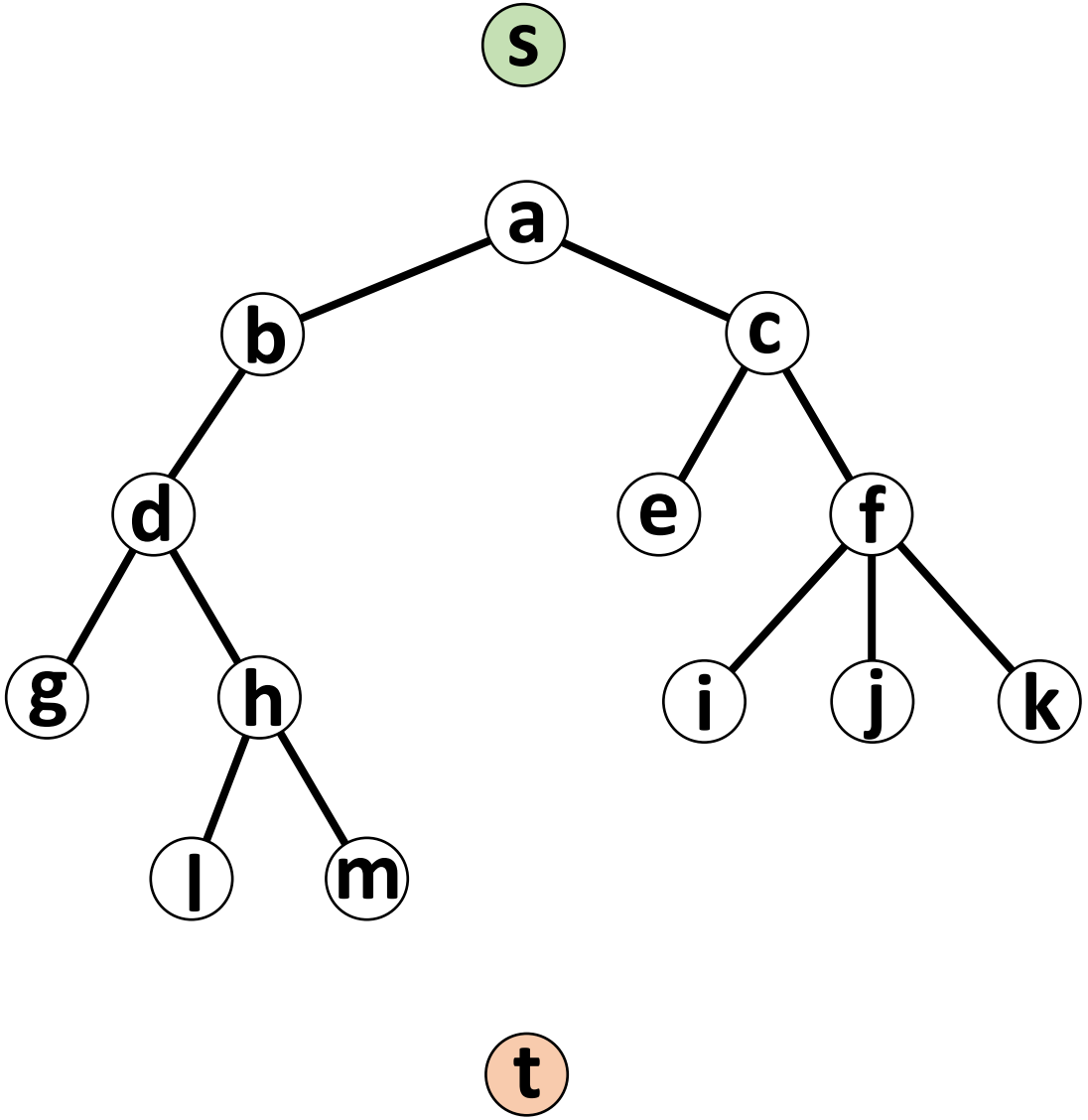


Maximum Matching

How do we do this in general?

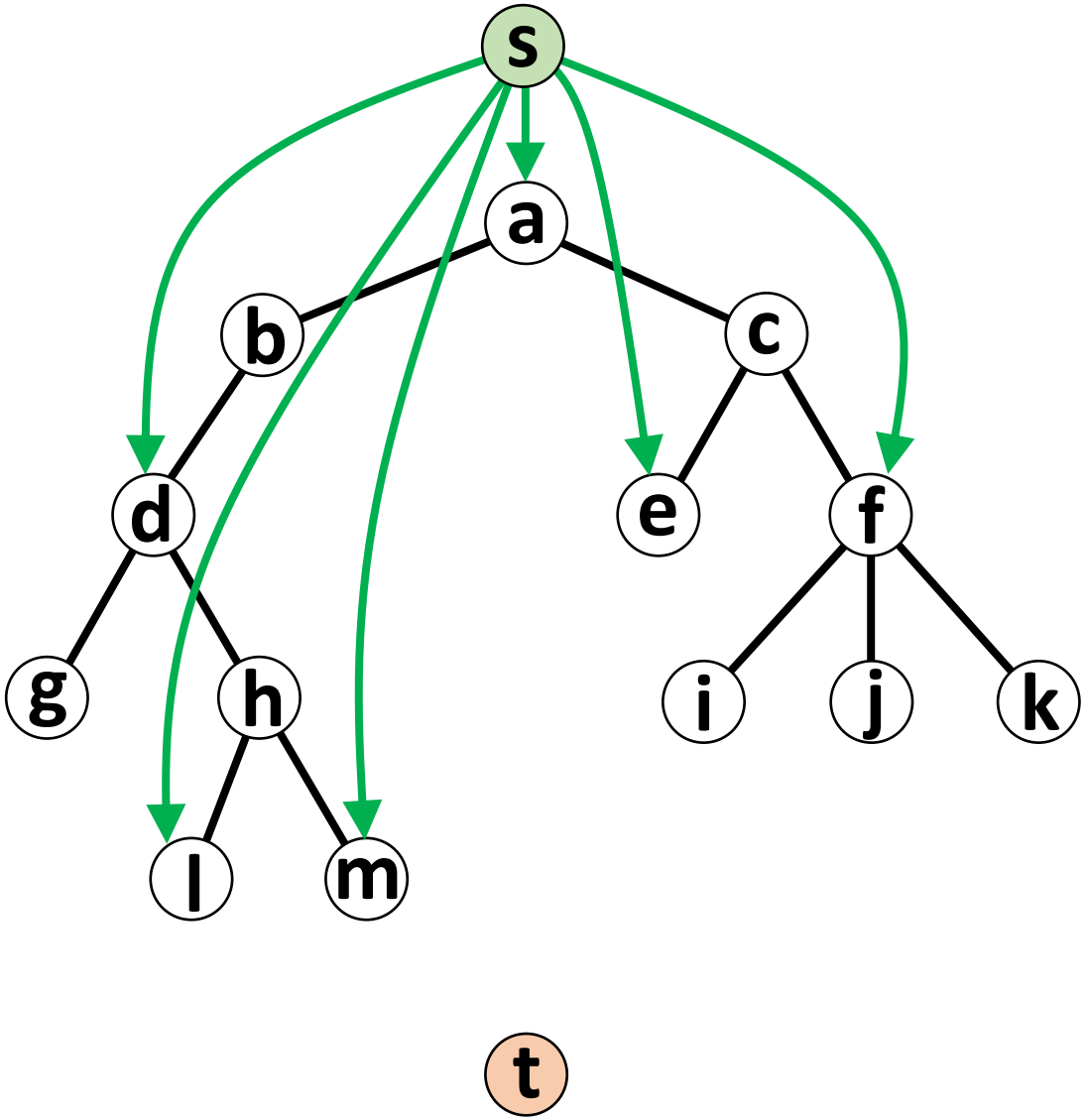


Maximum Matching



How do we do this in general?

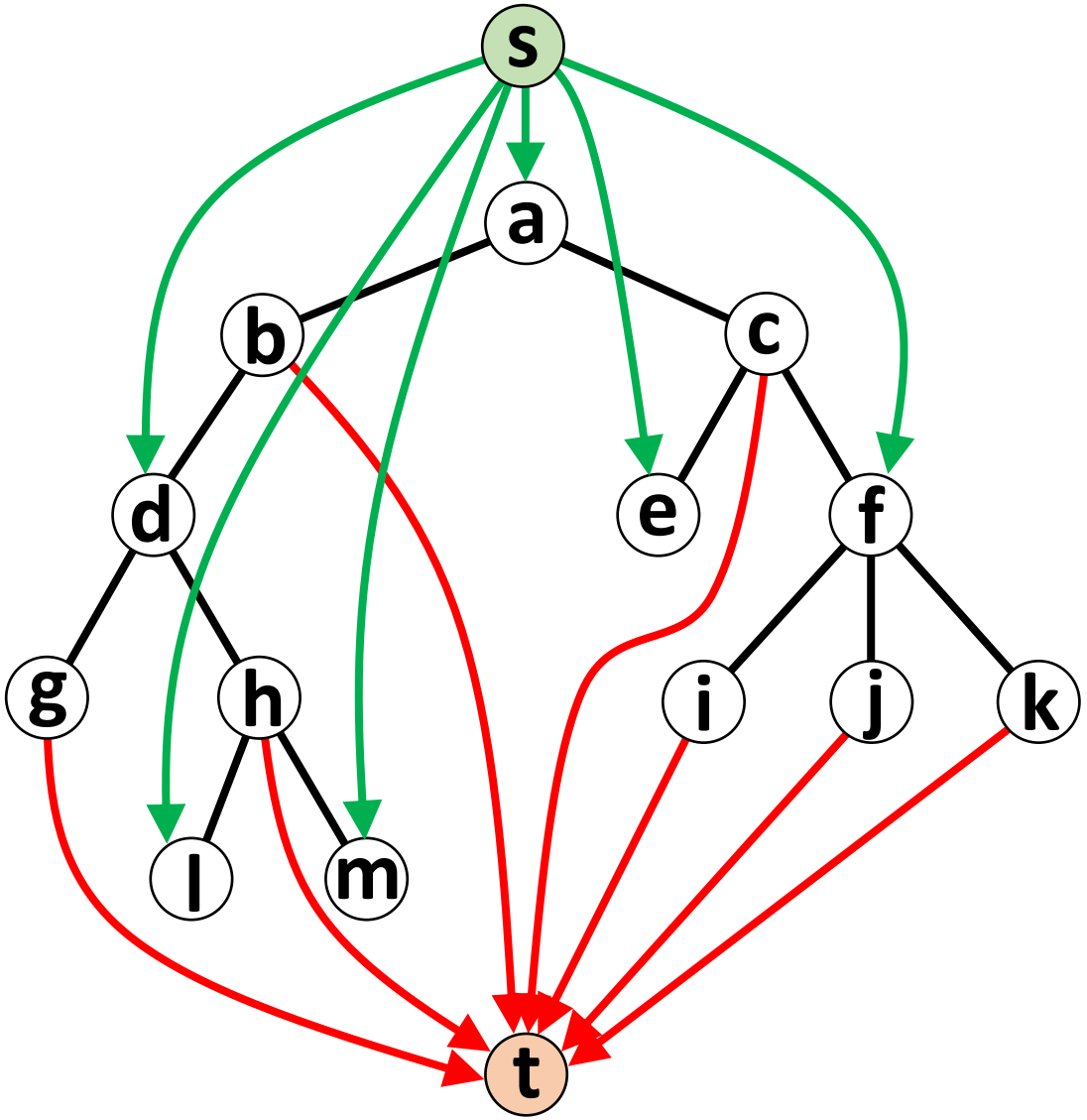
Maximum Matching



How do we do this in general?

1. Starting at the root, connect every other generation with edge from **s**.

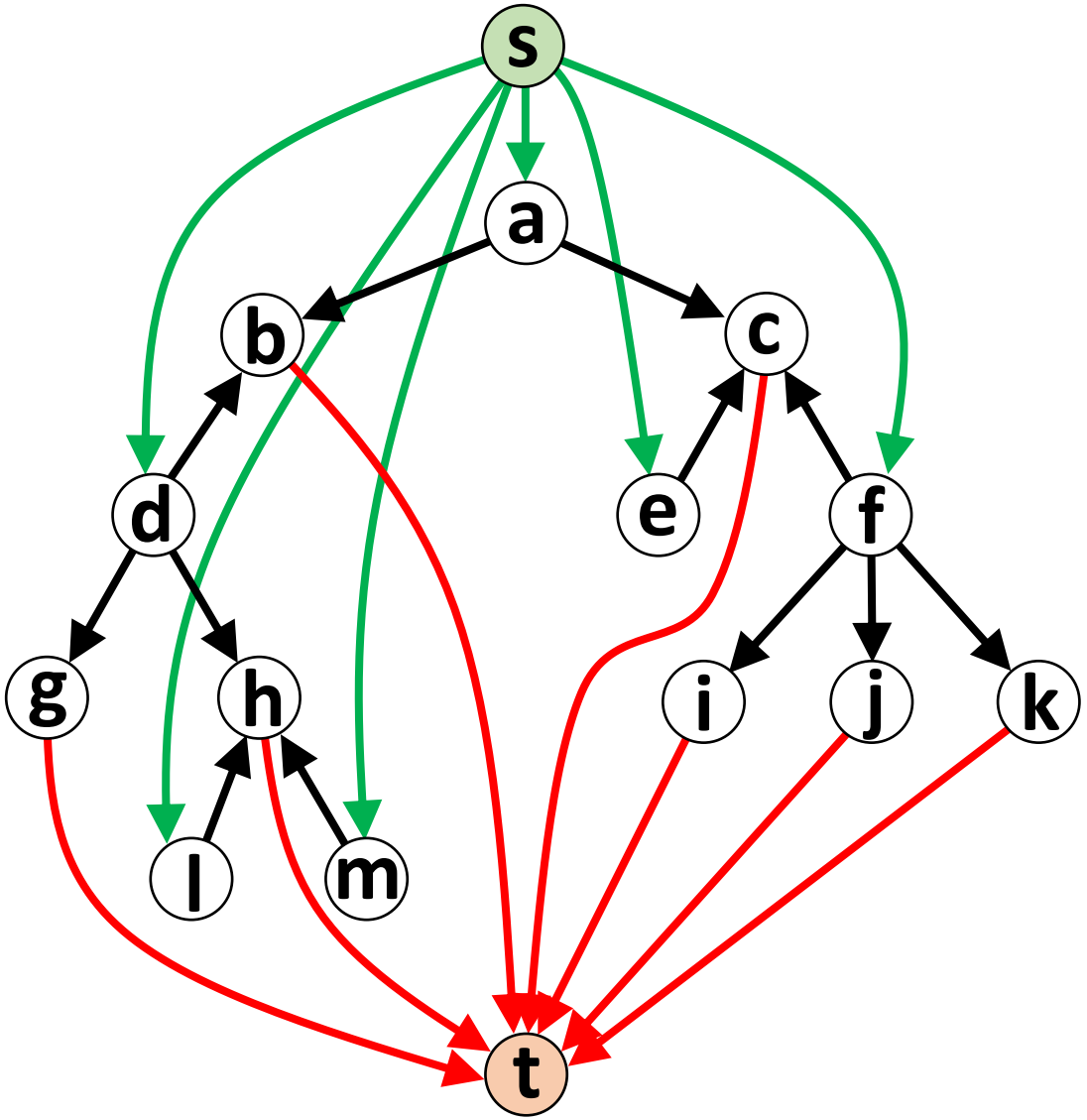
Maximum Matching



How do we do this in general?

1. Starting at the root, connect every other generation with edge from **s**.
2. Connect other generations with edge to **t**.

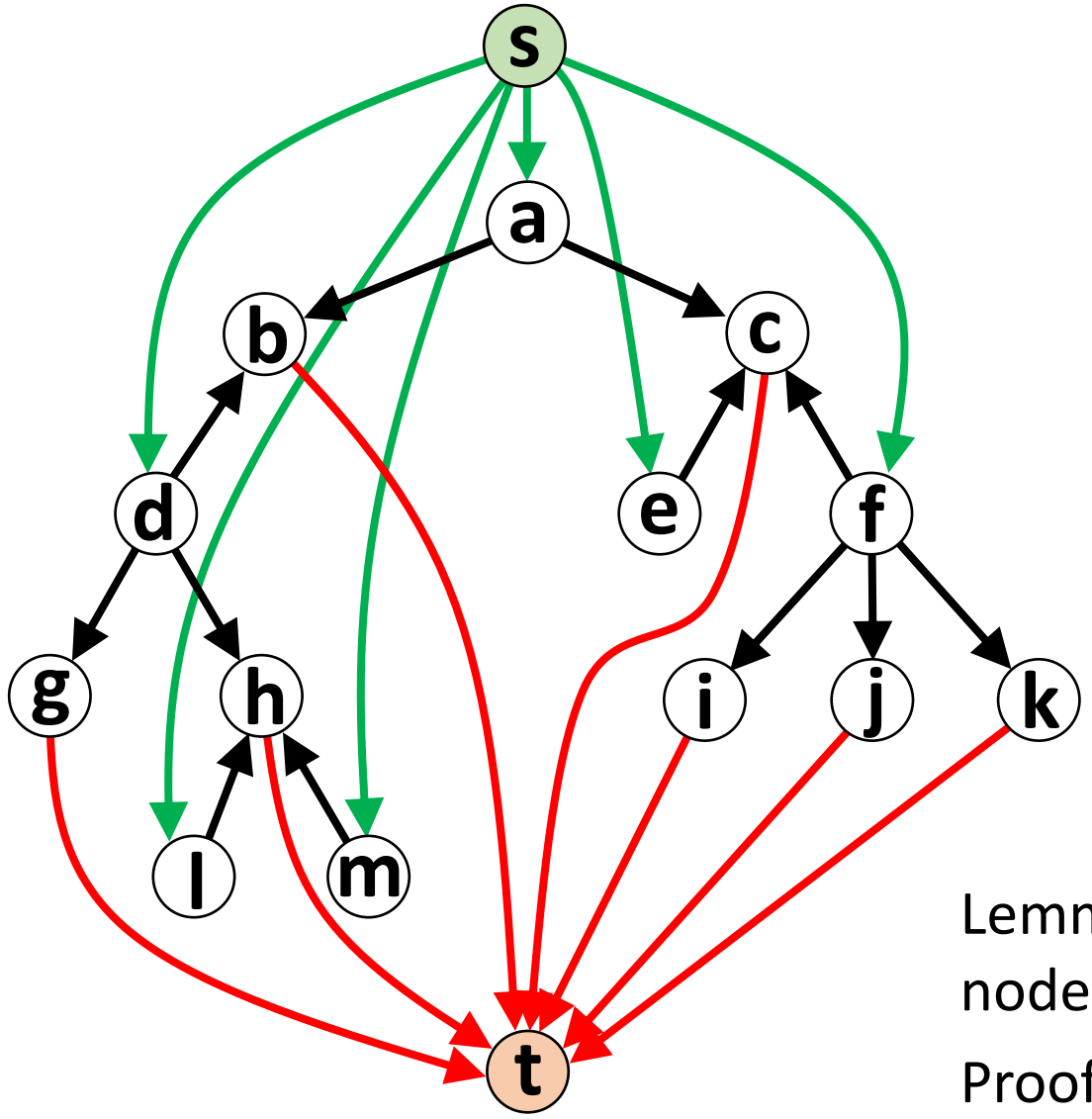
Maximum Matching



How do we do this in general?

1. Starting at the root, connect every other generation with edge from **s**.
2. Connect other generations with edge to **t**.
3. Make edges go from **s**-connected node to **t**-connected node.

Maximum Matching



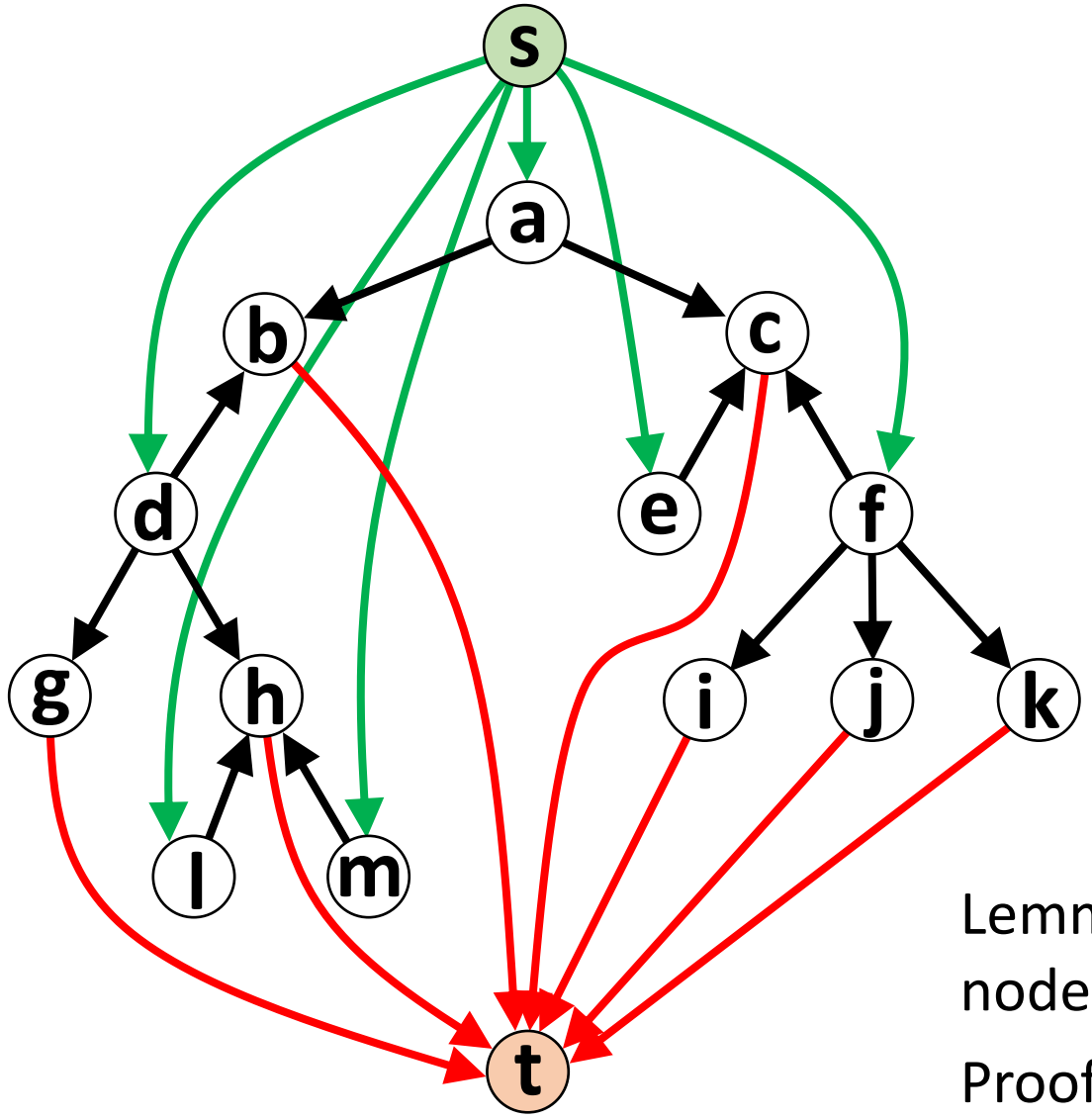
How do we do this in general?

1. Starting at the root, connect every other generation with edge from **s**.
2. Connect other generations with edge to **t**.
3. Make edges go from **s**-connected node to **t**-connected node.

Lemma: Every edge in the tree has an **s**-connected node and a **t**-connected node.

Proof: ?

Maximum Matching



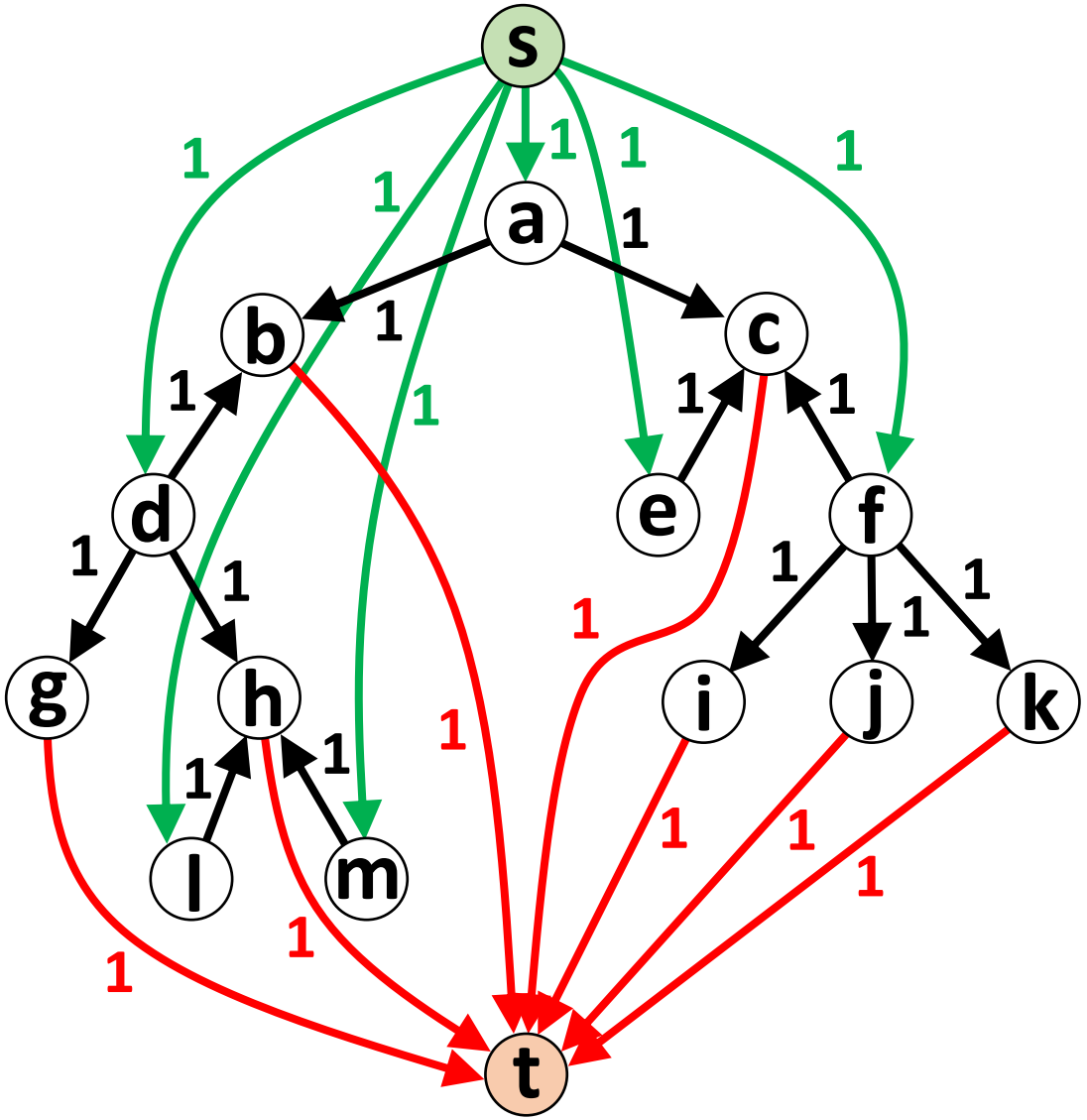
How do we do this in general?

1. Starting at the root, connect every other generation with edge from **s**.
2. Connect other generations with edge to **t**.
3. Make edges go from **s**-connected node to **t**-connected node.

Lemma: Every edge in the tree has an **s**-connected node and a **t**-connected node.

Proof: Edges define generations (parent-child), so they cannot be connected to the same **s**/**t** node.

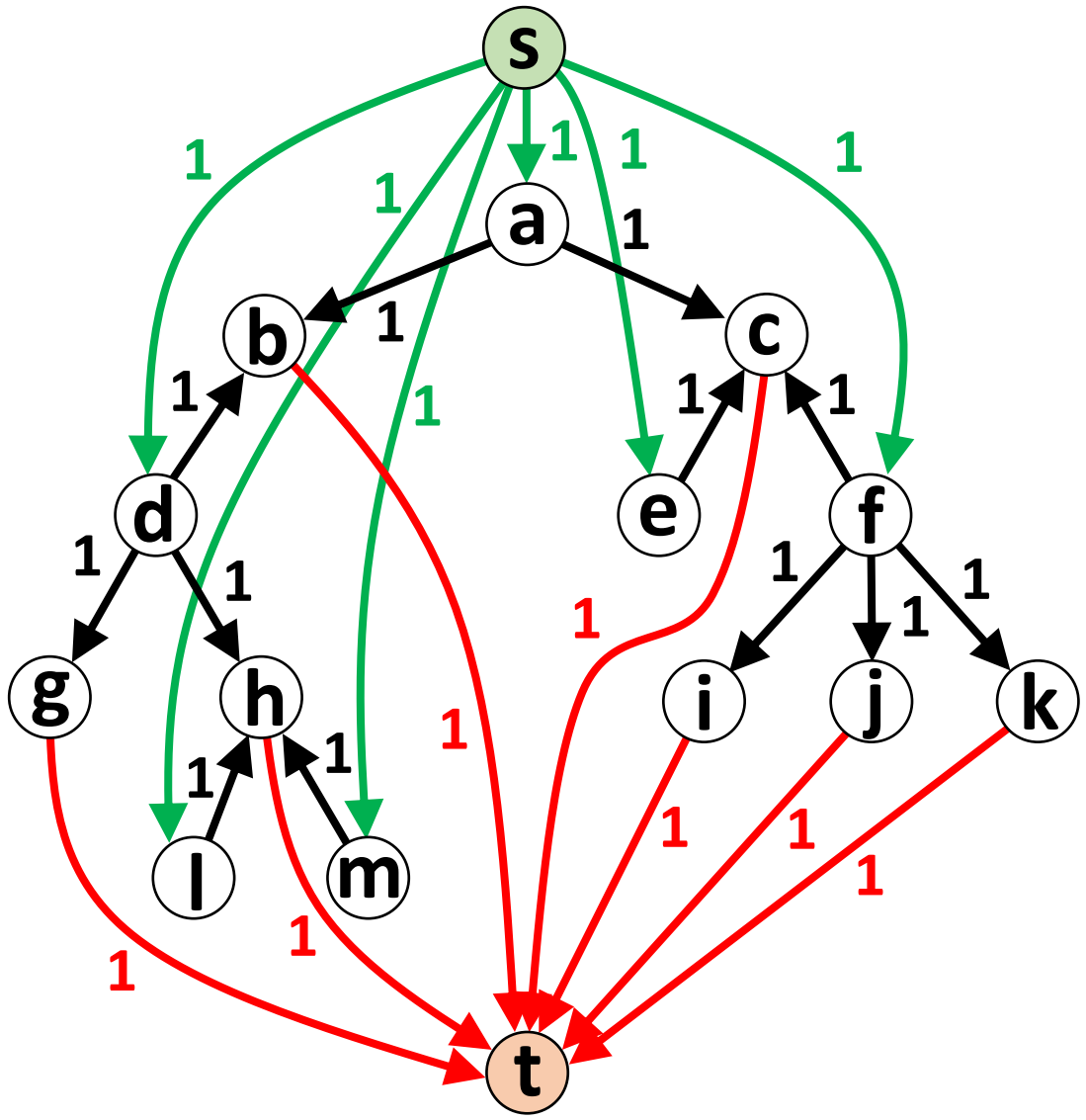
Maximum Matching



How do we do this in general?

1. Starting at the root, connect every other generation with edge from **s**.
2. Connect other generations with edge to **t**.
3. Make edges go from **s**-connected node to **t**-connected node.
4. Make all edge capacities 1.

Maximum Matching



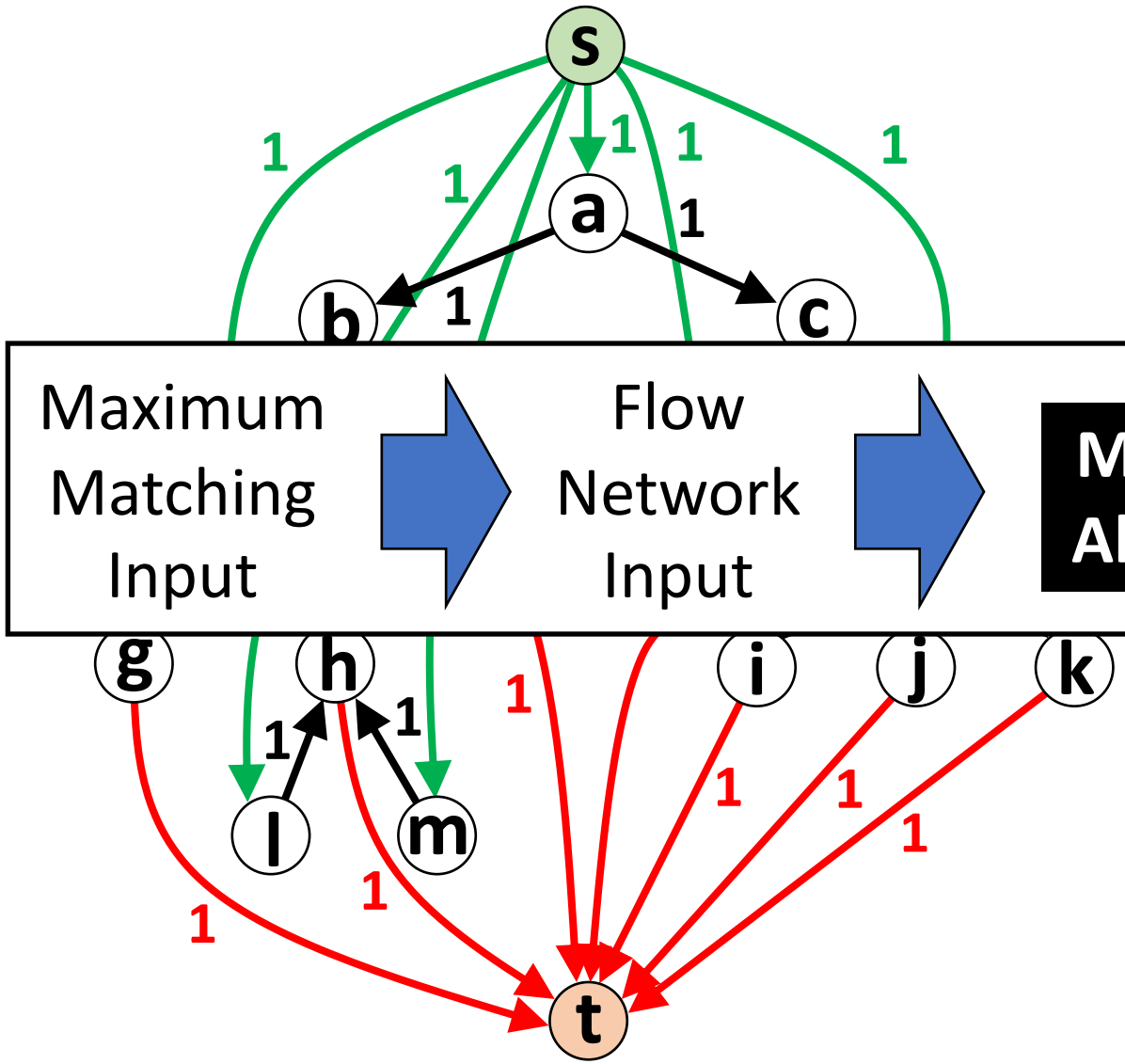
(A) Build Flow Network:

1. Starting at the root, connect every other generation with edge from **s**.
2. Connect other generations with edge to **t**.
3. Make edges go from **s**-connected node to **t**-connected node.
4. Make all edge capacities 1.

(B) Find Max Flow.

(C) If edge carries flow, select it.

Maximum Matching



Maximum Matching Input → Flow Network Input → **Max Flow Algorithm** → Max Flow → Maximum Matching

(A) Build Flow Network:

1. Starting at the root, connect every other generation with edge from **s**.

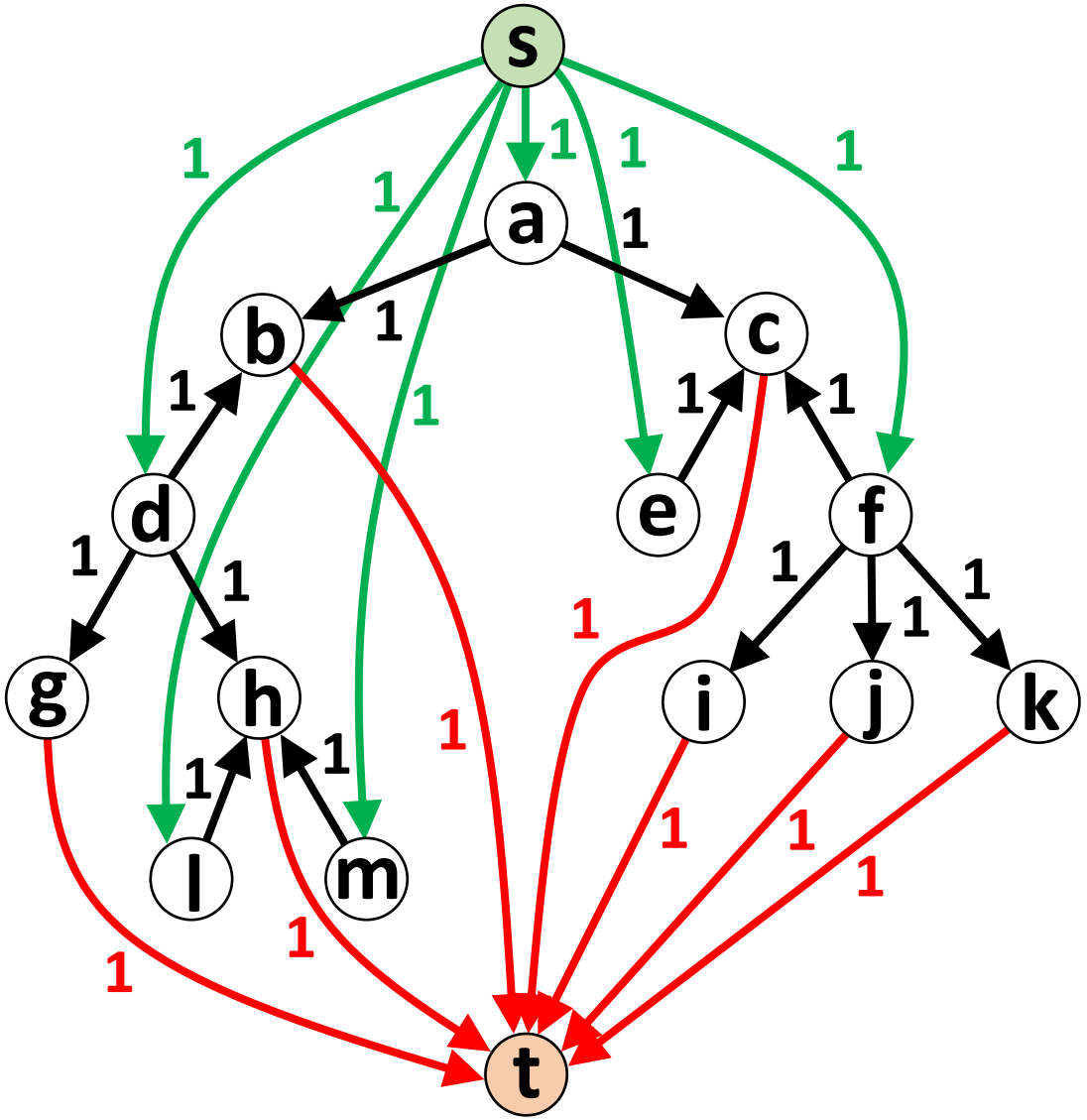
connected node to **t**-connected node.

4. Make all edge capacities 1.

(B) Find Max Flow.

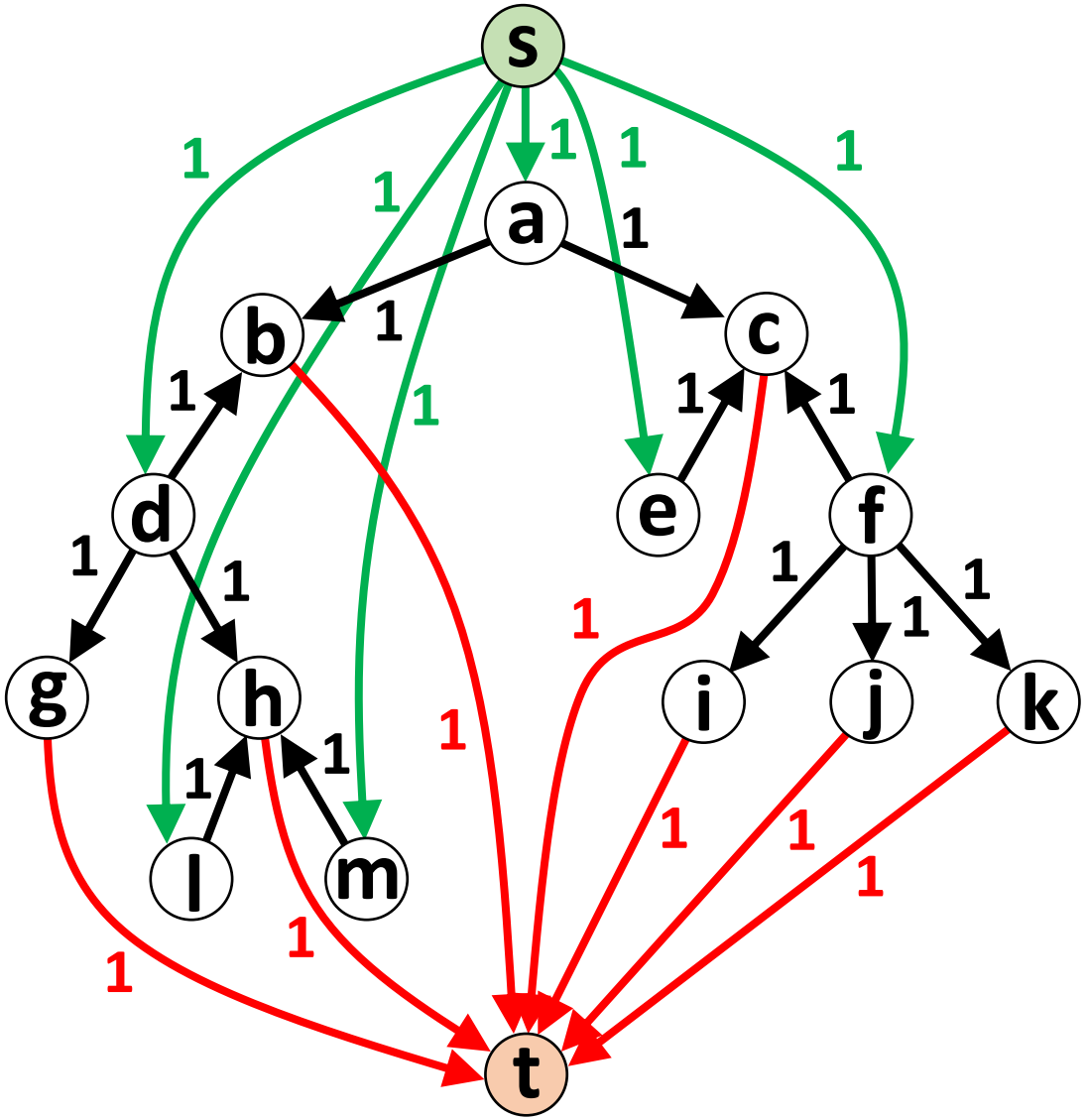
(C) If edge carries flow, select it.

Maximum Matching



How do we know the resulting set of edges is a matching?

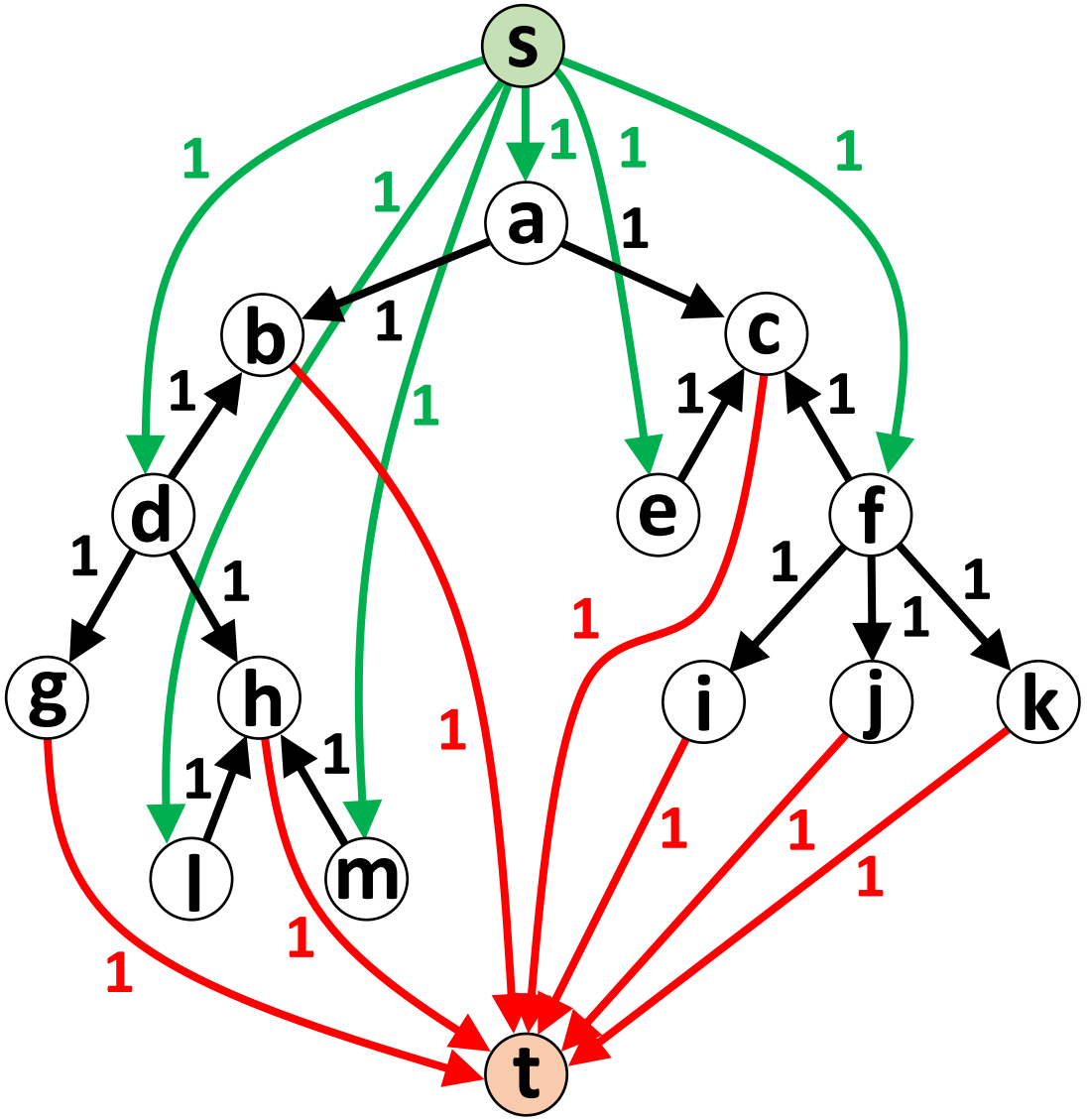
Maximum Matching



How do we know the resulting set of edges is a matching?

Suppose it is not. Suppose two edges share a node.

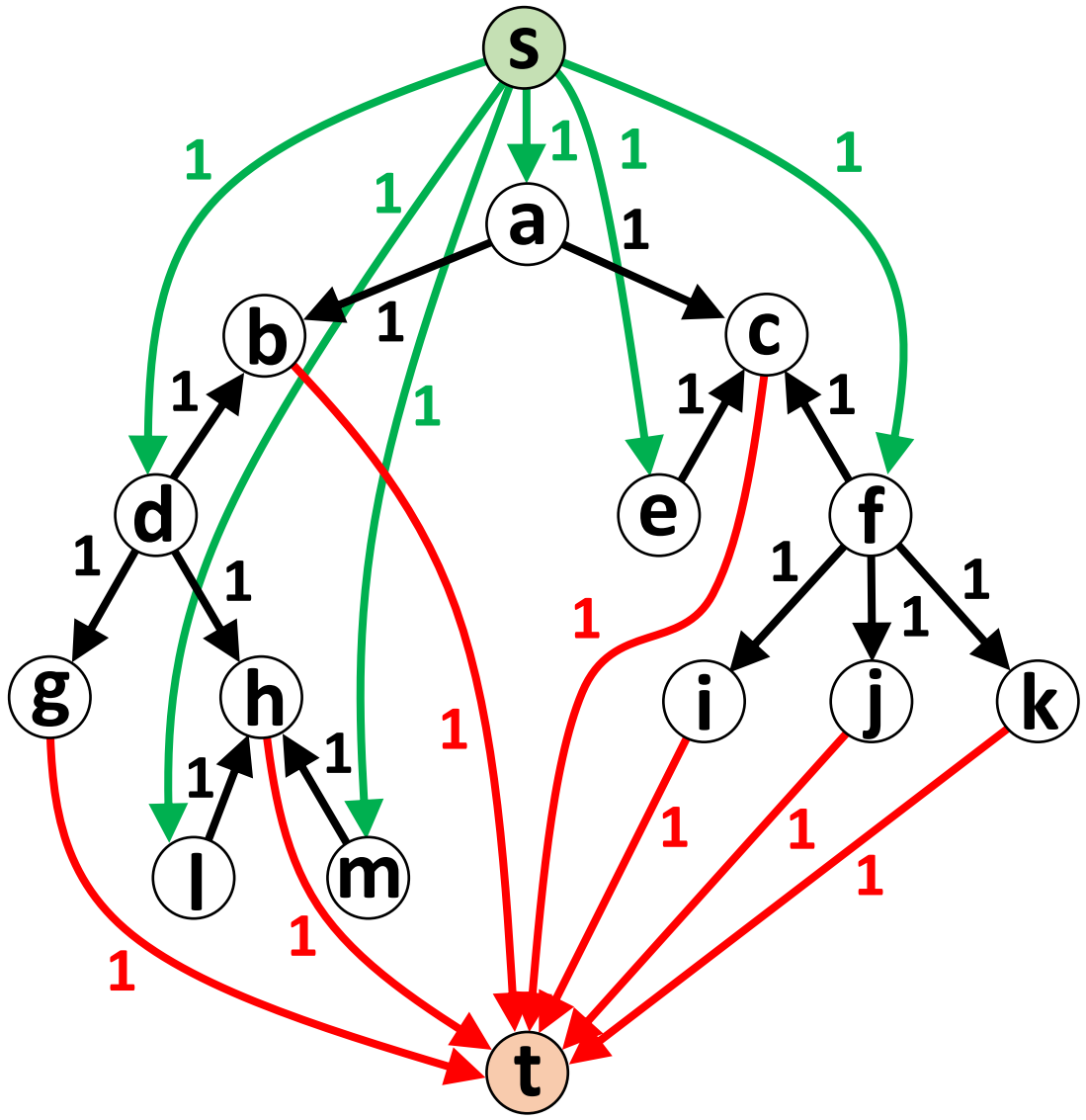
Maximum Matching



How do we know the resulting set of edges is a matching?

Suppose it is not. Suppose two edges share a node. At least one node (of the three) is attached to **s** and at least one is attached to **t**.

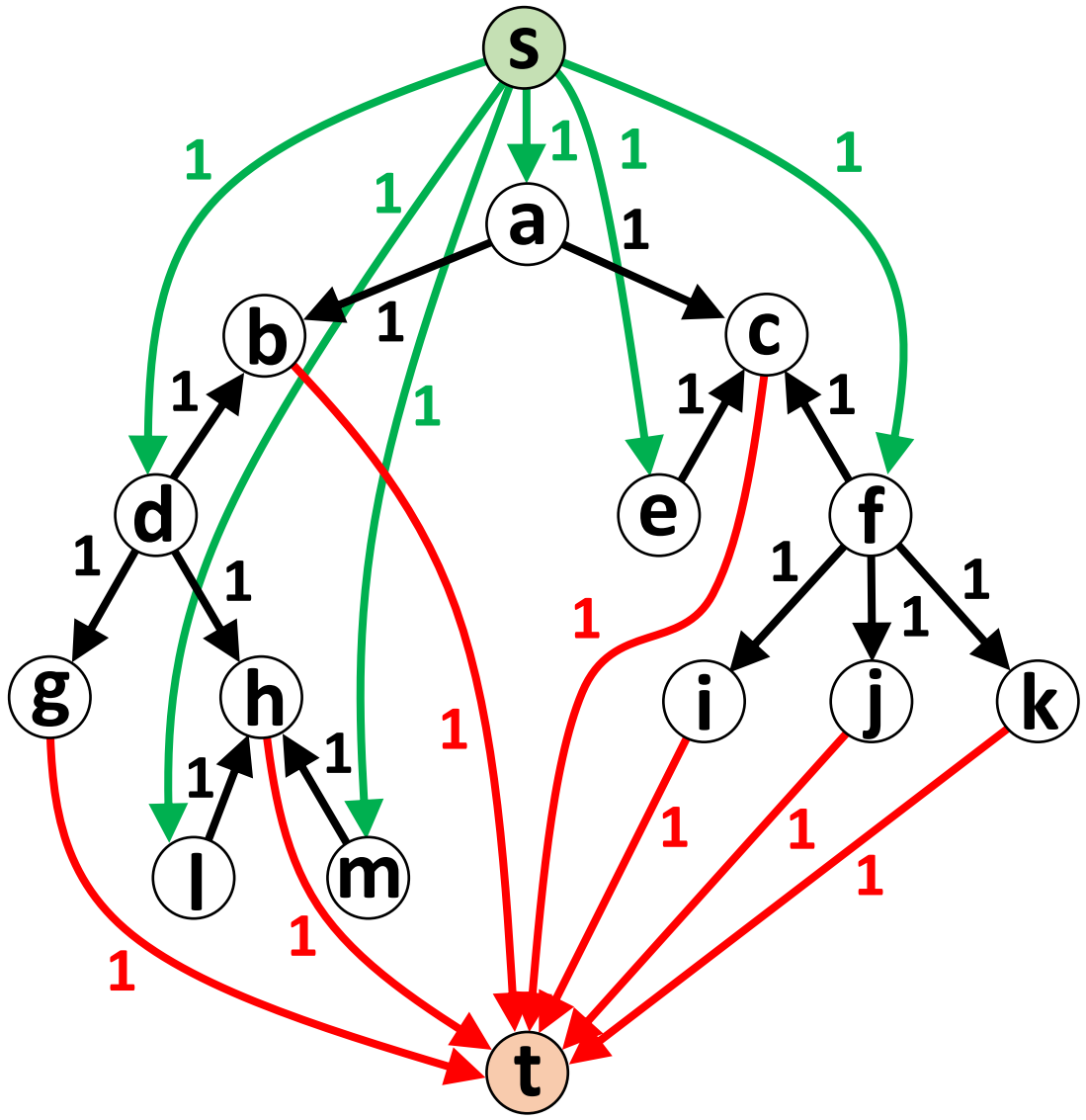
Maximum Matching



How do we know the resulting set of edges is a matching?

Suppose it is not. Suppose two edges share a node. At least one node (of the three) is attached to **s** and at least one is attached to **t**. That means the pair of edges either get one unit of flow (single node attached to **s**) or can send on one unit (single node attached to **t**).

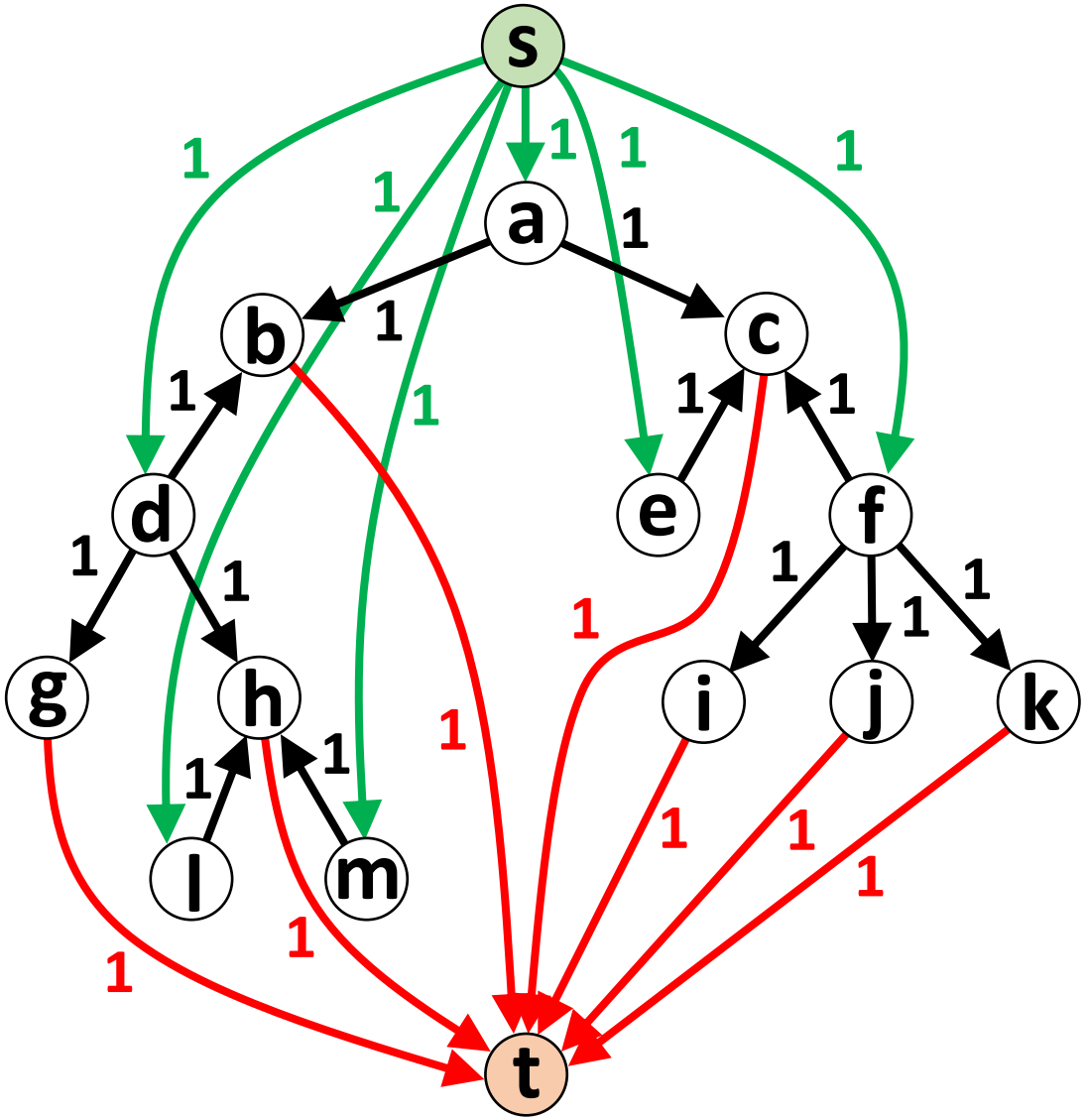
Maximum Matching



How do we know the resulting set of edges is a matching?

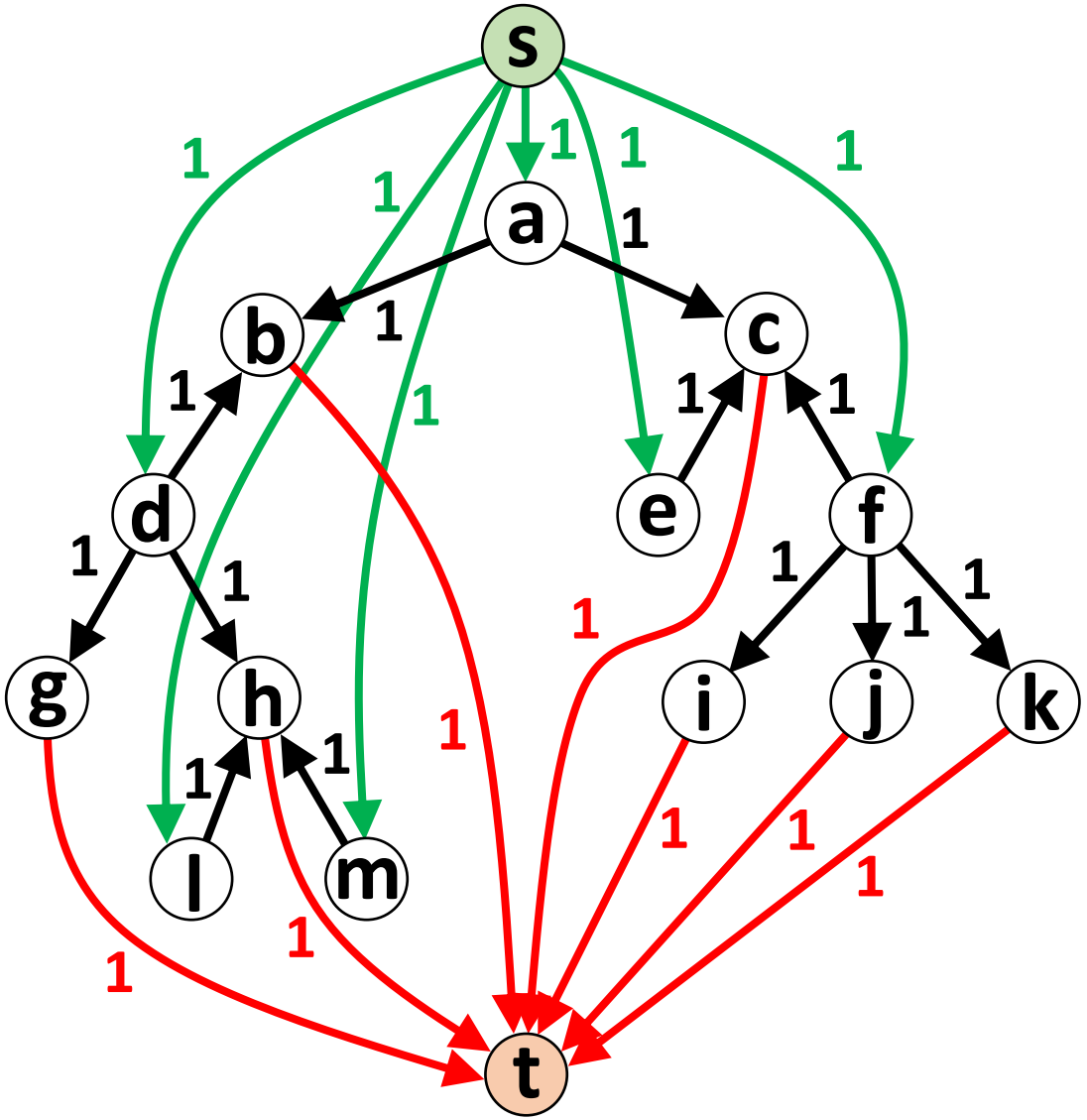
Suppose it is not. Suppose two edges share a node. At least one node (of the three) is attached to **s** and at least one is attached to **t**. That means the pair of edges either get one unit of flow (single node attached to **s**) or can send on one unit (single node attached to **t**). Either way, it is not possible for both edges to host flow.

Maximum Matching



How do we know the resulting set of edges is optimal?

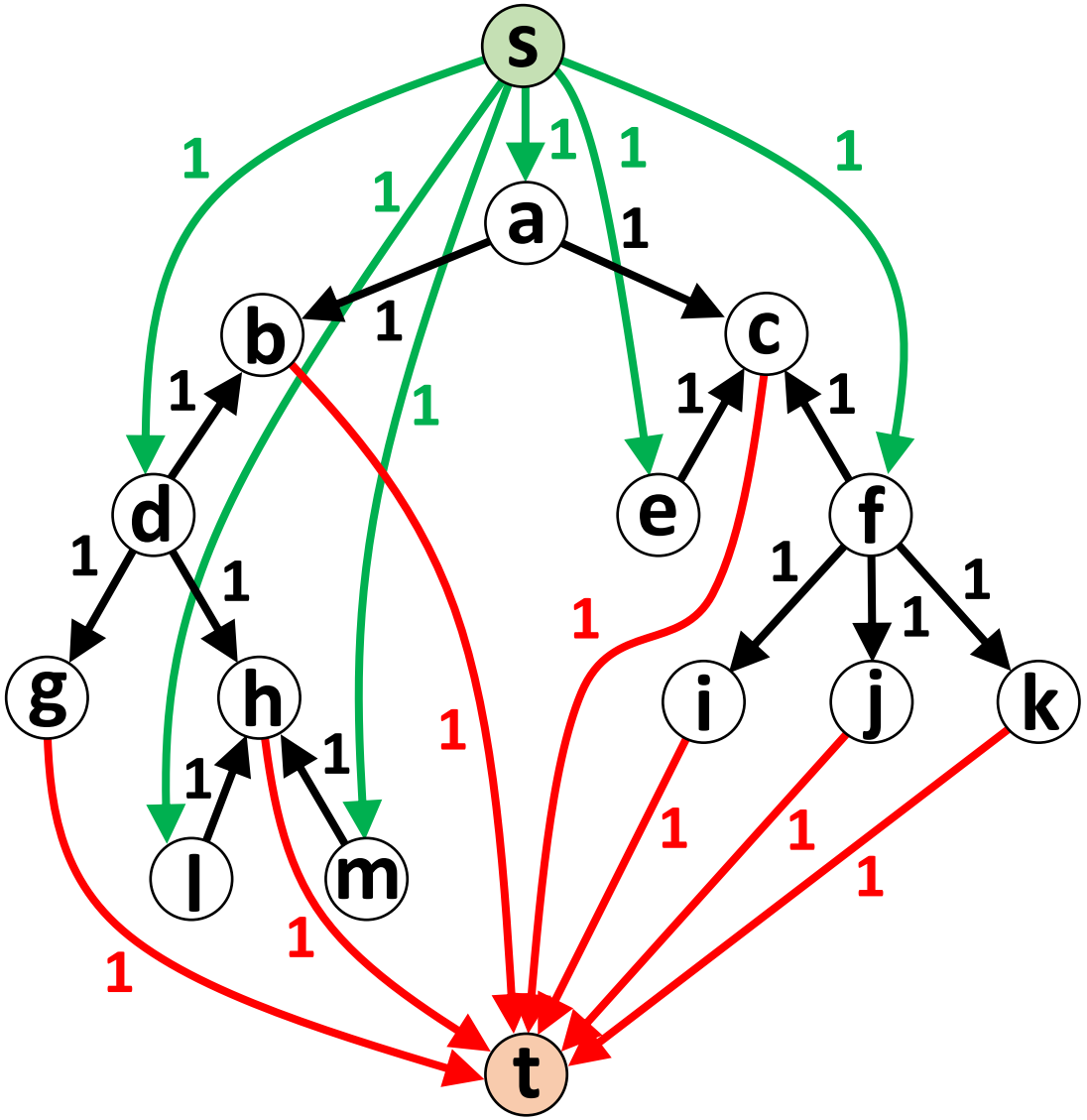
Maximum Matching



How do we know the resulting set of edges is optimal?

Suppose the set of edges found S is smaller than the optimal set S'.

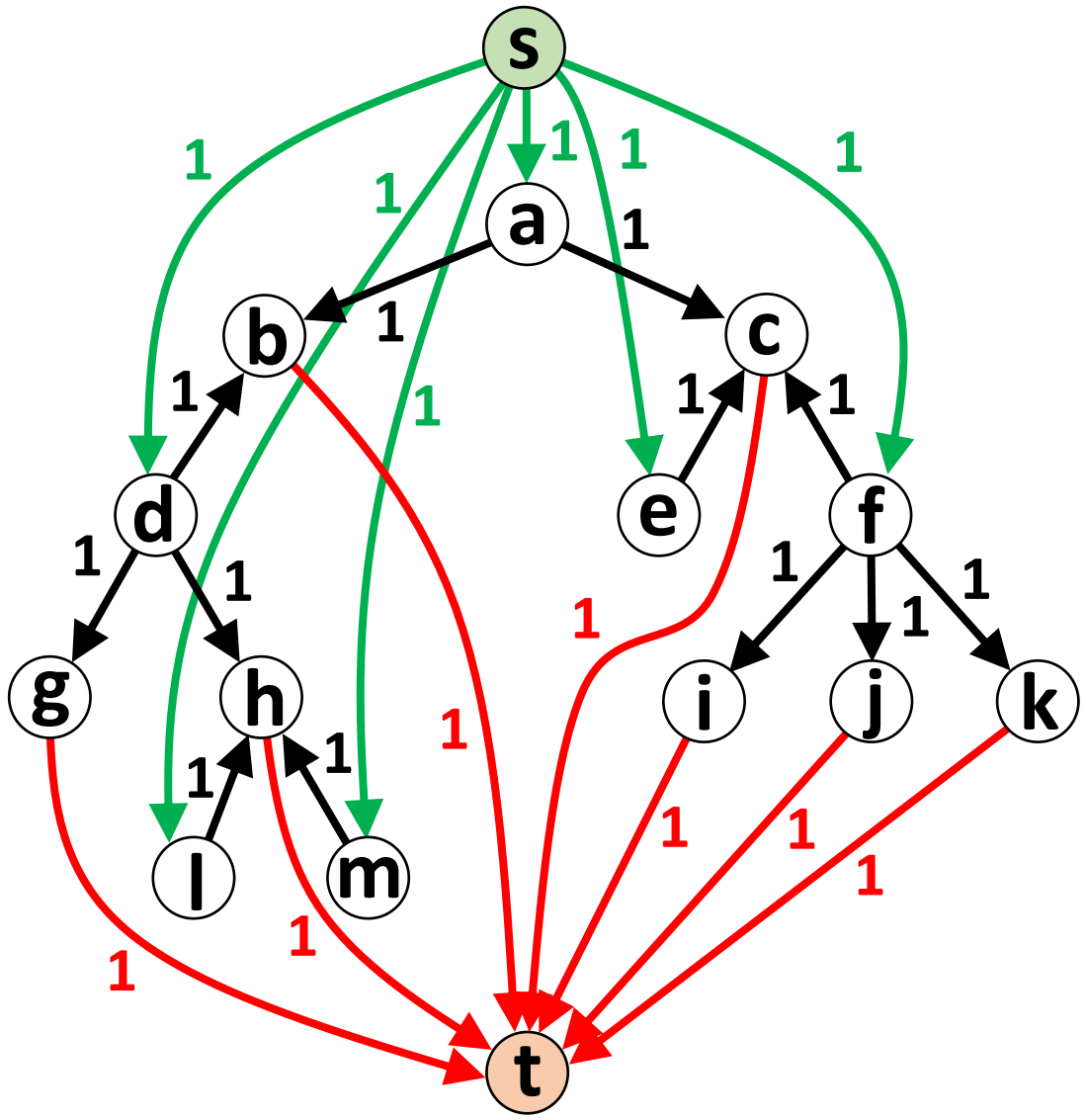
Maximum Matching



How do we know the resulting set of edges is optimal?

Suppose the set of edges found S is smaller than the optimal set S'. Since S' edges are independent, none share a node,

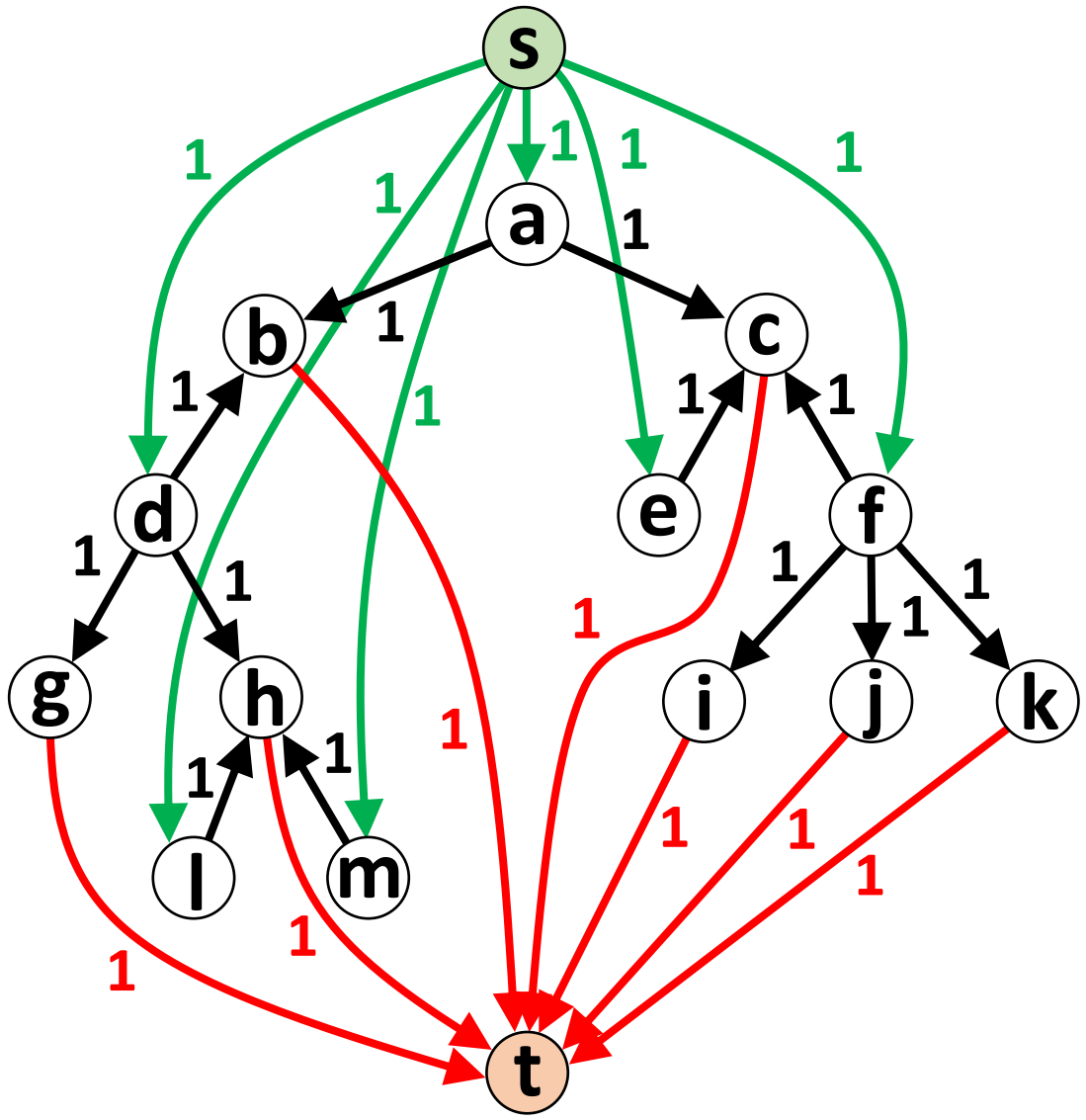
Maximum Matching



How do we know the resulting set of edges is optimal?

Suppose the set of edges found S is smaller than the optimal set S'. Since S' edges are independent, none share a node, so 1 unit of flow can be put on each path: $s \to e \in S' \to t$.

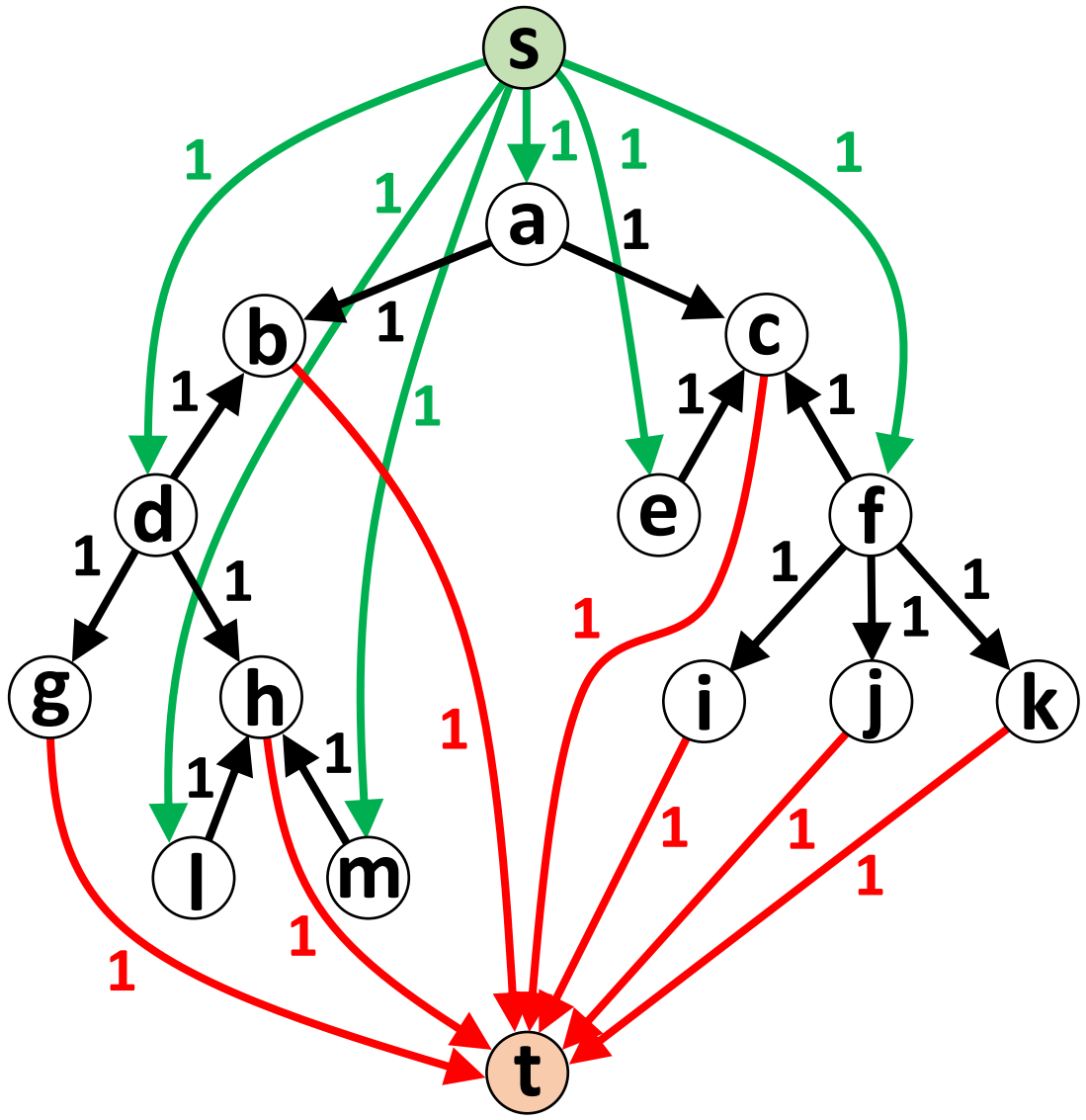
Maximum Matching



How do we know the resulting set of edges is optimal?

Suppose the set of edges found S is smaller than the optimal set S'. Since S' edges are independent, none share a node, so 1 unit of flow can be put on each path: $s \to e \in S' \to t$. This gives a flow (capacity?, conservation of flow?)

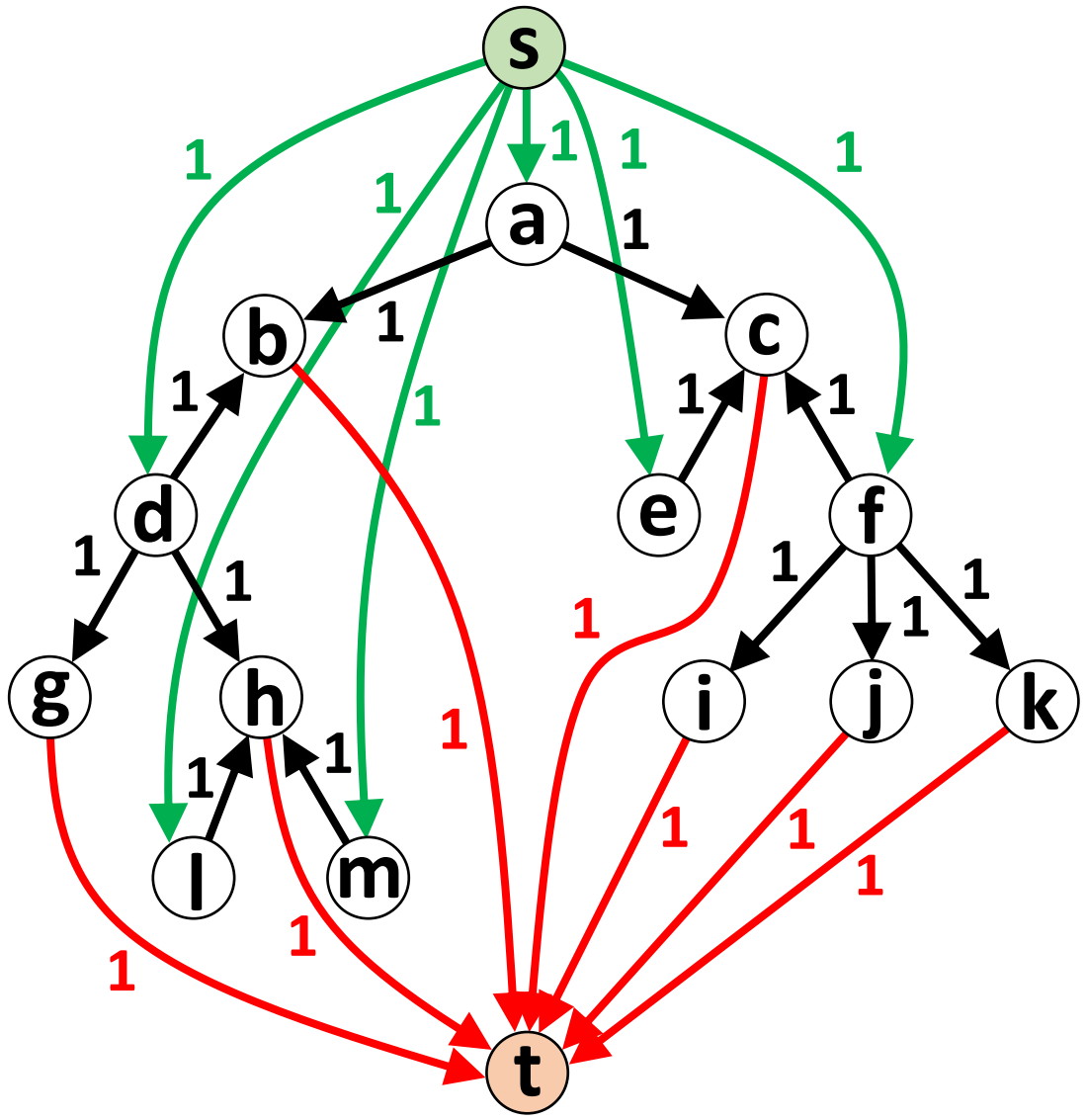
Maximum Matching



How do we know the resulting set of edges is optimal?

Suppose the set of edges found S is smaller than the optimal set S'. Since S' edges are independent, none share a node, so 1 unit of flow can be put on each path: $s \to e \in S' \to t$. This gives a flow (capacity?, conservation of flow?) larger than the max flow (since $|S| < |S'|$),

Maximum Matching



How do we know the resulting set of edges is optimal?

Suppose the set of edges found S is smaller than the optimal set S'. Since S' edges are independent, none share a node, so 1 unit of flow can be put on each path: $s \to e \in S' \to t$. This gives a flow (capacity?, conservation of flow?) larger than the max flow (since $|S| < |S'|$), which is a contradiction. Thus, the size of S is optimal.