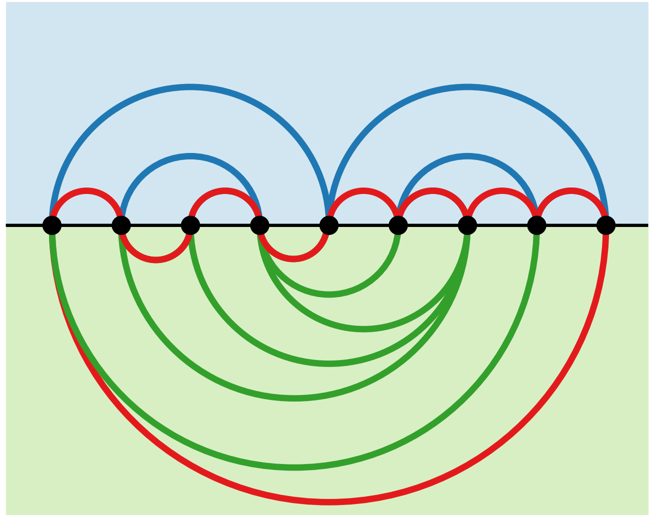
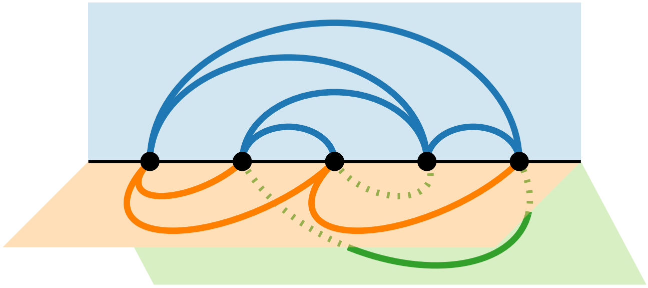
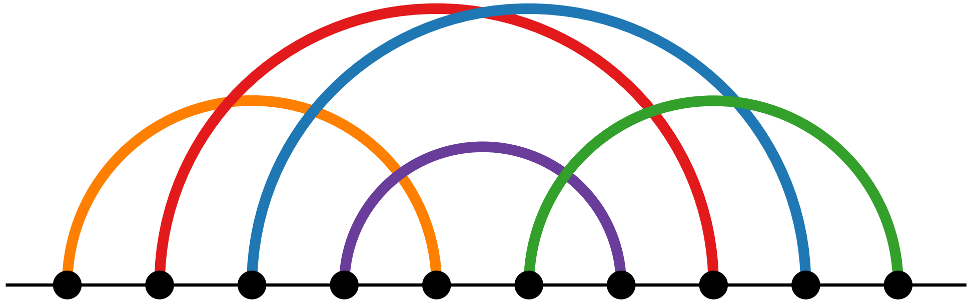
Visualization of Graphs

Lecture 12:
Linear Layouts
(Book Embeddings)



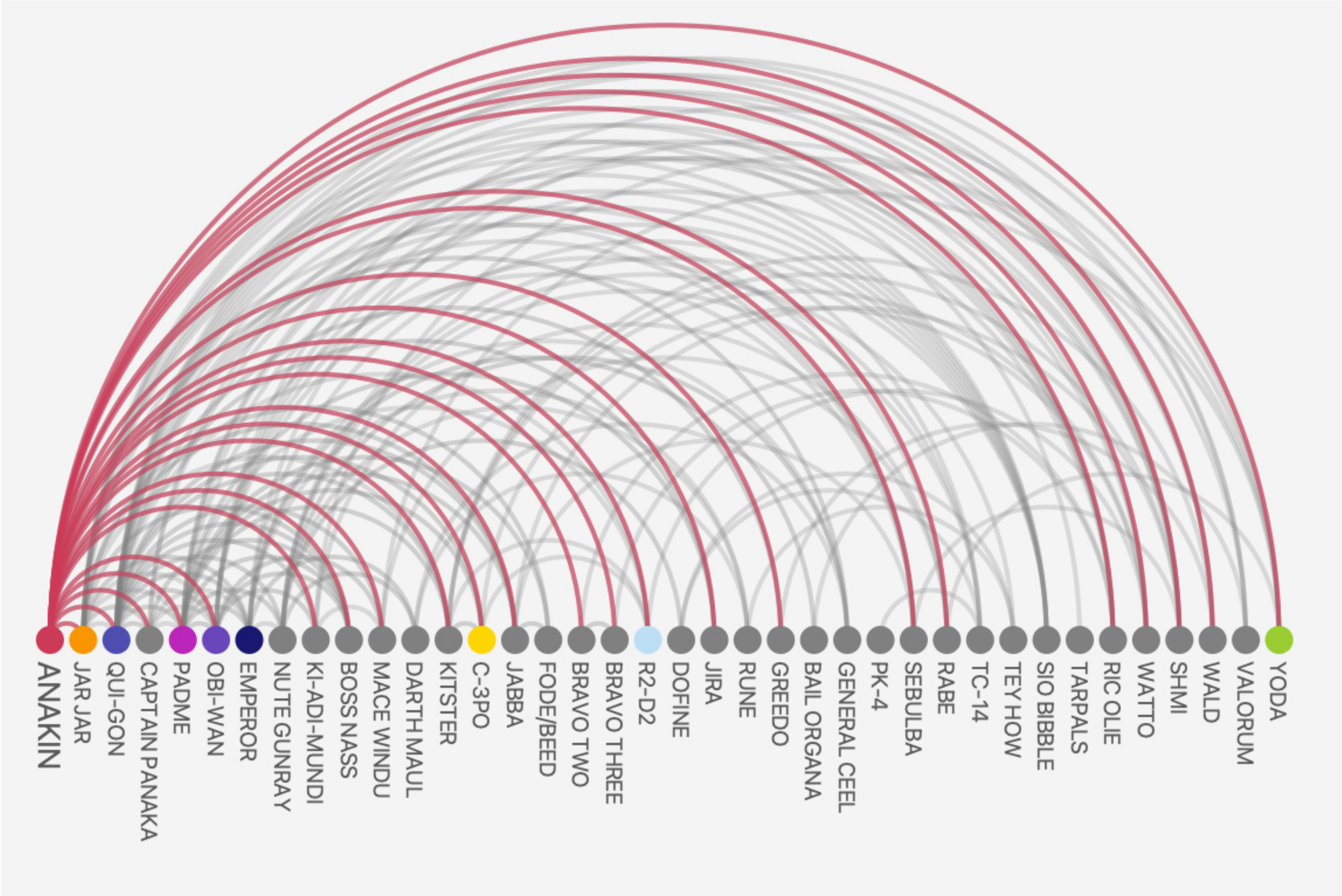




Alexander Wolff

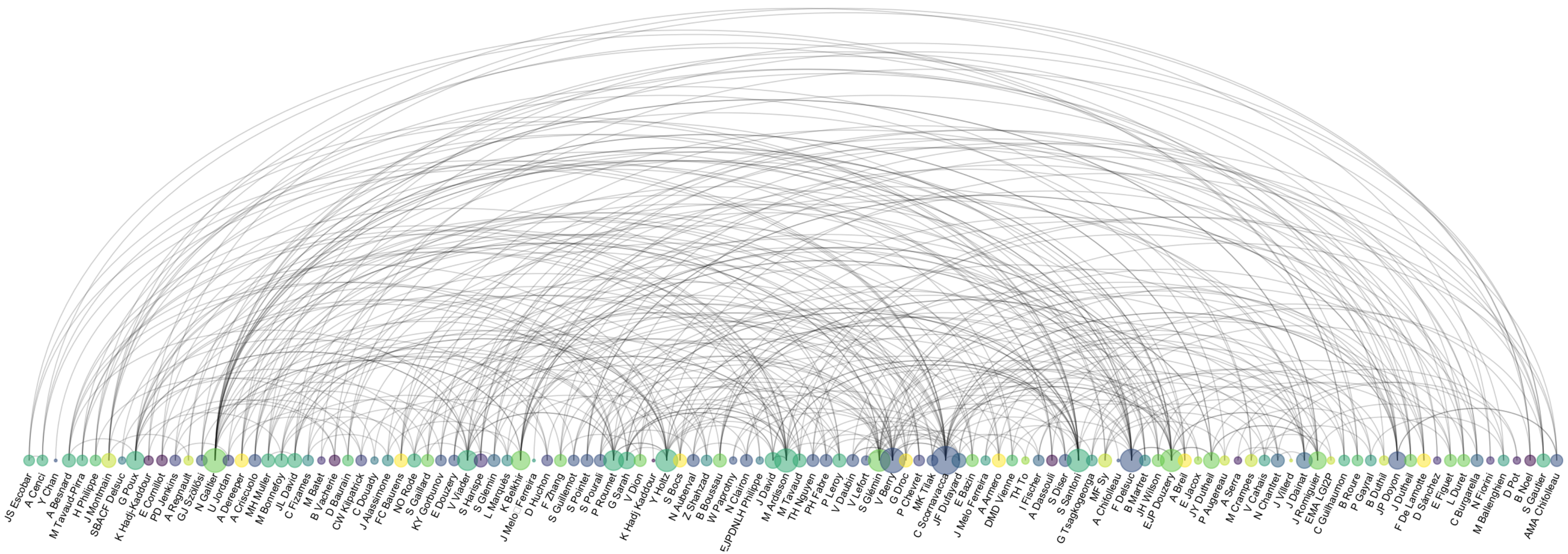
Summer term 2025

Drawing Style: Arc Diagrams



Interactions in Star Wars Episode I
[https://harmoniccode.blogspot.com/2020/11/arc-charts.html]

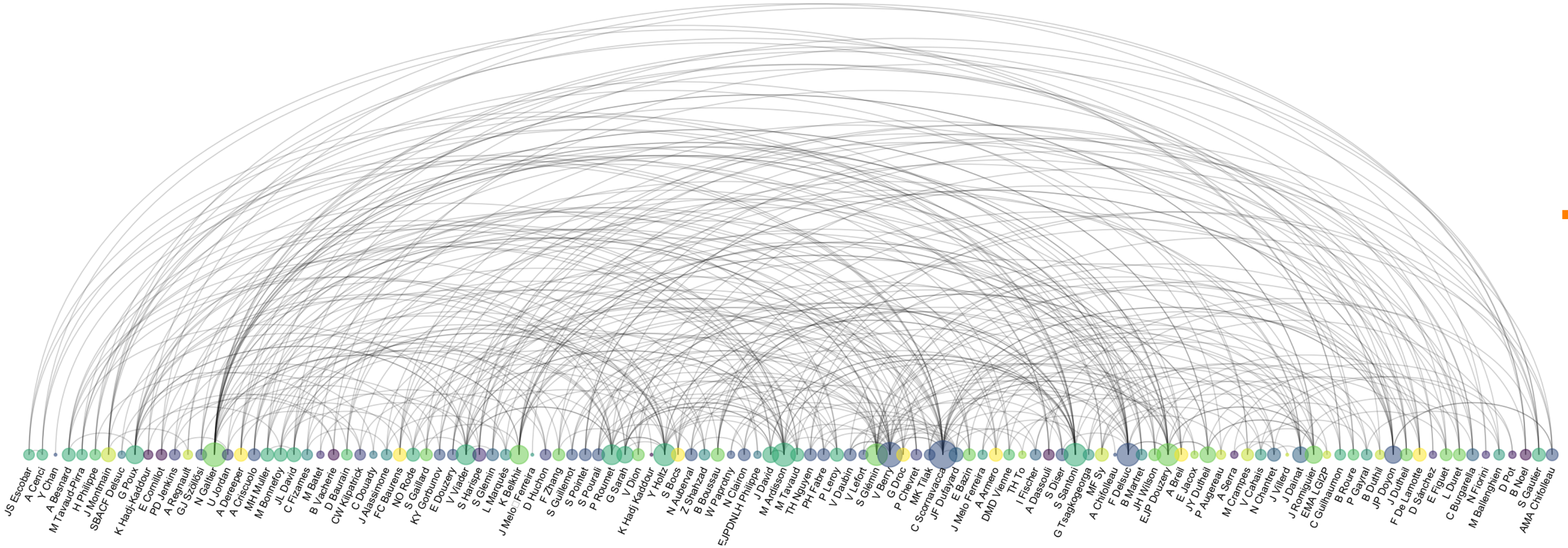
Drawing Style: Arc Diagrams



Network of co-authors of Vincent Ranwez (edge ⇔ co-authors)

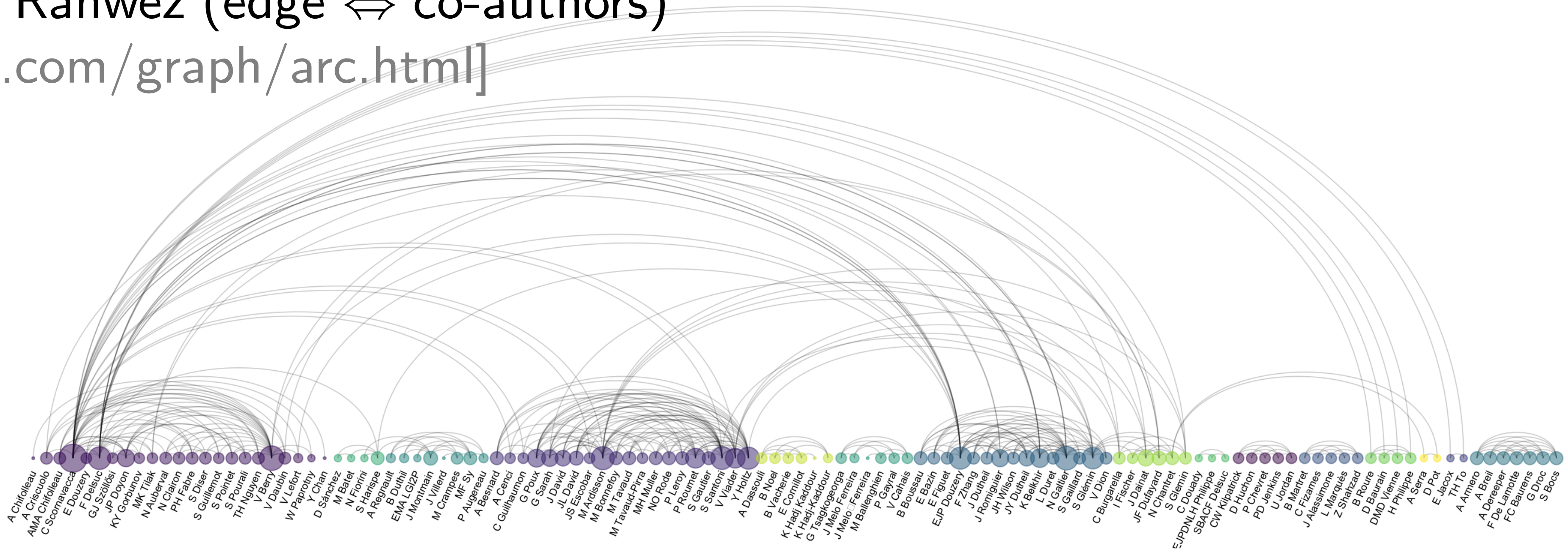
[https://www.data-to-viz.com/graph/arc.html]

Drawing Style: Arc Diagrams

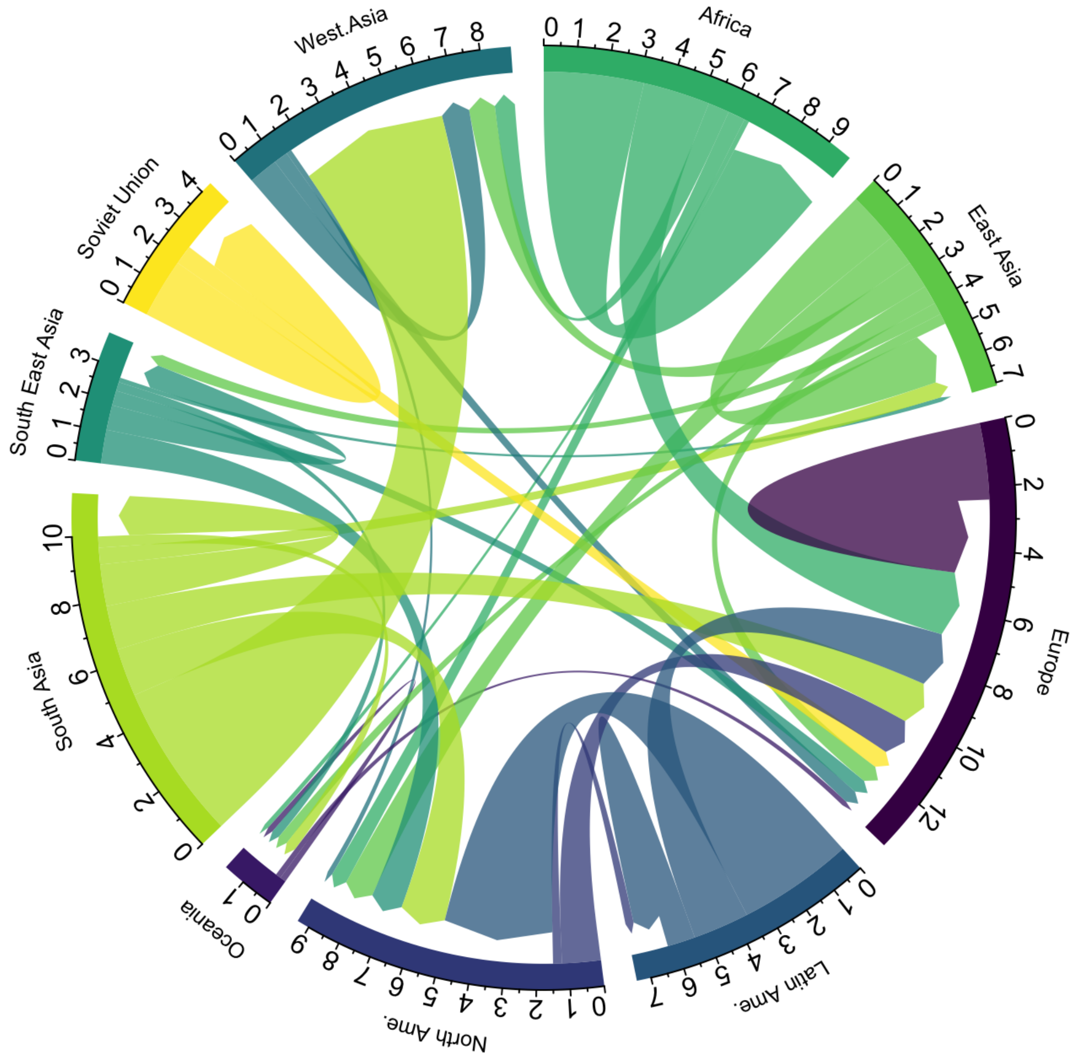


re-order vertices

Network of co-authors of Vincent Ranwez (edge $\Leftrightarrow$ co-authors)
[https://www.data-to-viz.com/graph/arc.html]



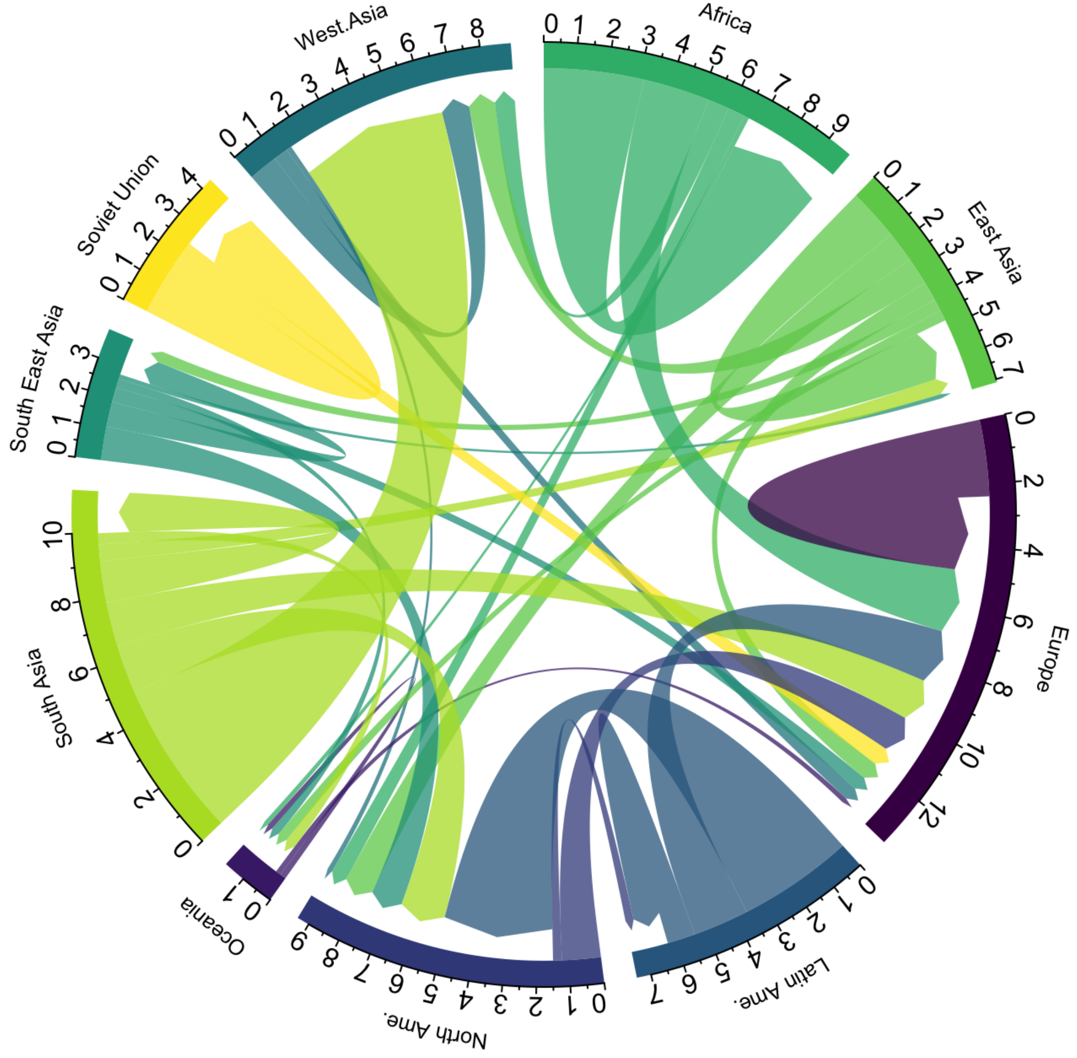
Drawing Style: Chord Diagrams



Migration between continents

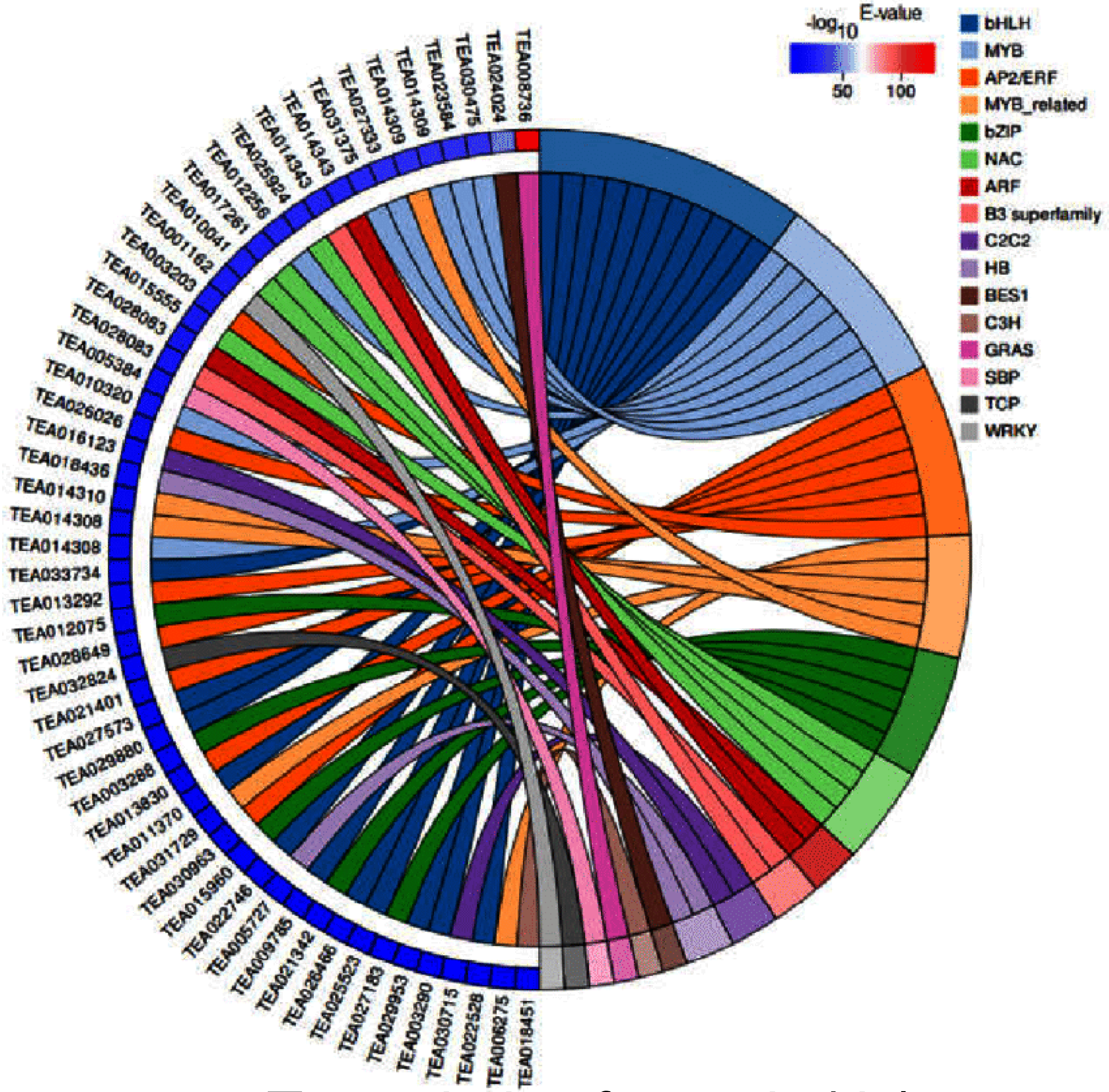
[https://www.data-to-viz.com/story/AdjacencyMatrix.html]

Drawing Style: Chord Diagrams



Migration between continents

[https://www.data-to-viz.com/story/AdjacencyMatrix.html]

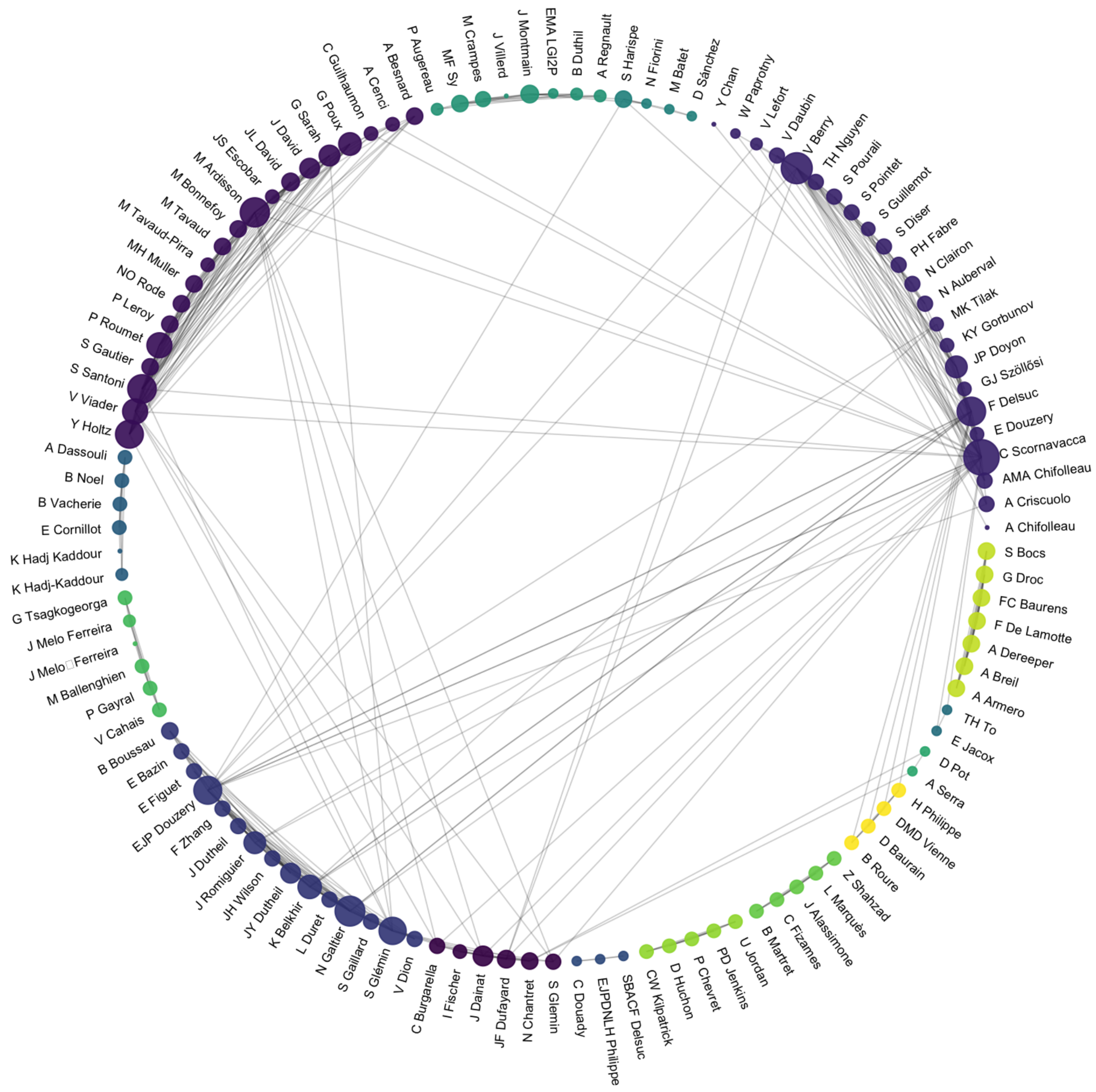


Transcription factors in biology

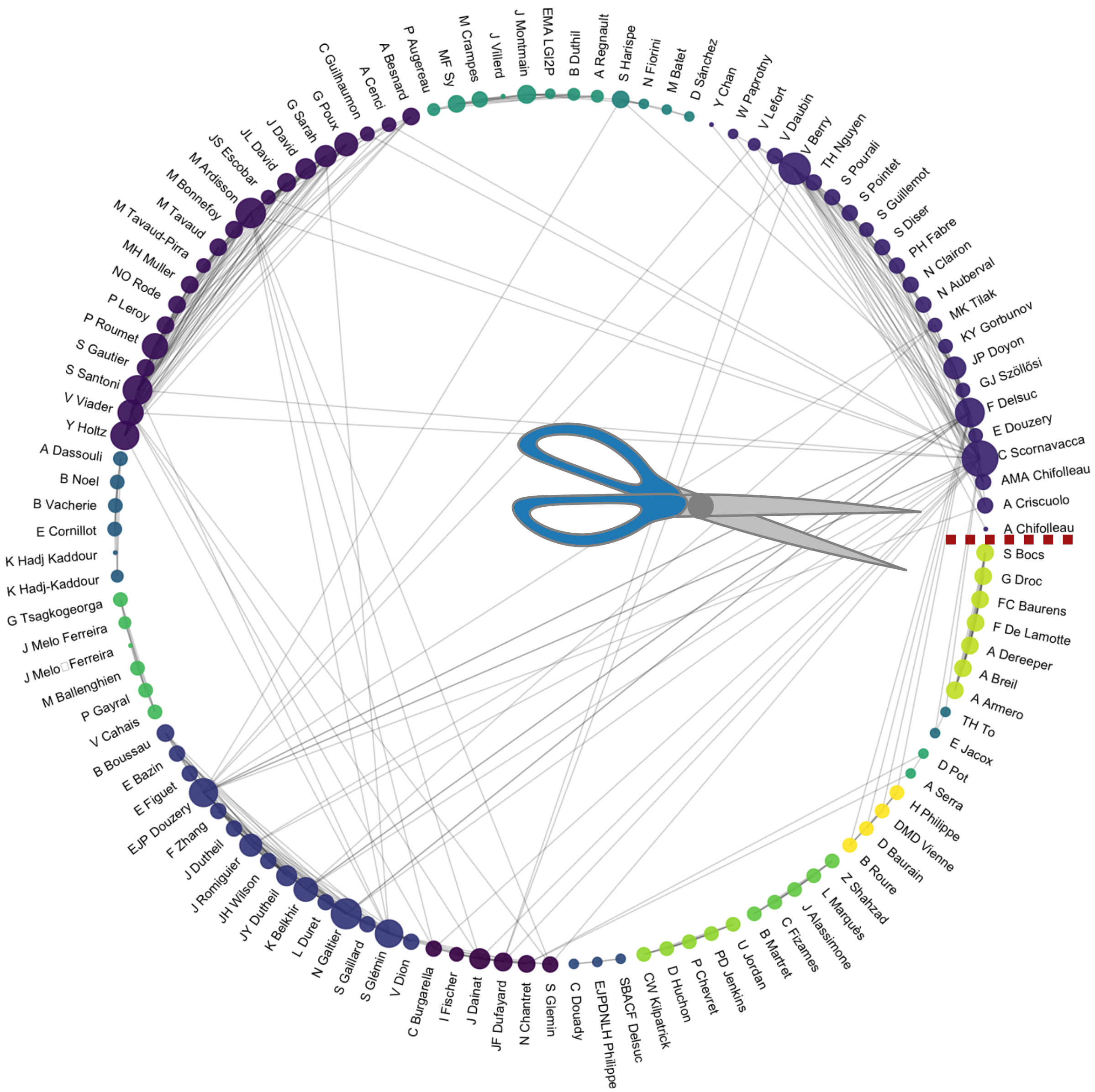
[Wang, Xuejin, Zhao 2020]

Exploration of the Effects of Different Blue LED Light Intensities on Flavonoid and Lipid Metabolism in Tea Plants via Transcriptomics and Metabolomics.

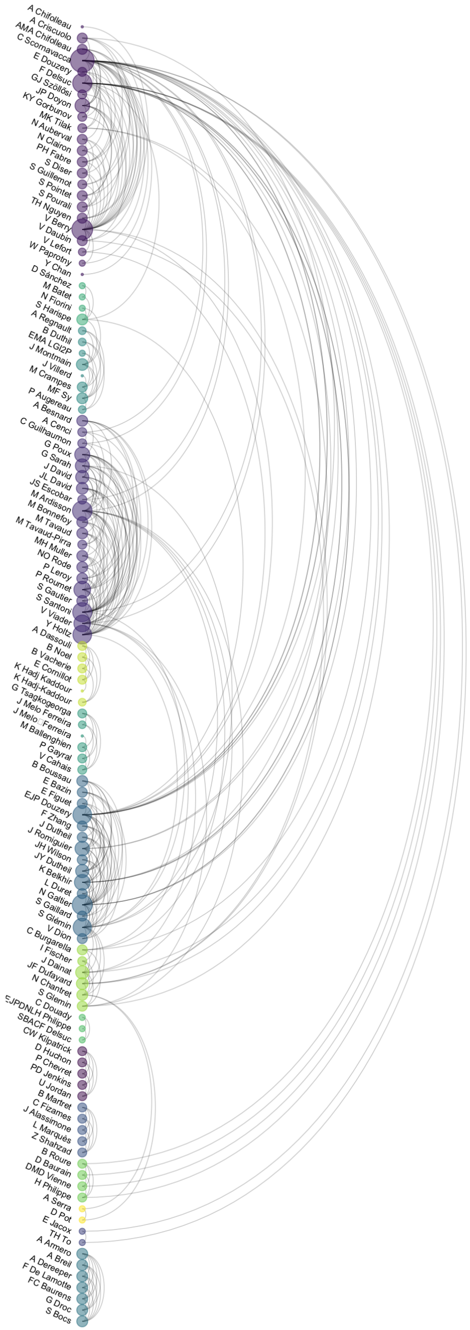
Drawing Style: Chord Diagrams



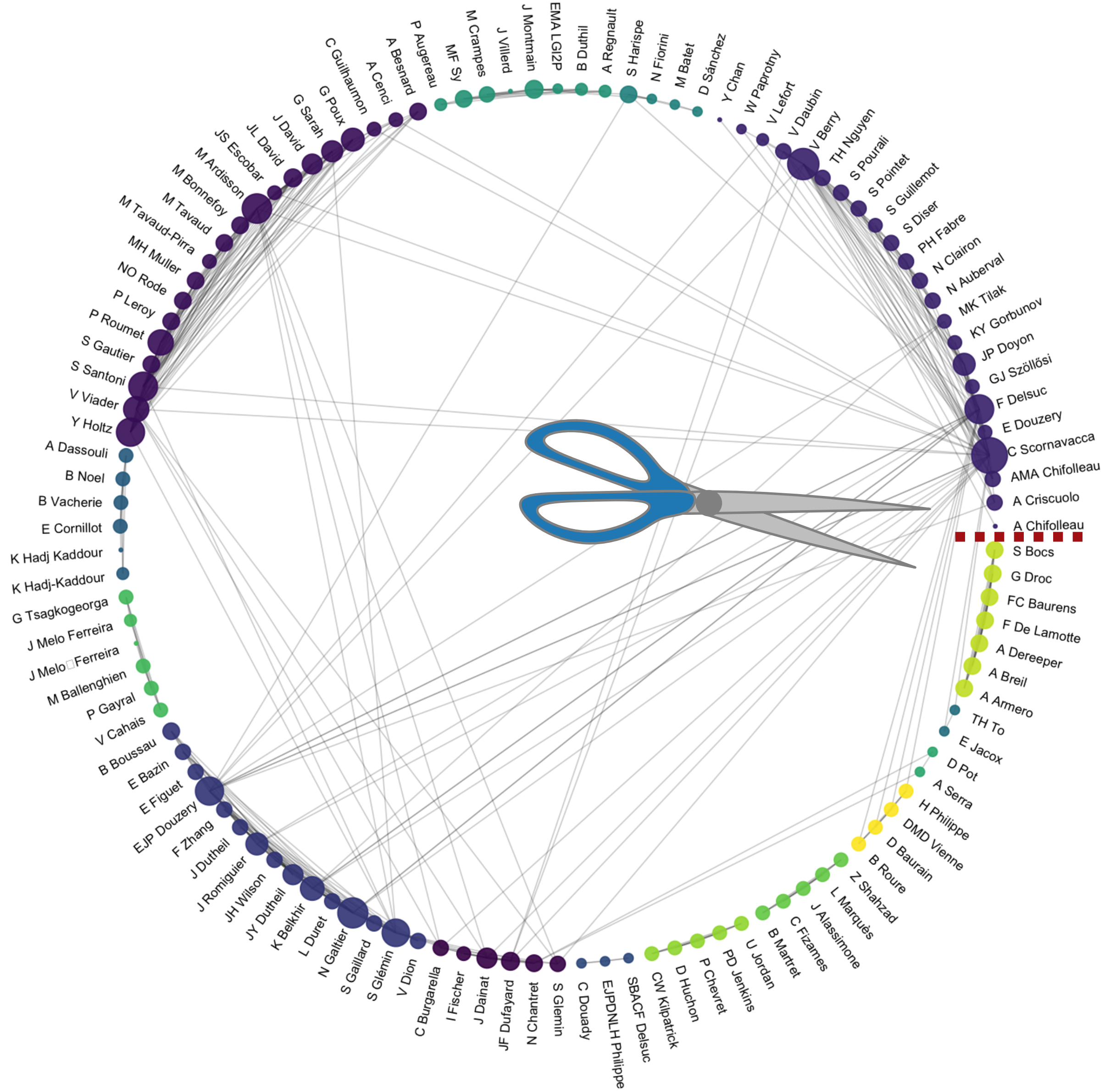
Drawing Style: Chord Diagrams



Drawing Style: Chord Diagrams

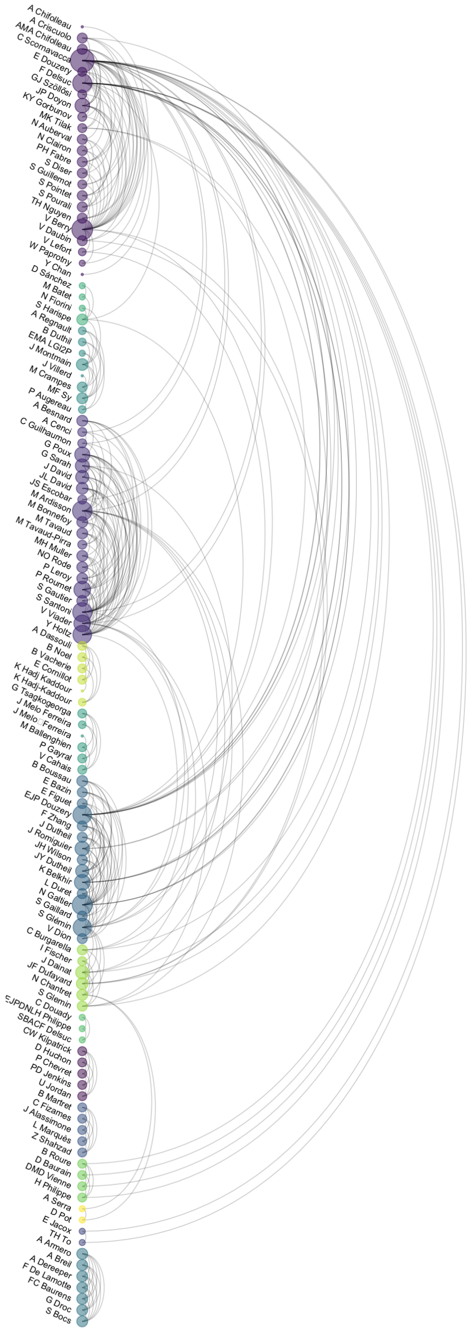


“cut” circle



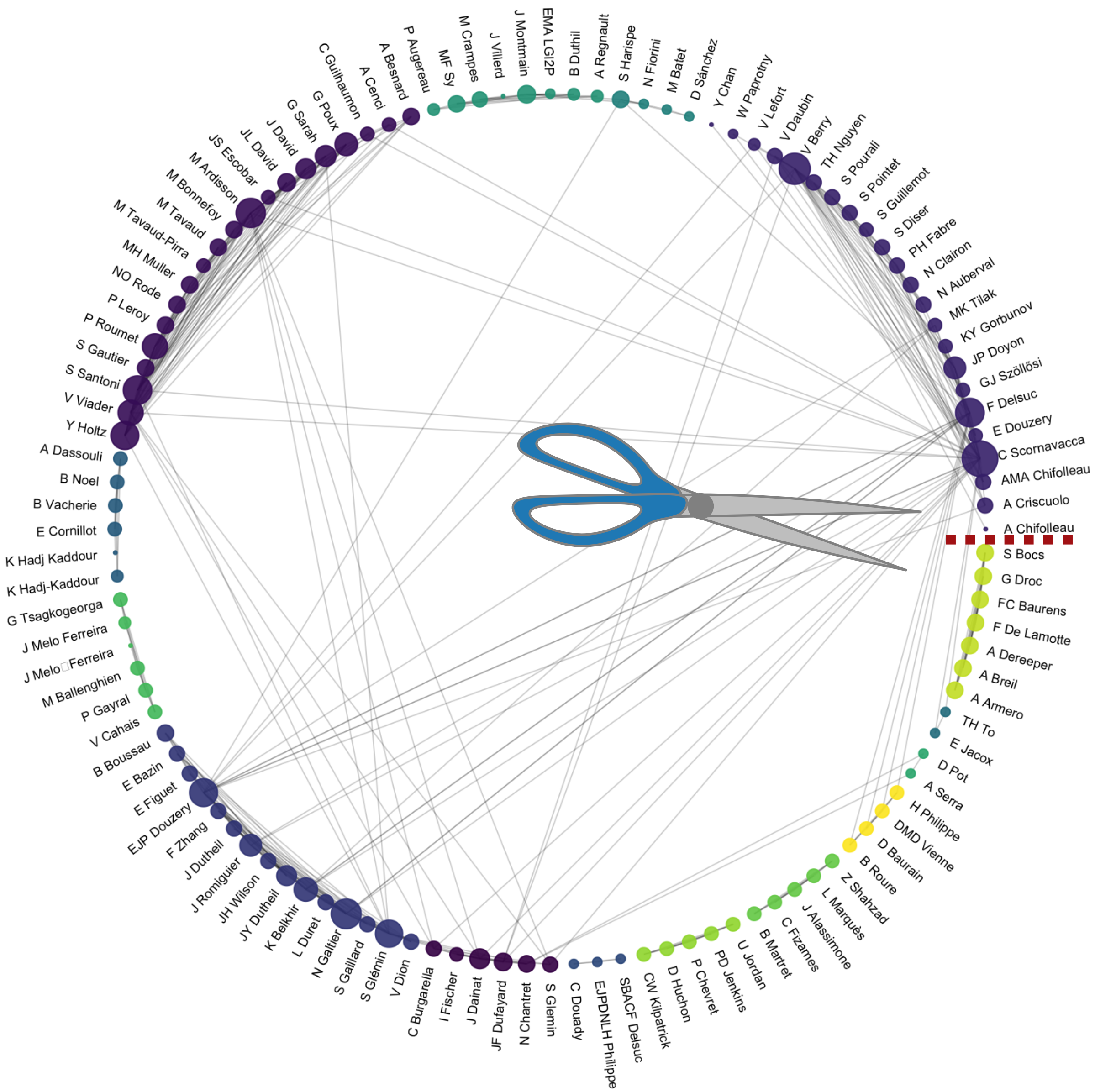
Network of co-authors of Vincent Ranwez (edge ⇔ co-authors)
[https://www.data-to-viz.com/story/AdjacencyMatrix.html]

Drawing Style: Chord Diagrams



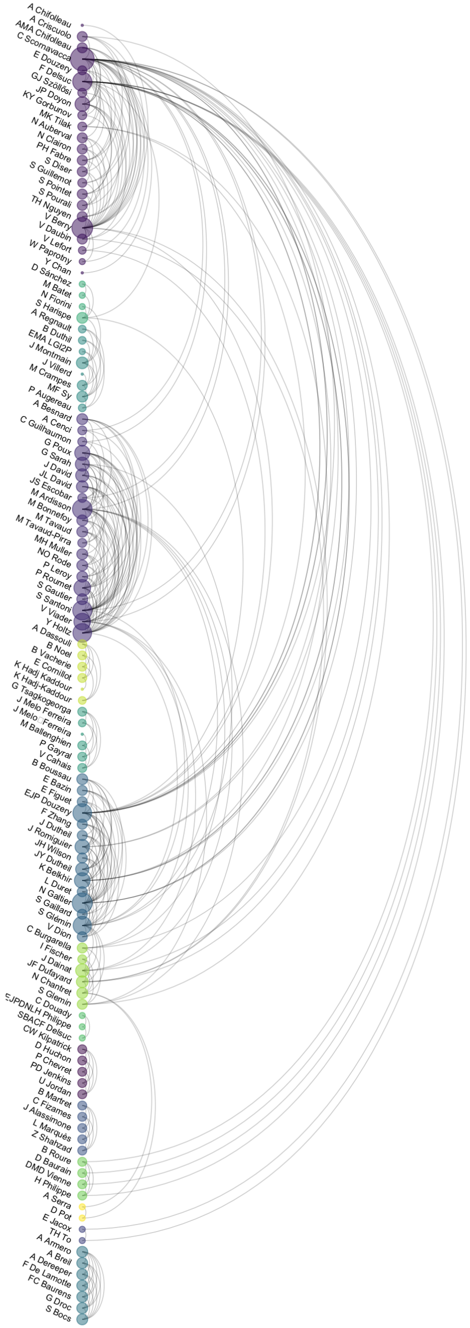
"cut" circle

"glue" sequence



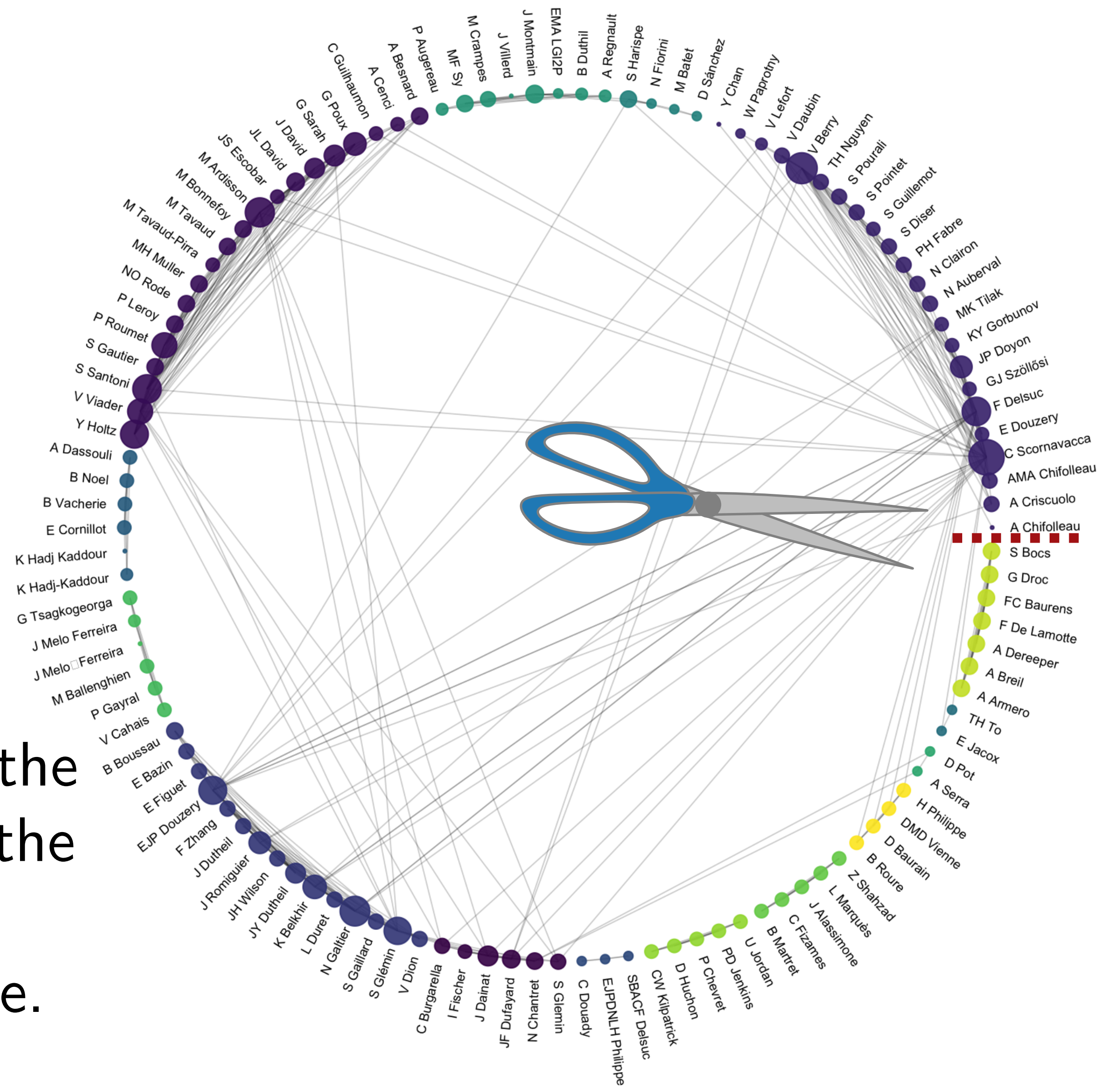
Network of co-authors of Vincent Ranwez (edge ⇔ co-authors)
[https://www.data-to-viz.com/story/AdjacencyMatrix.html]

Drawing Style: Chord Diagrams



“cut” circle

“glue” sequence



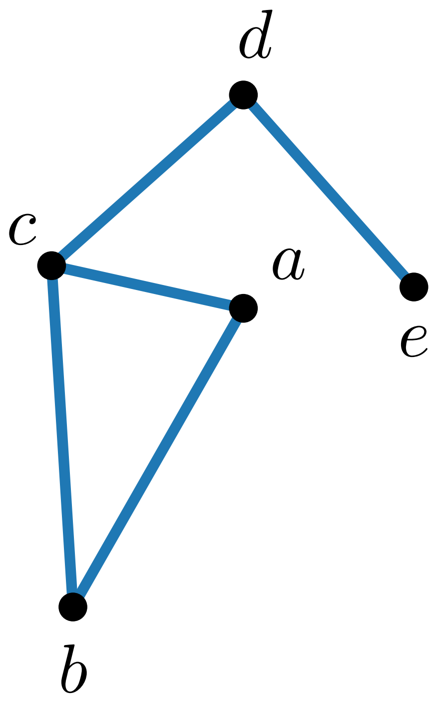
Note that, if we keep the order of the vertices, the crossings between the edges remain the same.

Network of co-authors of Vincent Ranwez (edge ⇔ co-authors)
[https://www.data-to-viz.com/story/AdjacencyMatrix.html]

Planarity + Arc/Chord Diagrams?

Planarity + Arc/Chord Diagrams?

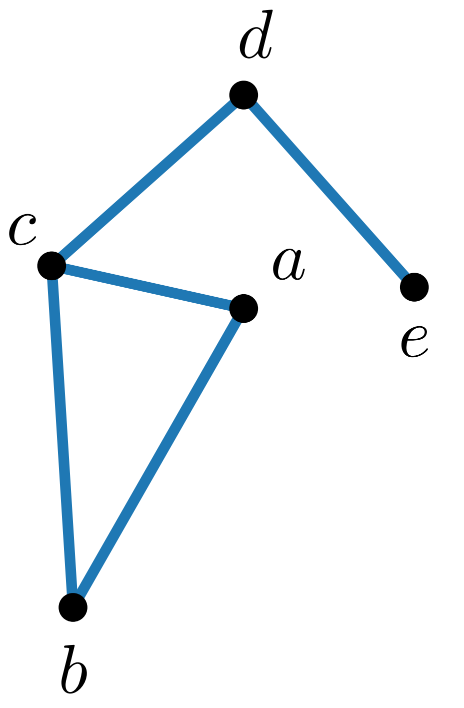
Given: ■ graph G



Planarity + Arc/Chord Diagrams?

Given: ■ graph G

Task: Find a linear order $\prec$ of $V(G)$

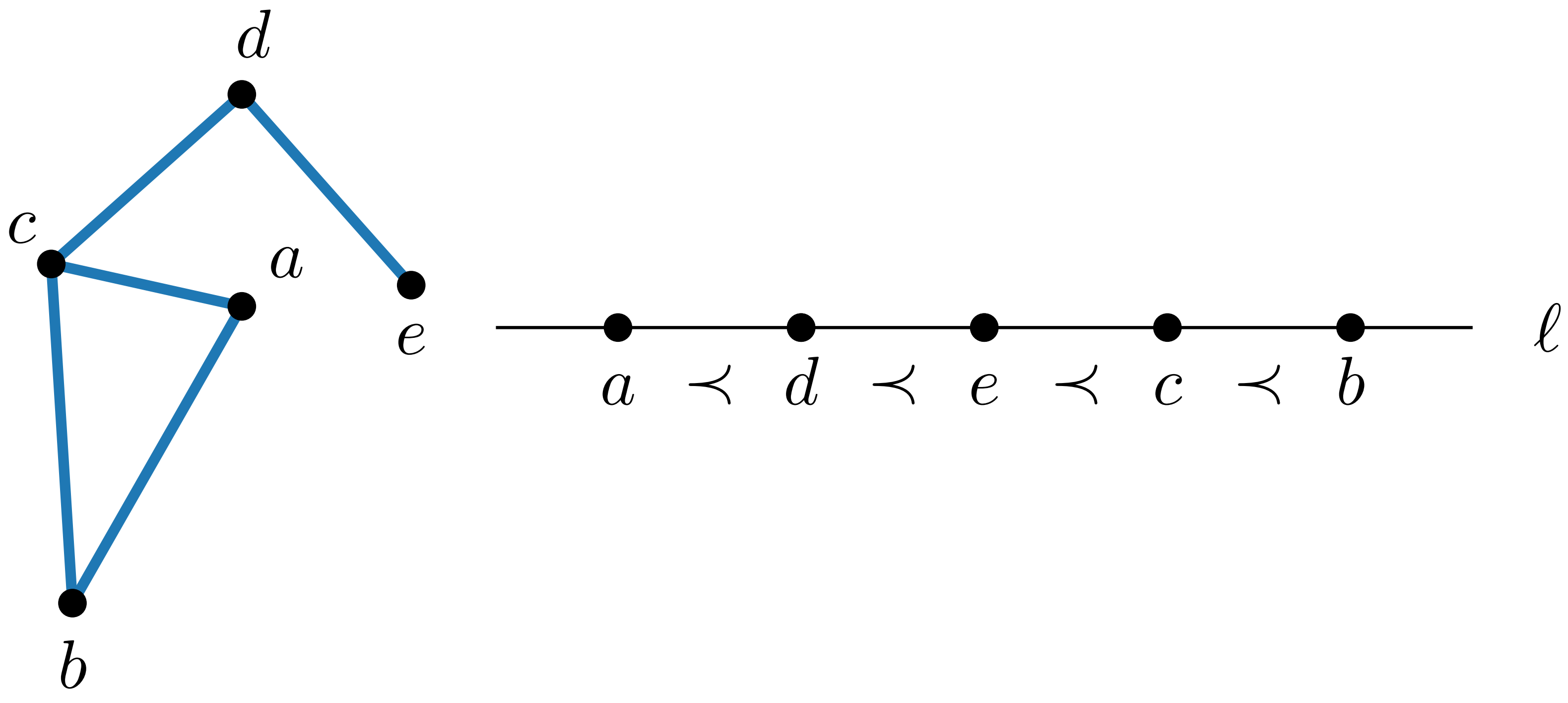


$$a \prec d \prec e \prec c \prec b$$

Planarity + Arc/Chord Diagrams?

Given:
- graph G

Task: Find a linear order $\prec$ of $V(G)$ such that there is a planar drawing where
- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ

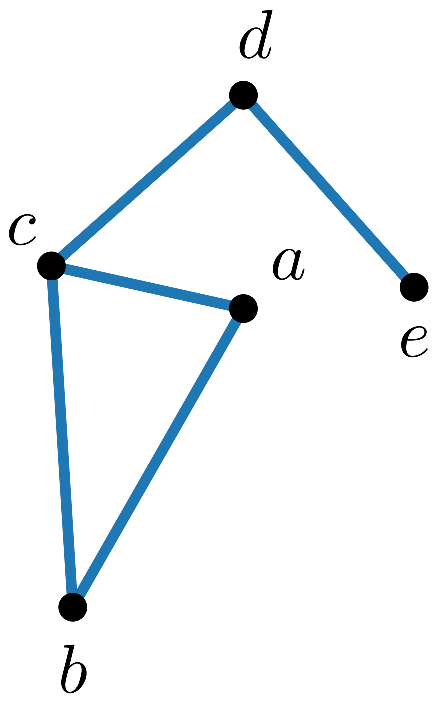


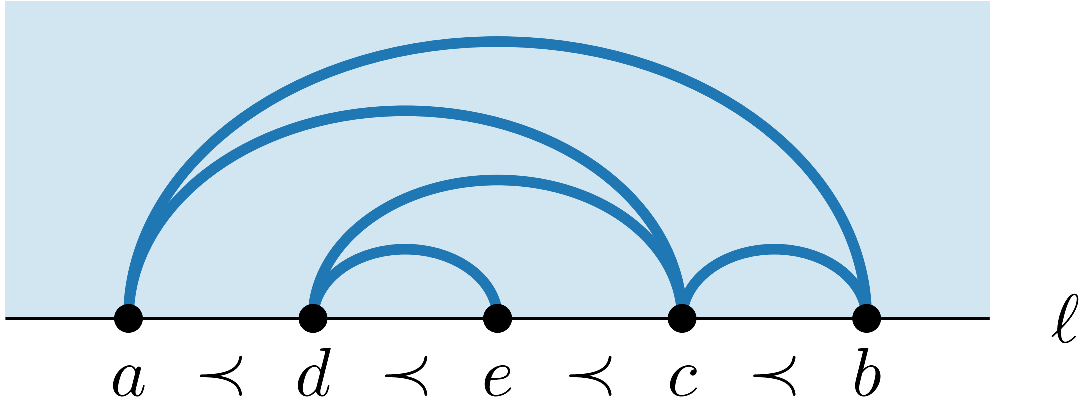
Planarity + Arc/Chord Diagrams?

Given: graph G

Task: Find a linear order $\prec$ of $V(G)$ such that there is a planar drawing where

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- the edges $E(G)$ are drawn as x-monotone arcs in the half plane above ℓ.

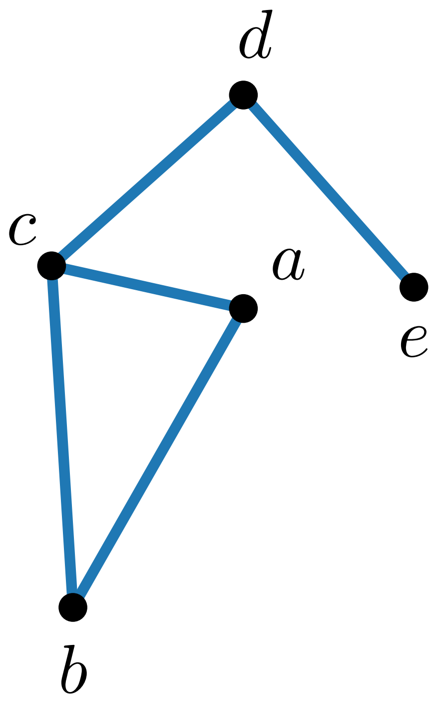


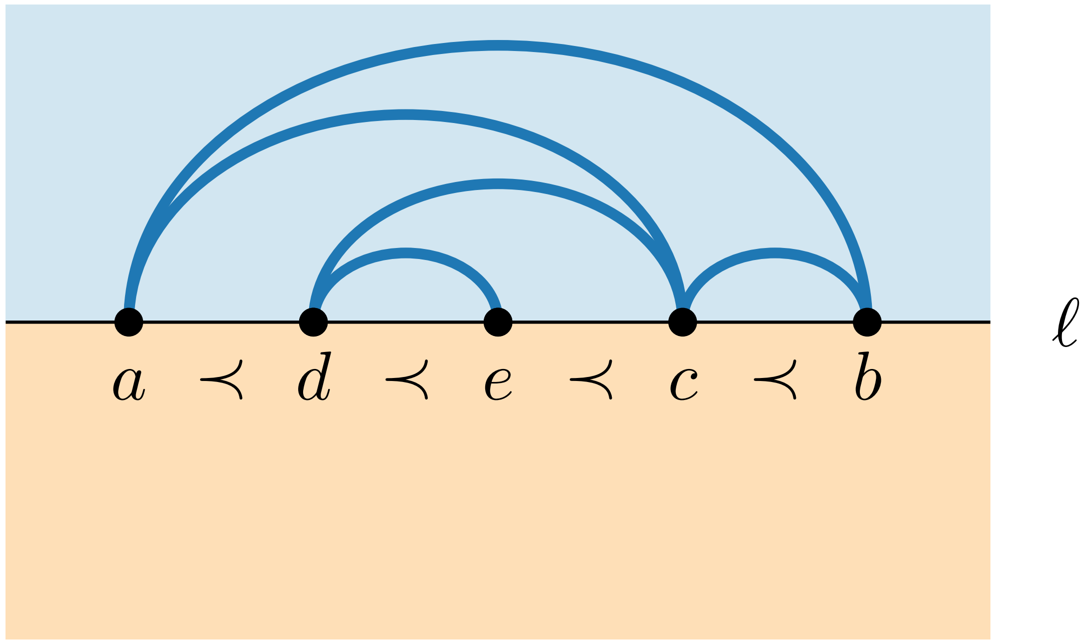


Planarity + Arc/Chord Diagrams?

Given: graph G

Task: Find a linear order $\prec$ of $V(G)$ such that there is a planar drawing where
- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- the edges $E(G)$ are drawn as x-monotone arcs in the half plane above ℓ.





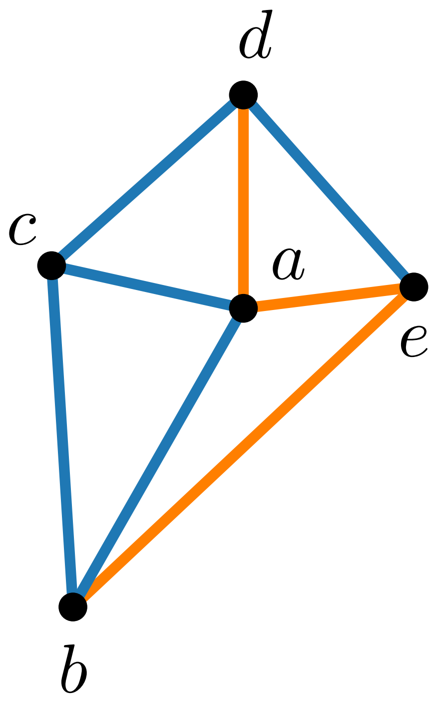
Why not using the half plane below ℓ?

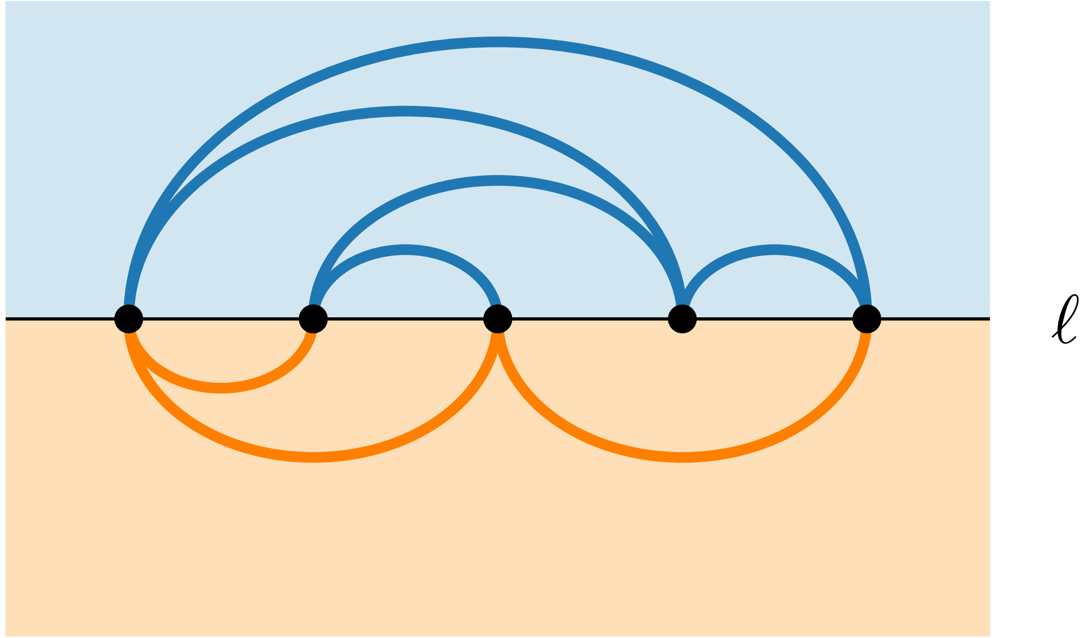
Planarity + Arc/Chord Diagrams?

Given: graph G

Task: Find a linear order $\prec$ of $V(G)$ such that there is a planar drawing where

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- the edges $E(G)$ are drawn as x-monotone arcs in the half plane above ℓ.



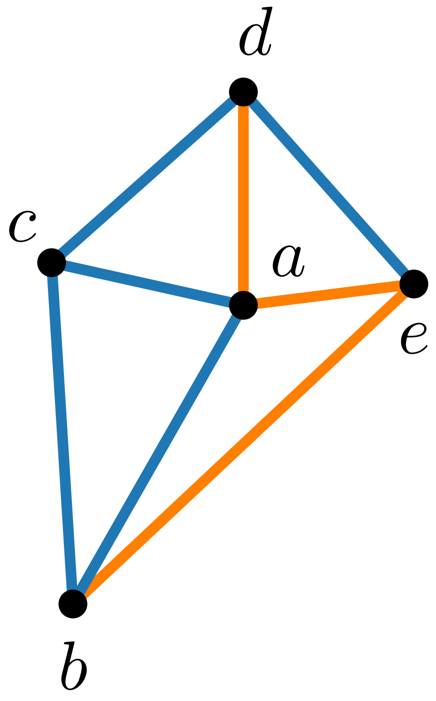


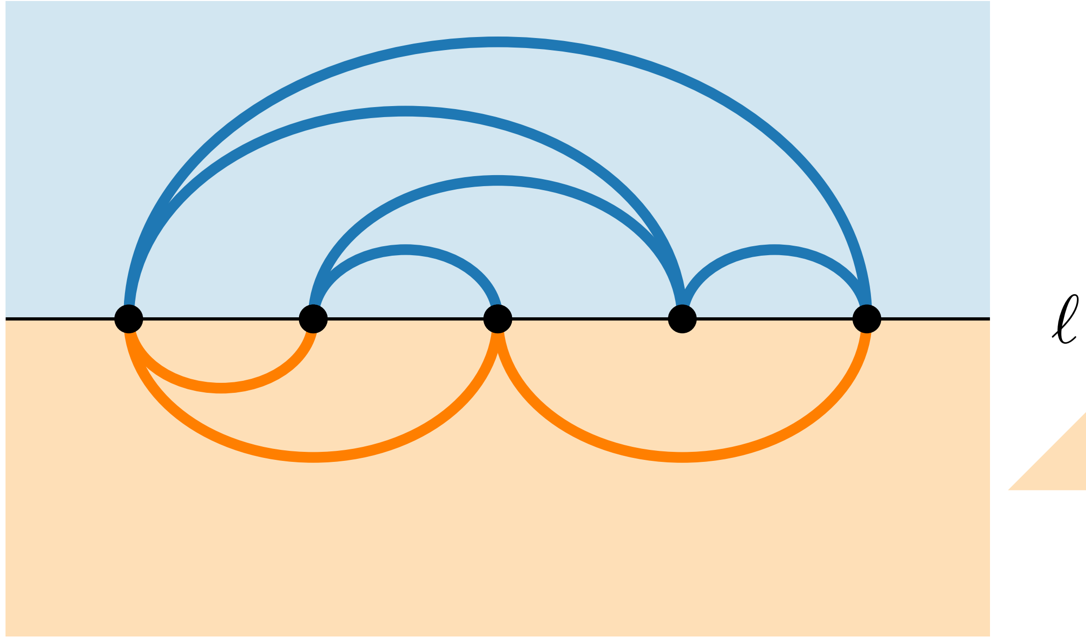
Why not using the half plane below ℓ?

Planarity + Arc/Chord Diagrams?

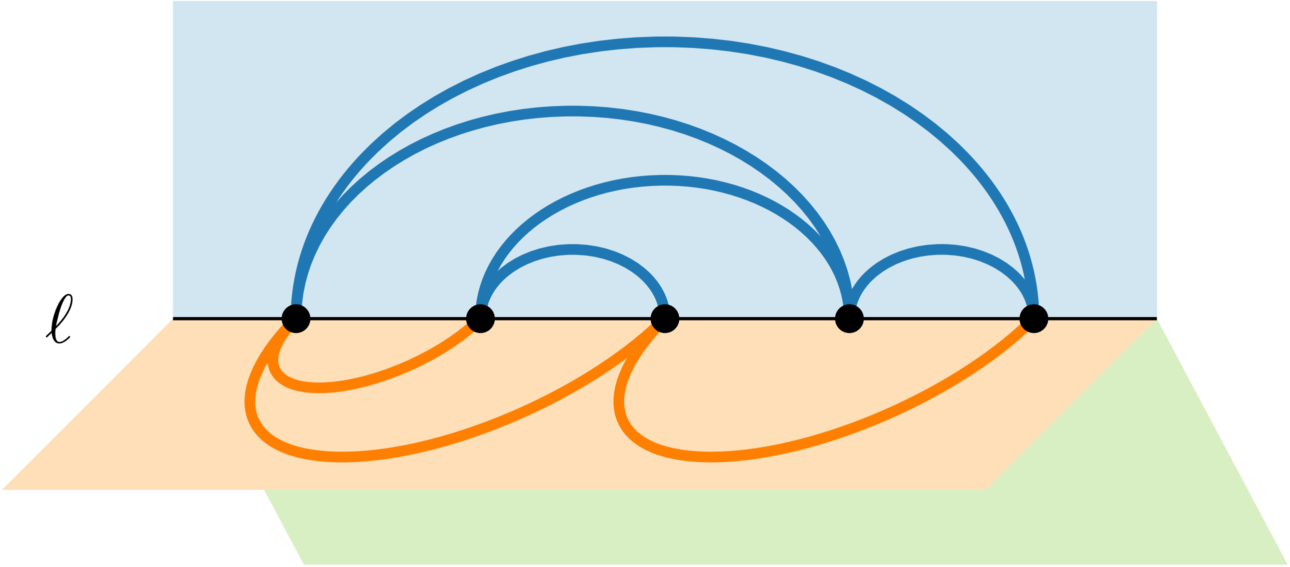
Given:
- graph G

Task: Find a linear order $\prec$ of $V(G)$ such that there is a planar drawing where
- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- the edges $E(G)$ are drawn as x-monotone arcs in the half plane above ℓ.





Why not using the half plane below ℓ?



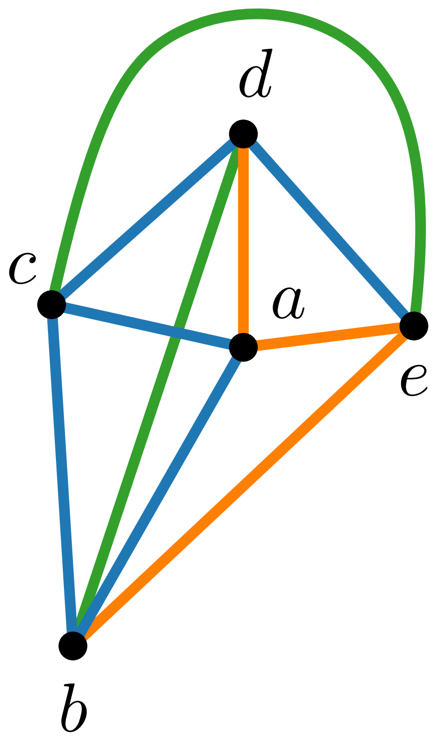
Or even more half planes?

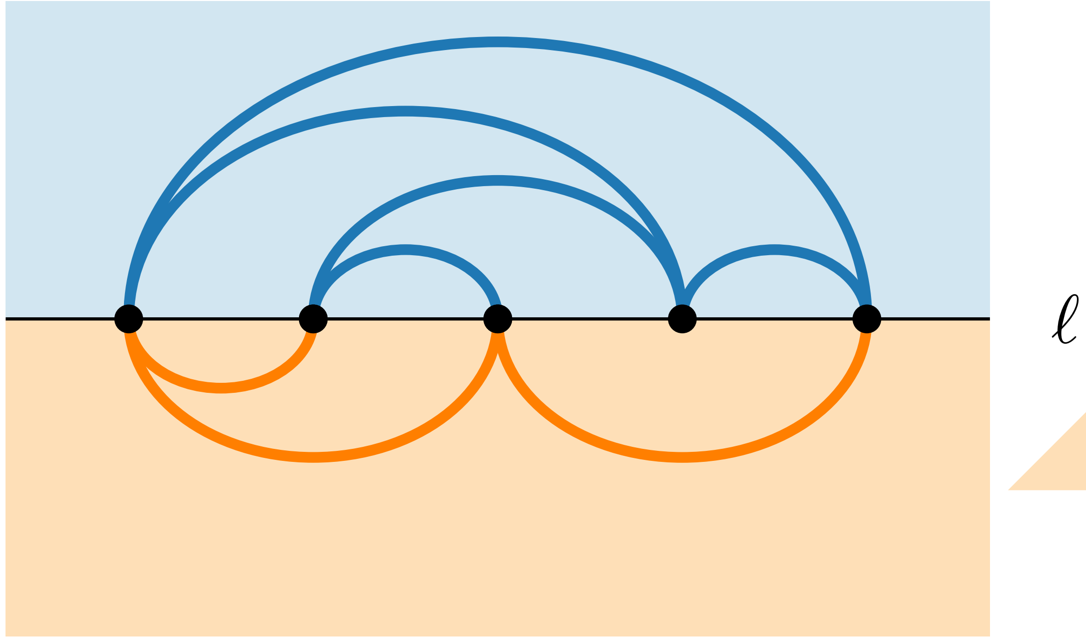
Planarity + Arc/Chord Diagrams?

Given: ■ graph G

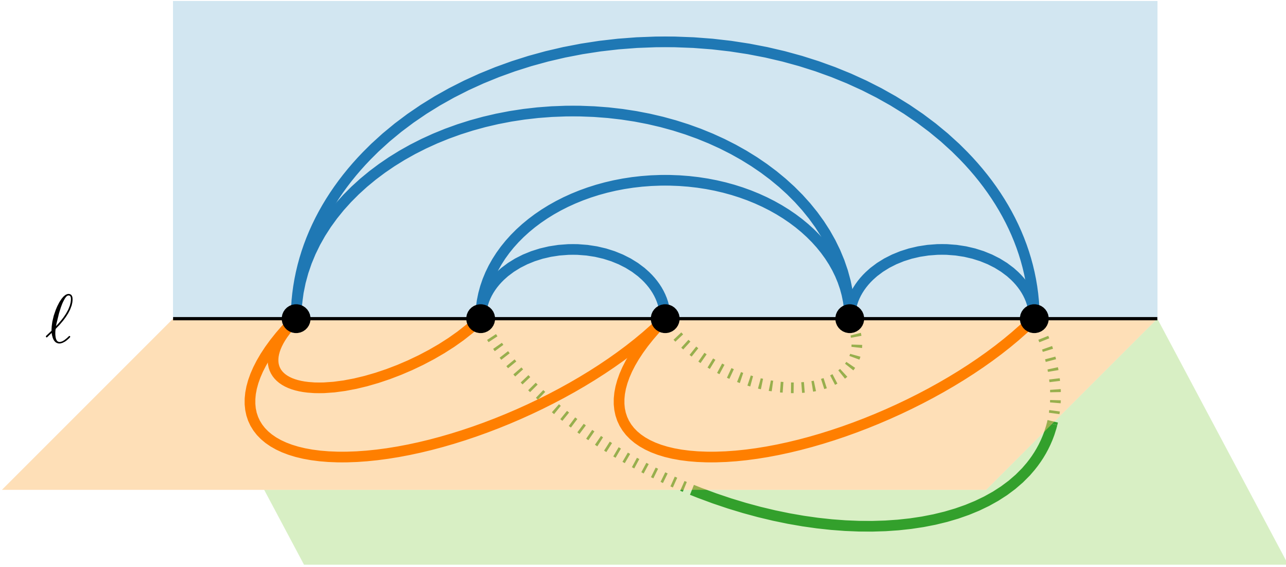
Task: Find a linear order $\prec$ of $V(G)$ such that there is a planar drawing where

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- the edges $E(G)$ are drawn as x-monotone arcs in the half plane above ℓ.





Why not using the half plane below ℓ?

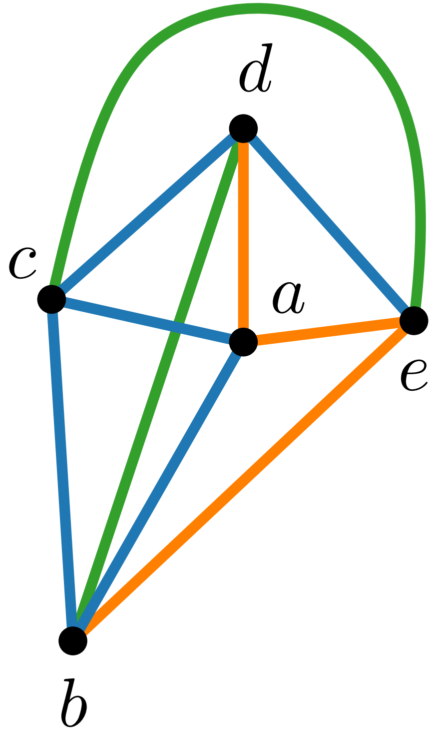


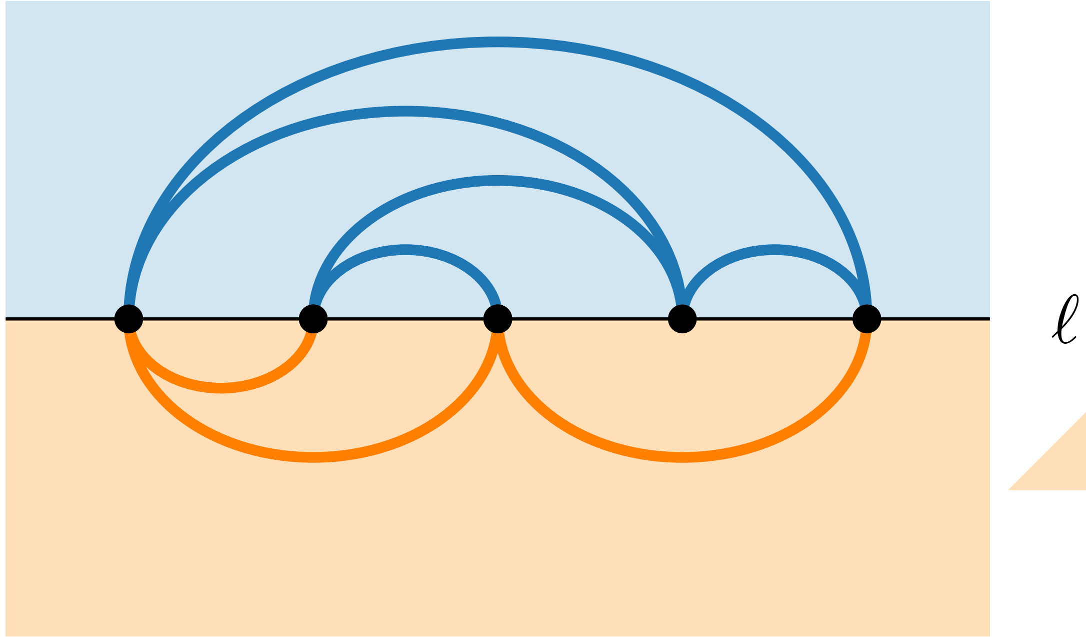
Or even more half planes?

Planarity + Arc/Chord Diagrams?

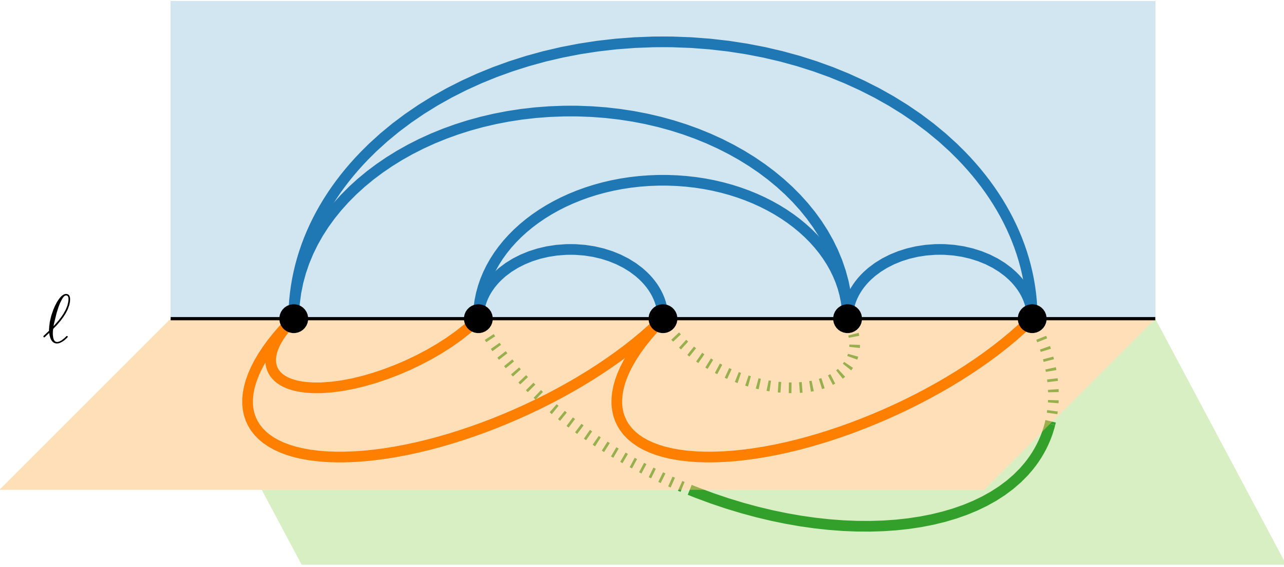
Given:
- graph G

Task: Find a linear order $\prec$ of $V(G)$ such that there is a planar drawing where
- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- the edges $E(G)$ are drawn as x-monotone arcs in the half plane above ℓ.





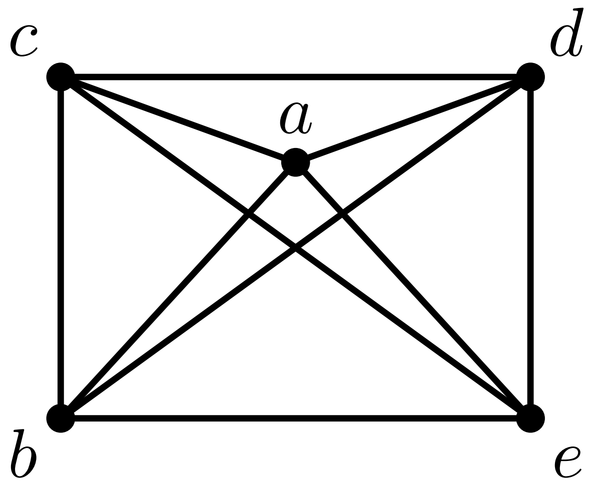
Why not using the half plane below ℓ?



Or even more half planes?
→ **book embeddings**

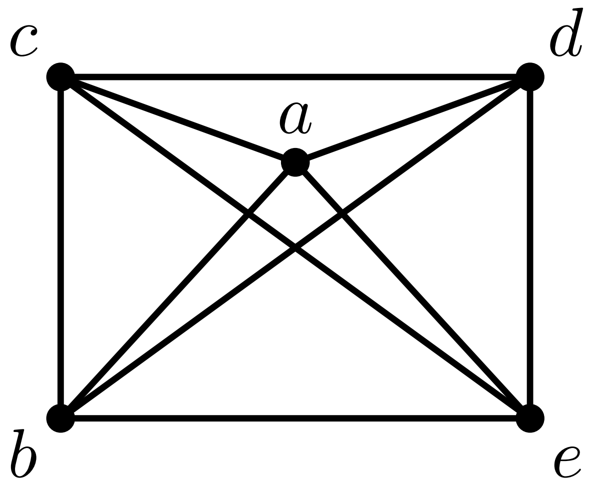
Book Embeddings

Given: ■ graph G



Book Embeddings

Given:
- graph G
- integer k



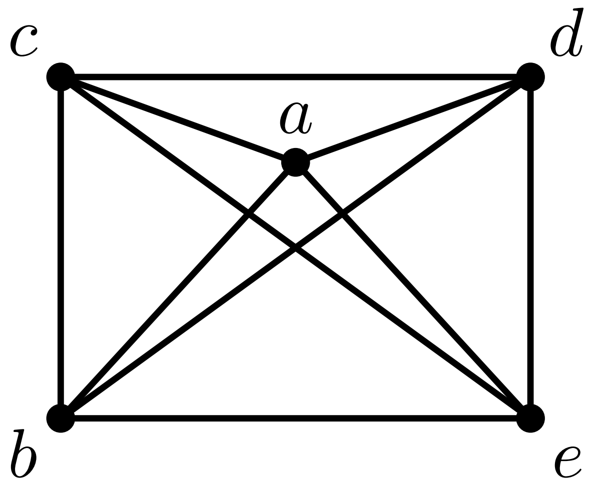
$k = 3$

Book Embeddings

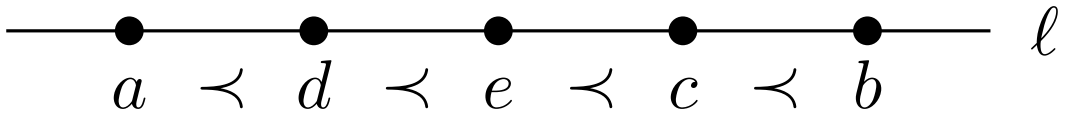
Given:
- graph G
- integer k

Task: Find a linear order $\prec$ of $V(G)$ such that . . .

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and



$k = 3$

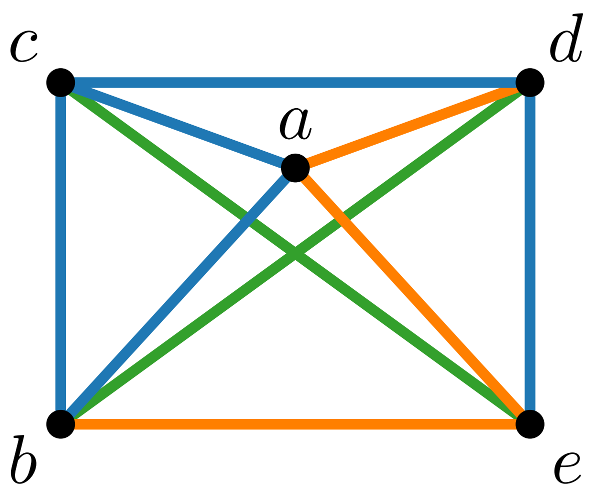


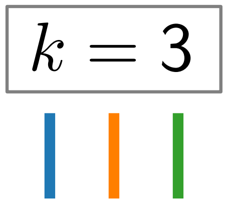
Book Embeddings

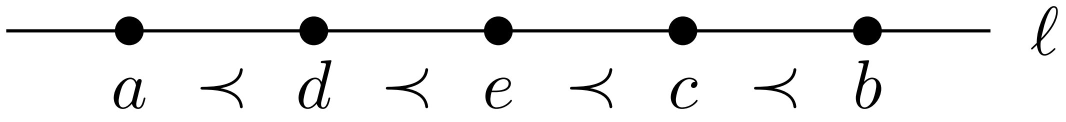
Given:
- graph G
- integer k

Task: Find (i) a linear order $\prec$ of $V(G)$ and (ii) an assigment $p\colon E(G) \to \{1, \ldots, k\}$ such that ...

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and





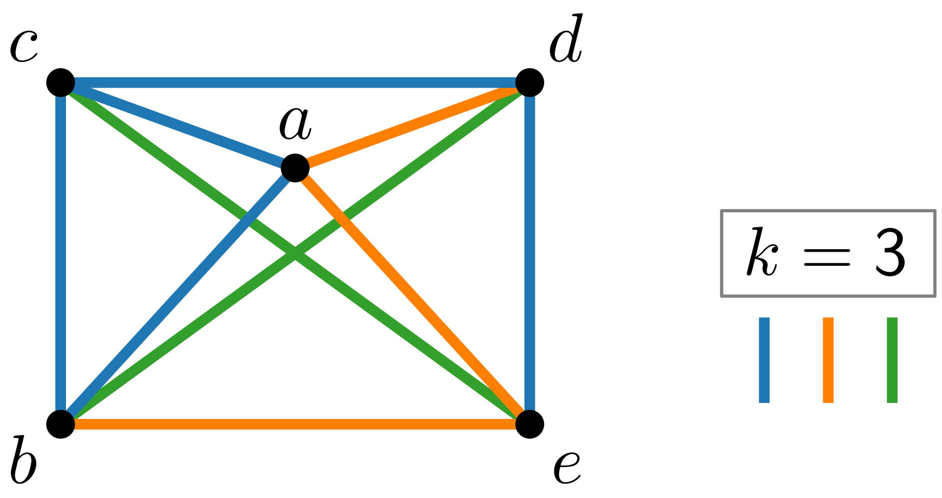


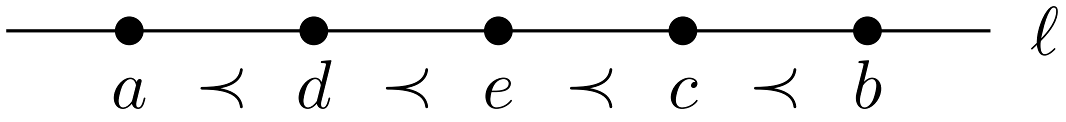
Book Embeddings

Given:
- graph G
- integer k

Task: Find (i) a linear order $\prec$ of $V(G)$ and (ii) an assigment $p\colon E(G) \to \{1, \dots, k\}$ such that ...

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- for each $i \in \{1, \dots, k\}$, the edges in $p^{-1}(i)$ are drawn as x-monotone arcs without crossings in a (separate) half plane delimited by ℓ.



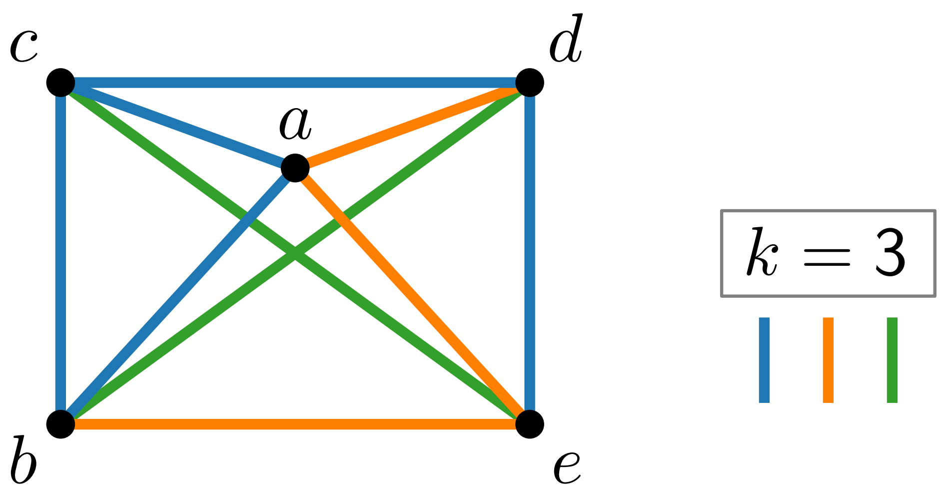


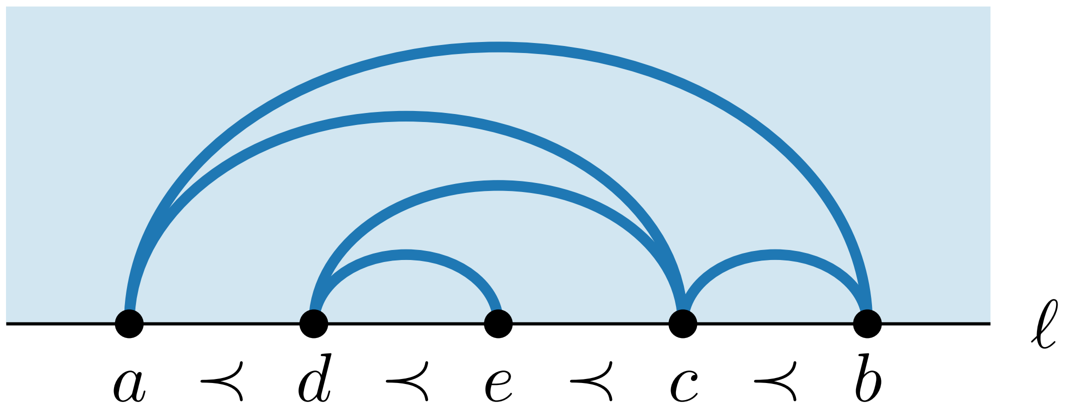
Book Embeddings

Given:
- graph G
- integer k

Task: Find (i) a linear order $\prec$ of $V(G)$ and (ii) an assigment $p\colon E(G) \to \{1, \ldots, k\}$ such that ...

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- for each $i \in \{1, \ldots, k\}$, the edges in $p^{-1}(i)$ are drawn as x-monotone arcs without crossings in a (separate) half plane delimited by ℓ.



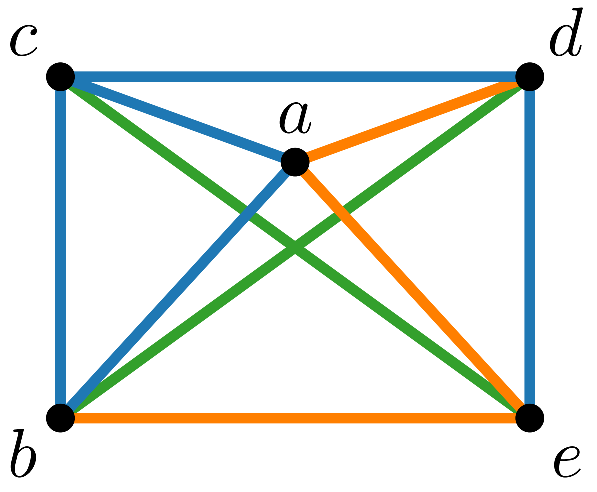


Book Embeddings

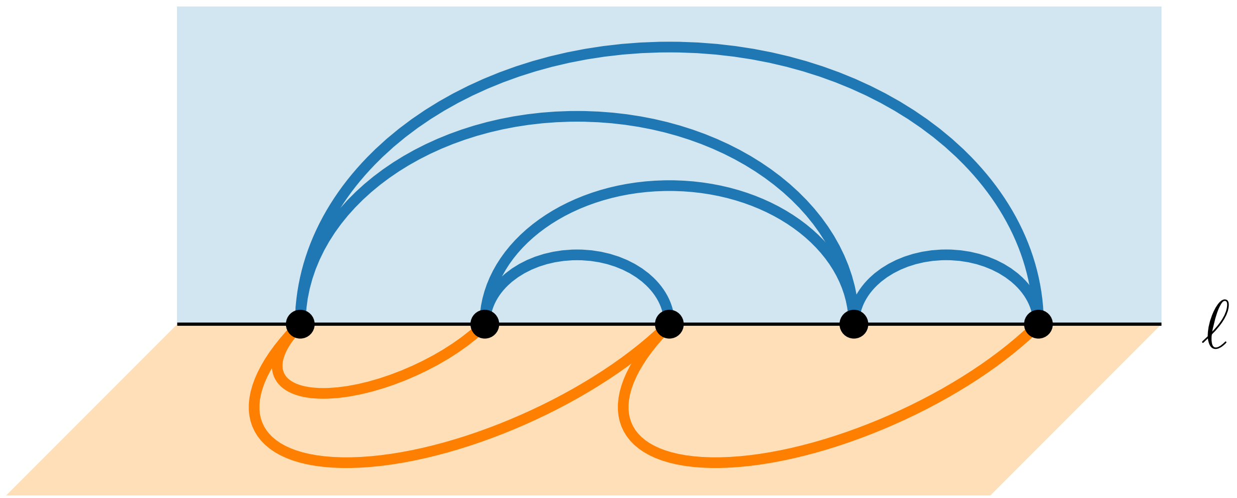
Given:
- graph G
- integer k

Task: Find (i) a linear order $\prec$ of $V(G)$ and (ii) an assigment $p\colon E(G) \to \{1,\dots,k\}$ such that ...

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- for each $i \in \{1,\dots,k\}$, the edges in $p^{-1}(i)$ are drawn as x-monotone arcs without crossings in a (separate) half plane delimited by ℓ.



$k = 3$

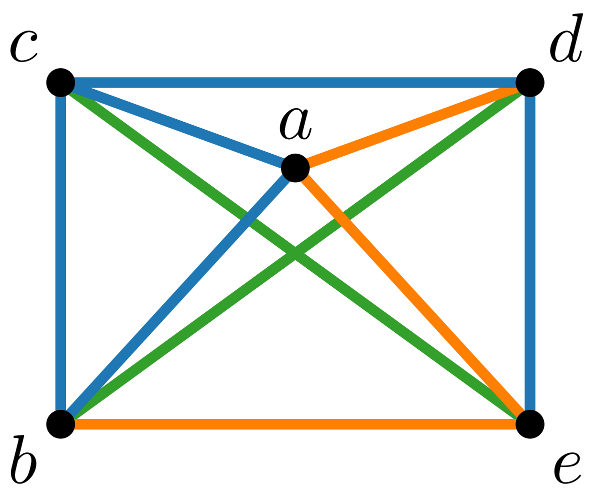


Book Embeddings

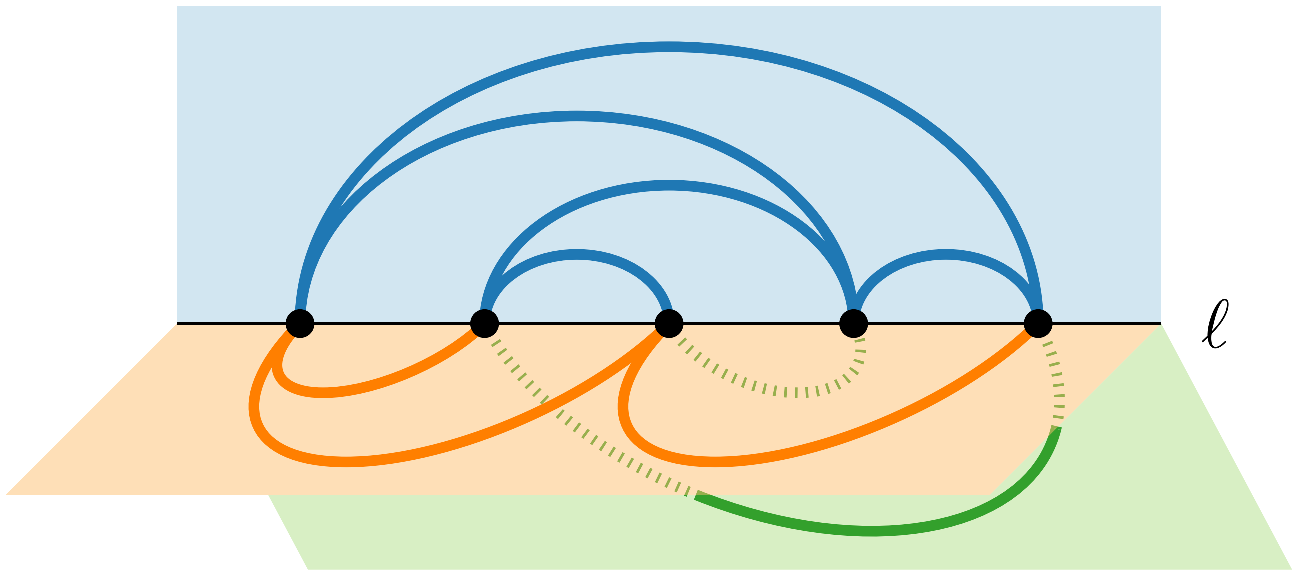
Given:
- graph G
- integer k

Task: Find (i) a linear order $\prec$ of $V(G)$ and (ii) an assigment $p\colon E(G) \to \{1, \ldots, k\}$ such that ...

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- for each $i \in \{1, \ldots, k\}$, the edges in $p^{-1}(i)$ are drawn as x-monotone arcs without crossings in a (separate) half plane delimited by ℓ.



$k = 3$

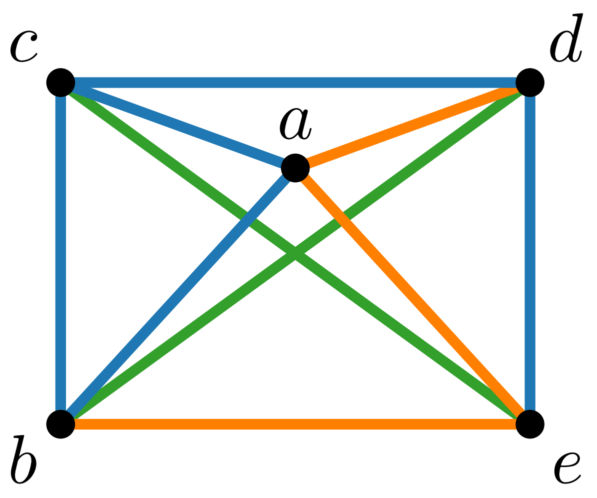


Book Embeddings

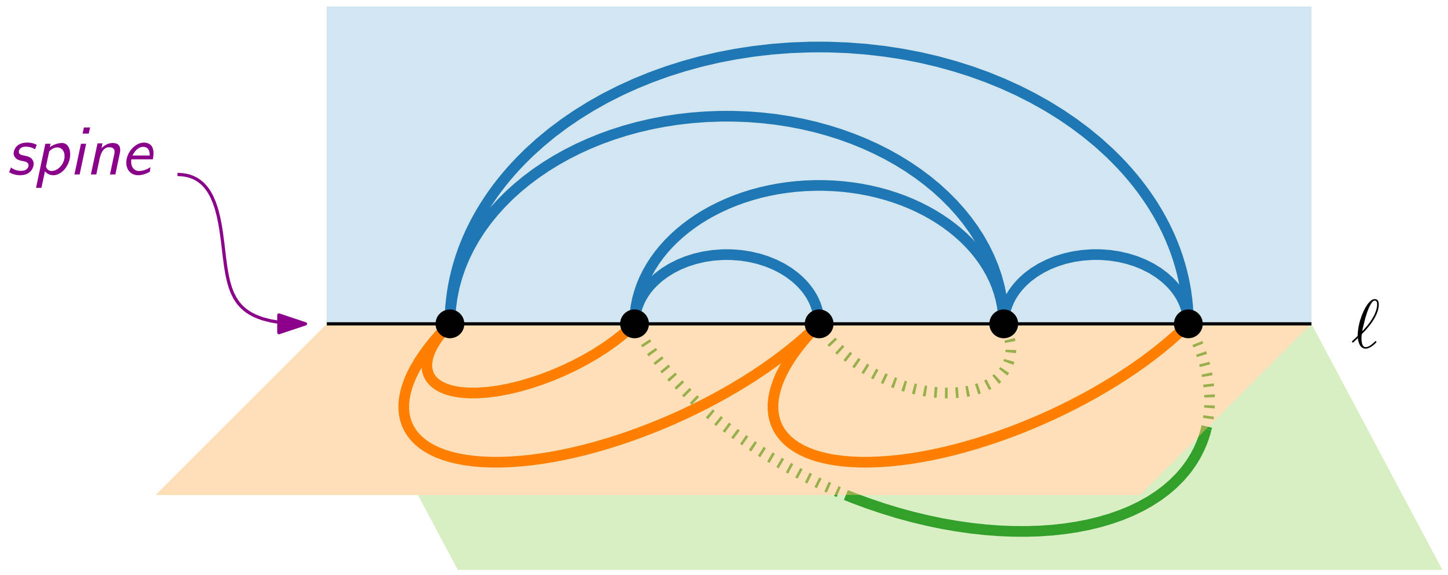
Given:
- graph G
- integer k

Task: Find (i) a linear order $\prec$ of $V(G)$ and (ii) an assigment $p\colon E(G) \to \{1, \ldots, k\}$ such that ...

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- for each $i \in \{1, \ldots, k\}$, the edges in $p^{-1}(i)$ are drawn as x-monotone arcs without crossings in a (separate) half plane delimited by ℓ.



$k = 3$

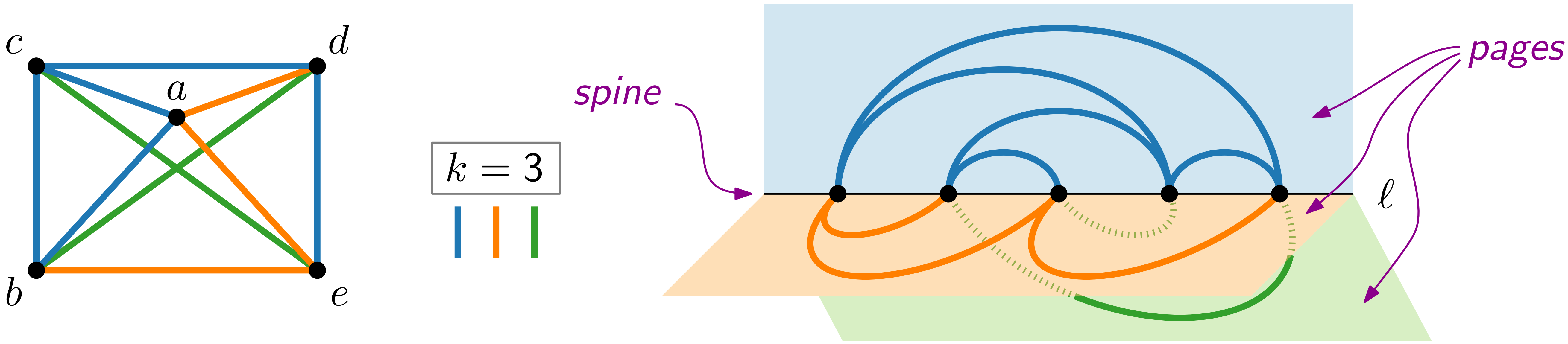


Book Embeddings

Given:
- graph G
- integer k

Task: Find (i) a linear order $\prec$ of $V(G)$ and (ii) an assigment $p\colon E(G) \to \{1, \ldots, k\}$ such that ...

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- for each $i \in \{1, \ldots, k\}$, the edges in $p^{-1}(i)$ are drawn as x-monotone arcs without crossings in a (separate) half plane delimited by ℓ.

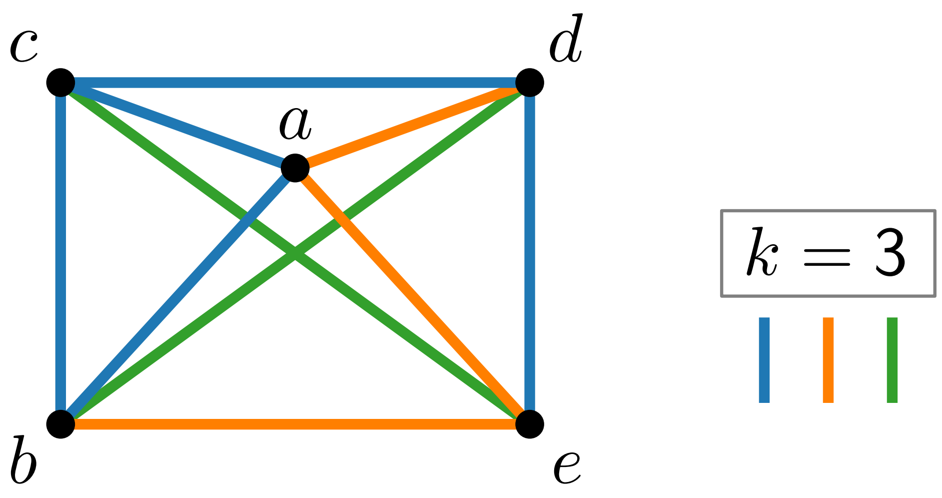


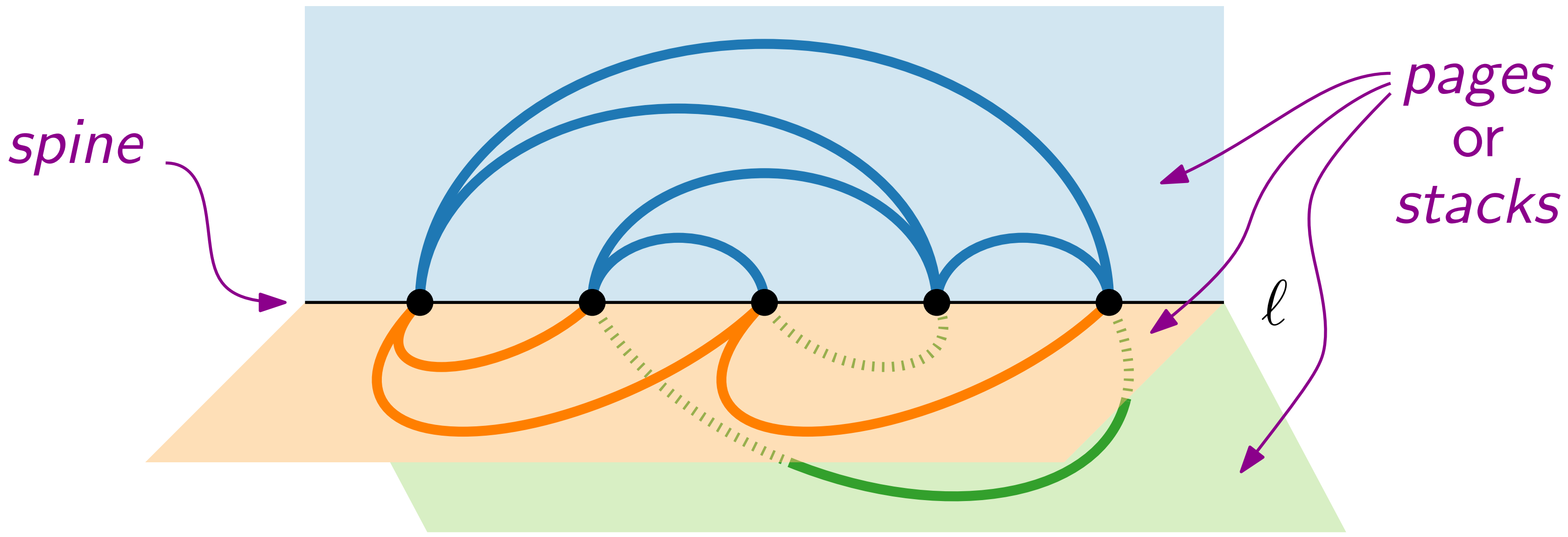
Book Embeddings (Stack Layouts)

Given:
- graph G
- integer k

Task: Find (i) a linear order $\prec$ of $V(G)$ and (ii) an assigment $p \colon E(G) \to \{1, \dots, k\}$ such that ...

- the vertices $V(G)$ in order $\prec$ are arranged along a horizontal line ℓ and
- for each $i \in \{1, \dots, k\}$, the edges in $p^{-1}(i)$ are drawn as x-monotone arcs without crossings in a (separate) half plane delimited by ℓ.





But Why *Stacks*?!

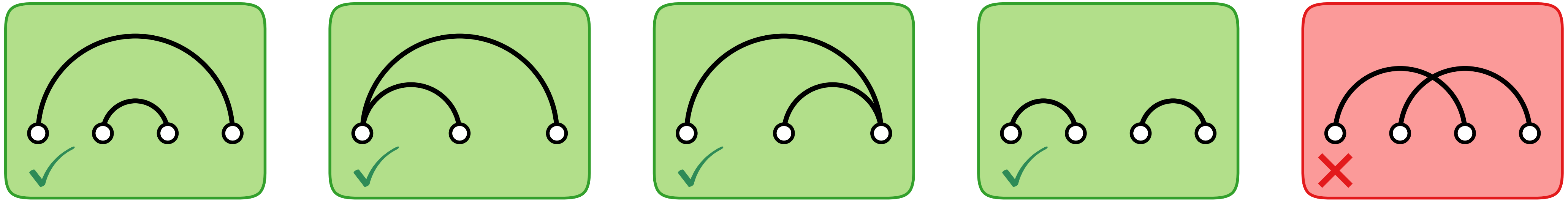
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

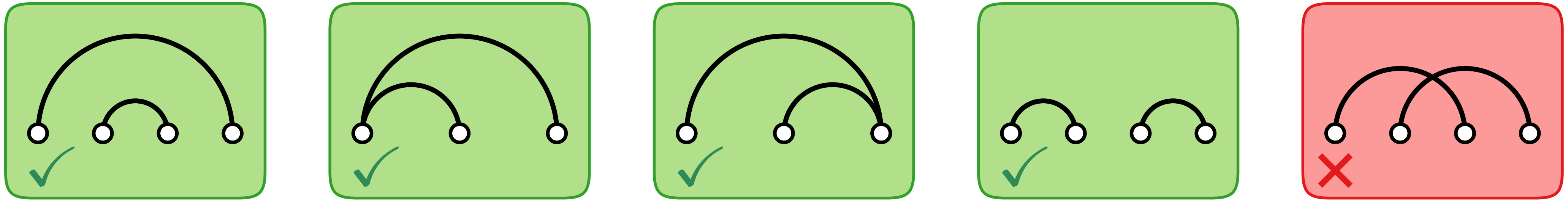
Stack Layouts:



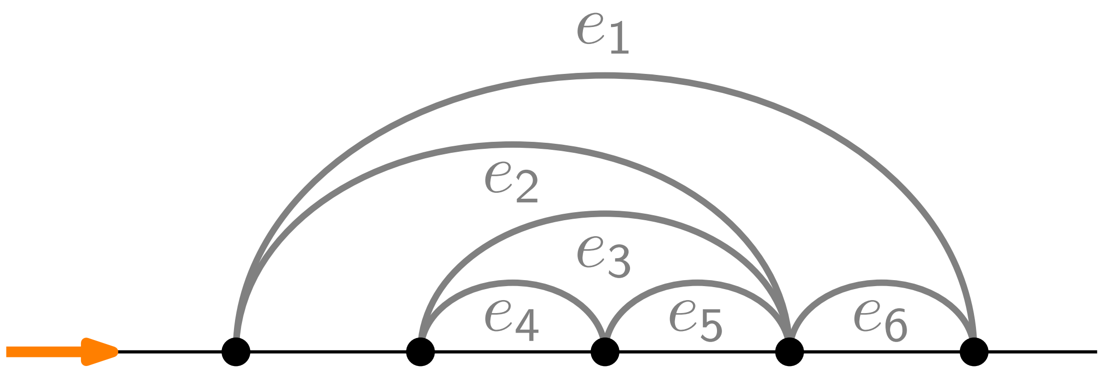
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



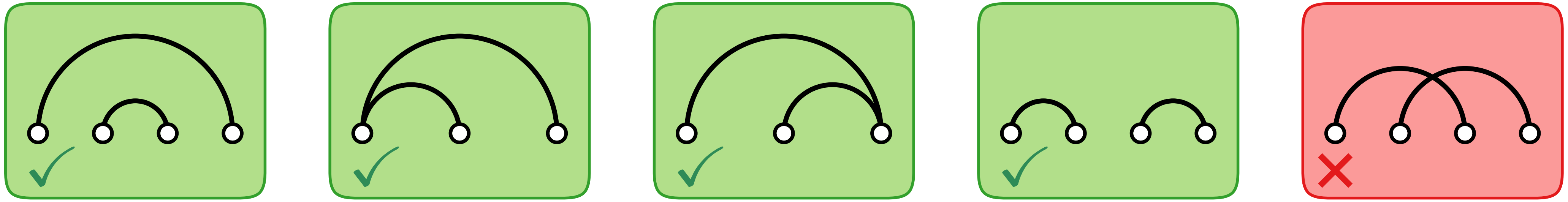
- For one stack, traverse the spine from left to right.



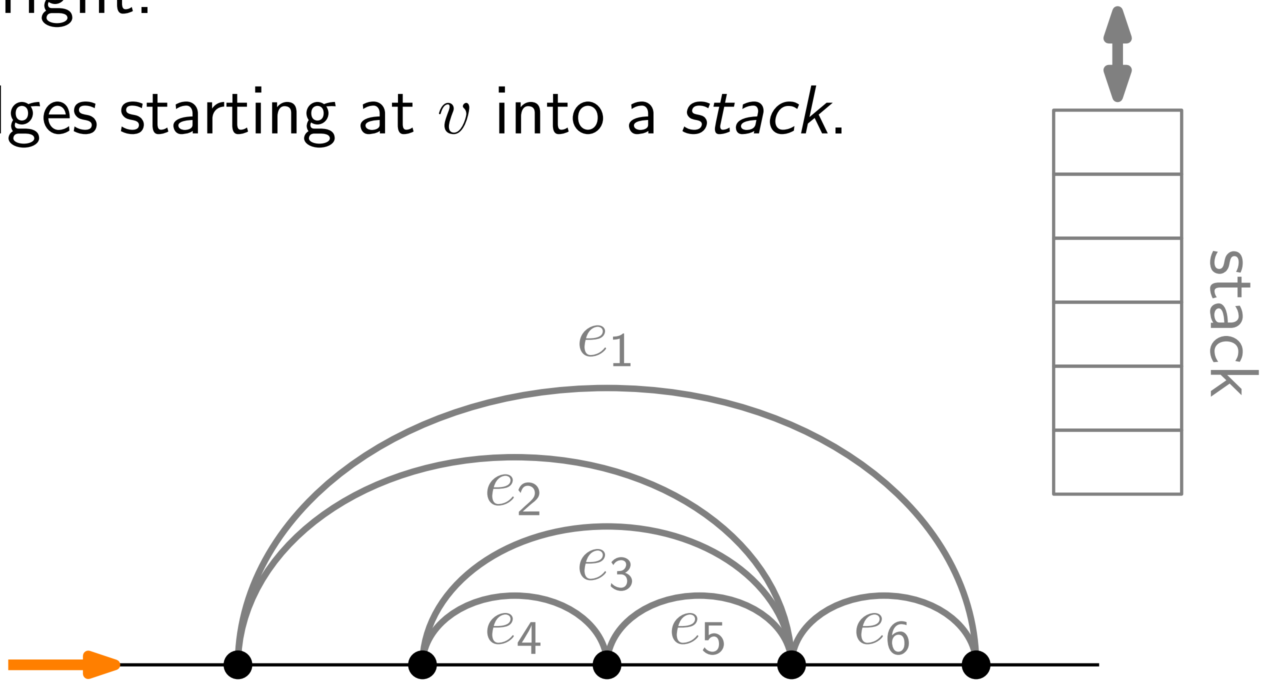
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



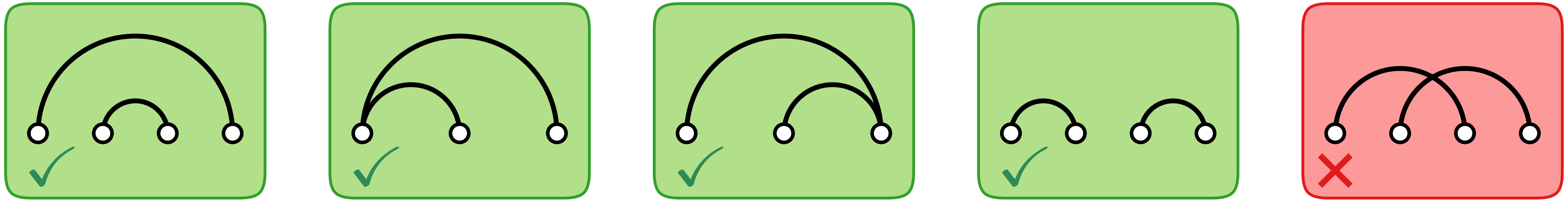
- For one stack, traverse the spine from left to right.
- Whenever we encounter a vertex v, put the edges starting at v into a *stack*.



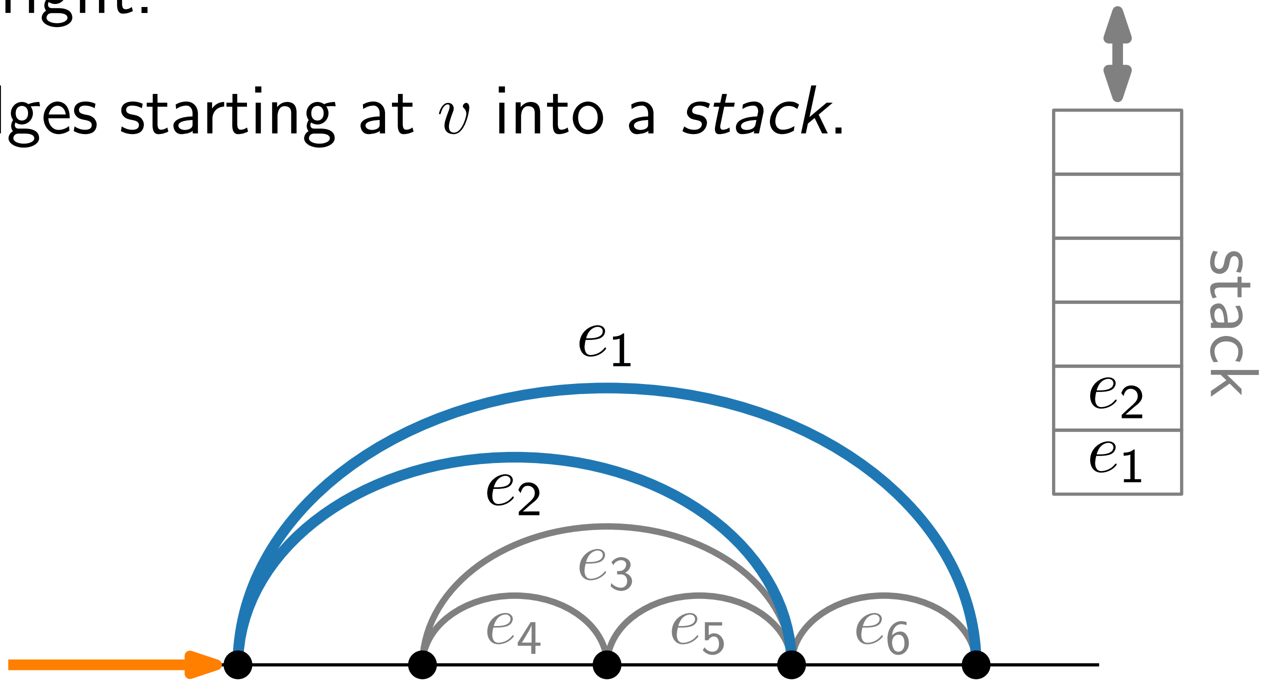
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



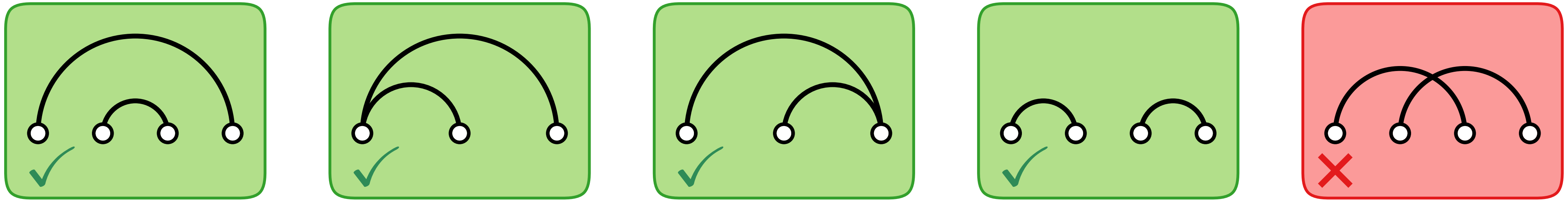
- For one stack, traverse the spine from left to right.
- Whenever we encounter a vertex v, put the edges starting at v into a *stack*.



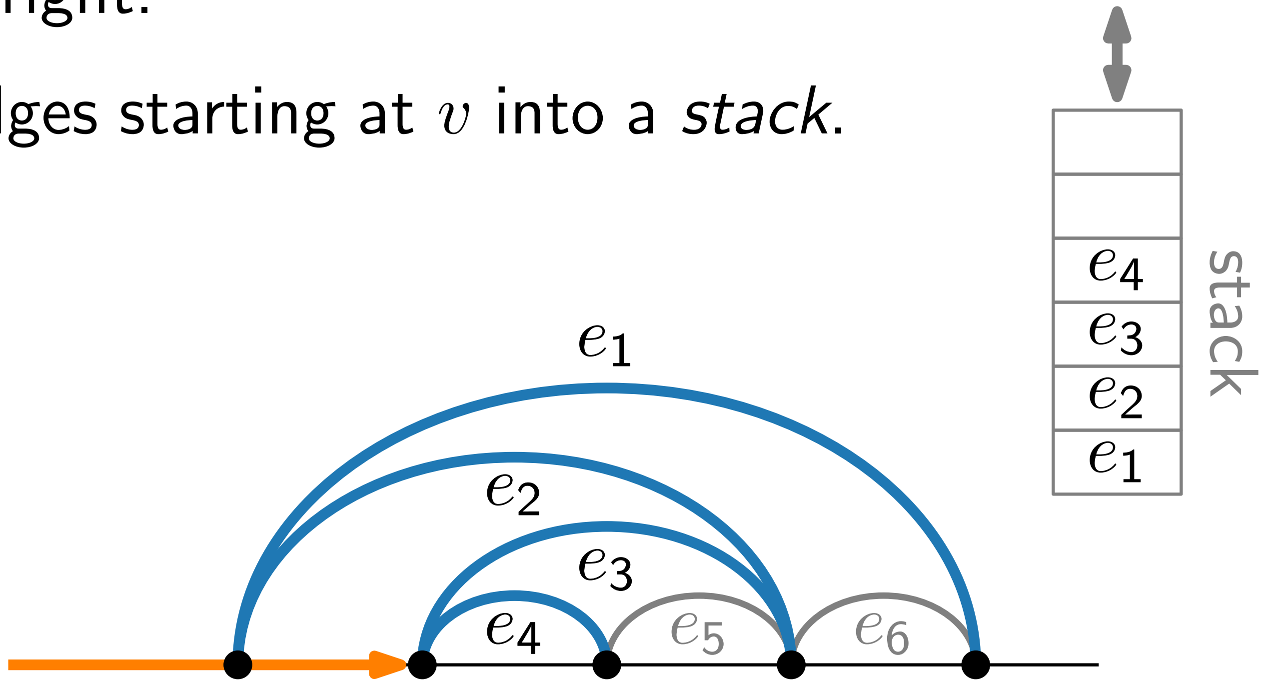
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



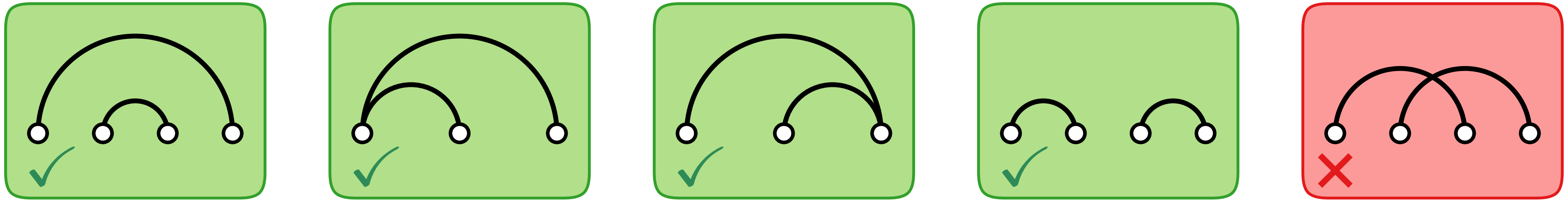
- For one stack, traverse the spine from left to right.
- Whenever we encounter a vertex v, put the edges starting at v into a *stack*.



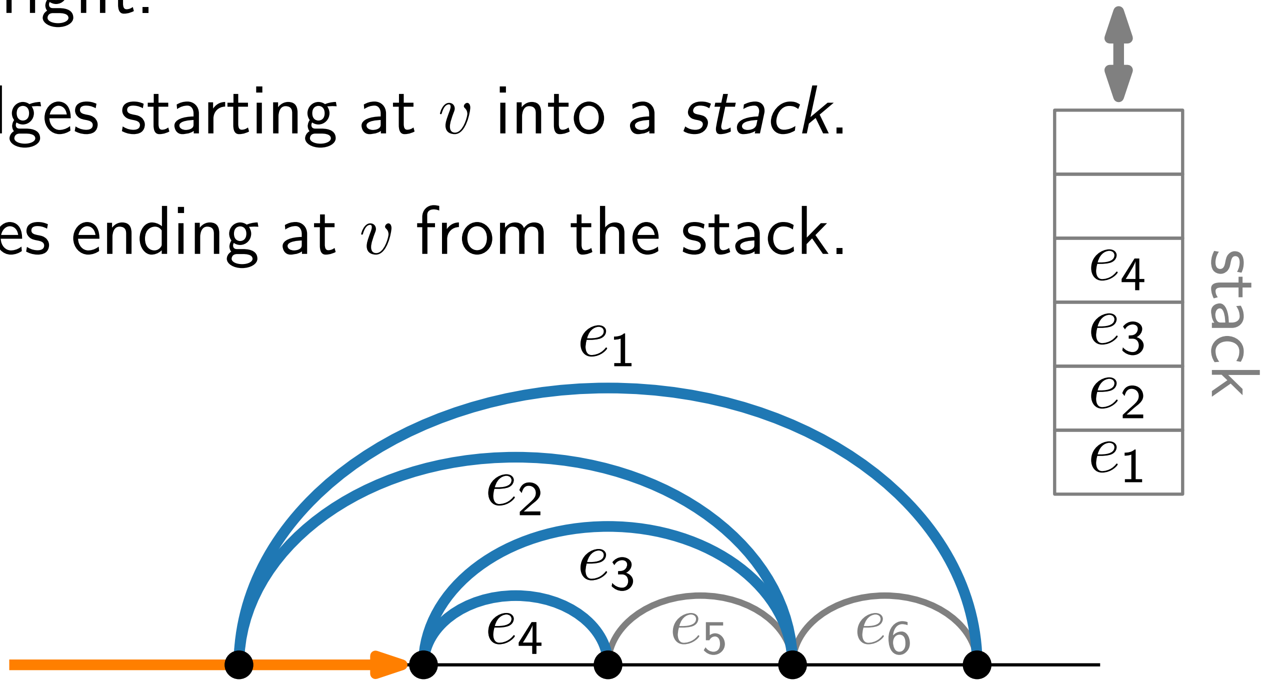
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



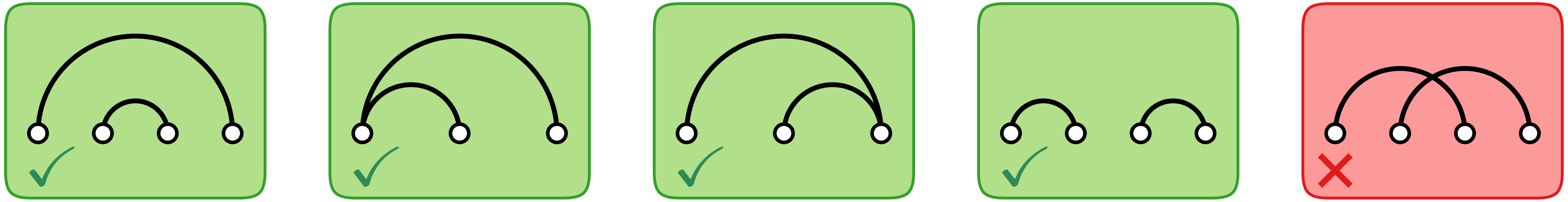
- For one stack, traverse the spine from left to right.
- Whenever we encounter a vertex v, put the edges starting at v into a *stack*.
- Before we put edges on the stack, we pop edges ending at v from the stack.



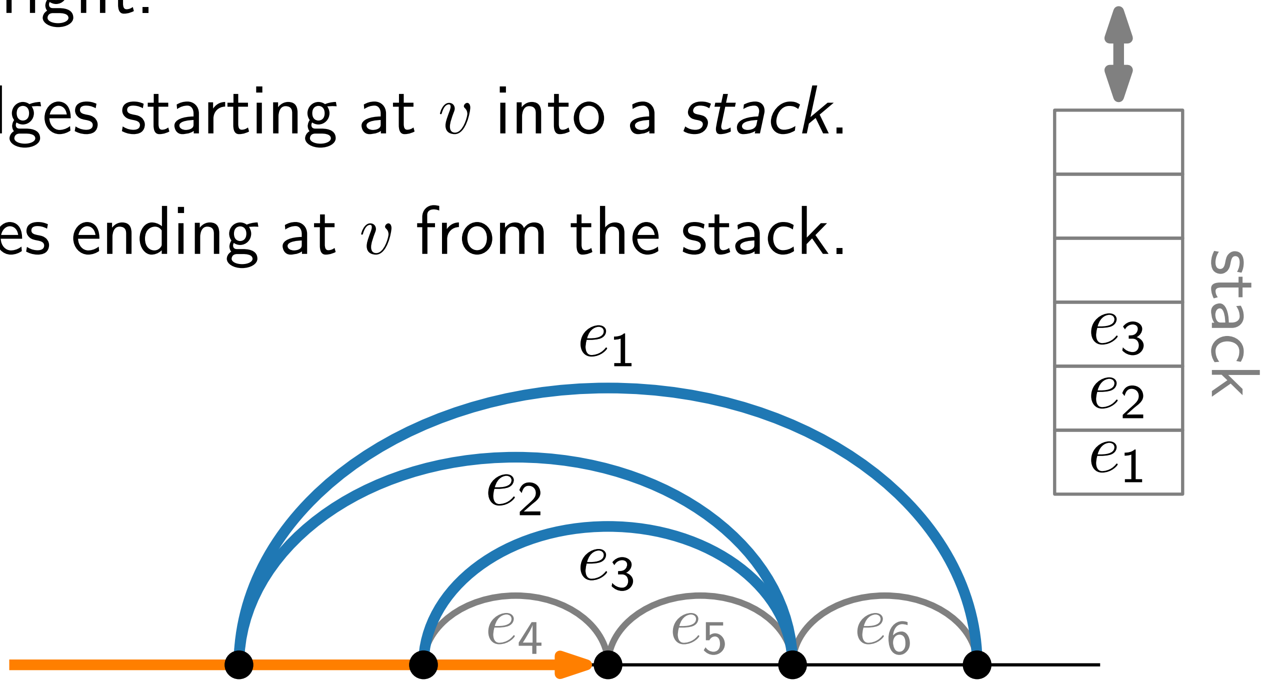
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



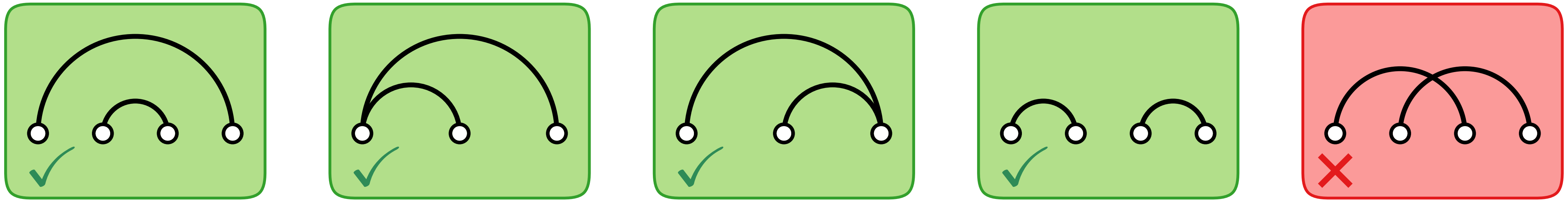
- For one stack, traverse the spine from left to right.
- Whenever we encounter a vertex v, put the edges starting at v into a *stack*.
- Before we put edges on the stack, we pop edges ending at v from the stack.



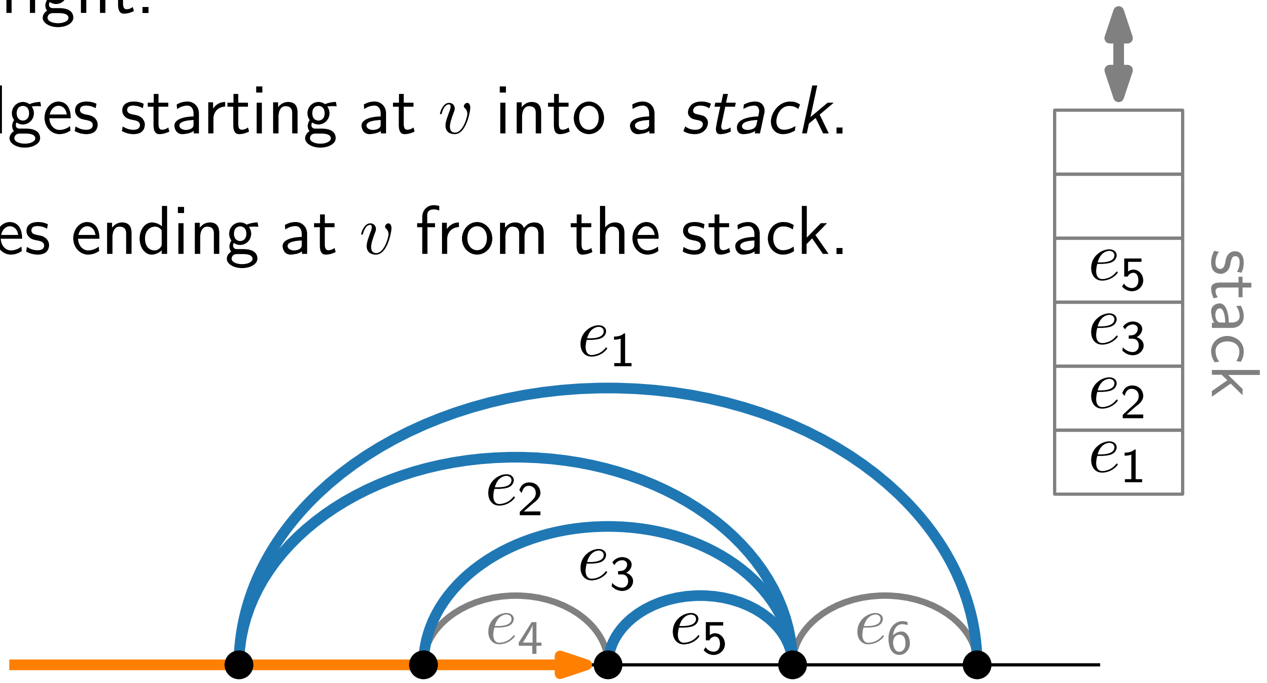
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



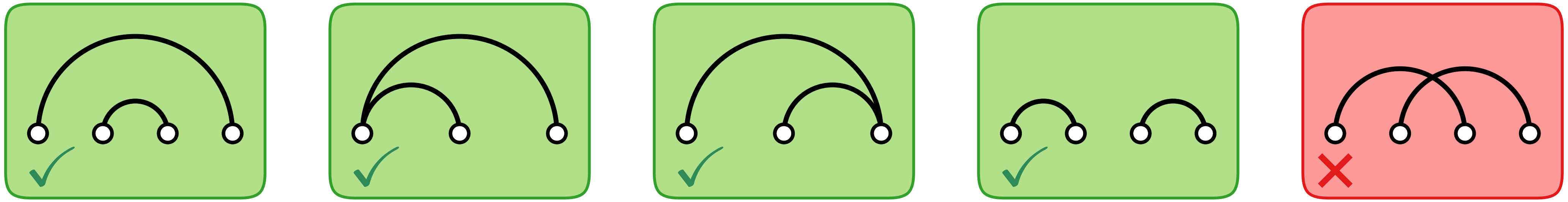
- For one stack, traverse the spine from left to right.
- Whenever we encounter a vertex v, put the edges starting at v into a *stack*.
- Before we put edges on the stack, we pop edges ending at v from the stack.



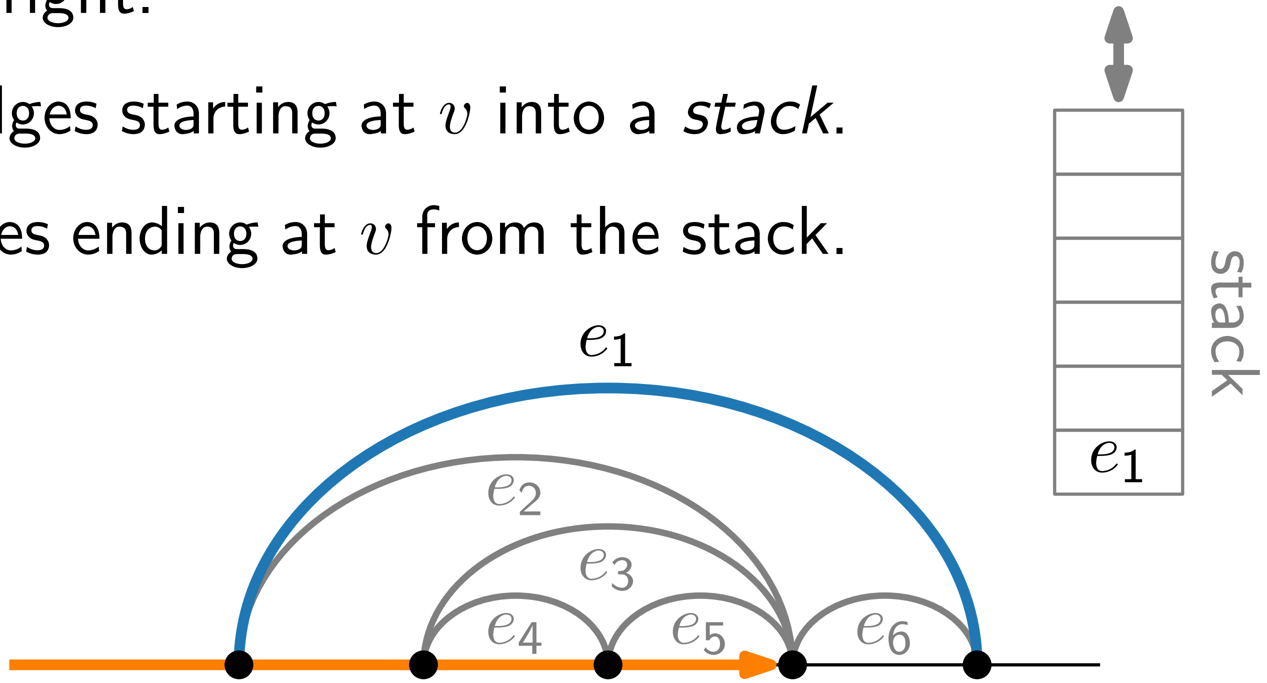
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



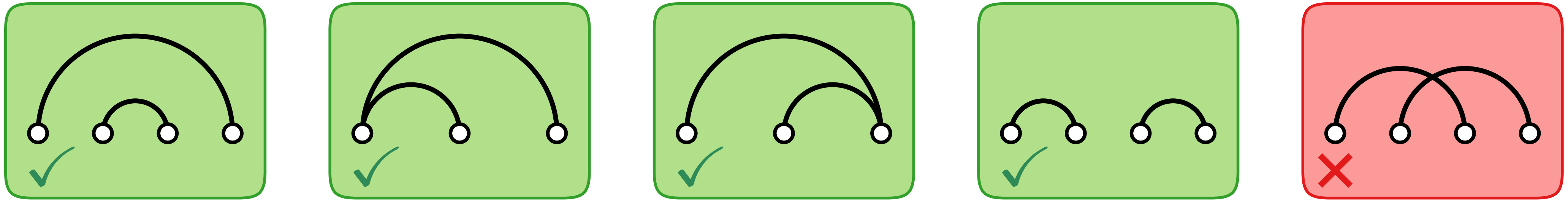
- For one stack, traverse the spine from left to right.
- Whenever we encounter a vertex v, put the edges starting at v into a *stack*.
- Before we put edges on the stack, we pop edges ending at v from the stack.



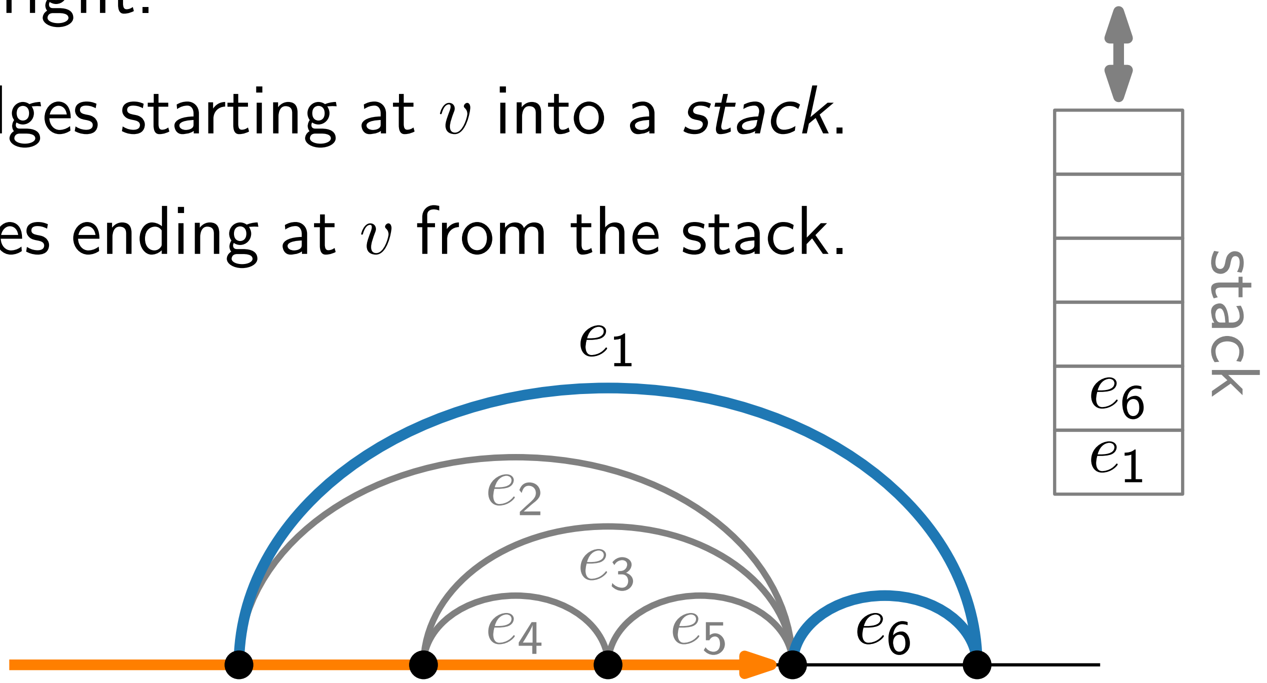
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



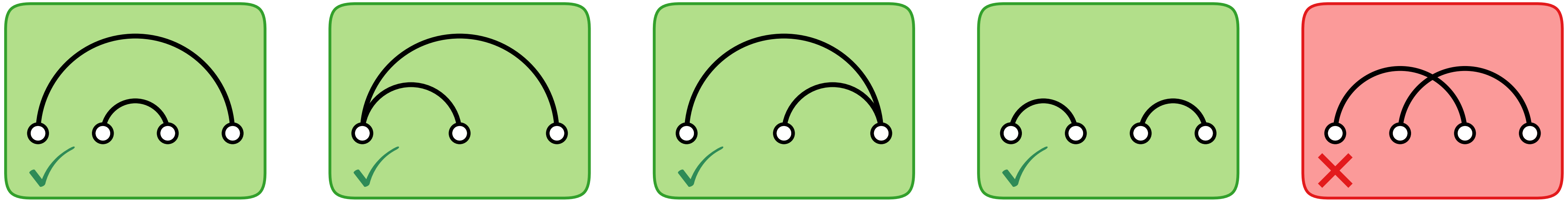
- For one stack, traverse the spine from left to right.
- Whenever we encounter a vertex v, put the edges starting at v into a *stack*.
- Before we put edges on the stack, we pop edges ending at v from the stack.



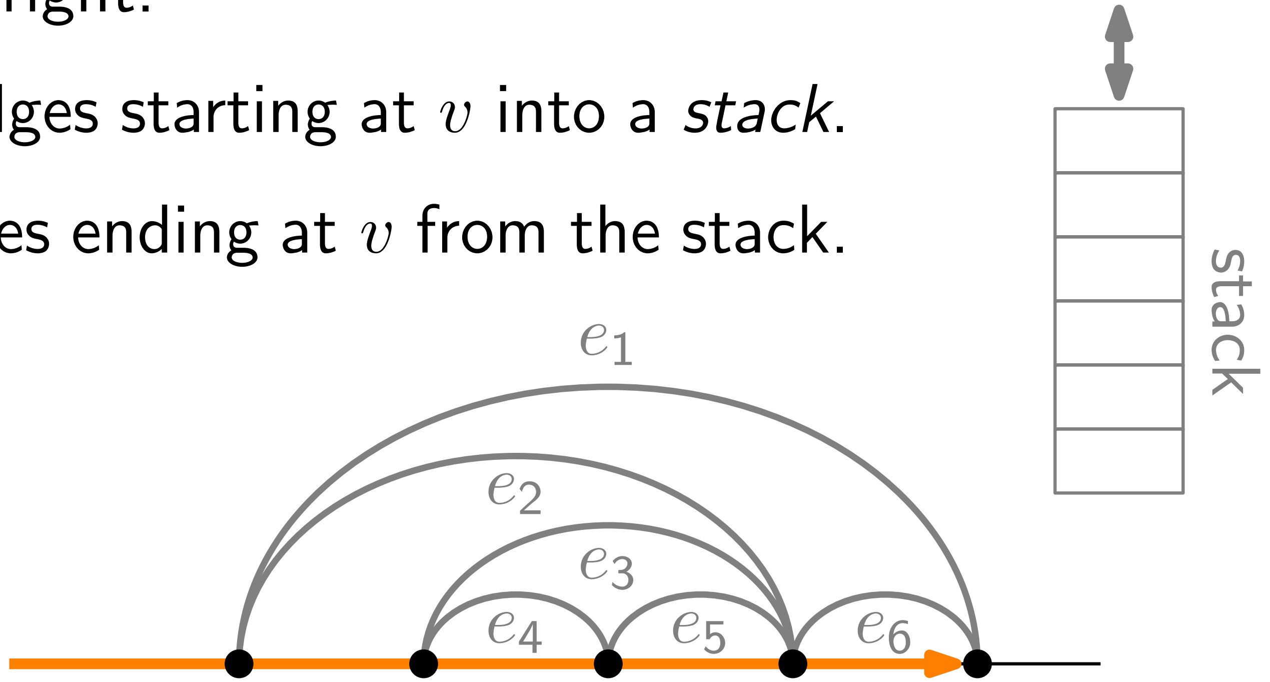
But Why *Stacks*?!

- Consider stack layouts purely combinatorially in terms of allowed and forbidden patterns.

Stack Layouts:



- For one stack, traverse the spine from left to right.
- Whenever we encounter a vertex v, put the edges starting at v into a *stack*.
- Before we put edges on the stack, we pop edges ending at v from the stack.



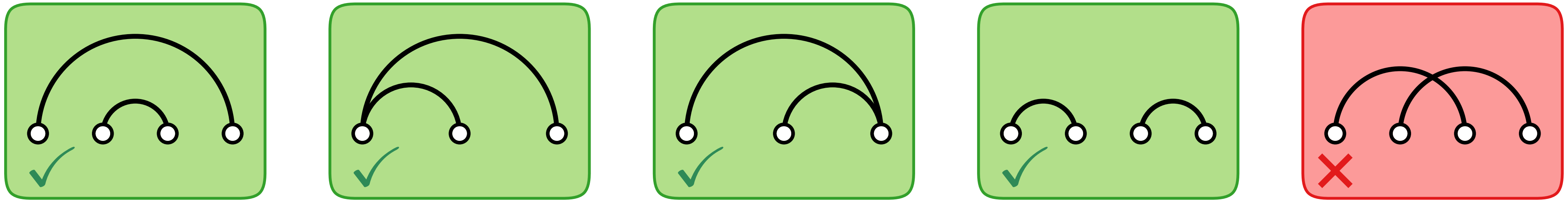
Have people studied linear layouts using other data structures than stacks?

Have people studied linear layouts using other data structures than stacks? Yes, queues!

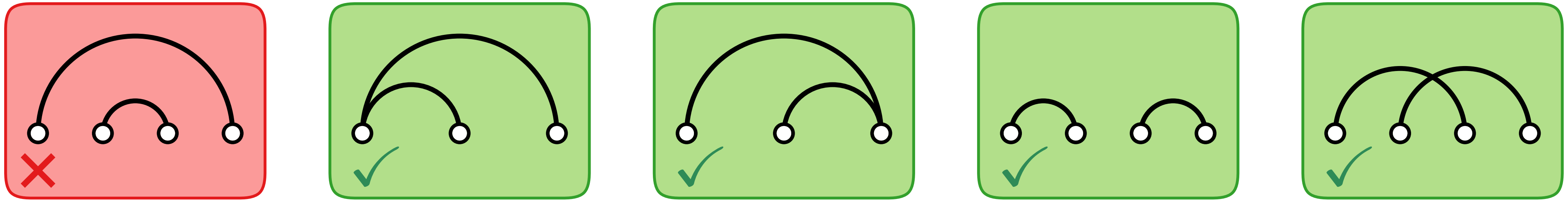
Queue Layouts

Have people studied linear layouts using other data structures than stacks? Yes, queues!

Stack Layouts:



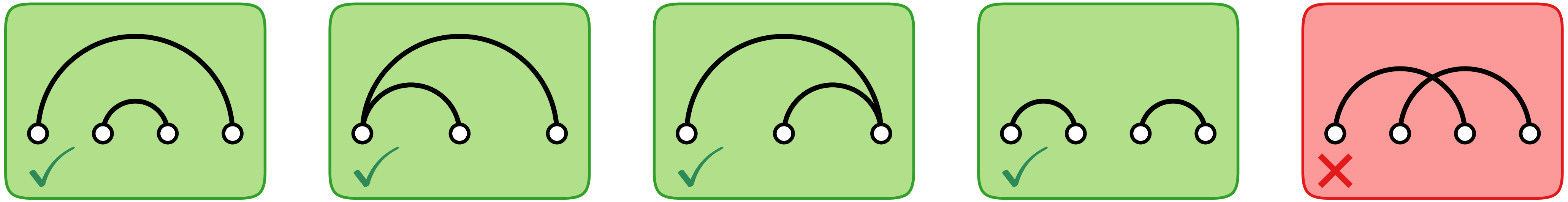
Queue Layouts:



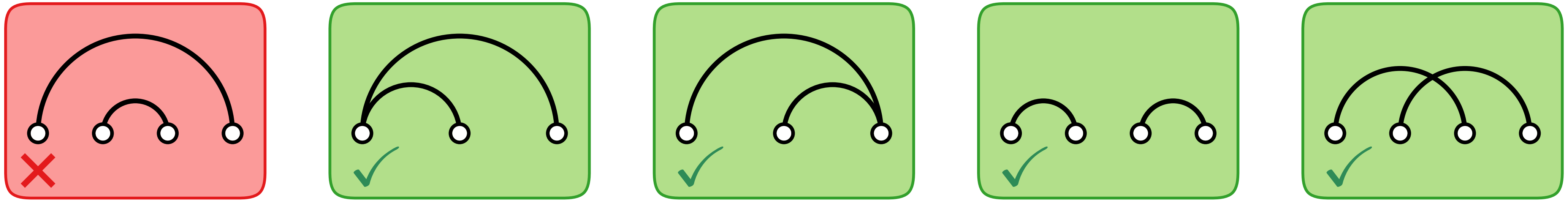
Queue Layouts

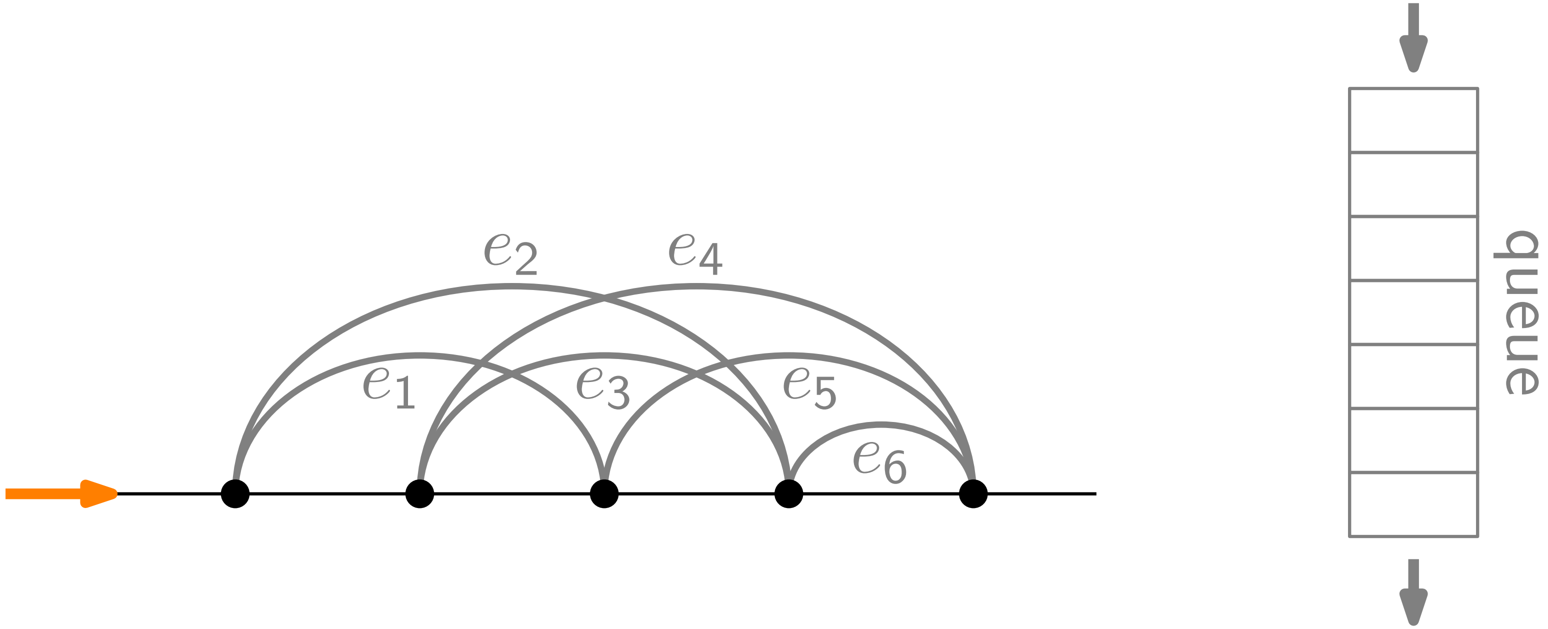
Have people studied linear layouts using other data structures than stacks? Yes, queues!

Stack Layouts:



Queue Layouts:

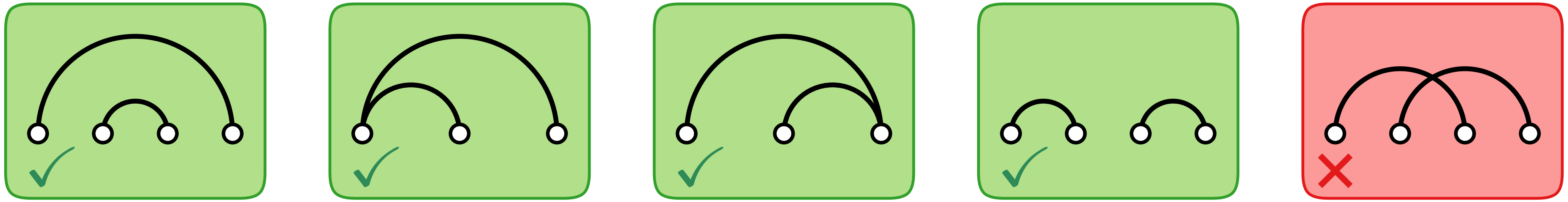




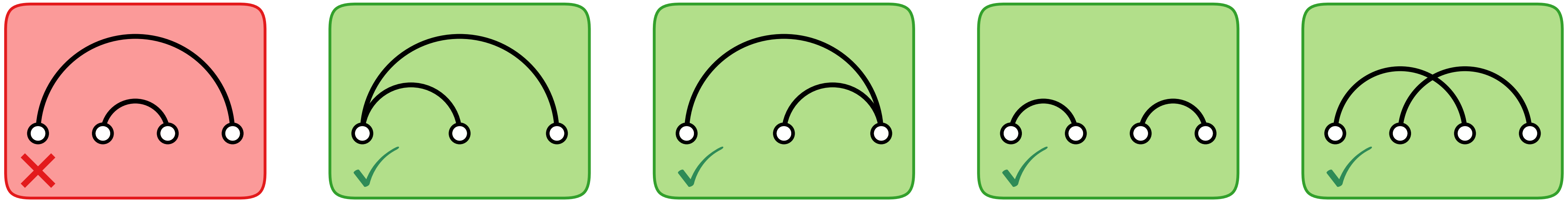
Queue Layouts

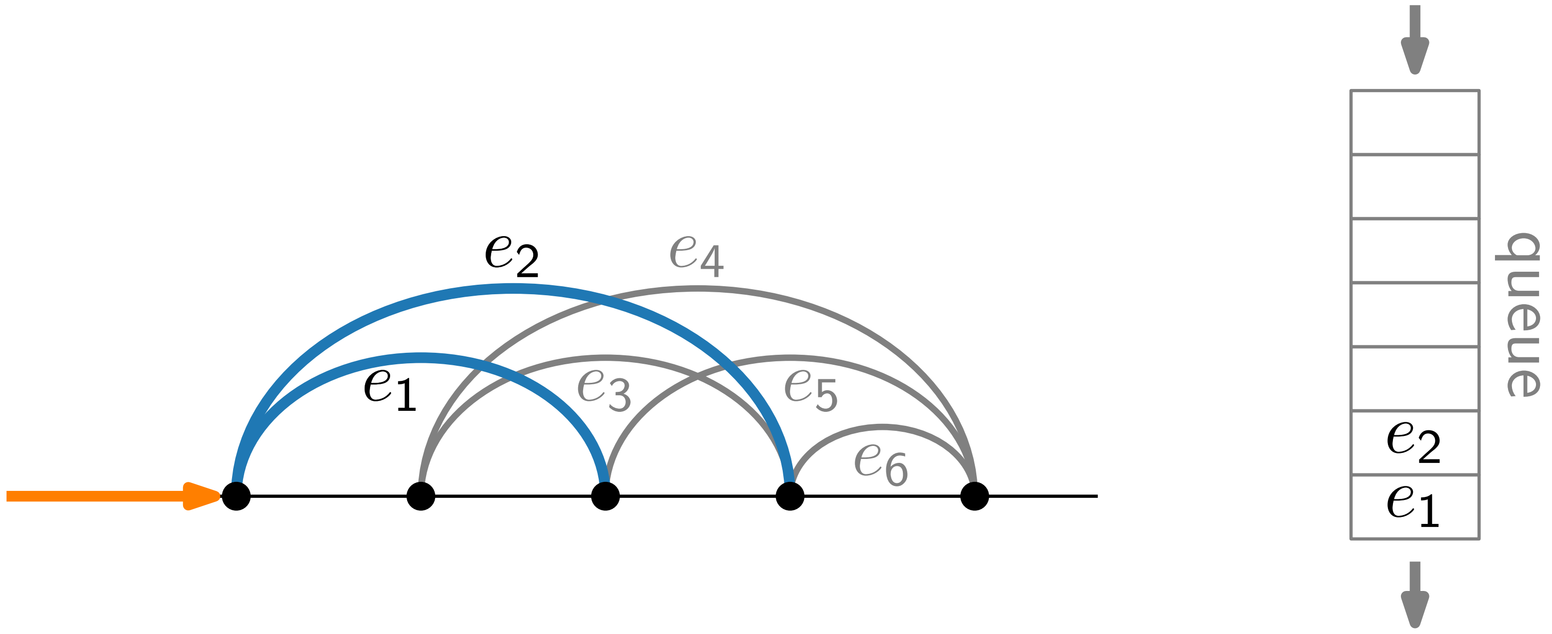
Have people studied linear layouts using other data structures than stacks? Yes, queues!

Stack Layouts:



Queue Layouts:

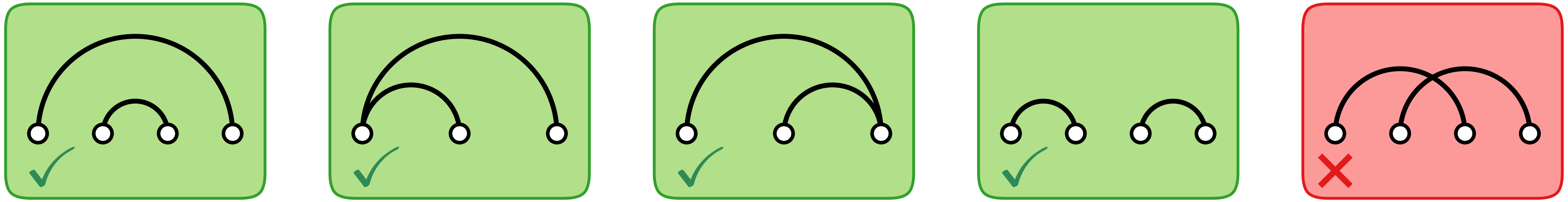




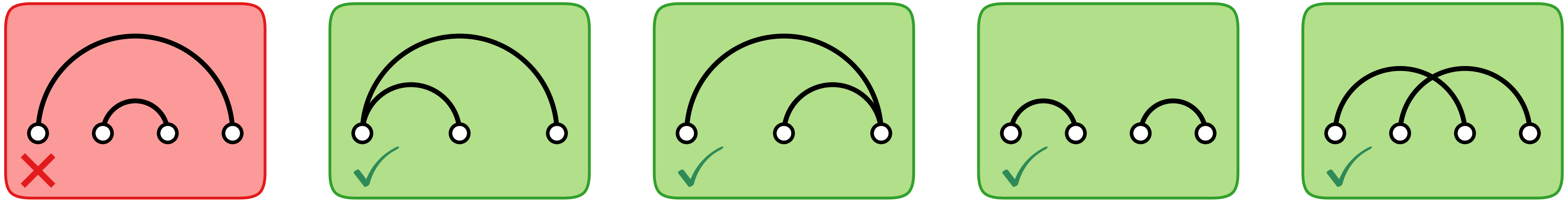
Queue Layouts

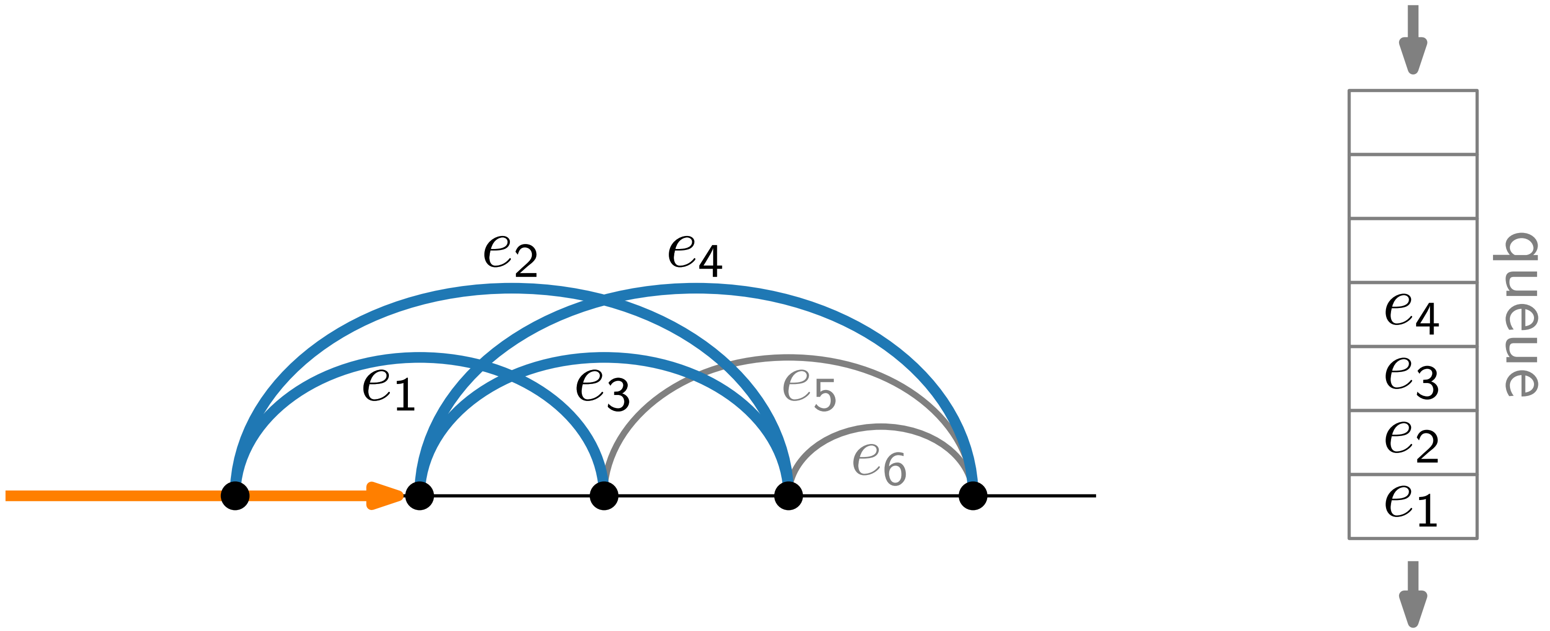
Have people studied linear layouts using other data structures than stacks? Yes, queues!

Stack Layouts:



Queue Layouts:

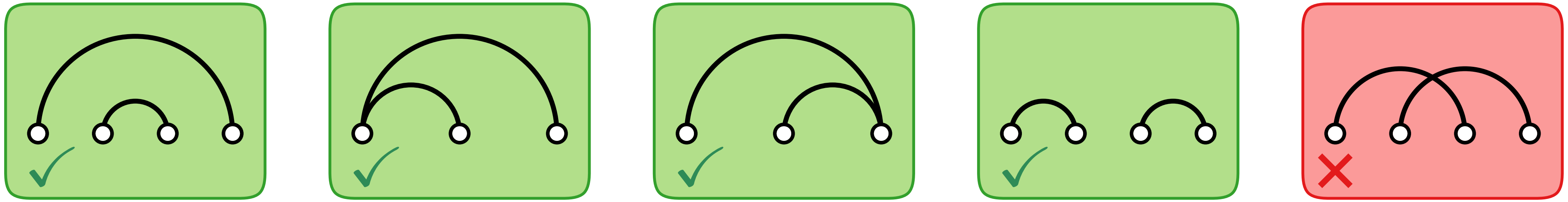




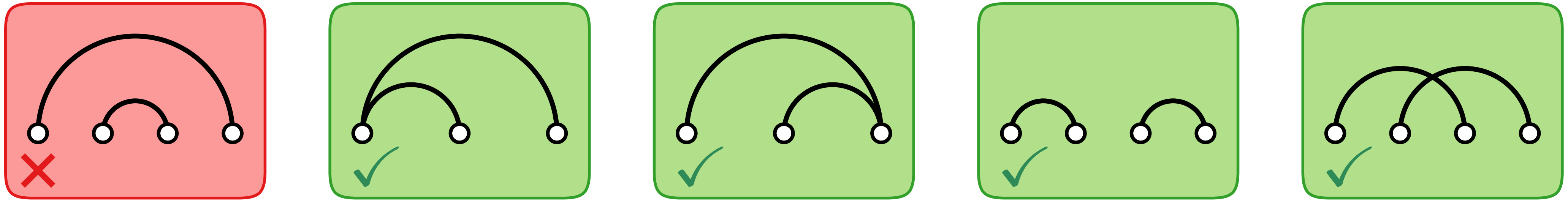
Queue Layouts

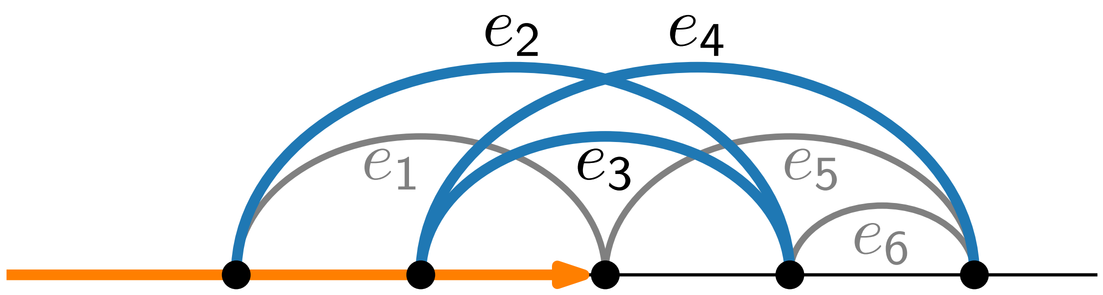
Have people studied linear layouts using other data structures than stacks? Yes, queues!

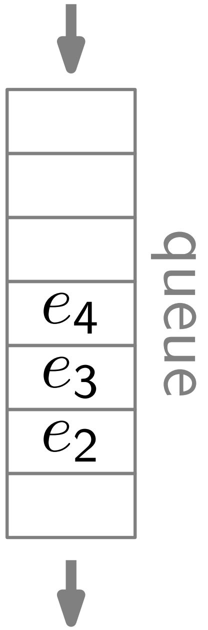
Stack Layouts:



Queue Layouts:



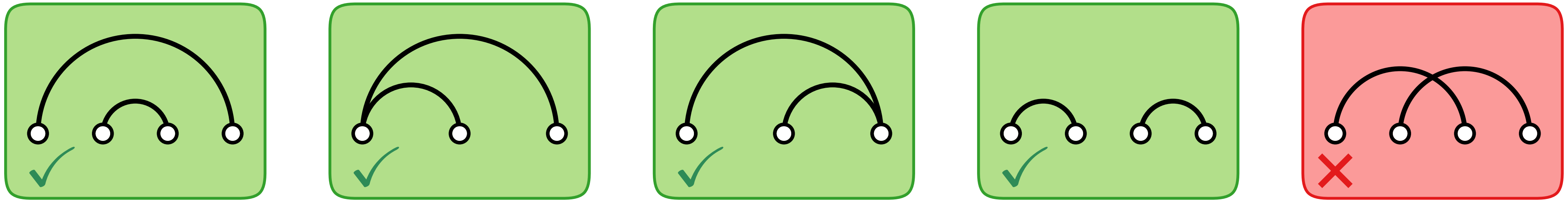




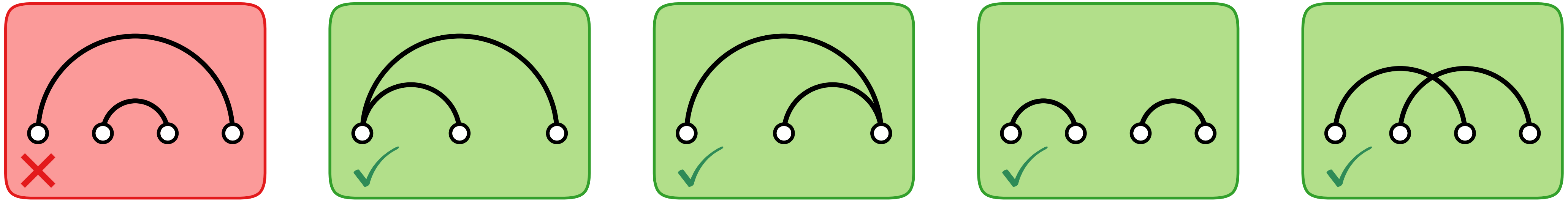
Queue Layouts

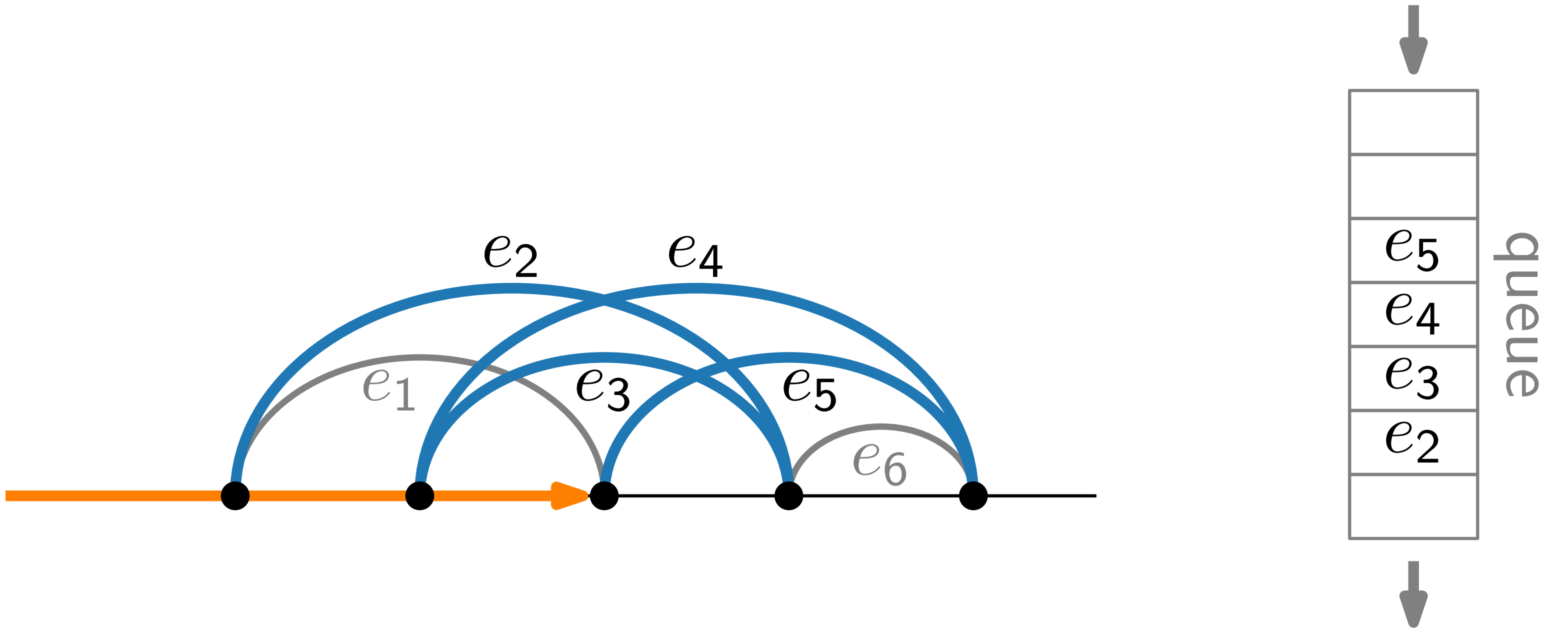
Have people studied linear layouts using other data structures than stacks? Yes, queues!

Stack Layouts:



Queue Layouts:



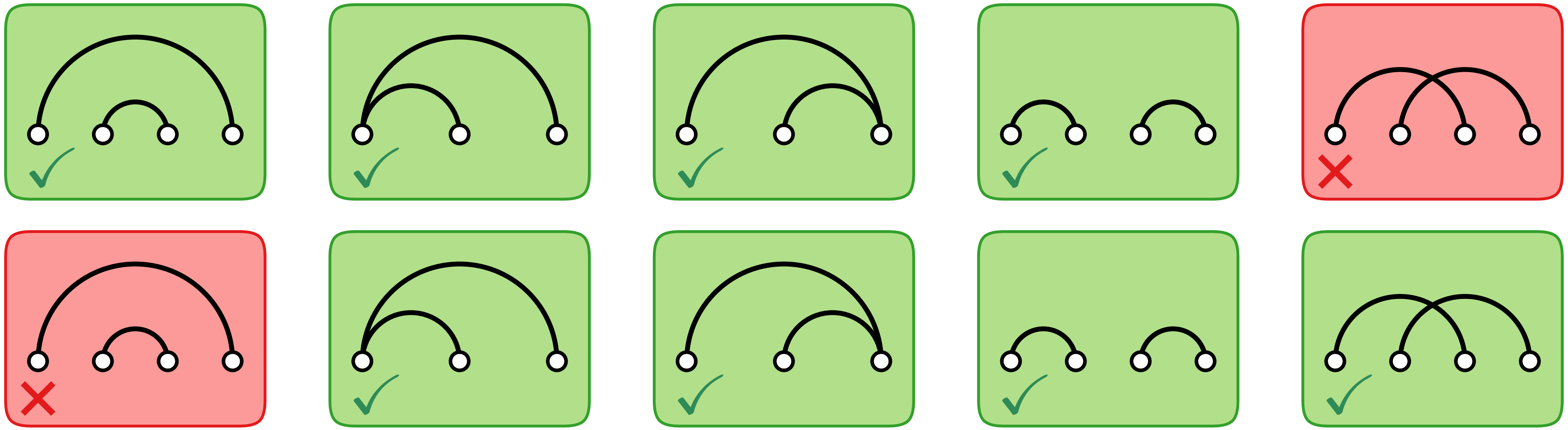


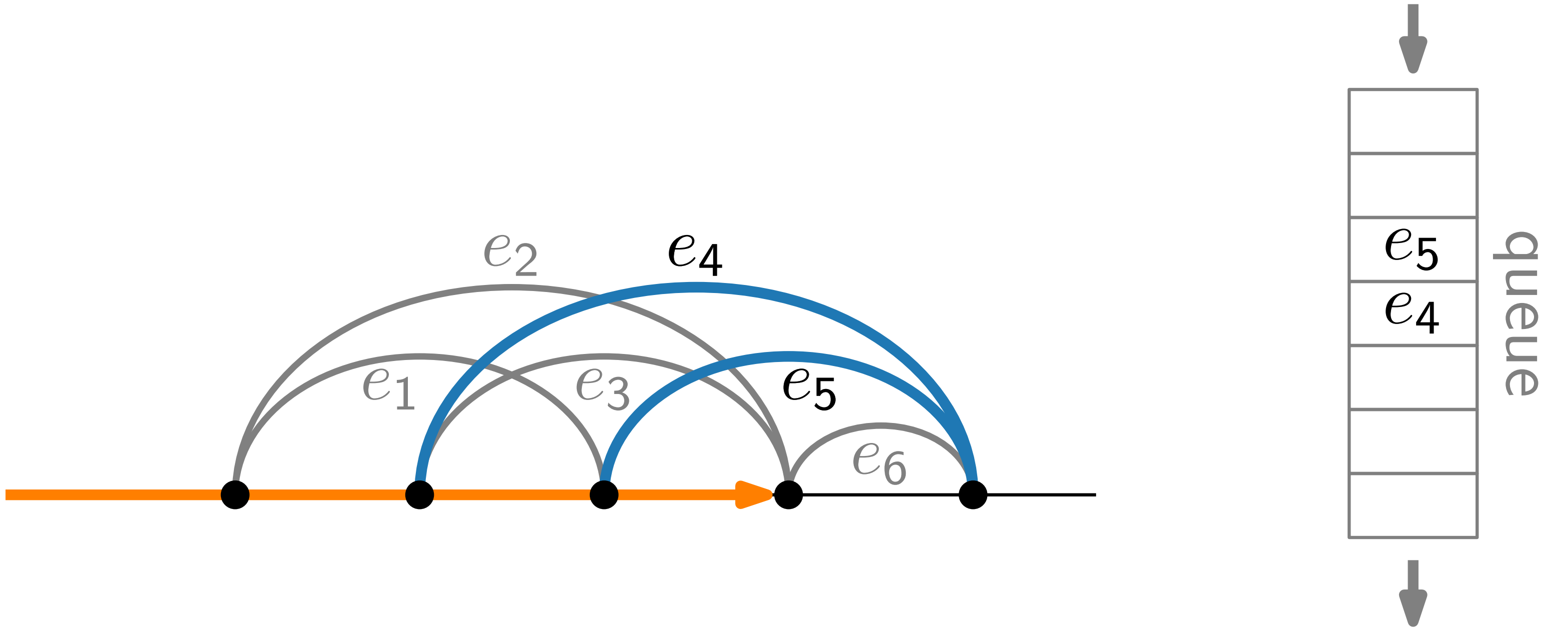
Queue Layouts

Have people studied linear layouts using other data structures than stacks? Yes, queues!

Stack Layouts:

Queue Layouts:

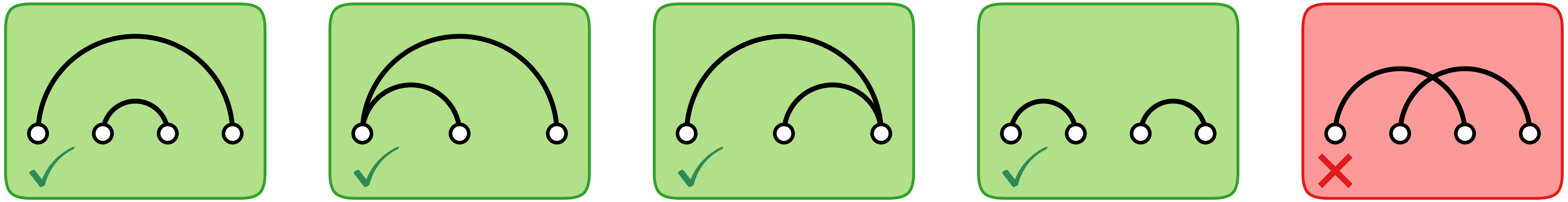




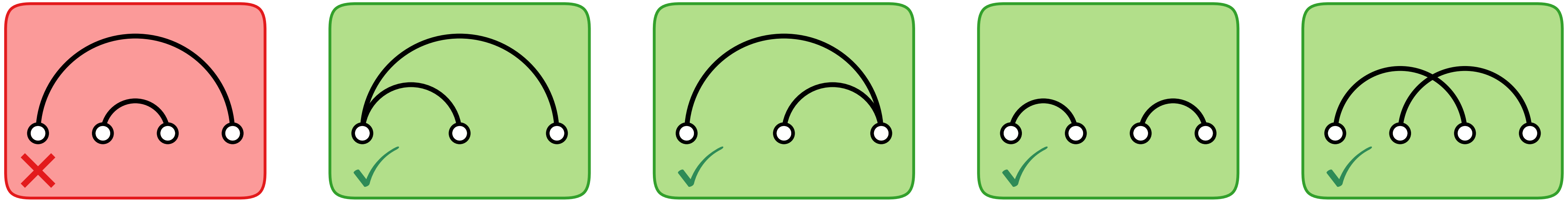
Queue Layouts

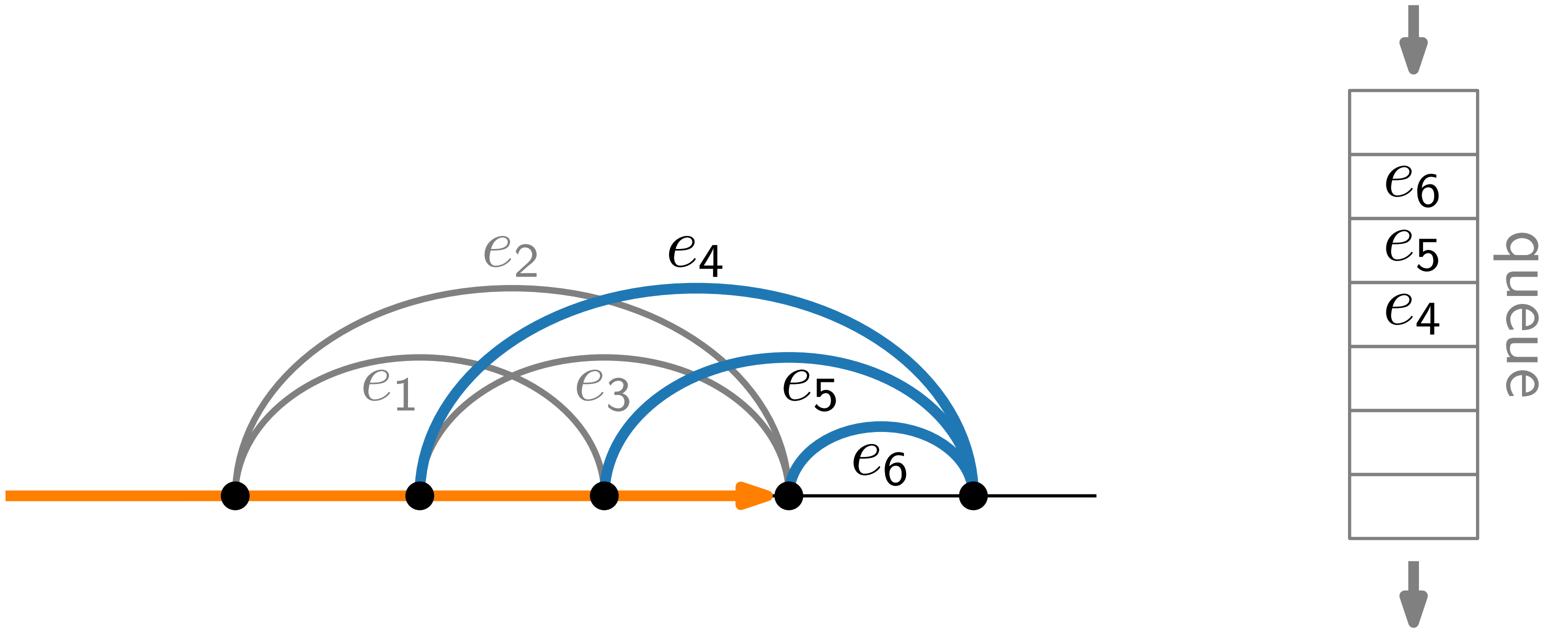
Have people studied linear layouts using other data structures than stacks? Yes, queues!

Stack Layouts:



Queue Layouts:

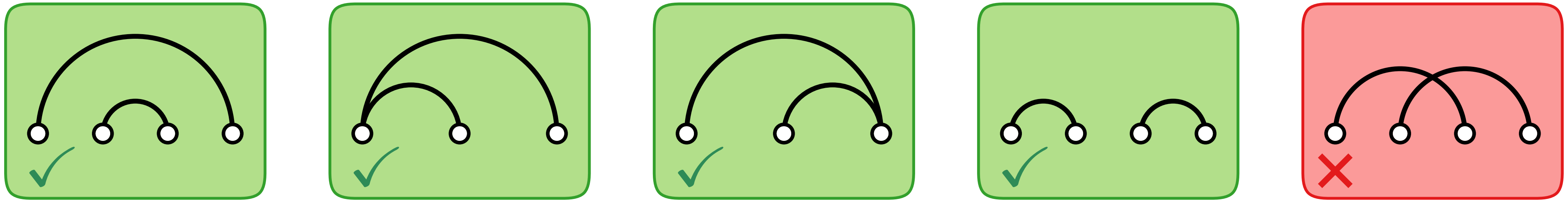




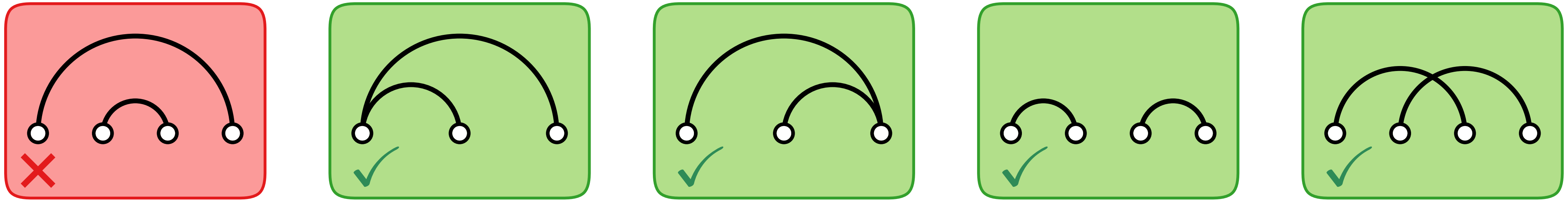
Queue Layouts

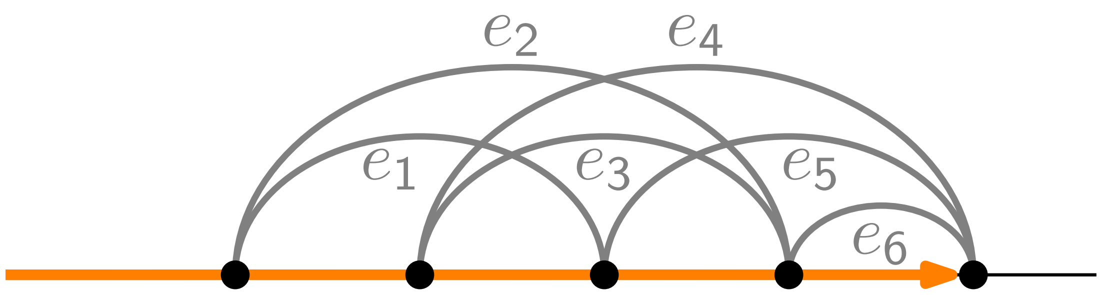
Have people studied linear layouts using other data structures than stacks? Yes, queues!

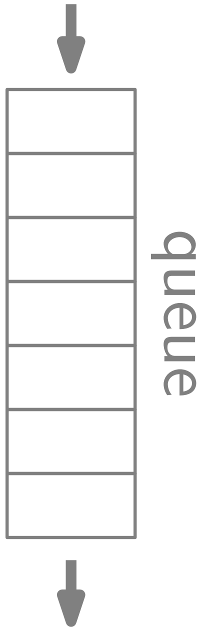
Stack Layouts:



Queue Layouts:







Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.

Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** $\text{sn}(G) = k$ (**queue number** $\text{qn}(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** $\mathrm{sn}(G) = k$ (**queue number** $\mathrm{qn}(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:

- We have seen that K_5 has a 3-page stack layout.

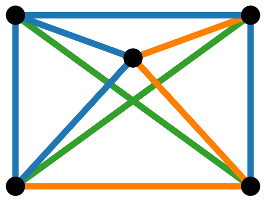
Stack Number and Queue Number

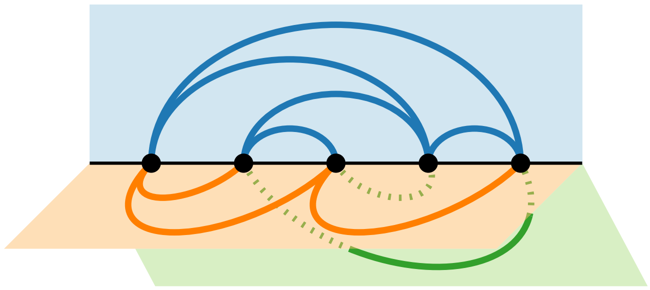
- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** $\text{sn}(G) = k$ (**queue number** $\text{qn}(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:

- We have seen that K_5 has a 3-page stack layout.





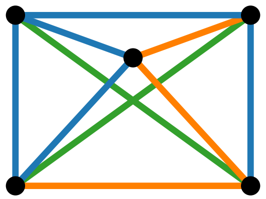
Stack Number and Queue Number

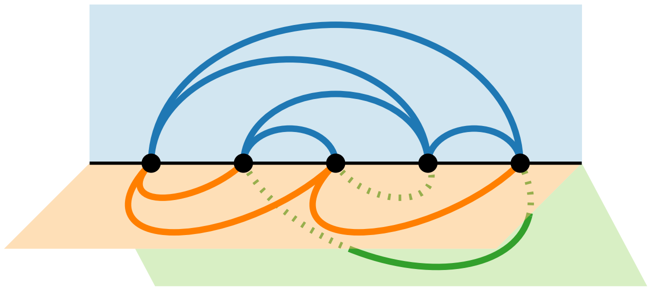
- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** $\mathrm{sn}(G) = k$ (**queue number** $\mathrm{qn}(G) = k$) if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:

- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?





Stack Number and Queue Number

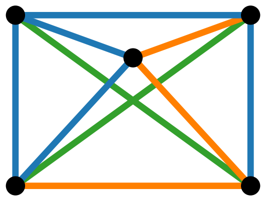
- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

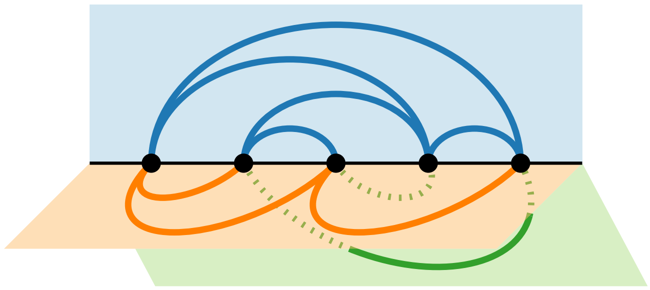
A graph G has **stack number** $\text{sn}(G) = k$ (**queue number** $\text{qn}(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:

- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?

  No, because this would be a planar drawing of K_5.





Stack Number and Queue Number

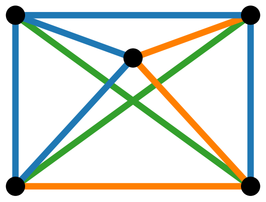
- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

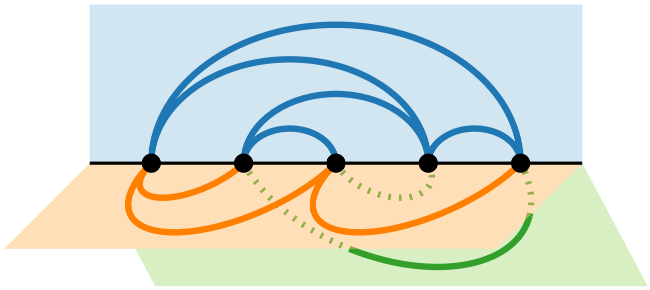
A graph G has **stack number** $\mathrm{sn}(G) = k$ (**queue number** $\mathrm{qn}(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:

- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?

No, because this would be a planar drawing of K_5. $\Rightarrow \mathrm{sn}(K_5) = 3$





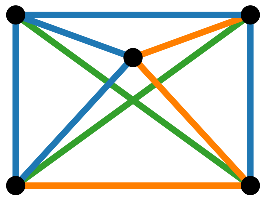
Stack Number and Queue Number

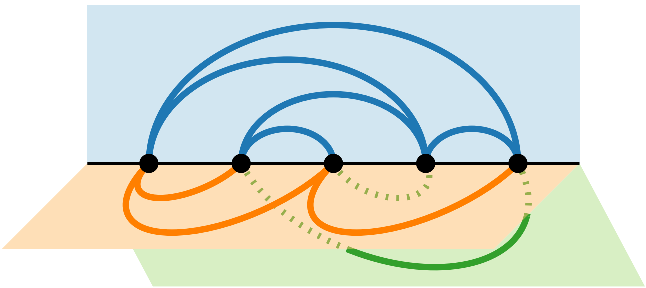
- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** $\mathrm{sn}(G) = k$ (**queue number** $\mathrm{qn}(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:

- We have seen that K_5 has a 3-page stack layout.





- Does K_5 have a 2-page stack layout?

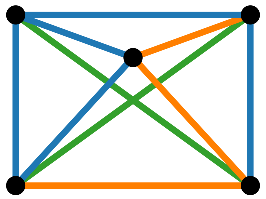
  No, because this would be a planar drawing of K_5. $\Rightarrow \mathrm{sn}(K_5) = 3$
- Does K_5 have a 1-page queue layout?

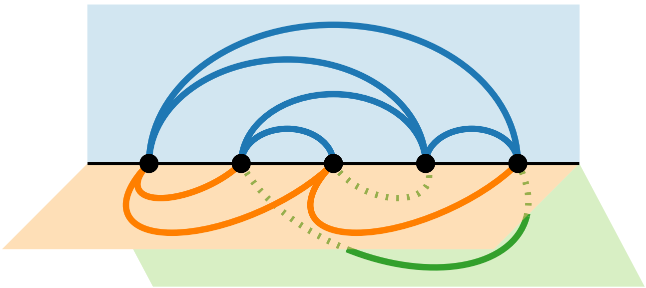
Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** sn$(G) = k$ (**queue number** qn$(G) = k$) if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:





- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?

  No, because this would be a planar drawing of K_5. $\Rightarrow$ sn$(K_5) = 3$
- Does K_5 have a 1-page queue layout?

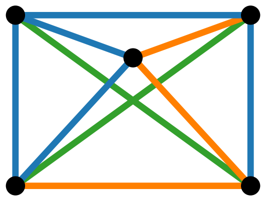
  No, because if we have all edges on one page, there are nestings.

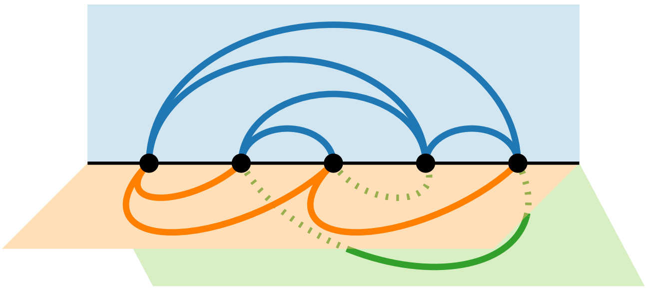
Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** $\text{sn}(G) = k$ (**queue number** $\text{qn}(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:





- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?

  No, because this would be a planar drawing of K_5. $\Rightarrow \text{sn}(K_5) = 3$
- Does K_5 have a 1-page queue layout?

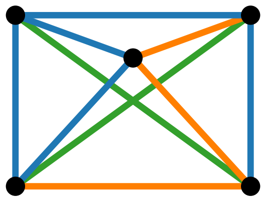
  No, because if we have all edges on one page, there are nestings.
- Does K_5 have a 2-page queue layout?

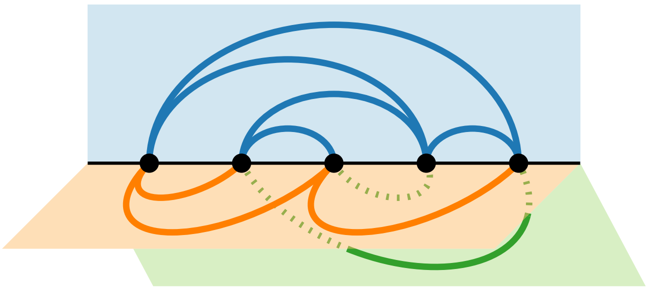
Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** sn$(G) = k$ (**queue number** qn$(G) = k$) if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:





- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?

  No, because this would be a planar drawing of K_5. $\Rightarrow$ sn$(K_5) = 3$
- Does K_5 have a 1-page queue layout?

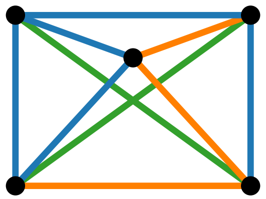
  No, because if we have all edges on one page, there are nestings.
- Does K_5 have a 2-page queue layout? Yes!

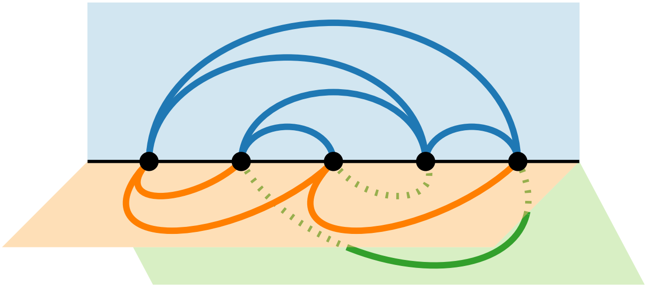
Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** $\text{sn}(G) = k$ (**queue number** $\text{qn}(G) = k$) if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

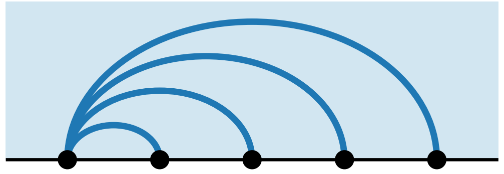
Example:





- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?

  No, because this would be a planar drawing of K_5. $\Rightarrow \text{sn}(K_5) = 3$



- Does K_5 have a 1-page queue layout?

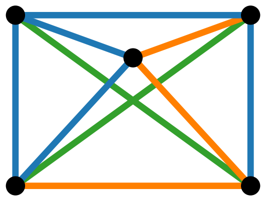
  No, because if we have all edges on one page, there are nestings.
- Does K_5 have a 2-page queue layout? Yes!

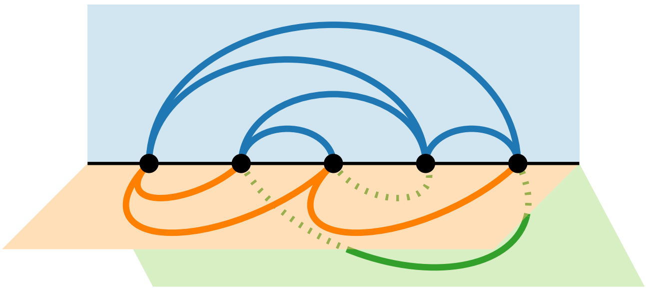
Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** sn$(G) = k$ (**queue number** qn$(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

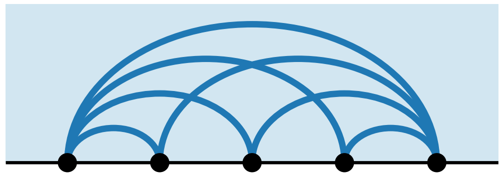
Example:





- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?

  No, because this would be a planar drawing of K_5. $\Rightarrow$ sn$(K_5) = 3$



- Does K_5 have a 1-page queue layout?

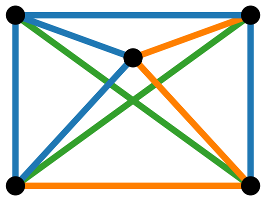
  No, because if we have all edges on one page, there are nestings.
- Does K_5 have a 2-page queue layout? Yes!

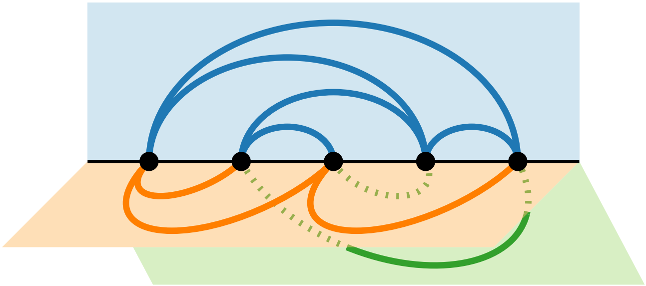
Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** sn$(G) = k$ (**queue number** qn$(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

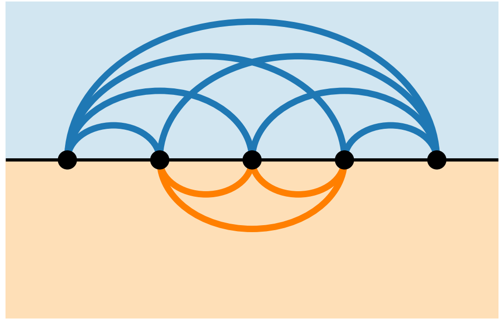
Example:





- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?

  No, because this would be a planar drawing of K_5. $\Rightarrow$ sn$(K_5) = 3$



- Does K_5 have a 1-page queue layout?

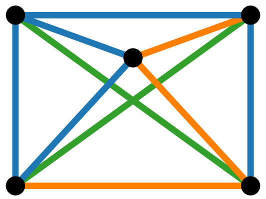
  No, because if we have all edges on one page, there are nestings.
- Does K_5 have a 2-page queue layout? Yes!

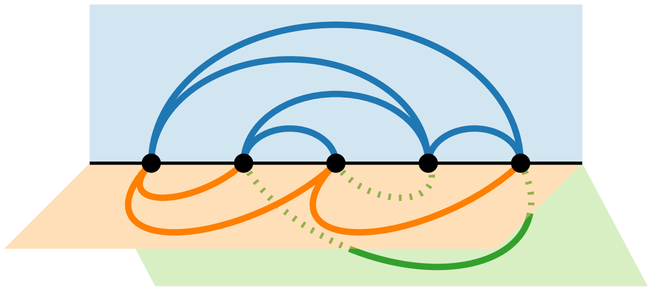
Stack Number and Queue Number

- Some graphs require more pages than other graphs to admit a stack (queue) layout.
- We seek for a measure how well a graph can be represented by a stack (queue) layout.

A graph G has **stack number** $\mathrm{sn}(G) = k$ (**queue number** $\mathrm{qn}(G) = k$)
if G admits a k-page stack (queue) layout but no $(k-1)$-page stack (queue) layout.

Example:

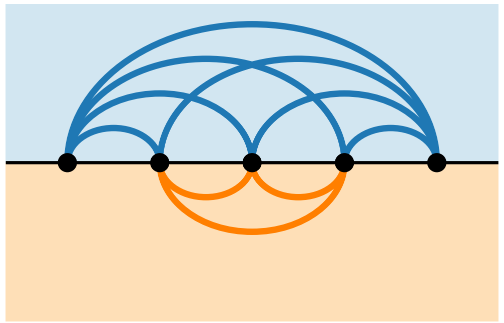




- We have seen that K_5 has a 3-page stack layout.
- Does K_5 have a 2-page stack layout?

  No, because this would be a planar drawing of K_5. $\Rightarrow \mathrm{sn}(K_5) = 3$
- Does K_5 have a 1-page queue layout?

  No, because if we have all edges on one page, there are nestings.



- Does K_5 have a 2-page queue layout? Yes! $\Rightarrow \mathrm{qn}(K_5) = 2$

1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\text{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

Proof Idea.

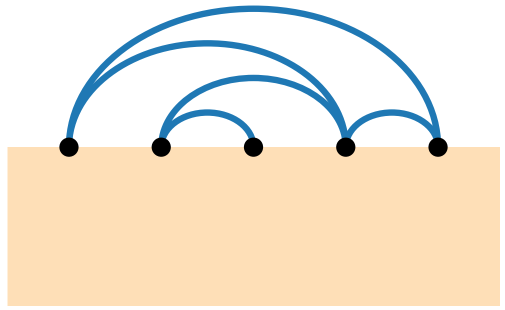
1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\text{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.



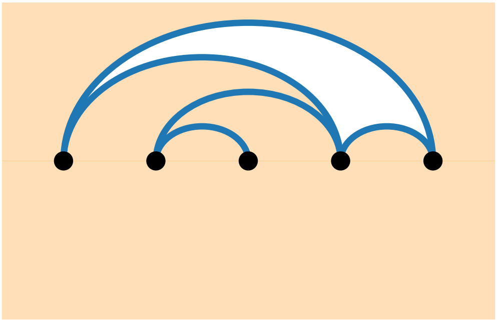
1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

Proof Idea.

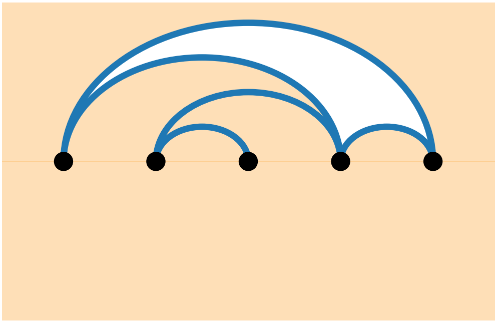
"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.



1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

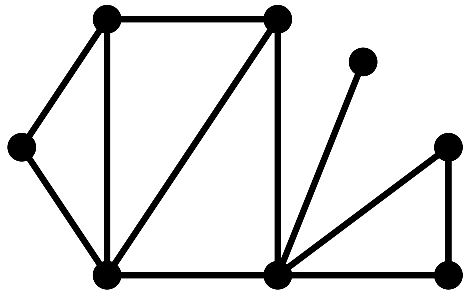
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.



Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

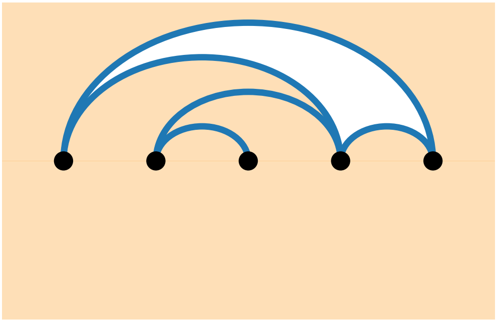
"$\Leftarrow$": Given an outerplanar drawing of G,



1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\text{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

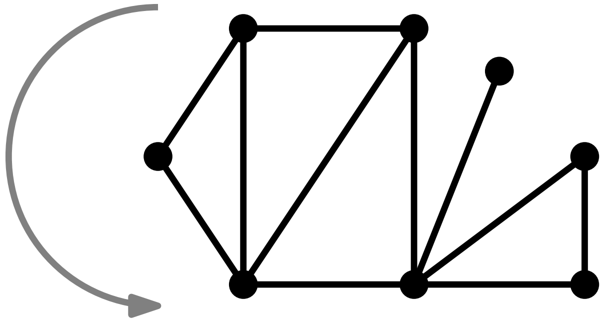
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.



Proof Idea.

“$\Rightarrow$”: Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

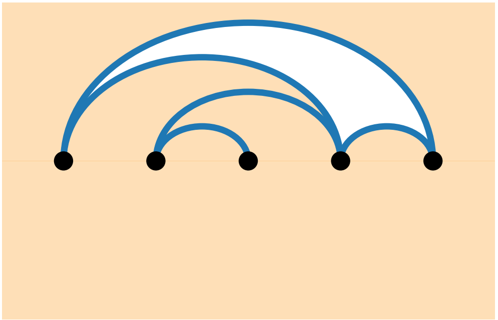
“$\Leftarrow$”: Given an outerplanar drawing of G,
traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.



1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

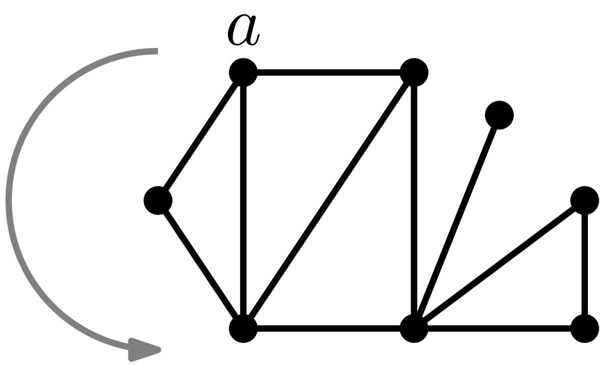
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.



Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

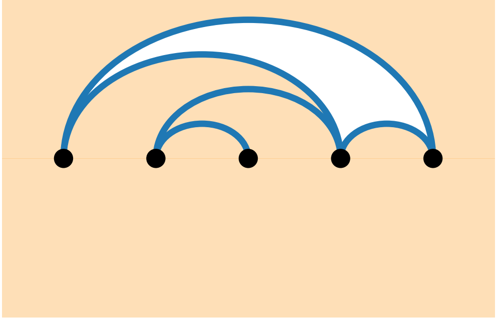
"$\Leftarrow$": Given an outerplanar drawing of G,
traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.



1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

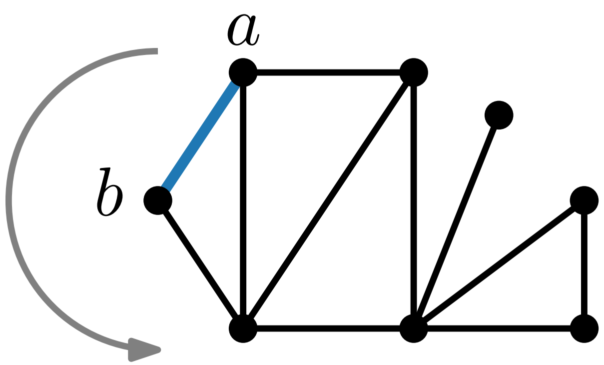
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.



Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

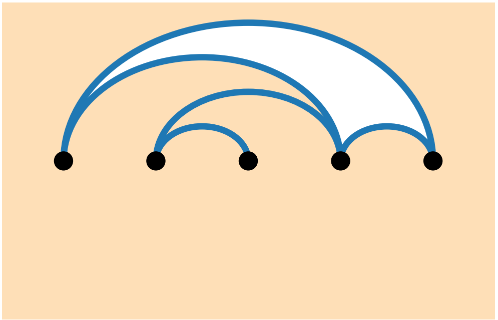
"$\Leftarrow$": Given an outerplanar drawing of G, traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.



1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: sn$(G) = 1 \Leftrightarrow G$ is outerplanar

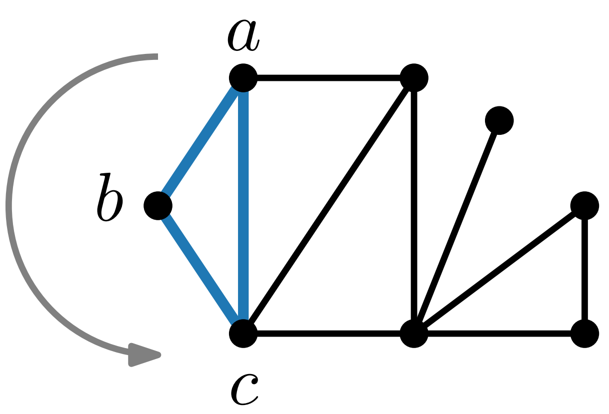
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

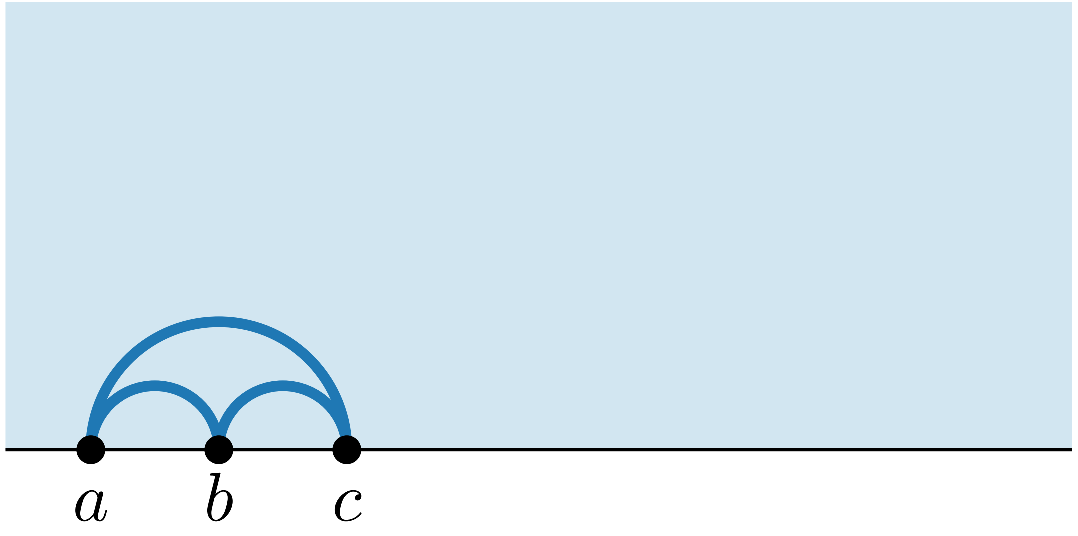


Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

"$\Leftarrow$": Given an outerplanar drawing of G,
traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.

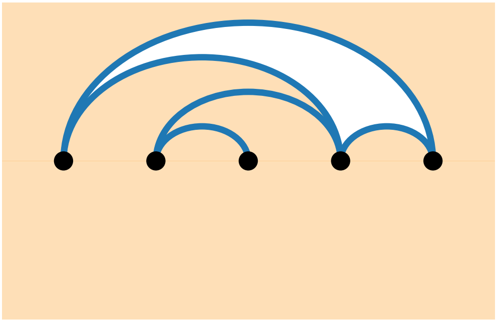




1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

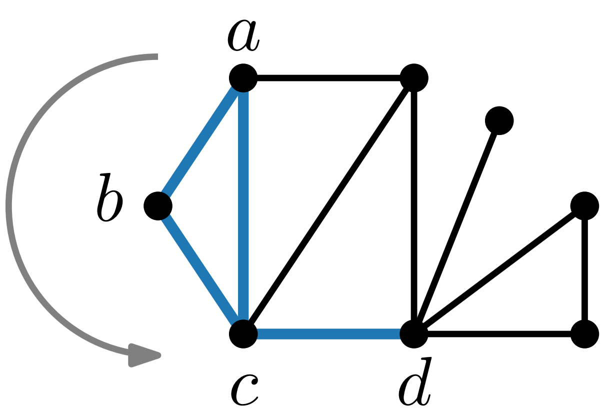
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

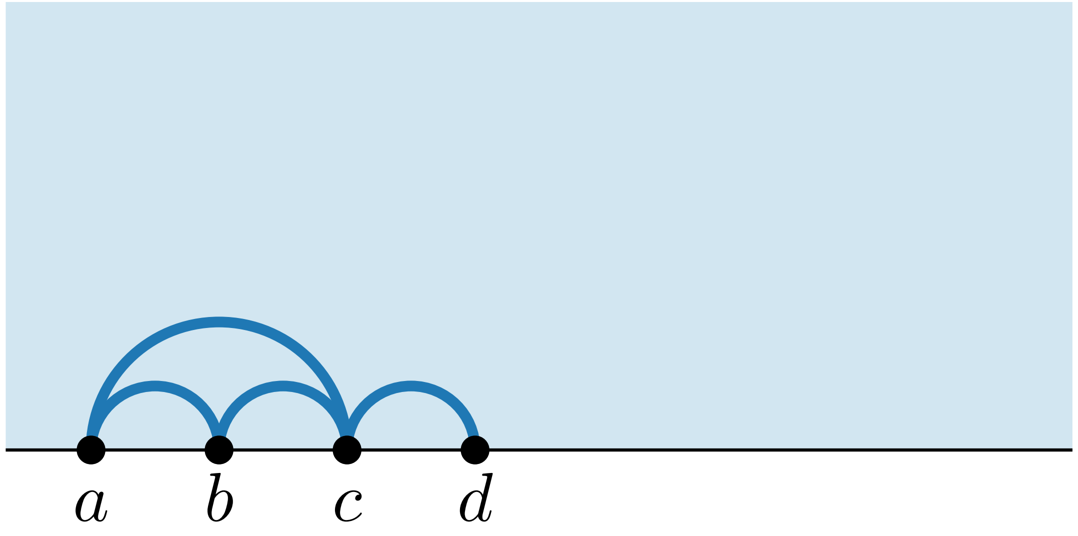


Proof Idea.

"⇒": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

"⇐": Given an outerplanar drawing of G,
traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.

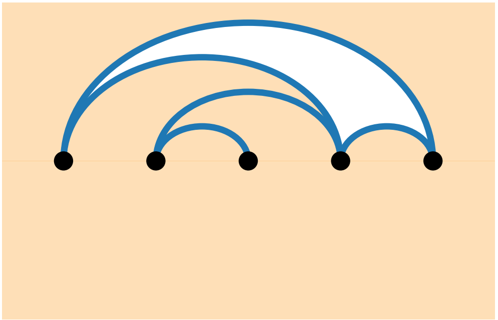




1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

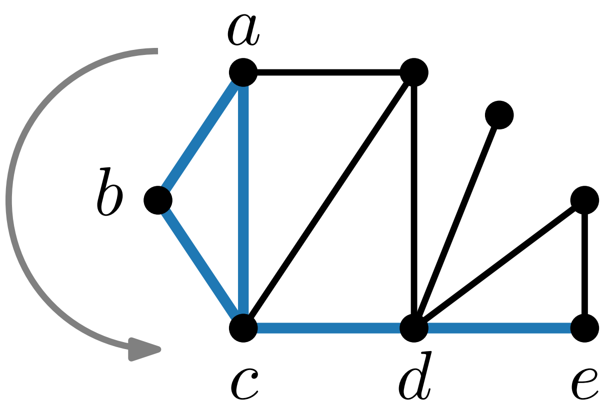
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

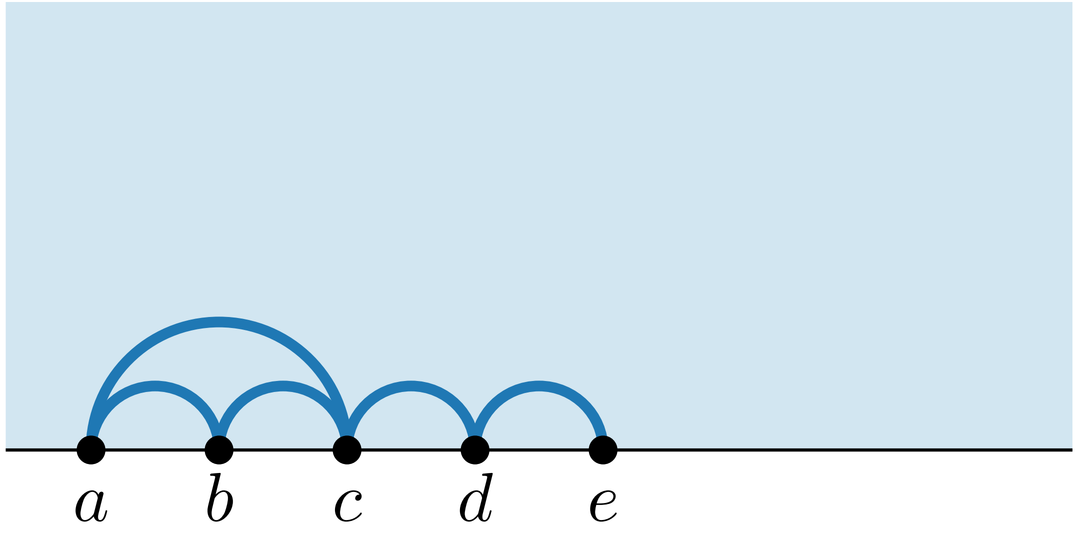


Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

"$\Leftarrow$": Given an outerplanar drawing of G,
traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.

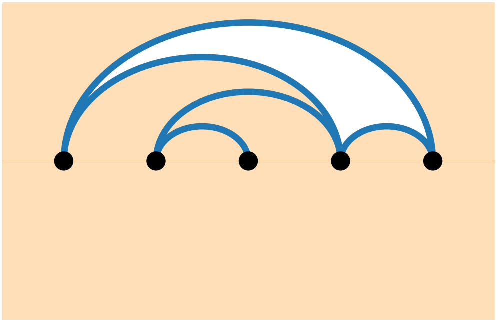




1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\text{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

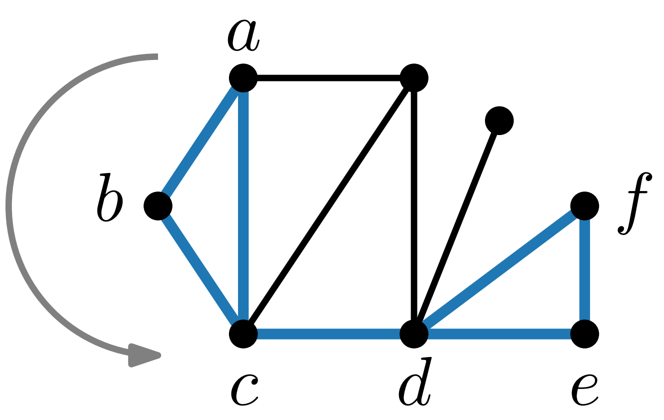
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

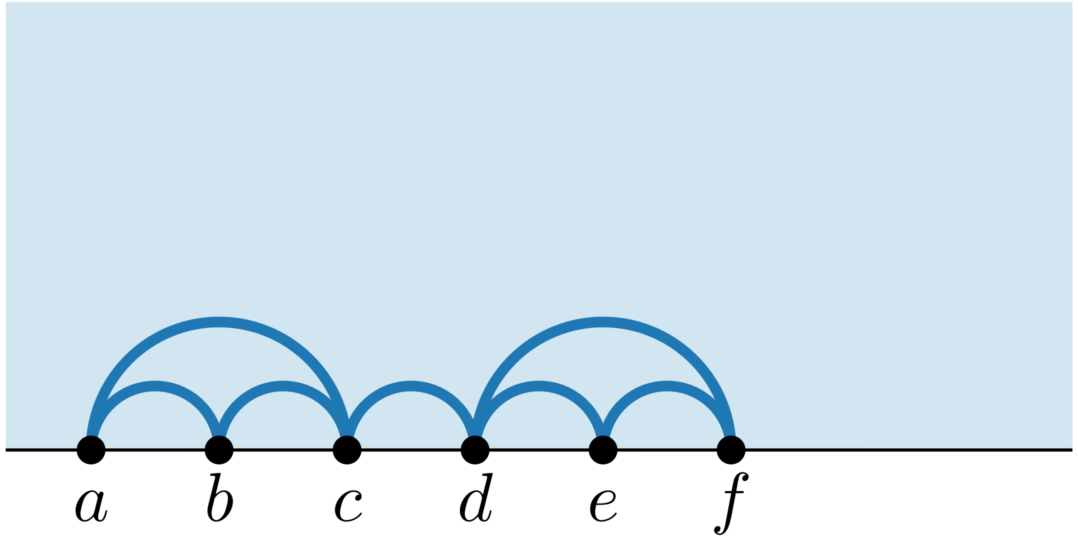


Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

"$\Leftarrow$": Given an outerplanar drawing of G,
traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.

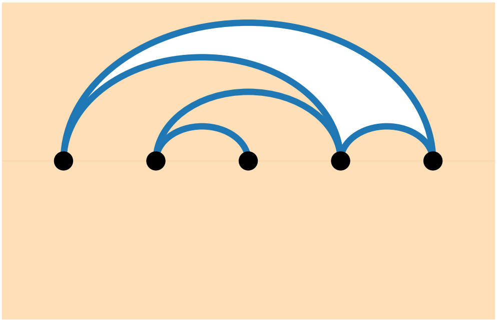




1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: sn$(G) = 1 \Leftrightarrow G$ is outerplanar

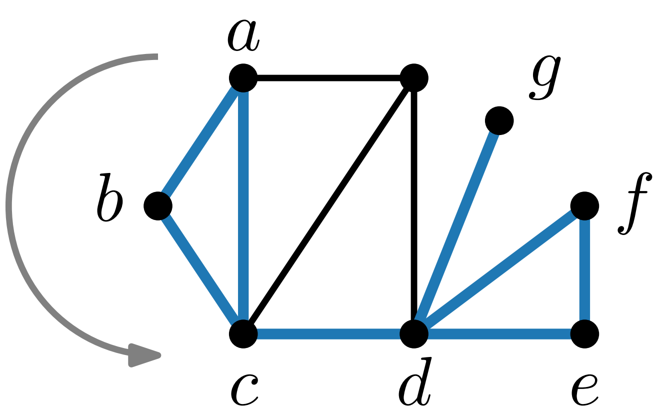
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

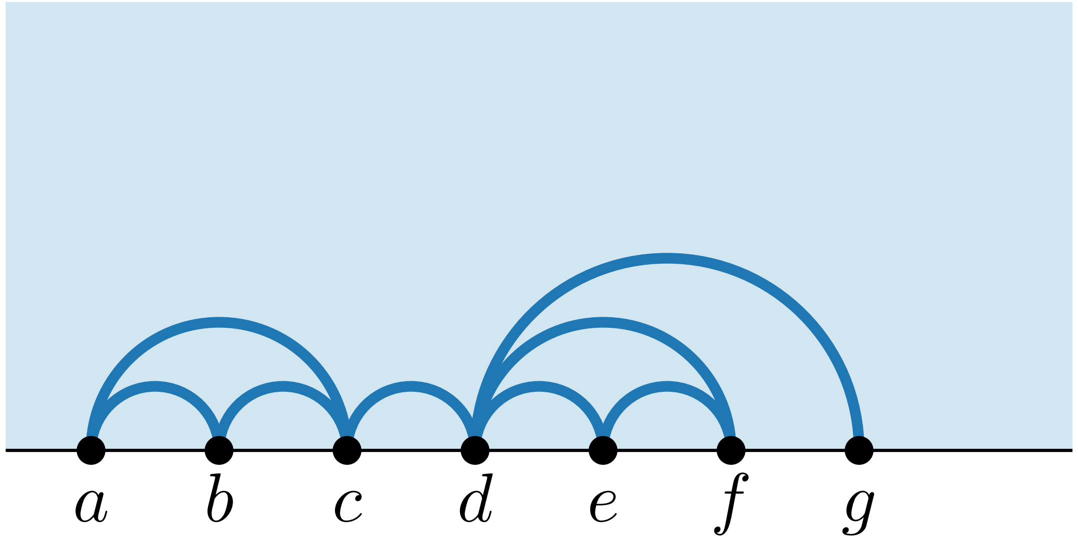


Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

"$\Leftarrow$": Given an outerplanar drawing of G, traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.

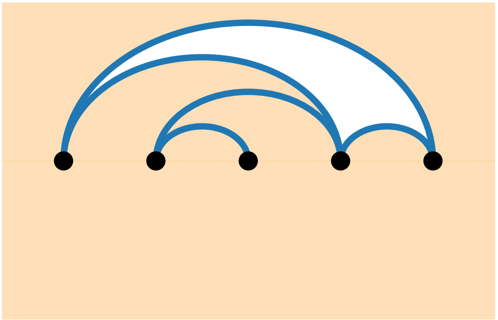




1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: sn$(G) = 1 \Leftrightarrow G$ is outerplanar

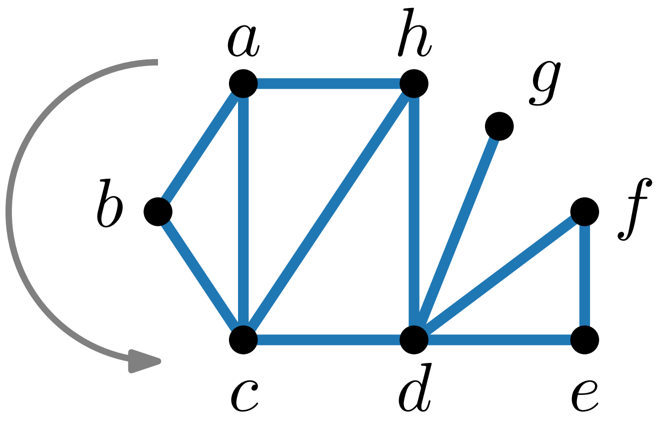
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

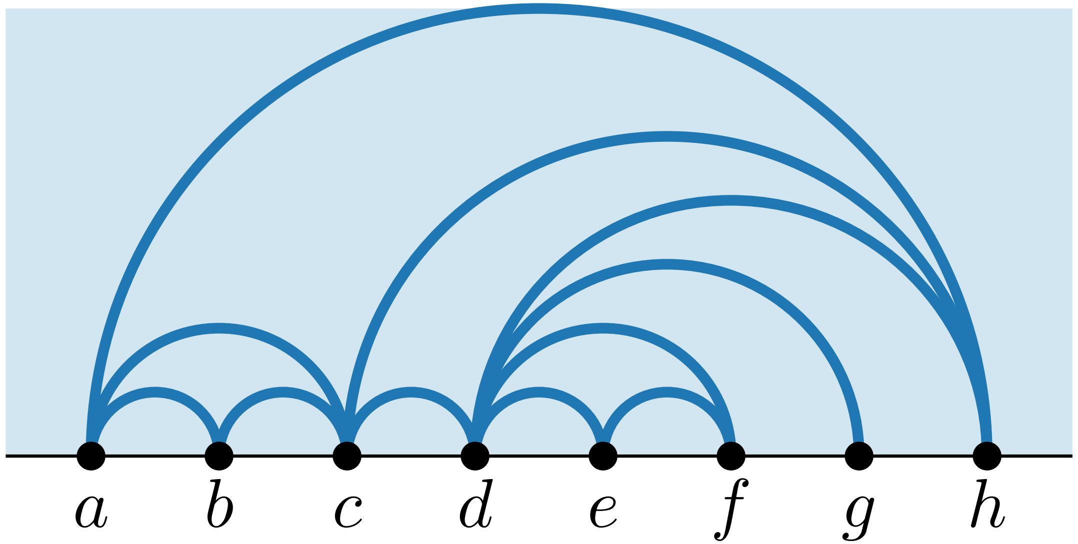


Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

"$\Leftarrow$": Given an outerplanar drawing of G, traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.

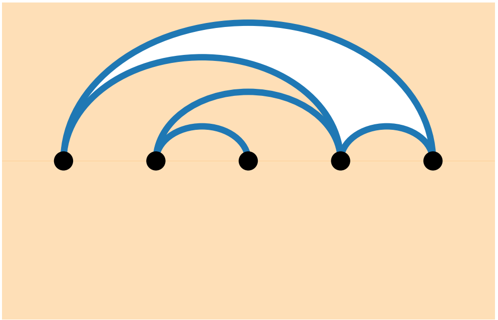




1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\text{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

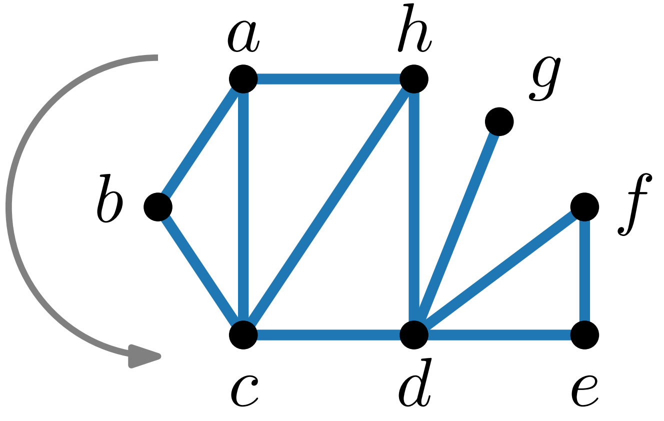
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

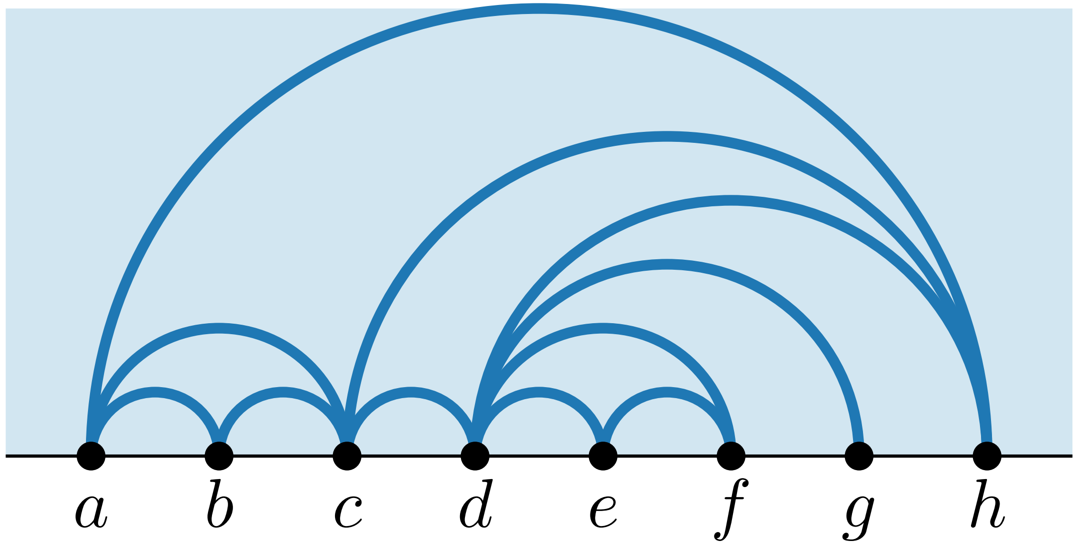


Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

"$\Leftarrow$": Given an outerplanar drawing of G, traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.



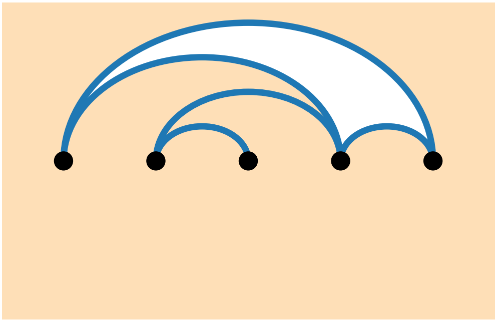


Note that the planar embedding is preserved.

1-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\text{sn}(G) = 1 \Leftrightarrow G$ is outerplanar

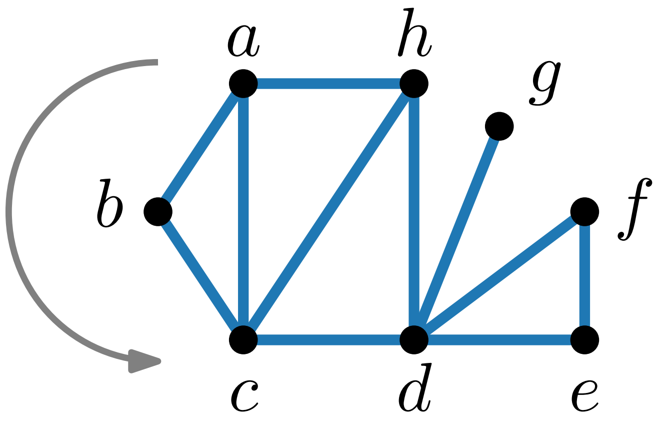
Recall: a graph is outerplanar if it has a planar drawing where every vertex is incident to the outer face.

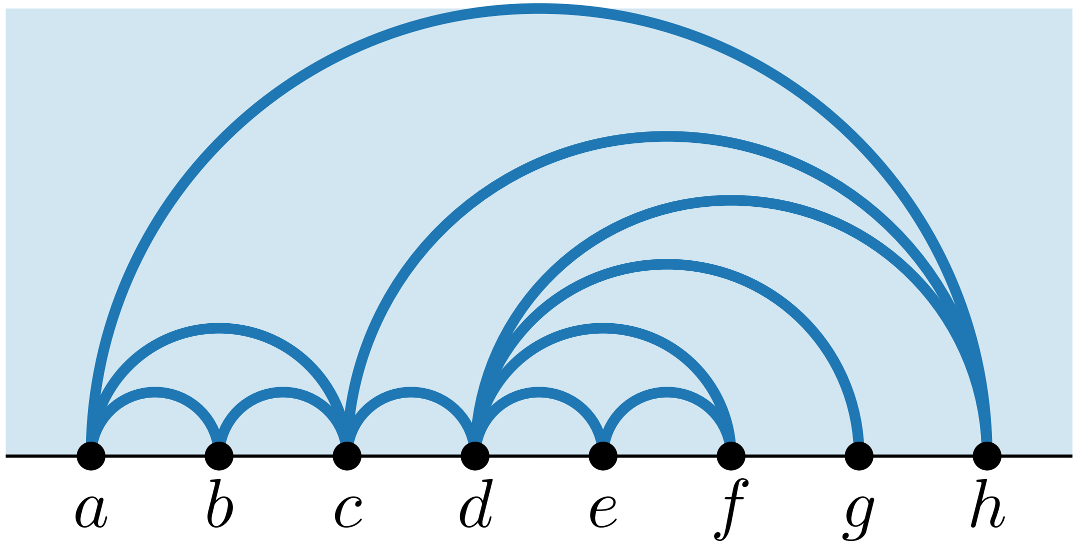


Proof Idea.

"$\Rightarrow$": Clearly, a 1-page stack layout can be perceived as a planar drawing where the vertices lie at the outer face.

"$\Leftarrow$": Given an outerplanar drawing of G, traverse the outer face in counterclockwise order and place the vertices in this order onto the spine.





Note that the planar embedding is preserved.

We can think of "morphing" the one drawing into the other.

□

2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]

For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\text{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

Proof.

i.e., a graph that has a Hamiltonian cycle

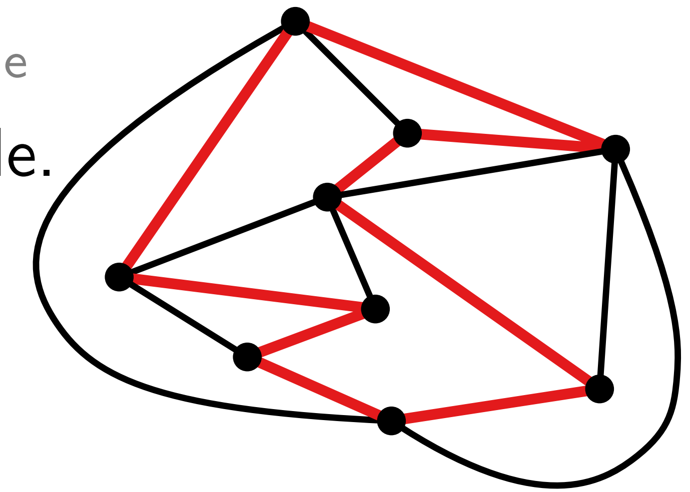
2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Leftarrow$": ■ Let Γ be a planar drawing of a graph with a Hamiltonian cycle.



2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]

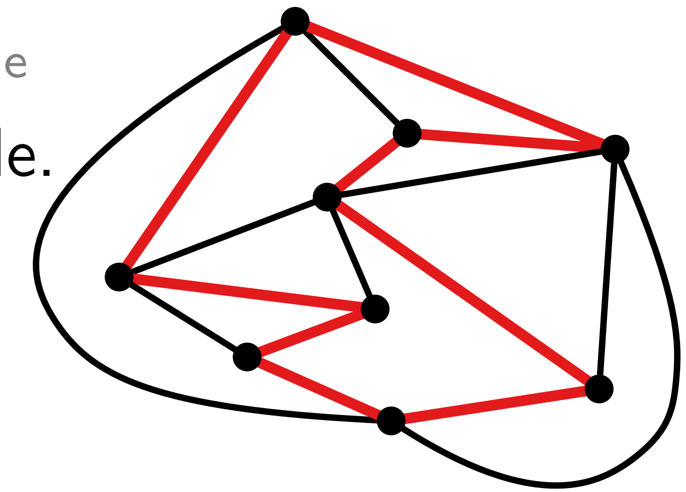
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.



2-Page Stack Layouts

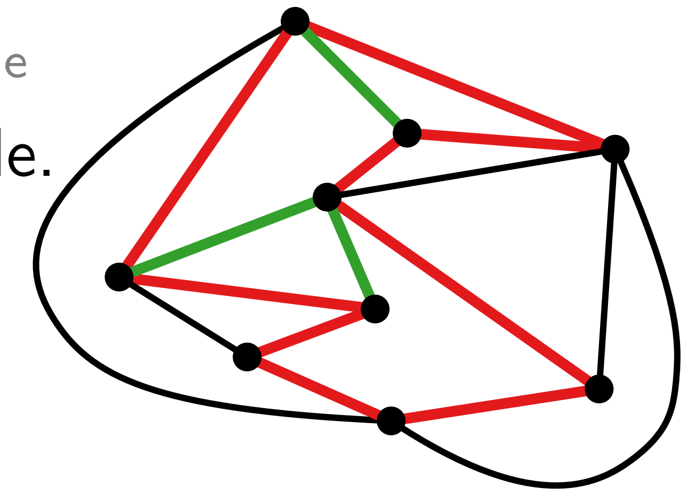
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.



2-Page Stack Layouts

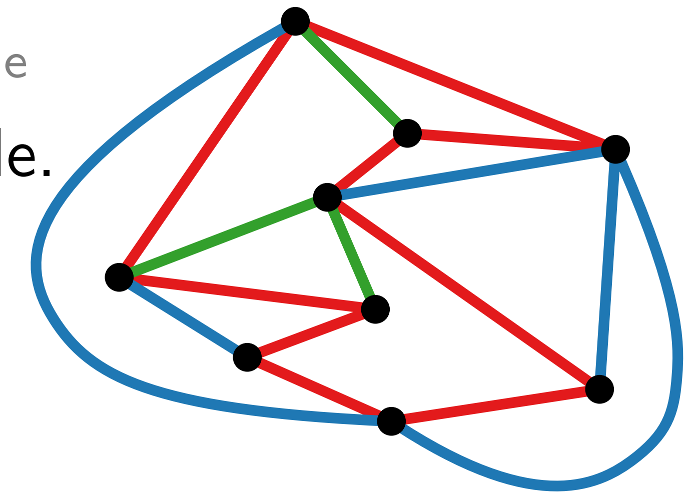
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.



2-Page Stack Layouts

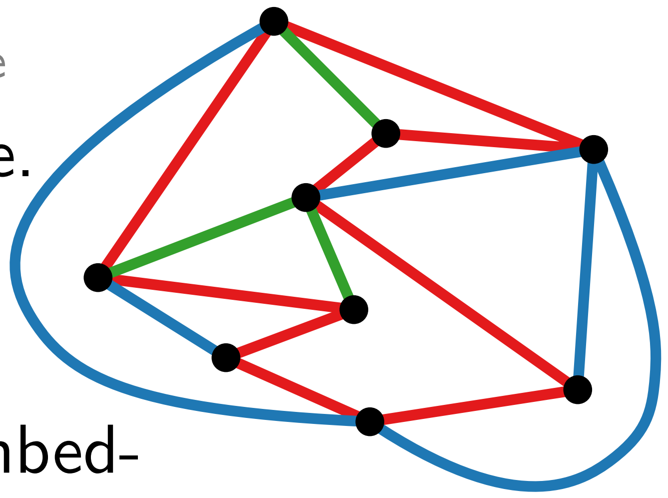
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.
- The red–green / red–blue edges induce two outerplanar embeddings with the same cyclic order of the vertices on the outer face.

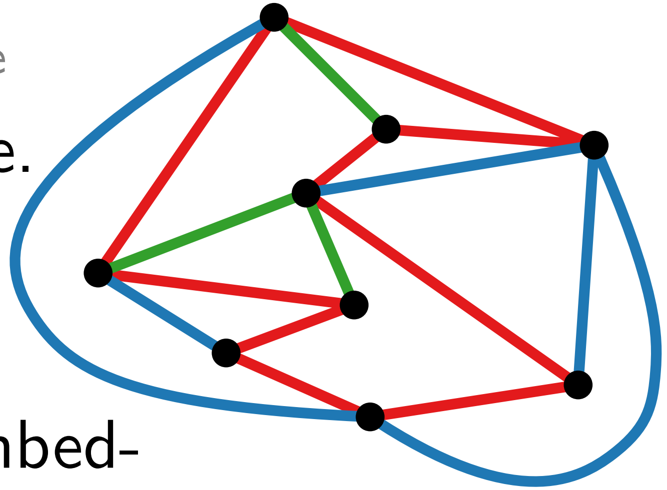


2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\text{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

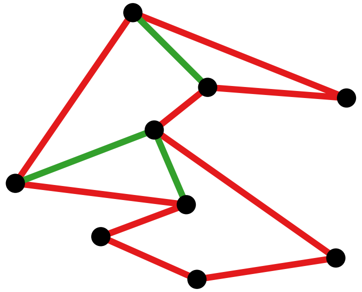
i.e., a graph that has a Hamiltonian cycle

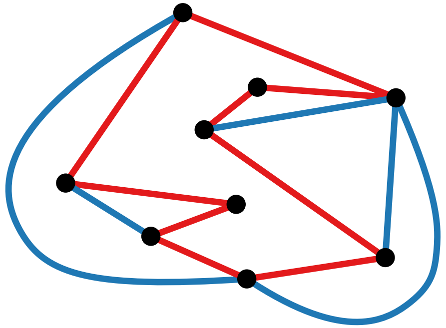
Proof.



"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.
- The red–green / red–blue edges induce two outerplanar embeddings with the same cyclic order of the vertices on the outer face.





2-Page Stack Layouts

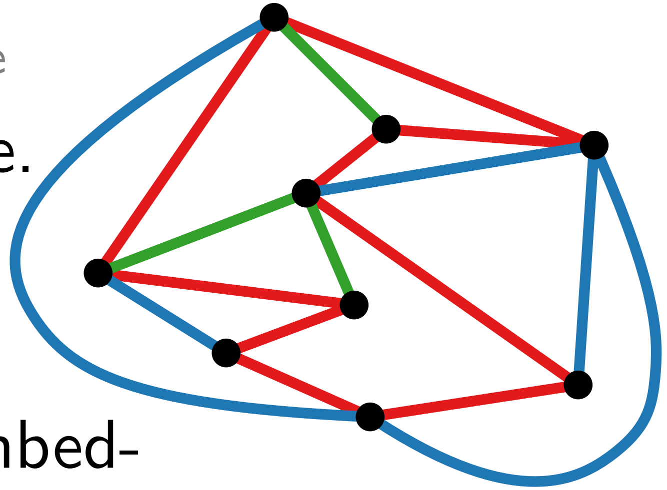
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

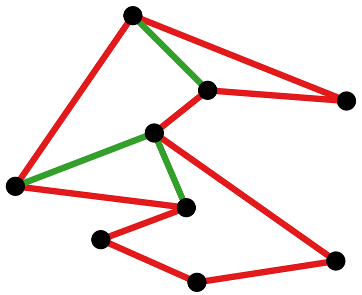
i.e., a graph that has a Hamiltonian cycle

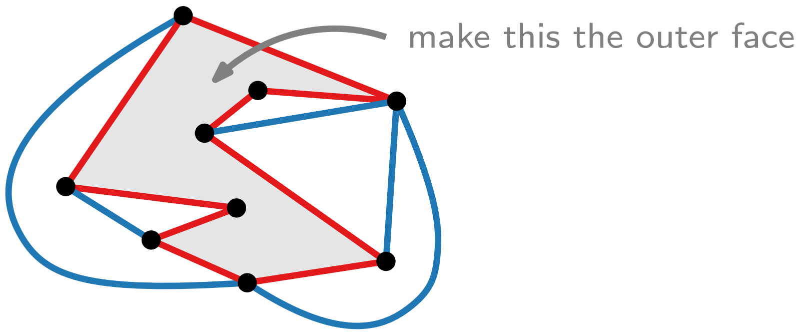
Proof.

"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.
- The red–green / red–blue edges induce two outerplanar embeddings with the same cyclic order of the vertices on the outer face.







2-Page Stack Layouts

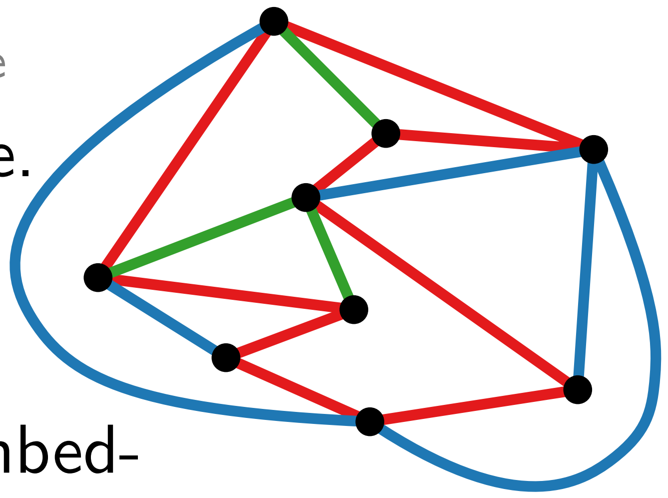
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\text{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

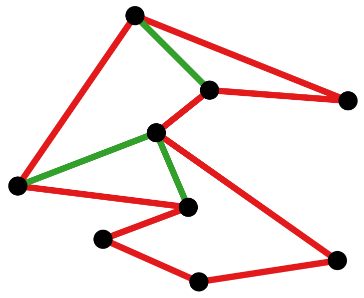
i.e., a graph that has a Hamiltonian cycle

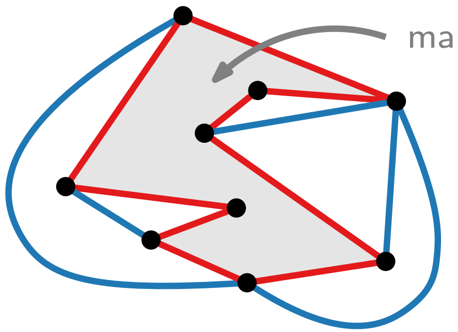
Proof.

"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.
- The red–green / red–blue edges induce two outerplanar embeddings with the same cyclic order of the vertices on the outer face.
- Put each one into a separate stack (same order of vertices on the spine).







2-Page Stack Layouts

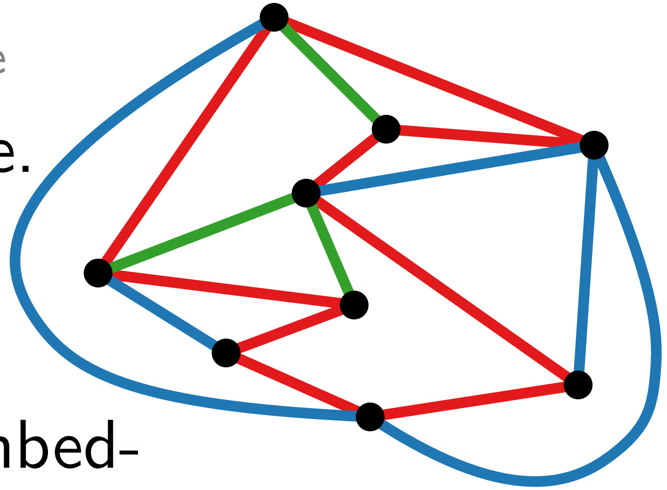
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

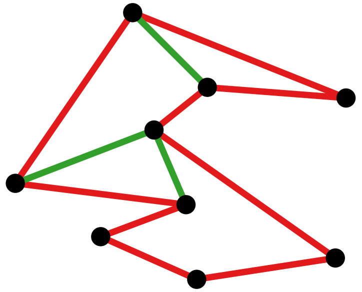
i.e., a graph that has a Hamiltonian cycle

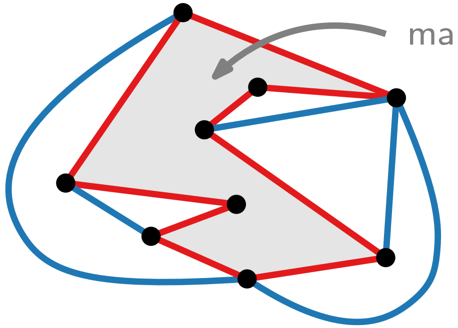
Proof.

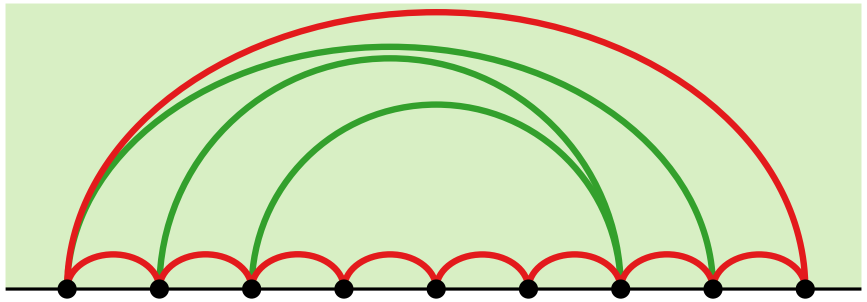
"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.
- The red–green / red–blue edges induce two outerplanar embeddings with the same cyclic order of the vertices on the outer face.
- Put each one into a separate stack (same order of vertices on the spine).









2-Page Stack Layouts

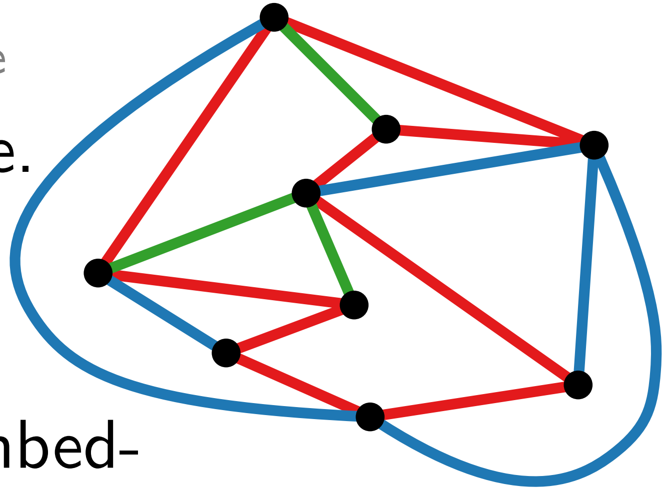
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

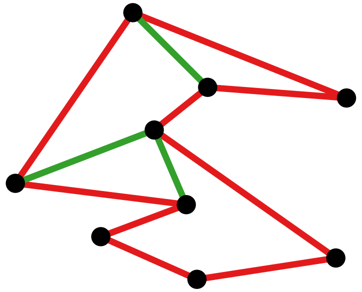
i.e., a graph that has a Hamiltonian cycle

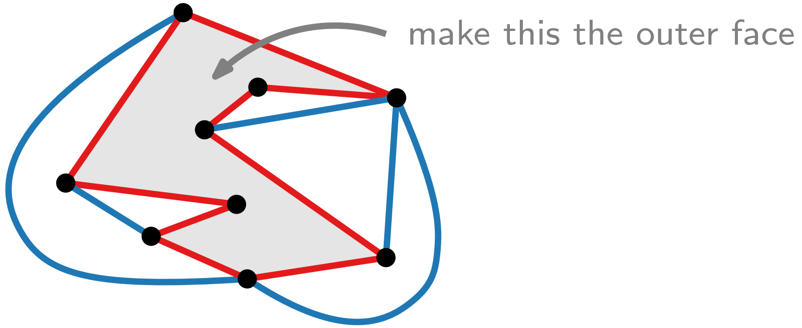
Proof.

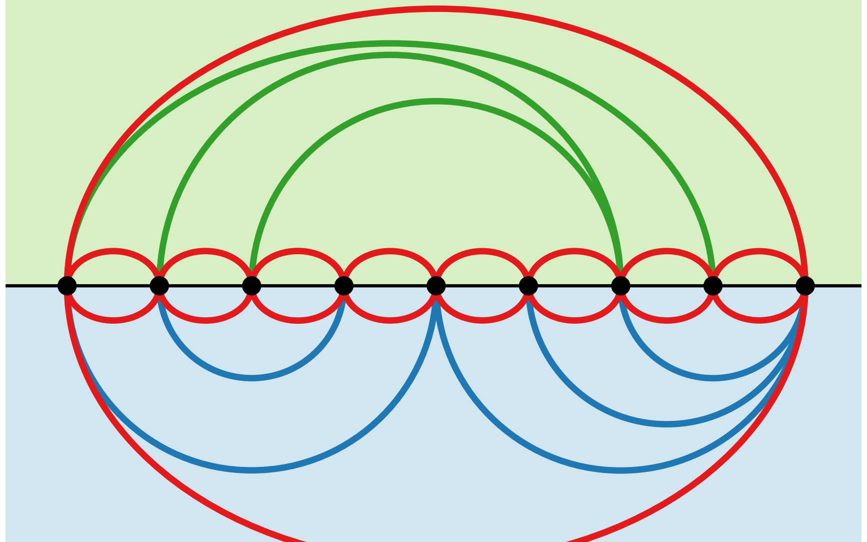
"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.
- The red–green / red–blue edges induce two outerplanar embeddings with the same cyclic order of the vertices on the outer face.
- Put each one into a separate stack (same order of vertices on the spine).





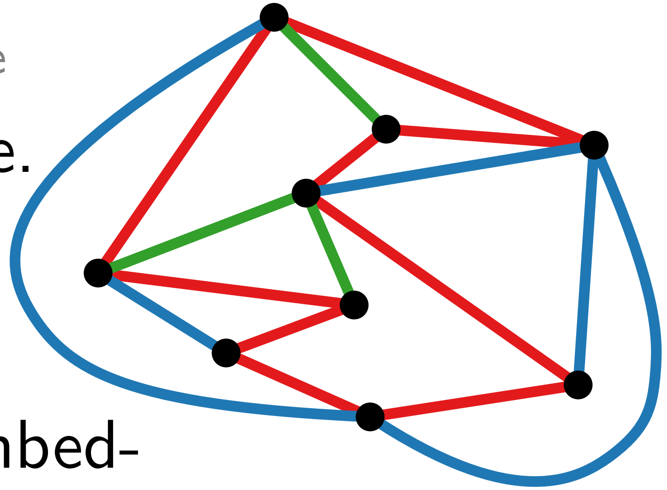




2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

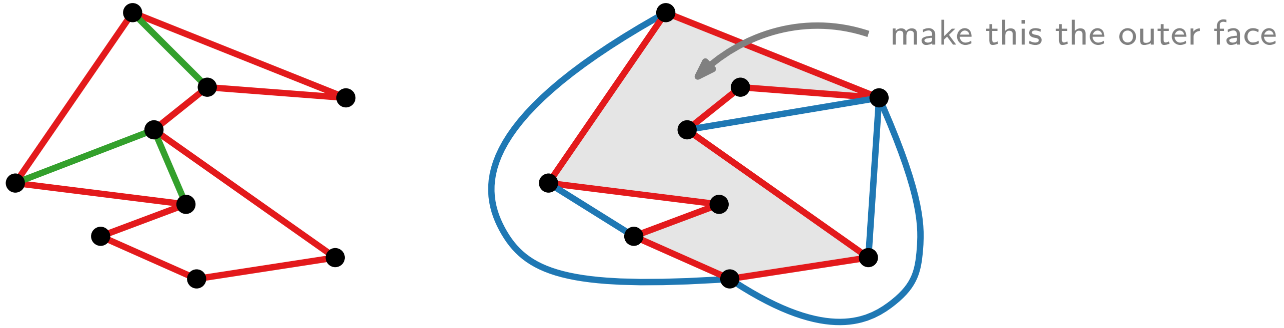
i.e., a graph that has a Hamiltonian cycle

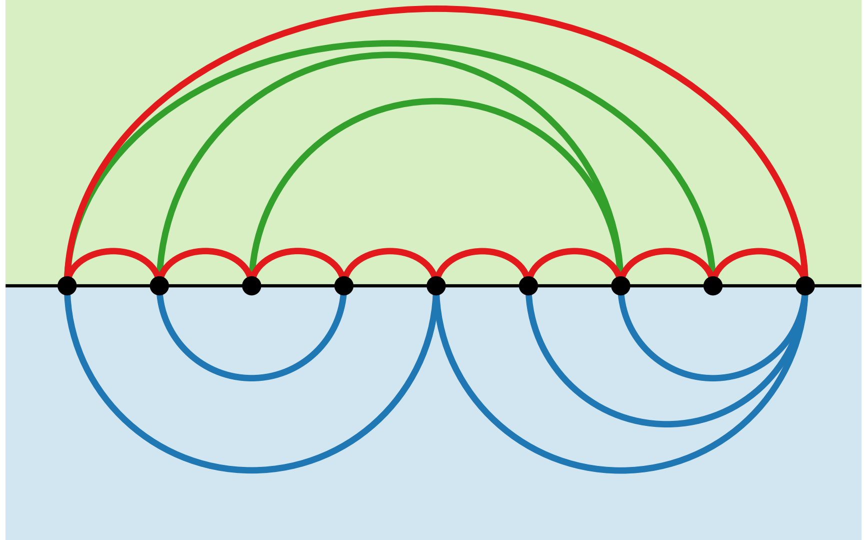


Proof.

"$\Leftarrow$":

- Let Γ be a planar drawing of a graph with a Hamiltonian cycle.
- In Γ, color the edges of the Hamiltonian cycle red, the edges inside green, and the edges outside blue.
- The red–green / red–blue edges induce two outerplanar embeddings with the same cyclic order of the vertices on the outer face.
- Put each one into a separate stack (same order of vertices on the spine).





2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Rightarrow$":
- Consider a 2-page stack layout as a drawing Γ.

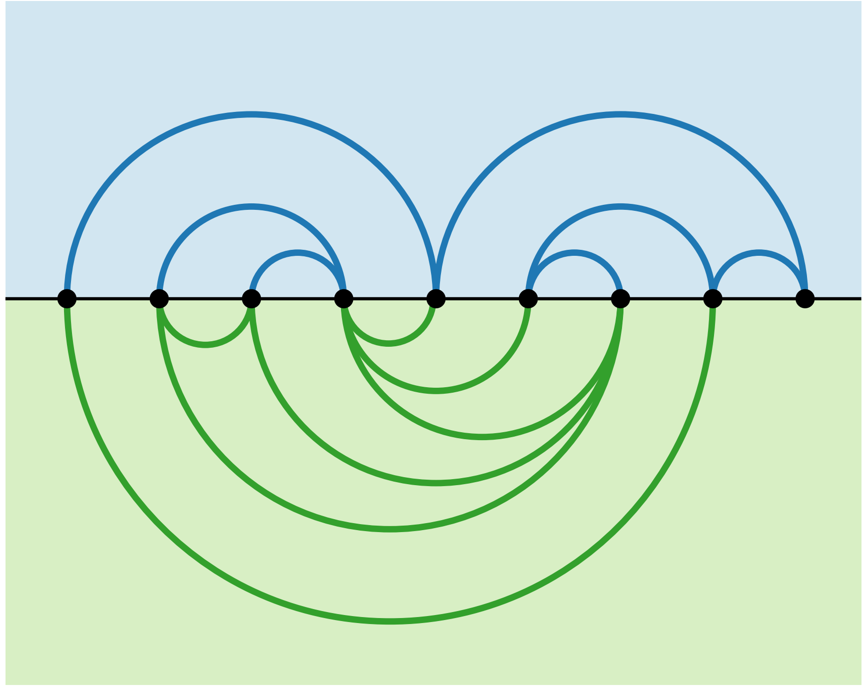
2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"⇒":
- Consider a 2-page stack layout as a drawing Γ.



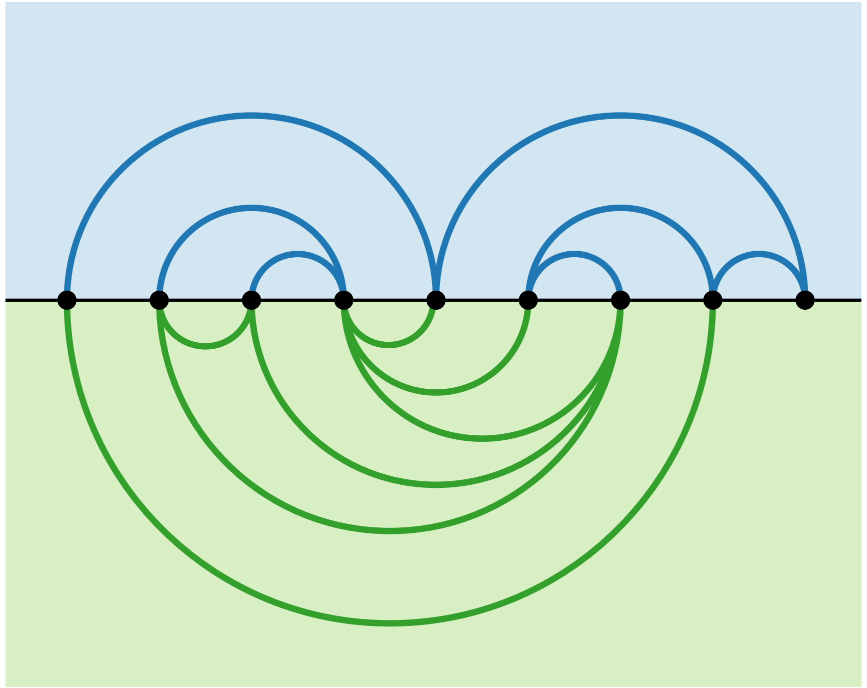
2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: sn(G) $\leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Rightarrow$":
- Consider a 2-page stack layout as a drawing Γ.
- Clearly, Γ is planar.



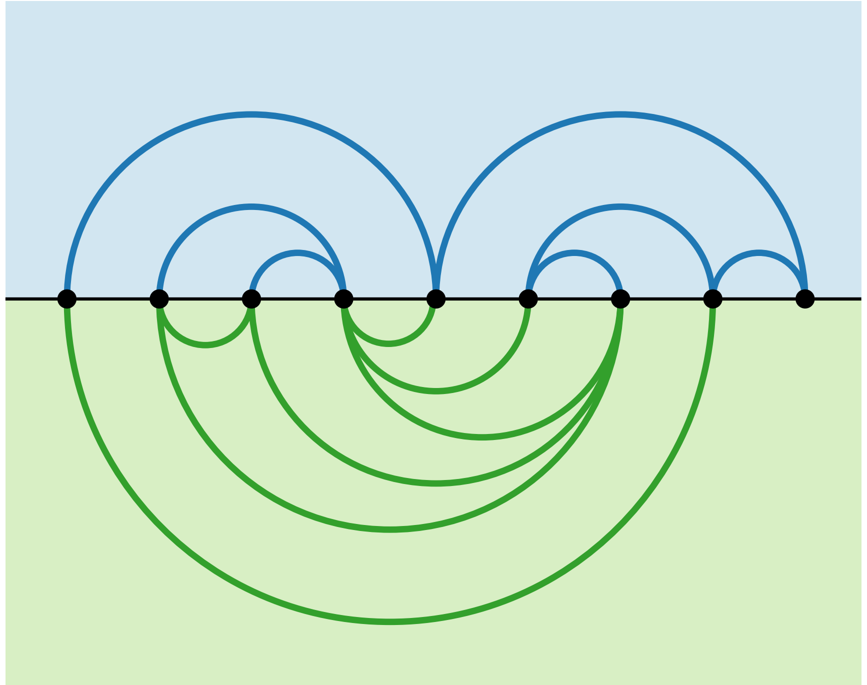
2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: sn(G) $\leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"⇒":
- Consider a 2-page stack layout as a drawing Γ.
- Clearly, Γ is planar.
- Add missing edges such that all pairs of neighboring vertices on the spine are connected (always possible).



2-Page Stack Layouts

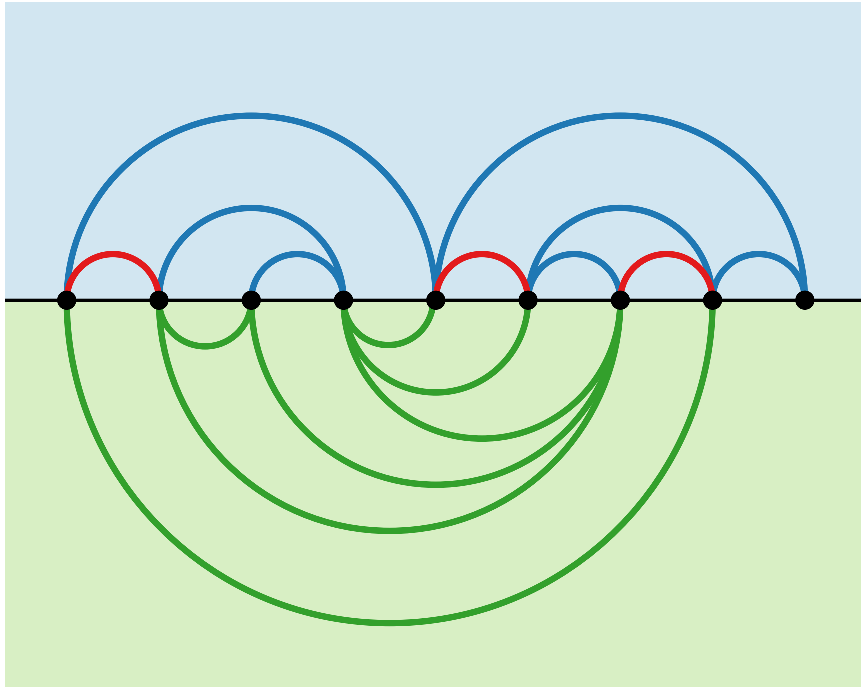
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: sn$(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Rightarrow$":

- Consider a 2-page stack layout as a drawing Γ.
- Clearly, Γ is planar.
- Add missing edges such that all pairs of neighboring vertices on the spine are connected (always possible).



2-Page Stack Layouts

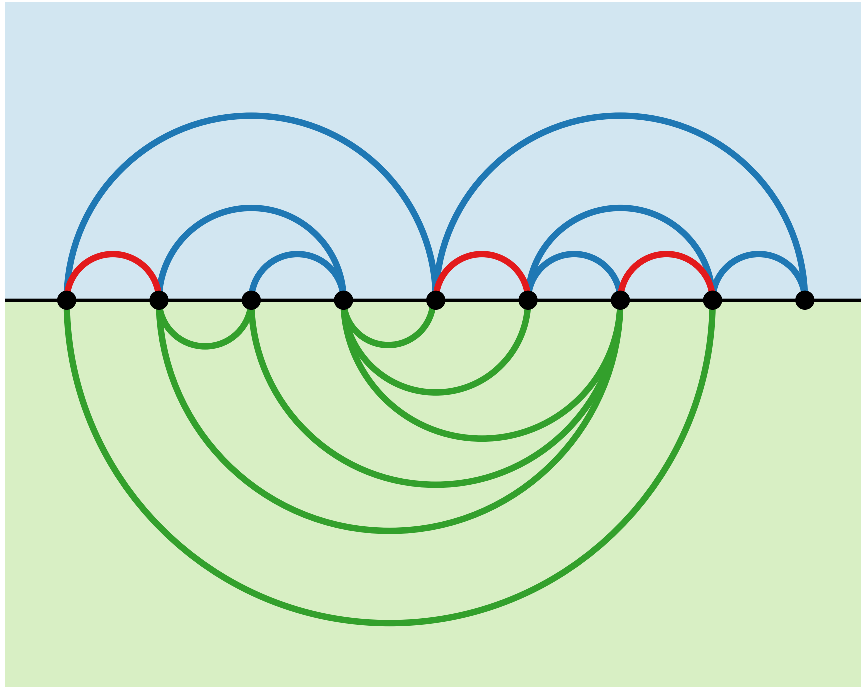
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Rightarrow$":

- Consider a 2-page stack layout as a drawing Γ.
- Clearly, Γ is planar.
- Add missing edges such that all pairs of neighboring vertices on the spine are connected (always possible).
- If absent, add edge from the first to the last vertex (always possible).



2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]

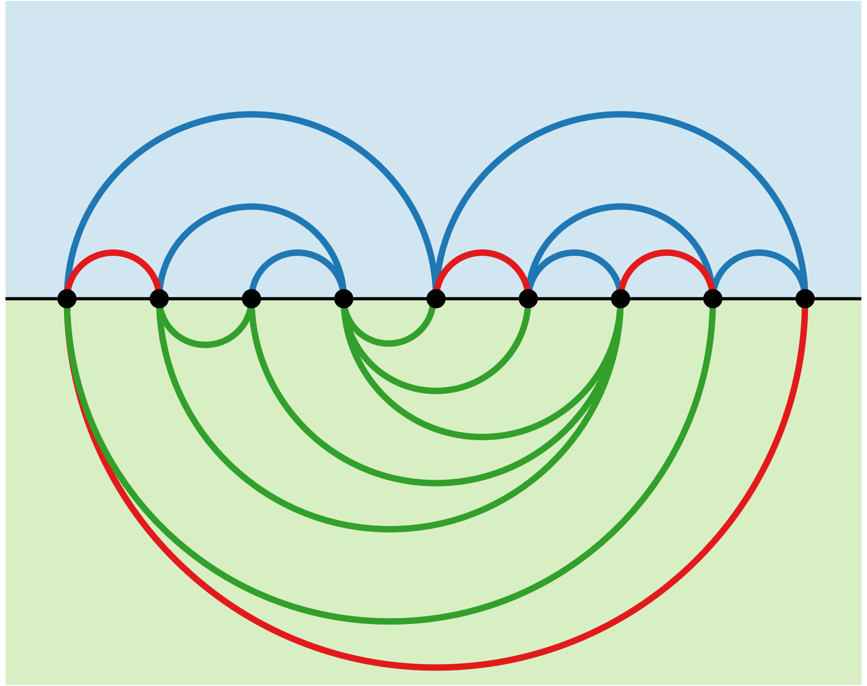
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"$\Rightarrow$":

- Consider a 2-page stack layout as a drawing Γ.
- Clearly, Γ is planar.
- Add missing edges such that all pairs of neighboring vertices on the spine are connected (always possible).
- If absent, add edge from the first to the last vertex (always possible).



2-Page Stack Layouts

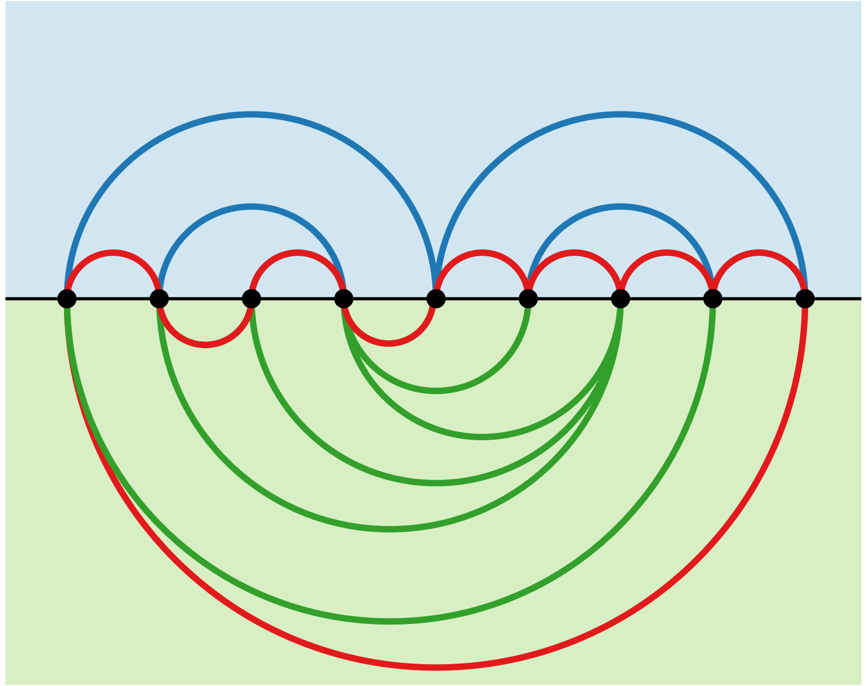
Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

i.e., a graph that has a Hamiltonian cycle

Proof.

"⇒":
- Consider a 2-page stack layout as a drawing Γ.
- Clearly, Γ is planar.
- Add missing edges such that all pairs of neighboring vertices on the spine are connected (always possible).
- If absent, add edge from the first to the last vertex (always possible).
- The Hamiltonian cycle traverses all vertices on the spine in order.

□



2-Page Stack Layouts

Theorem. [Bernhart & Kainen 1979]
For a graph G holds: $\mathrm{sn}(G) \leq 2 \Leftrightarrow G$ is a subgraph of a planar Hamiltonian graph

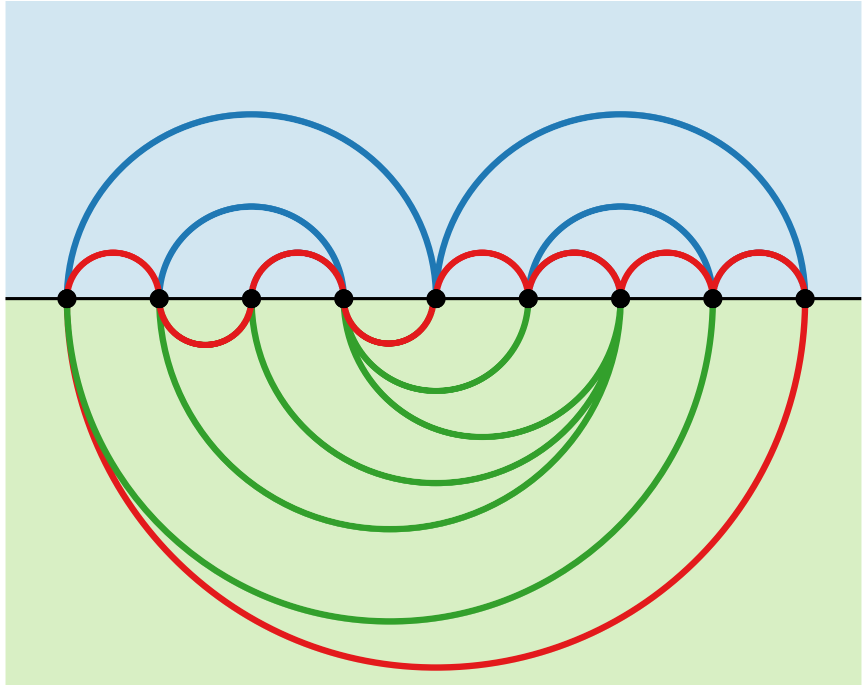
i.e., a graph that has a Hamiltonian cycle

Proof.

"⇒":

- Consider a 2-page stack layout as a drawing Γ.
- Clearly, Γ is planar.
- Add missing edges such that all pairs of neighboring vertices on the spine are connected (always possible).
- If absent, add edge from the first to the last vertex (always possible).
- The Hamiltonian cycle traverses all vertices on the spine in order.

□



This result includes planar bipartite and series-parallel graphs.

Stack Layouts of Planar Graphs

We have seen that the outerplanar graphs have stack number 1 and specific planar graphs stack number 2. What is the maximum stack number of any n-vertex **planar** graph?

Stack Layouts of Planar Graphs

We have seen that the outerplanar graphs have stack number 1 and specific planar graphs stack number 2. What is the maximum stack number of any n-vertex **planar** graph?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Stack Layouts of Planar Graphs

We have seen that the outerplanar graphs have stack number 1 and specific planar graphs stack number 2. What is the maximum stack number of any n-vertex **planar** graph?

~~2~~ 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

(not all planar graphs can be extended to have a Hamiltonian cycle)

Stack Layouts of Planar Graphs

We have seen that the outerplanar graphs have stack number 1 and specific planar graphs stack number 2. What is the maximum stack number of any n-vertex **planar** graph?

~~2~~ 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Conjecture. [Bernhart & Kainen 1979]
For $n \to \infty$, there are n-vertex planar graphs such that $\mathrm{sn}(G) \to \infty$.
(The stack number of planar graphs is not bounded by a constant.)

Stack Layouts of Planar Graphs

We have seen that the outerplanar graphs have stack number 1 and specific planar graphs stack number 2. What is the maximum stack number of any n-vertex **planar** graph?

~~2~~ 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

~~**Conjecture.** [Bernhart & Kainen 1979]~~
~~For $n \to \infty$, there are n-vertex planar graphs such that $\text{sn}(G) \to \infty$.~~
~~(The stack number of planar graphs is not bounded by a constant.)~~

Theorem. [Buss & Shor 1984]
For every planar graph G, $\text{sn}(G) \leq 9$.

Stack Layouts of Planar Graphs

We have seen that the outerplanar graphs have stack number 1 and specific planar graphs stack number 2. What is the maximum stack number of any n-vertex **planar** graph?

~~2~~ 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

~~**Conjecture.** [Bernhart & Kainen 1979]~~
~~For $n \to \infty$, there are n-vertex planar graphs such that $\text{sn}(G) \to \infty$.~~
~~(The stack number of planar graphs is not bounded by a constant.)~~

Theorem. [Buss & Shor 1984]
For every planar graph G, $\text{sn}(G) \leq 9$.

Theorem. [Heath 1984]
For every planar graph G, $\text{sn}(G) \leq 7$.

Stack Layouts of Planar Graphs

We have seen that the outerplanar graphs have stack number 1 and specific planar graphs stack number 2. What is the maximum stack number of any n-vertex **planar** graph?

~~2~~ 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

~~**Conjecture.** [Bernhart & Kainen 1979]~~
~~For $n \to \infty$, there are n-vertex planar graphs such that $\text{sn}(G) \to \infty$.~~
~~(The stack number of planar graphs is not bounded by a constant.)~~

Theorem. [Buss & Shor 1984]
For every planar graph G, $\text{sn}(G) \leq 9$.

Theorem. [Heath 1984]
For every planar graph G, $\text{sn}(G) \leq 7$.

Theorem. [Yannakakis 1986]
For every planar graph G, $\text{sn}(G) \leq 4$.

Stack Layouts of Planar Graphs

We have seen that the outerplanar graphs have stack number 1 and specific planar graphs stack number 2. What is the maximum stack number of any n-vertex **planar** graph?

~~2~~ 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

~~**Conjecture.** [Bernhart & Kainen 1979]~~
~~For $n \to \infty$, there are n-vertex planar graphs such that $\text{sn}(G) \to \infty$.~~
~~(The stack number of planar graphs is not bounded by a constant.)~~

Theorem. [Buss & Shor 1984]
For every planar graph G, $\text{sn}(G) \leq 9$.

Theorem. [Heath 1984]
For every planar graph G, $\text{sn}(G) \leq 7$.

Theorem. [Yannakakis 1986]
For every planar graph G, $\text{sn}(G) \leq 4$.

But are there planar graphs that need 4 stacks?

Stack Layouts of Planar Graphs

We have seen that the outerplanar graphs have stack number 1 and specific planar graphs stack number 2. What is the maximum stack number of any n-vertex **planar** graph?

~~2~~ 3 **(4)** 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

~~**Conjecture.** [Bernhart & Kainen 1979]
For $n \to \infty$, there are n-vertex planar graphs such that $\text{sn}(G) \to \infty$.
(The stack number of planar graphs is not bounded by a constant.)~~

Theorem. [Buss & Shor 1984]
For every planar graph G, $\text{sn}(G) \leq 9$.

Theorem. [Heath 1984]
For every planar graph G, $\text{sn}(G) \leq 7$.

Theorem. [Yannakakis 1986]
For every planar graph G, $\text{sn}(G) \leq 4$.

Theorem. [Yannakakis 2020, Bekos, Kaufmann, Klute, Pupyrev, Raftopoulou & Ueckerdt 2020]
There is a planar graph G with $\text{sn}(G) \geq 4$.

But are there planar graphs that need 4 stacks?

Yes! (The planar graph presented by Bekos et al. has 275 vertices and 819 edges.)

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number?

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number.

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We first show that $\text{sn}(K_n) \geq n/2$.

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

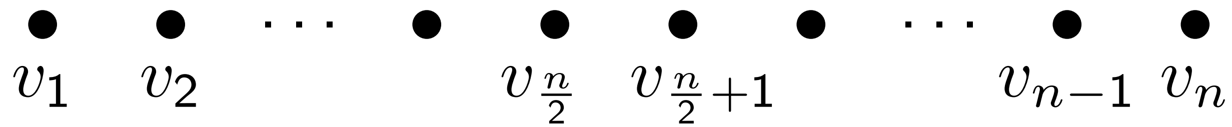
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We first show that $\mathrm{sn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \ldots, v_n$ accordingly.



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

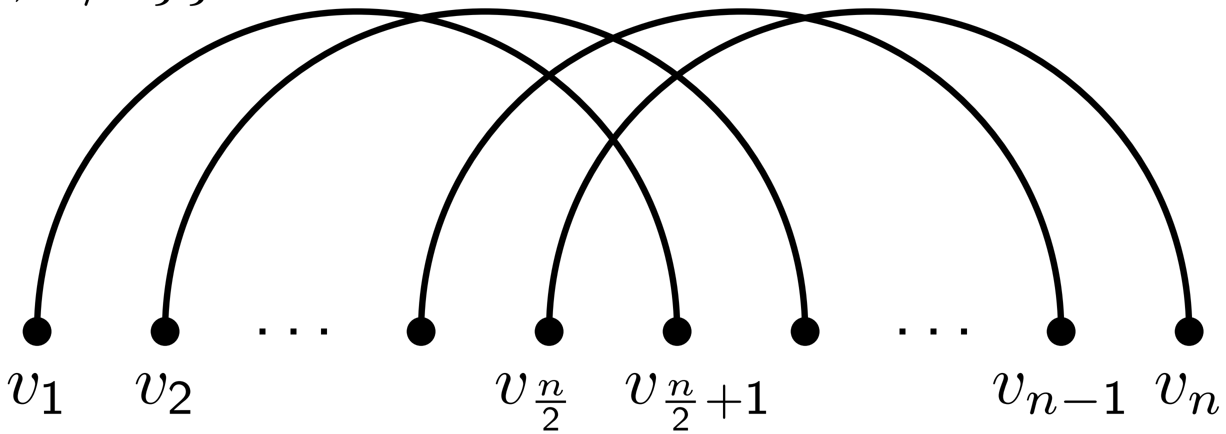
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We first show that $\text{sn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \ldots, v_n$ accordingly.
- Consider the set of edges $\{v_i v_{n/2+i} \mid i \in \{1, \ldots, n/2\}\}$.



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

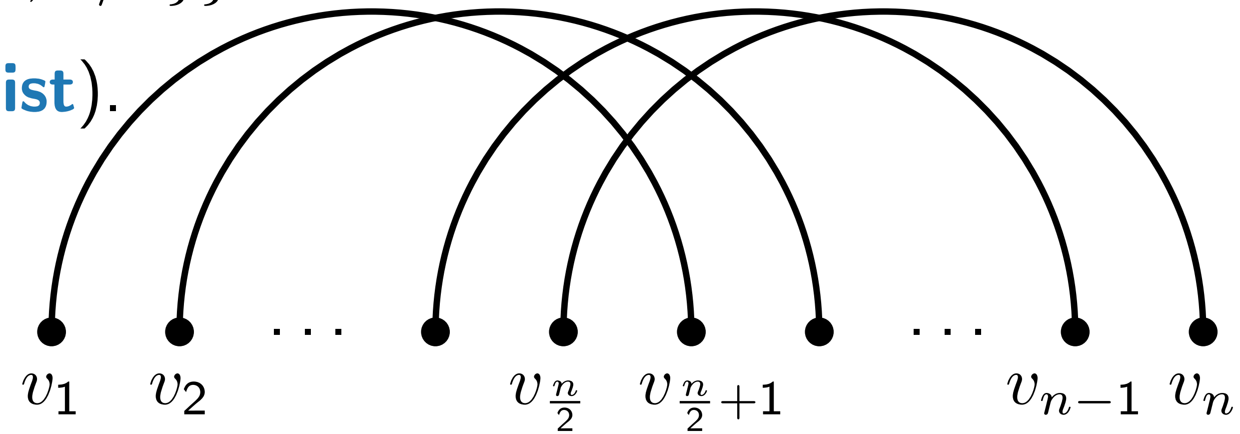
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We first show that $\mathrm{sn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \dots, v_n$ accordingly.
- Consider the set of edges $\{v_i v_{n/2+i} \mid i \in \{1, \dots, n/2\}\}$.
- These are $n/2$ pairwise crossing edges (**$n/2$-twist**).



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

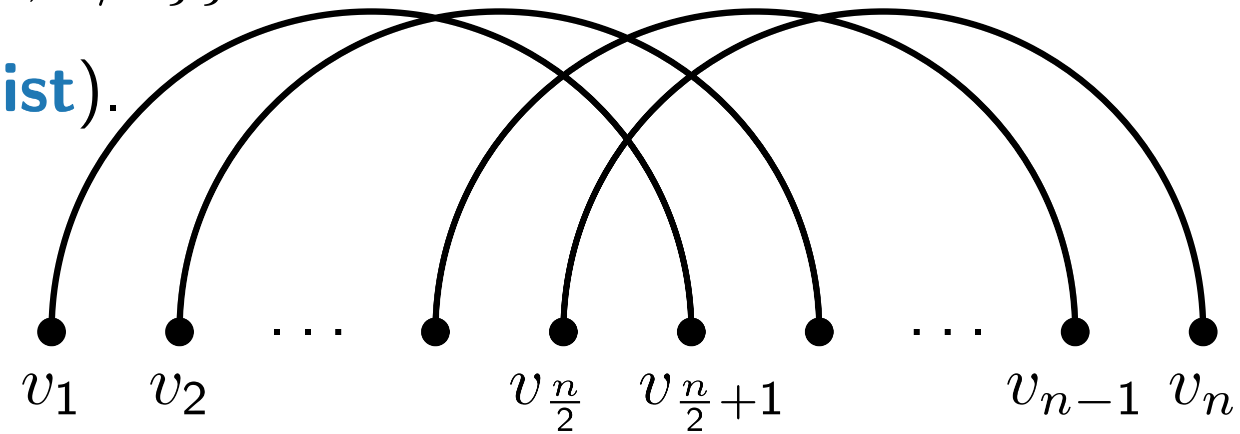
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We first show that $\text{sn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \ldots, v_n$ accordingly.
- Consider the set of edges $\{v_i v_{n/2+i} \mid i \in \{1, \ldots, n/2\}\}$.
- These are $n/2$ pairwise crossing edges (**$n/2$-twist**).
- Each of these edges needs a separate stack.



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\text{sn}(K_n) \leq n/2$.

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

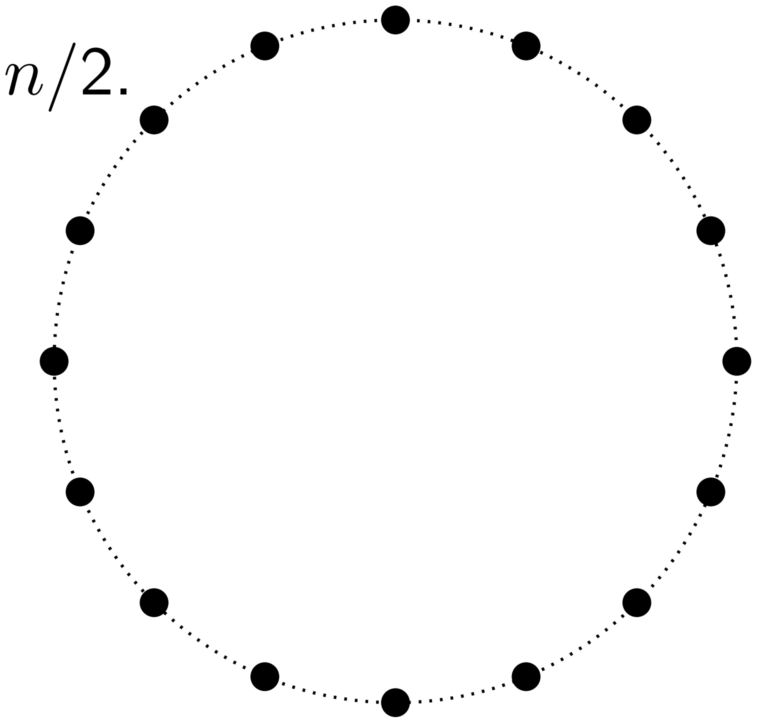
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\mathrm{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

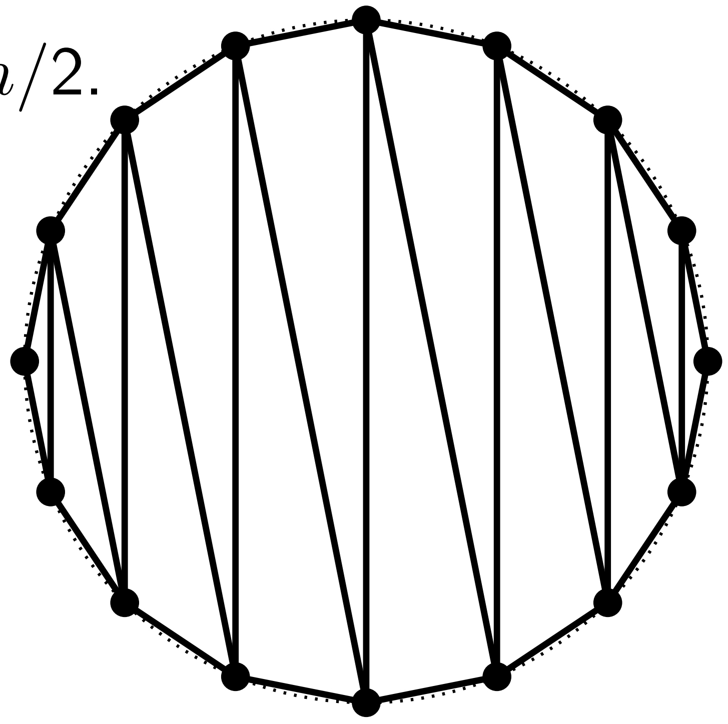
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\mathrm{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

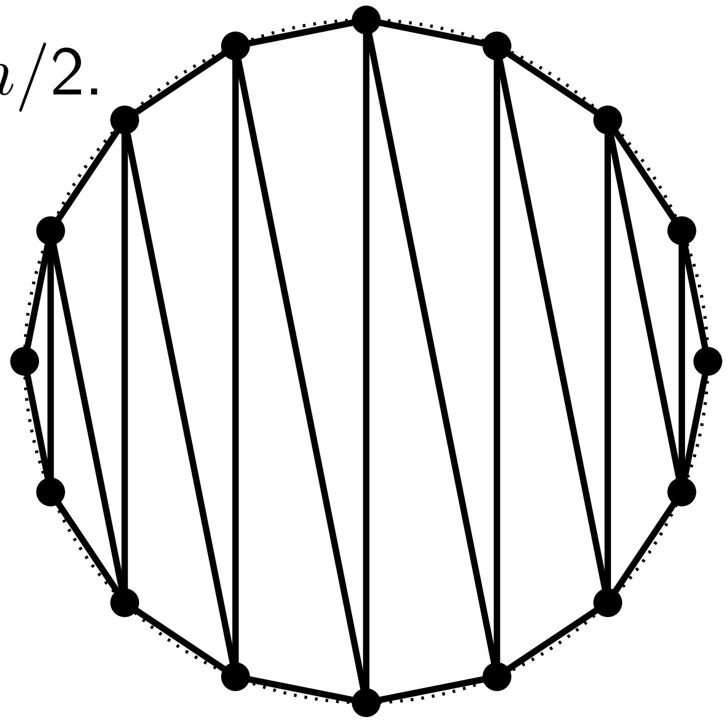
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\text{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

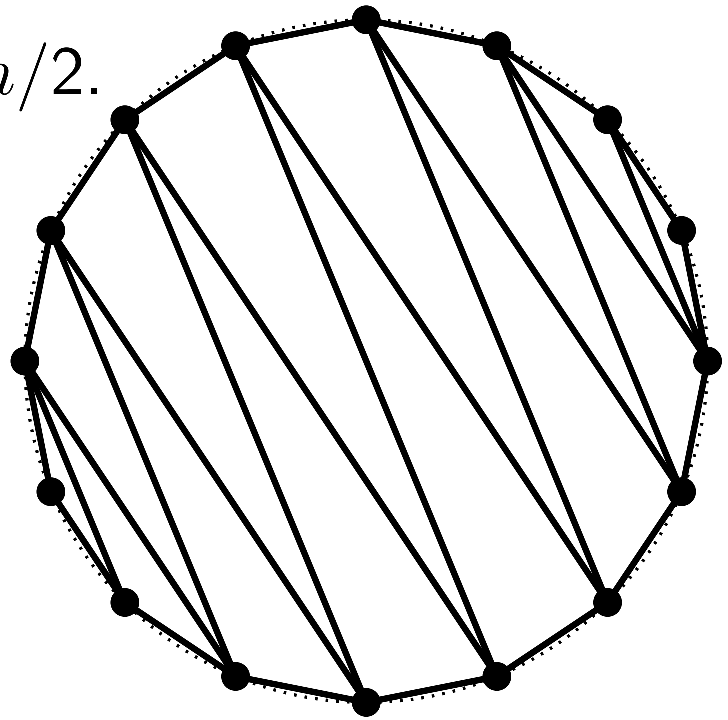
2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\text{sn}(K_n) \leq n/2$.



- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.
- "Rotate" the inner diagonals by 1, 2, ..., $n/2 - 1$ position(s).

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

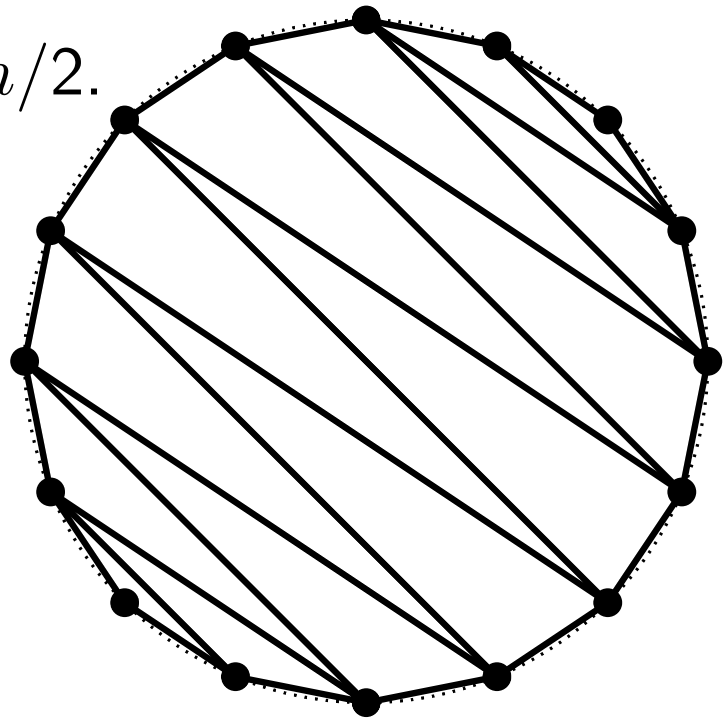
2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.
Assume that n is even (the case for odd n is similar).

We now show that $\text{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.
- "Rotate" the inner diagonals by 1, 2, . . . , $n/2 - 1$ position(s).



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

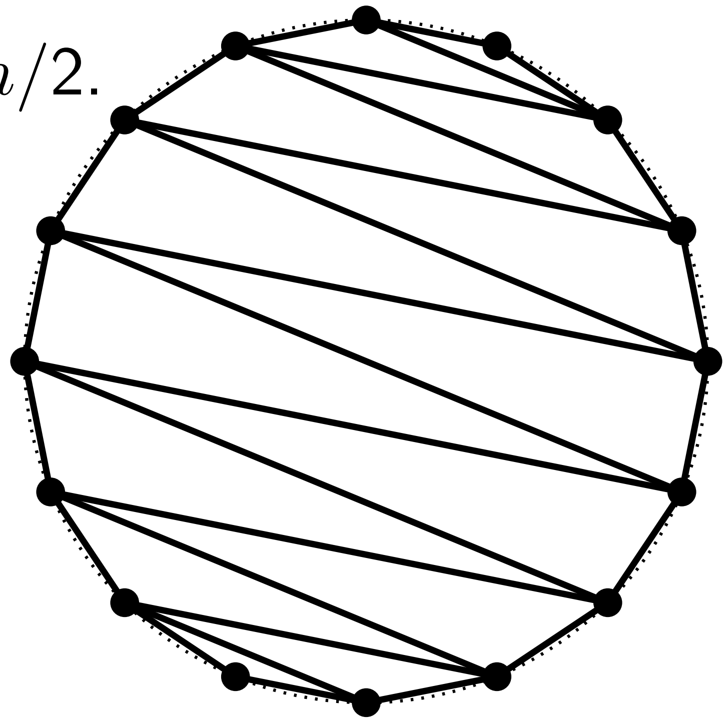
2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\text{sn}(K_n) \leq n/2$.



- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.
- "Rotate" the inner diagonals by 1, 2, . . . , $n/2 - 1$ position(s).

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

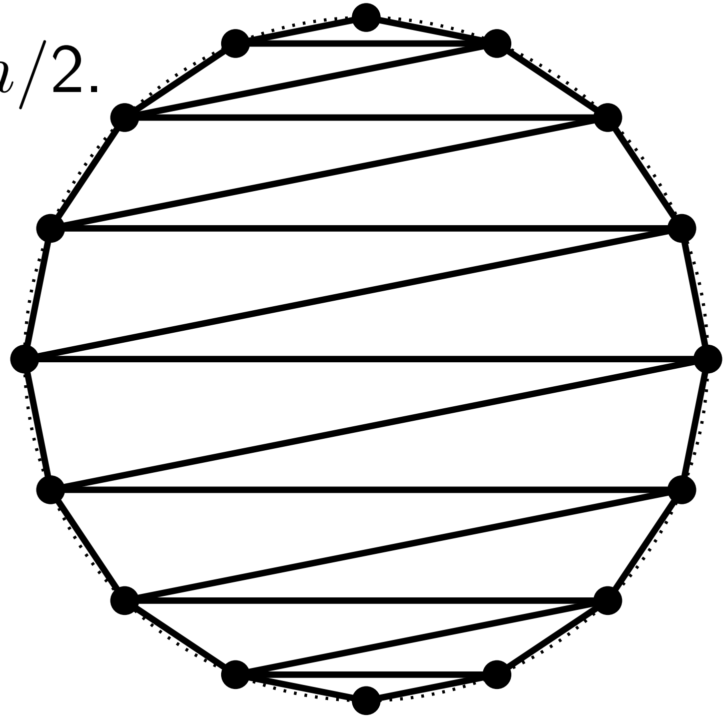
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\text{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.
- "Rotate" the inner diagonals by 1, 2, ..., $n/2 - 1$ position(s).



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

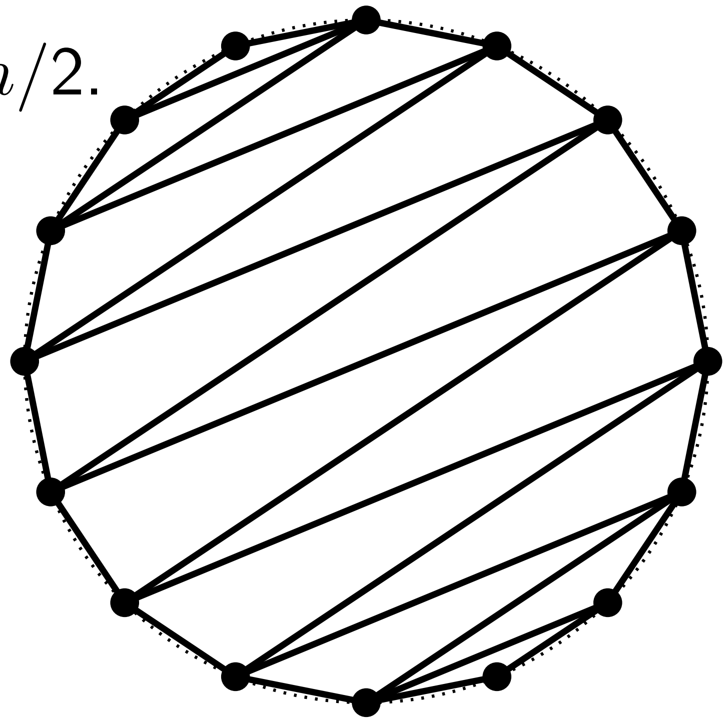
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\mathrm{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.
- "Rotate" the inner diagonals by 1, 2, ..., $n/2 - 1$ position(s).



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

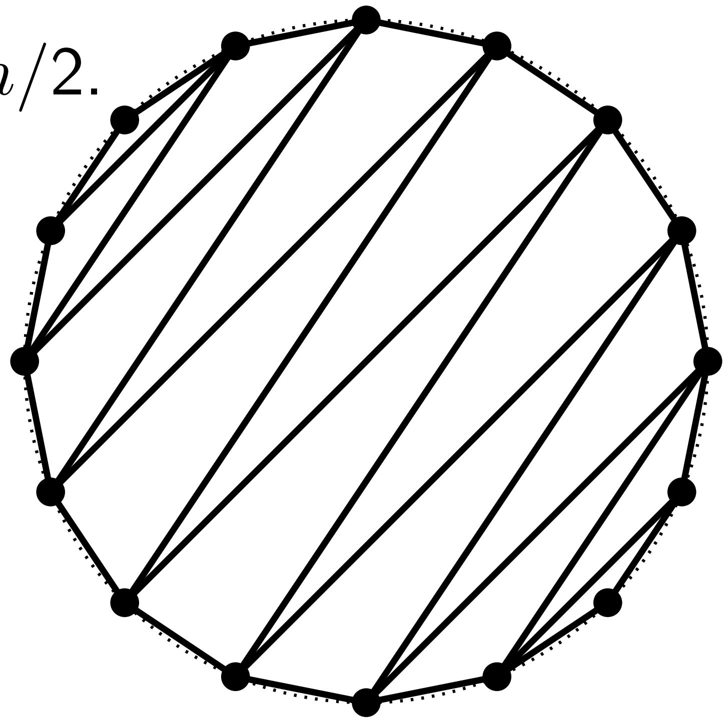
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\mathrm{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.
- "Rotate" the inner diagonals by 1, 2, ..., $n/2 - 1$ position(s).



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

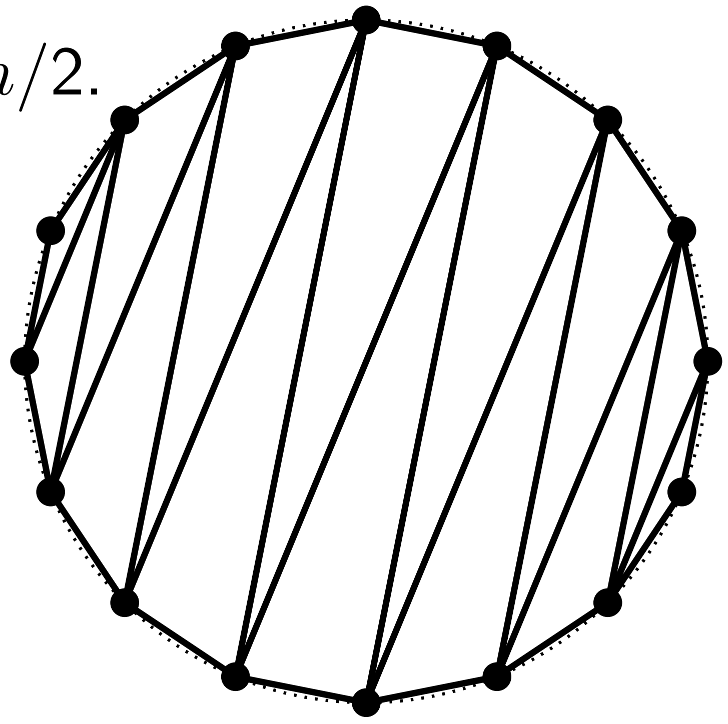
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\mathrm{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.
- "Rotate" the inner diagonals by 1, 2, ..., $n/2 - 1$ position(s).



Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

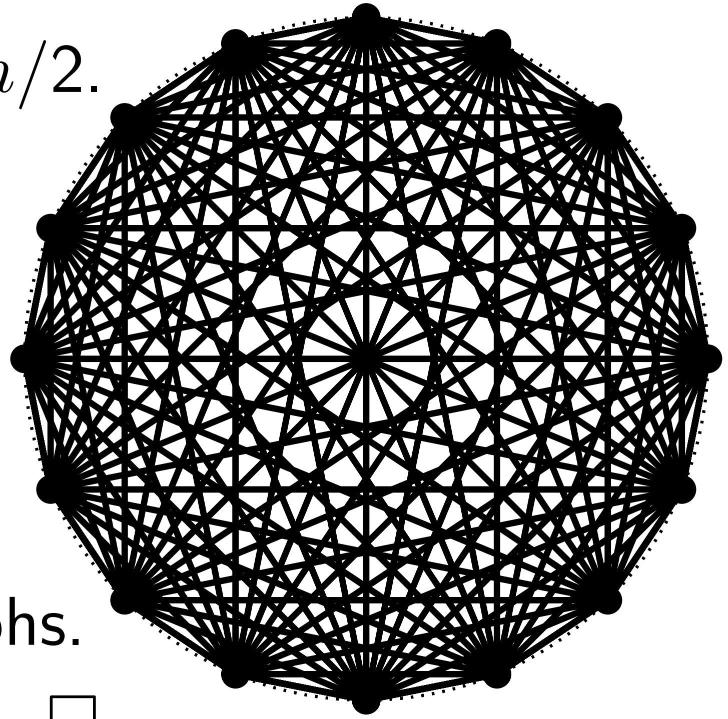
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\text{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\text{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.
- "Rotate" the inner diagonals by 1, 2, . . . , $n/2 - 1$ position(s).
- Clearly, every edge appears in some of these $n/2$ outerplanar graphs.



□

Stack Layouts of Complete Graphs

We have seen that outerplanar and planar graphs have constant stack number.
Do all graphs have constant stack number? Clearly, complete graphs have the largest stack number. What is the stack number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

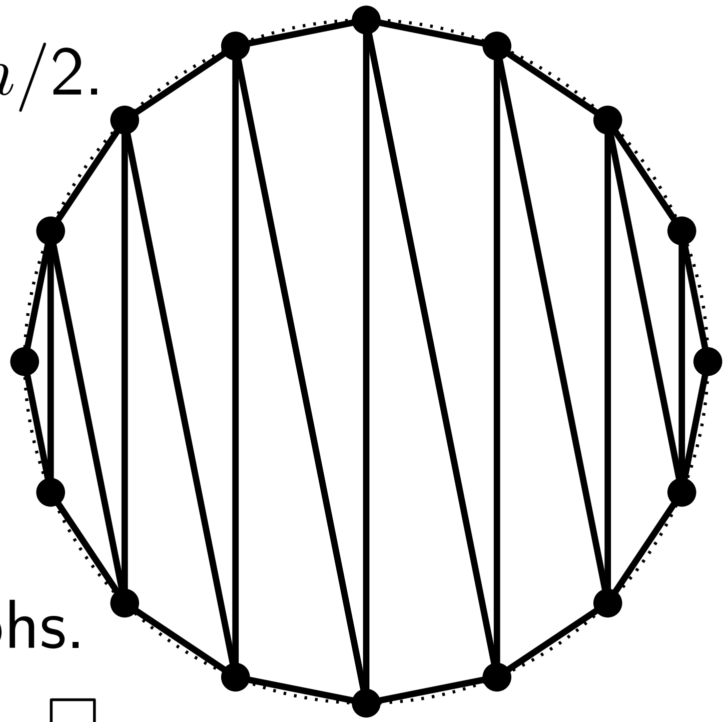
Theorem. [Bernhart & Kainen 1979]
For $n \geq 4$, $\mathrm{sn}(K_n) = \lceil n/2 \rceil$.

Proof.

Assume that n is even (the case for odd n is similar).

We now show that $\mathrm{sn}(K_n) \leq n/2$.

- Arrange the vertices of K_n on a circle.
- Add boundary edges and inner diagonals as follows:
- This is an outerplanar graph drawing and can go to one stack.
- "Rotate" the inner diagonals by 1, 2, ..., $n/2 - 1$ position(s).
- Clearly, every edge appears in some of these $n/2$ outerplanar graphs.



□

1-Page Queue Layouts

1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, $\mathrm{qn}(T) = 1$.

1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, $\mathrm{qn}(T) = 1$.

Proof.

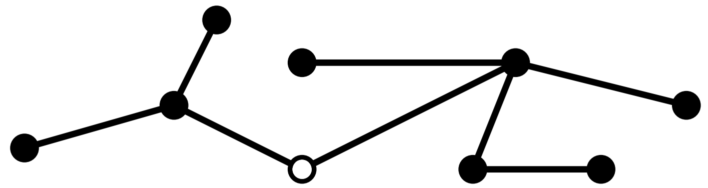
- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.

1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, qn$(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.

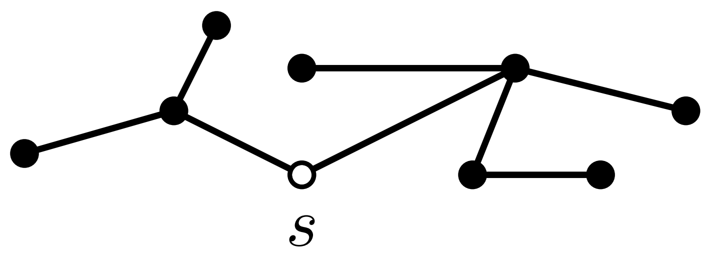


1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, qn$(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.

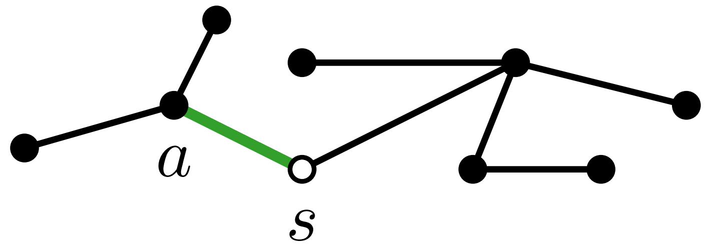


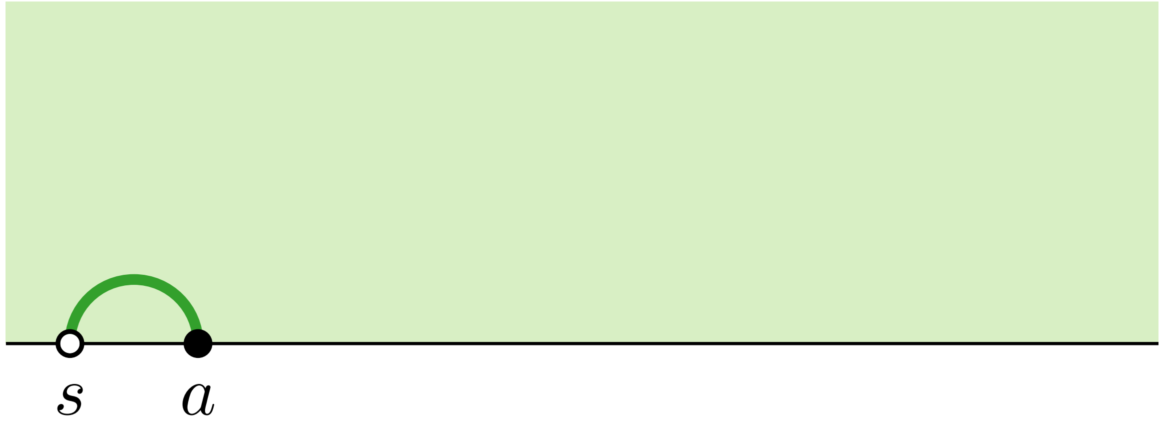
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, $\text{qn}(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.



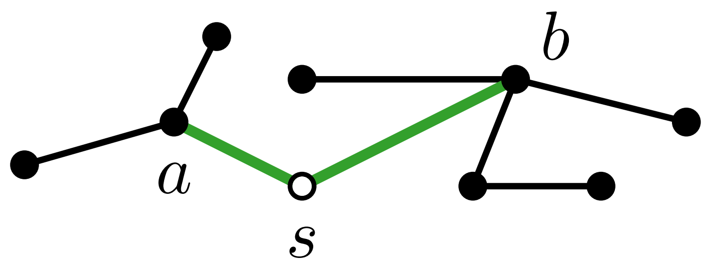


1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, qn$(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.

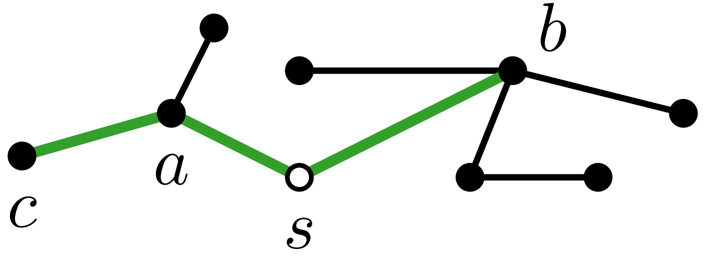


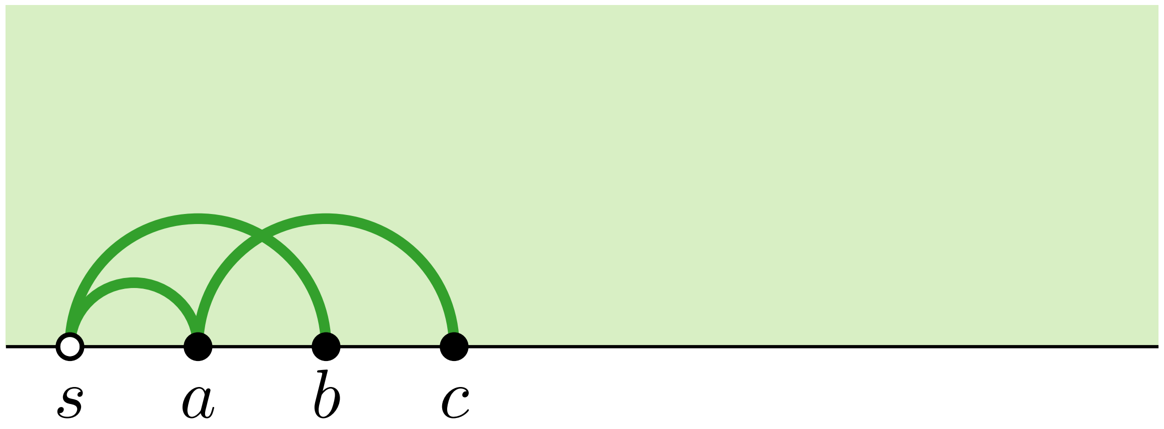
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, qn$(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.



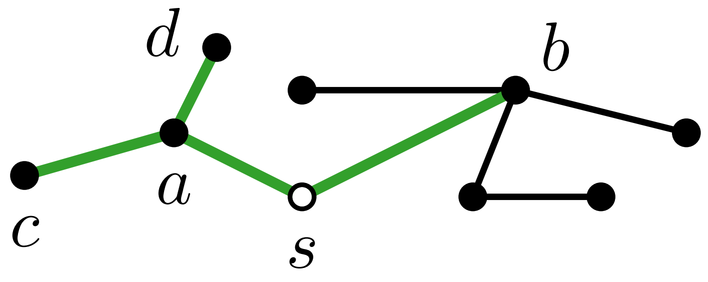


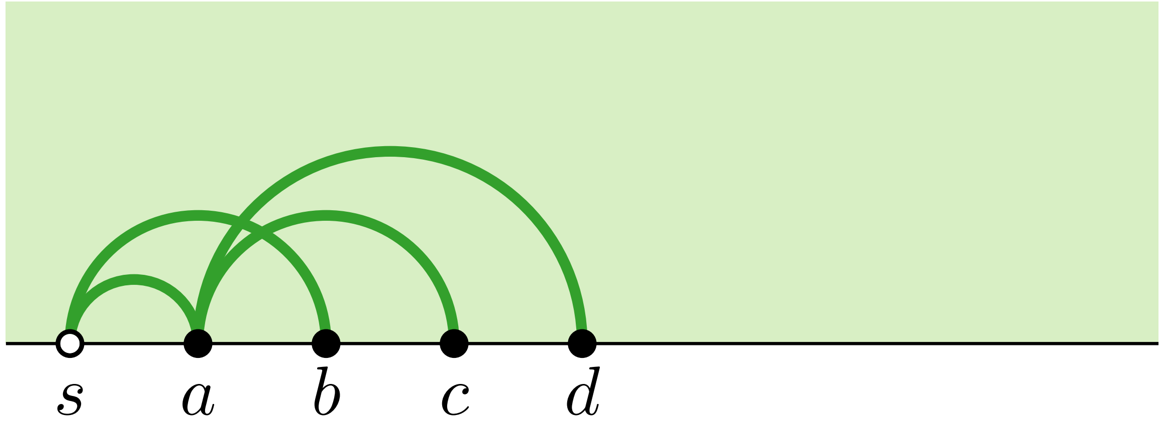
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, $\text{qn}(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.



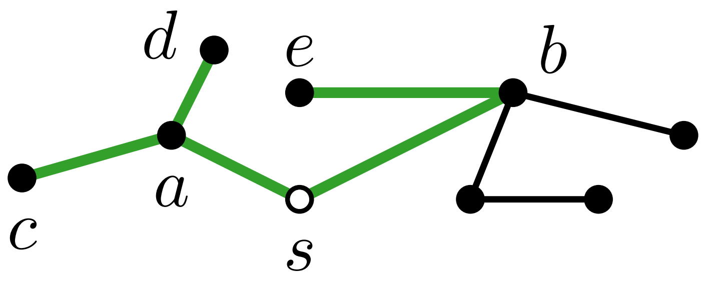


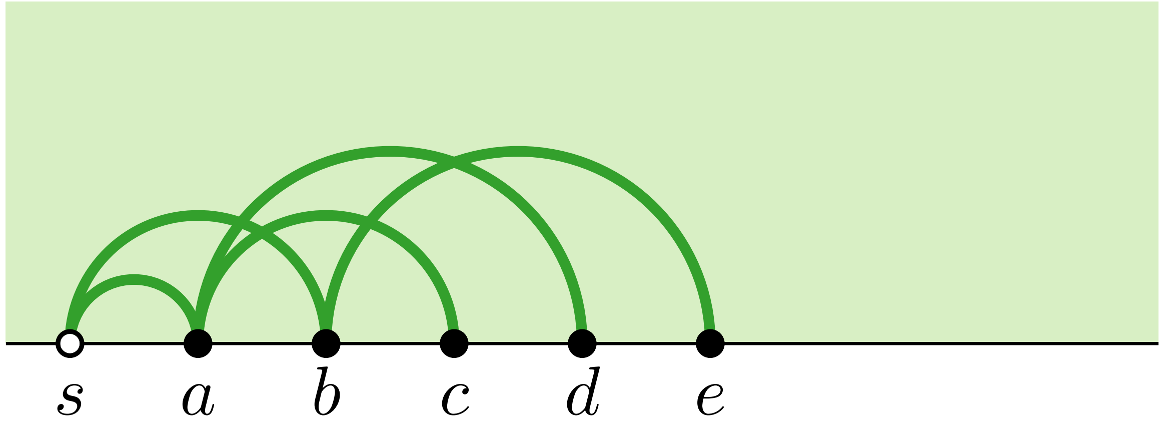
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, qn$(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.



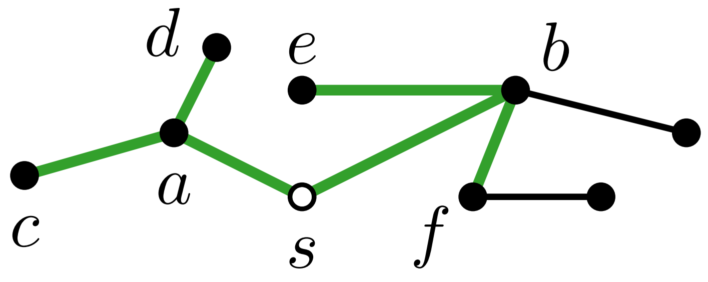


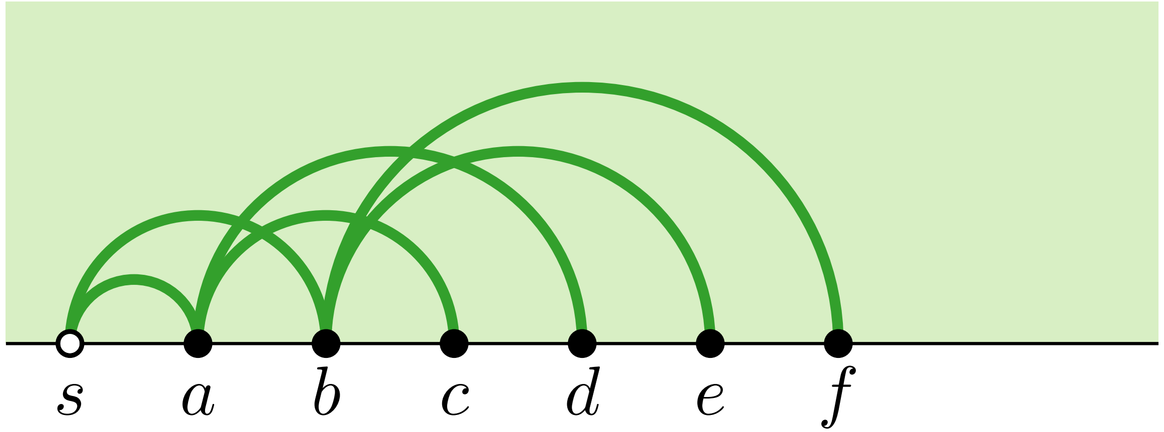
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, $\mathrm{qn}(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.



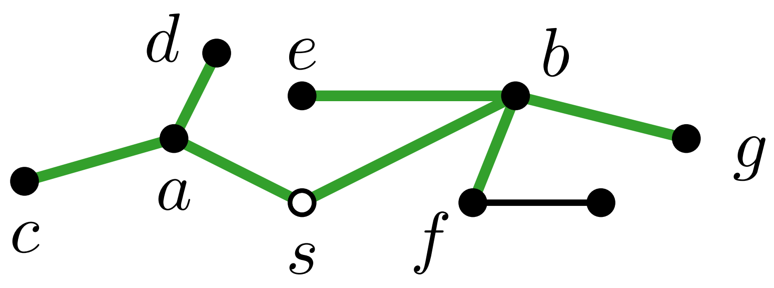


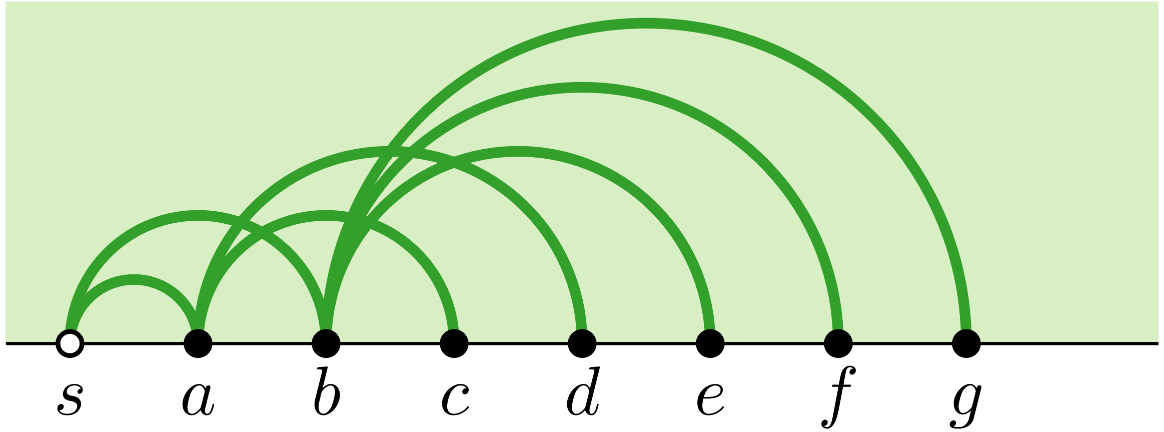
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, qn$(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.



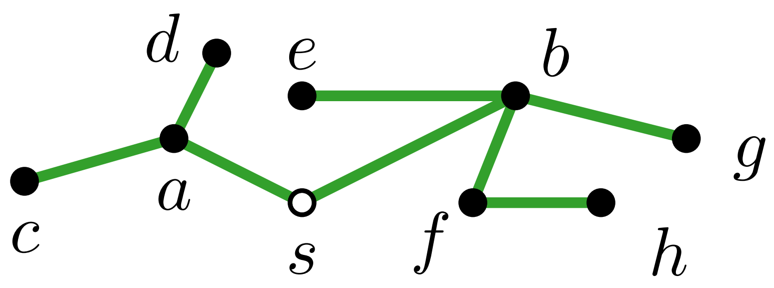


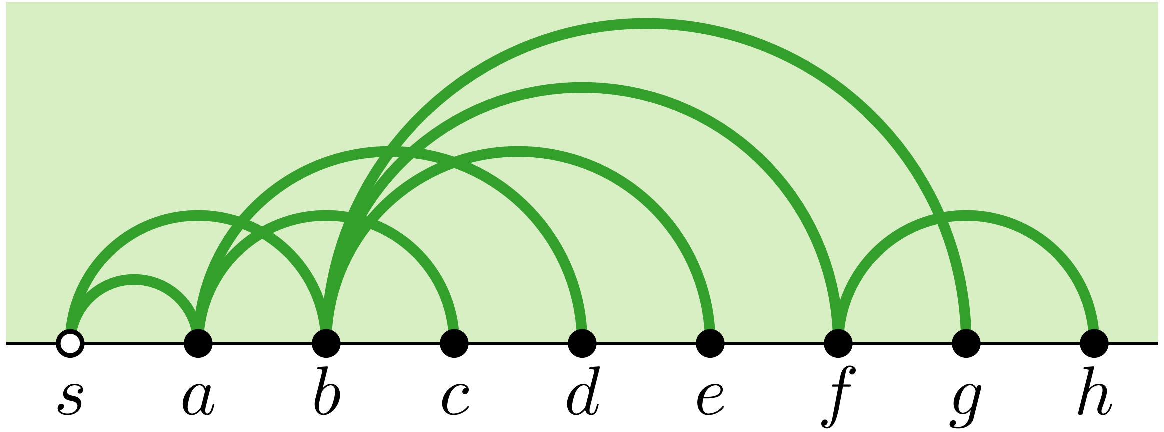
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, qn$(T) = 1$.

Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.



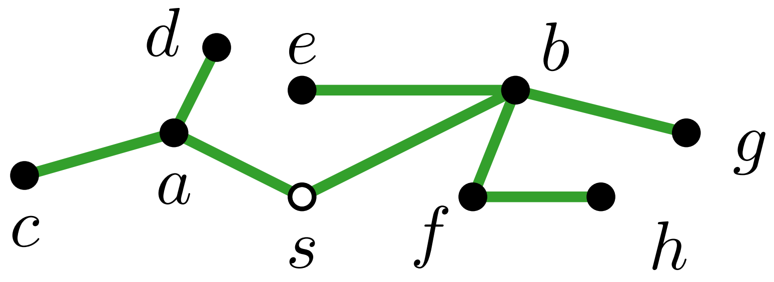


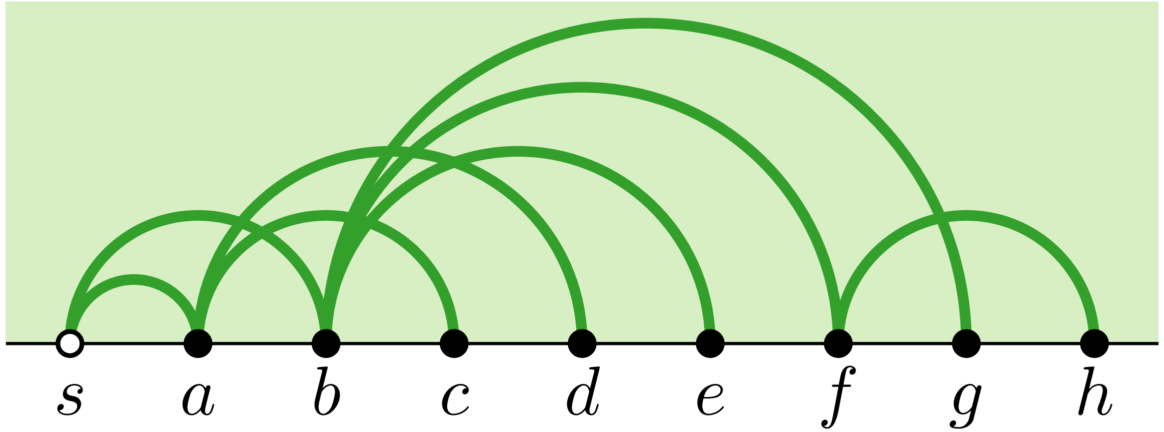
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every tree T, qn$(T) = 1$.

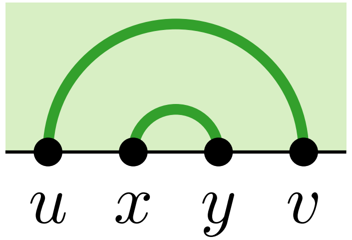
Proof.

- The exploration order in a breadth-first search (BFS) traversal yields a queue layout.





- If there was a nesting uv above xy, we would find u before x in the BFS, but discover a neighbor of x before a neighbor of u.



□

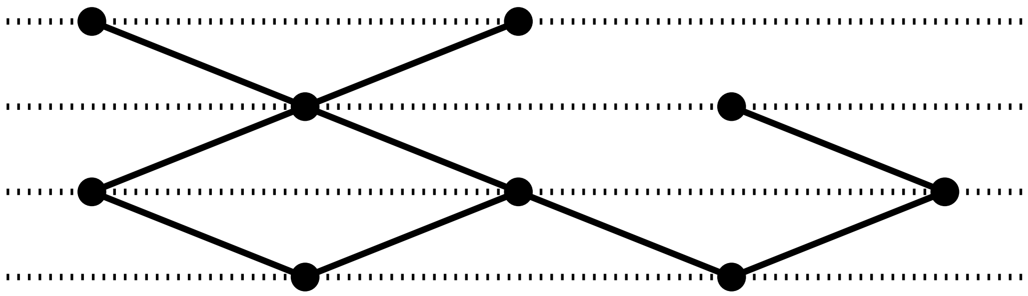
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\text{qn}(G) = 1$.

1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\mathrm{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

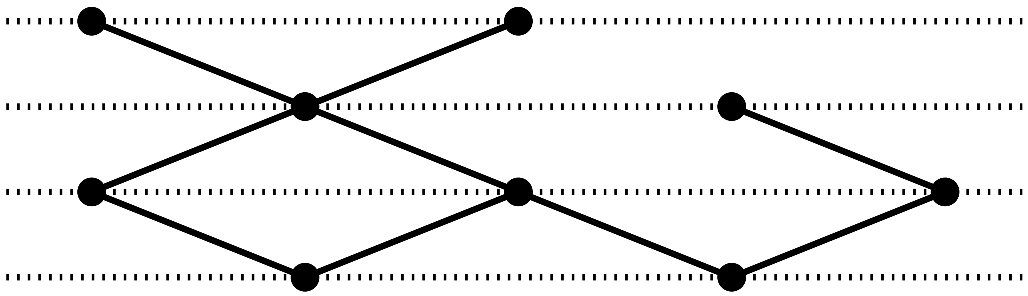


1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\mathrm{qn}(G) = 1$.

Proof.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.



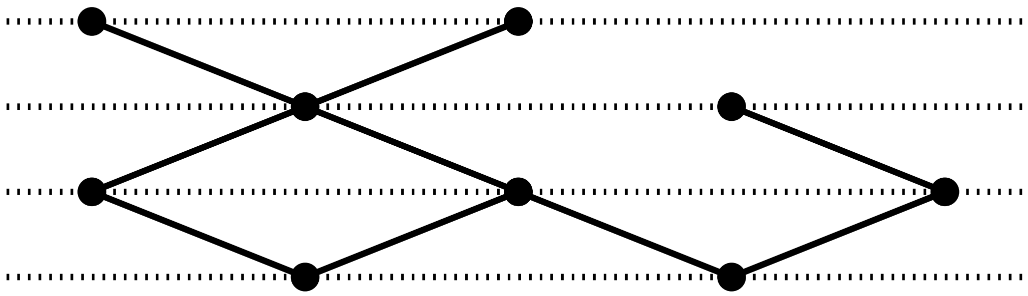
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\text{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.



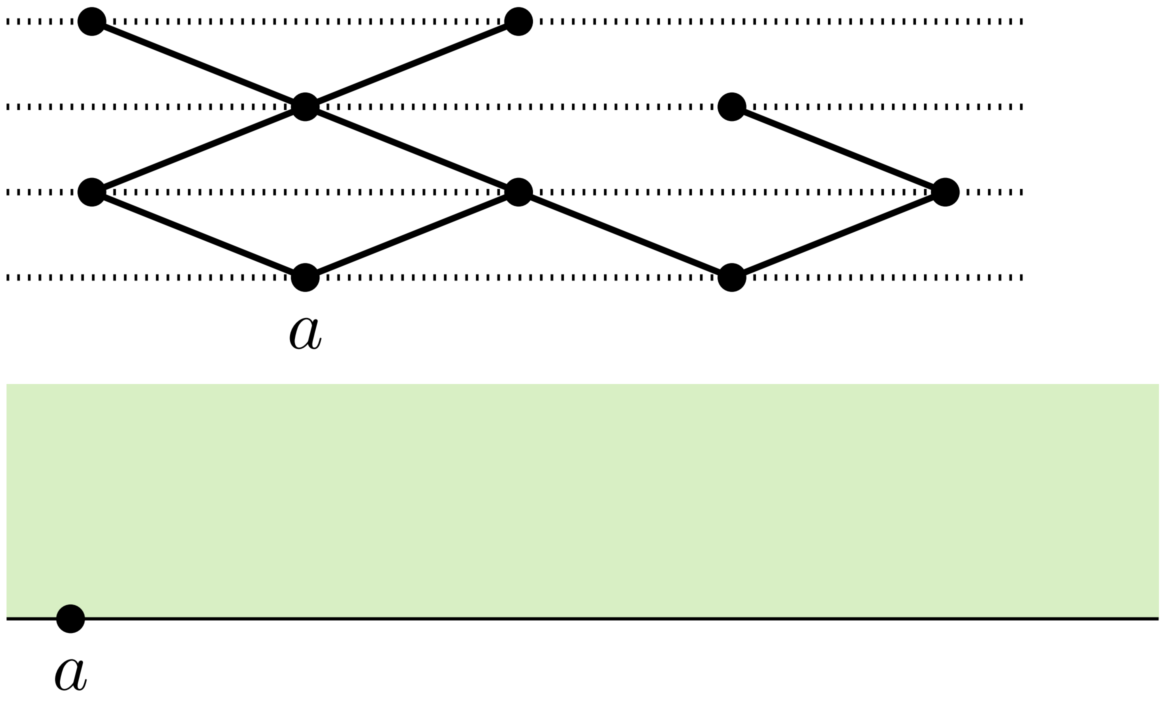
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, qn$(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.



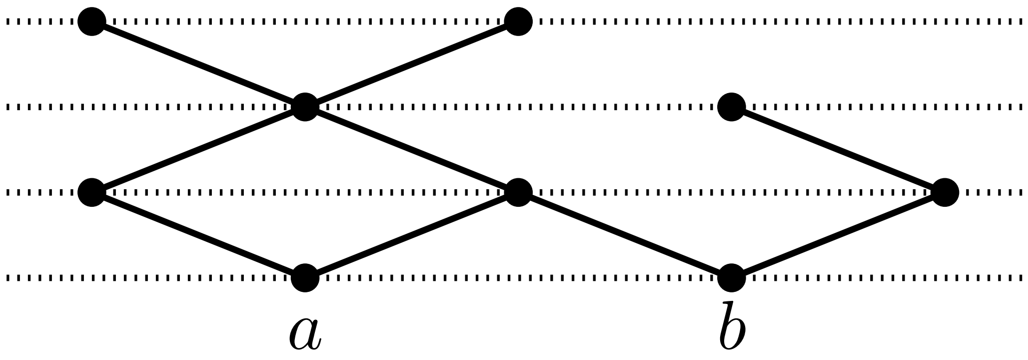
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\mathrm{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.



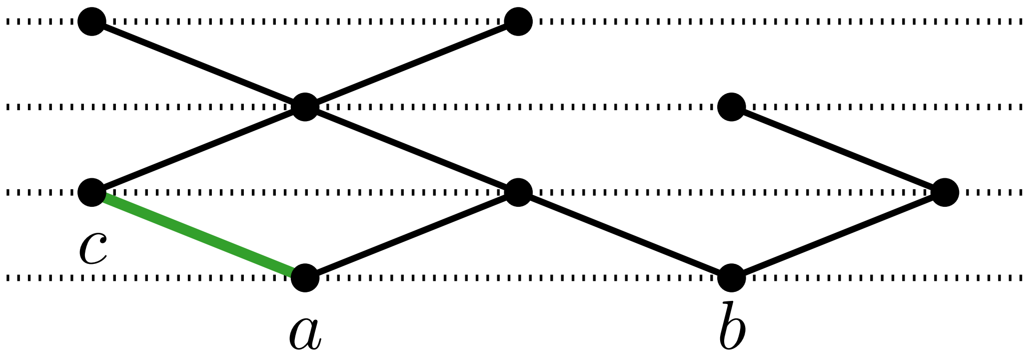
1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\text{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.



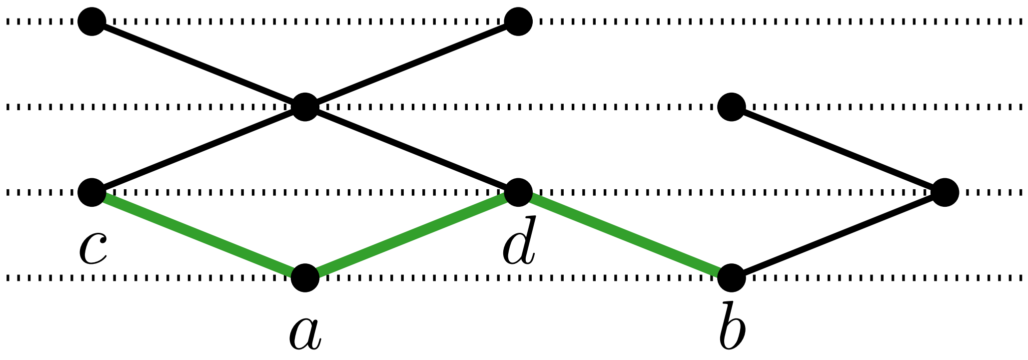
1-Page Queue Layouts

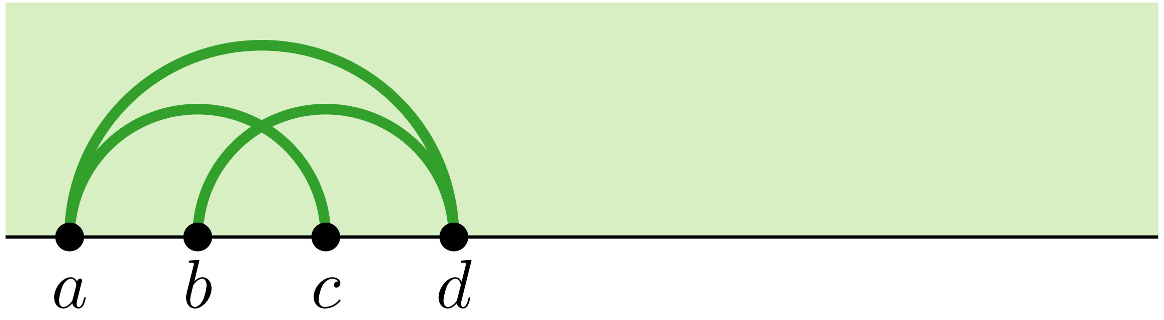
Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\mathrm{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.





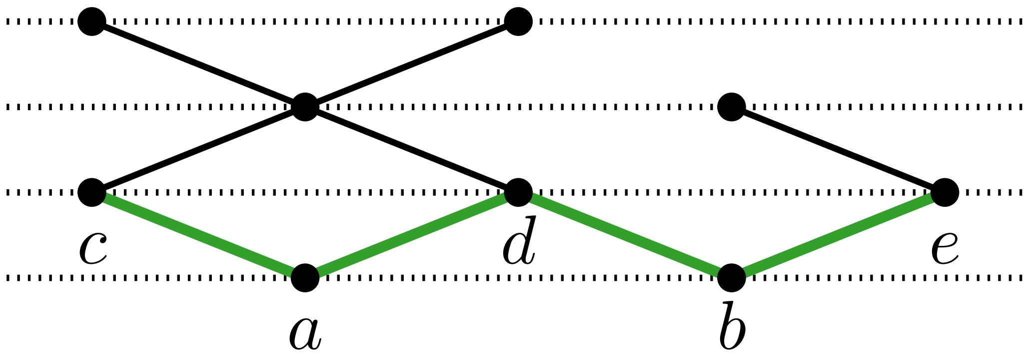
1-Page Queue Layouts

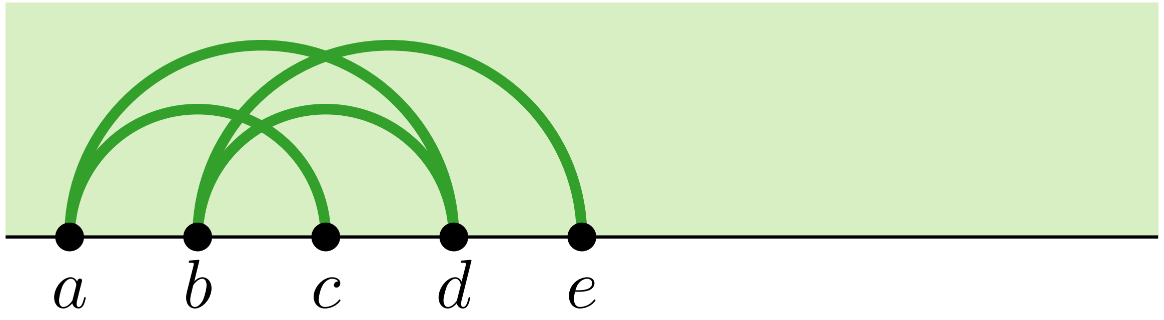
Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\text{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.





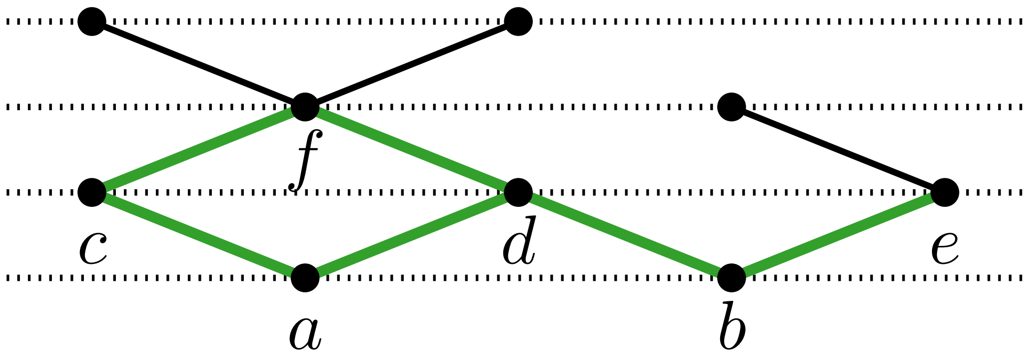
1-Page Queue Layouts

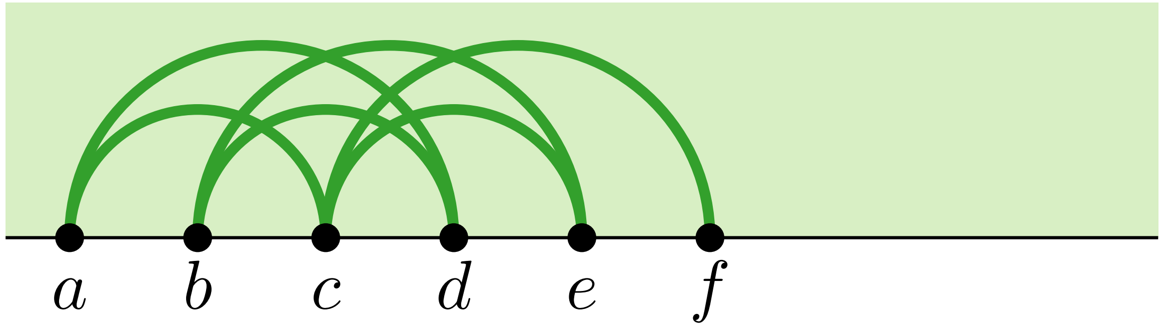
Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\mathrm{qn}(G) = 1$.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.





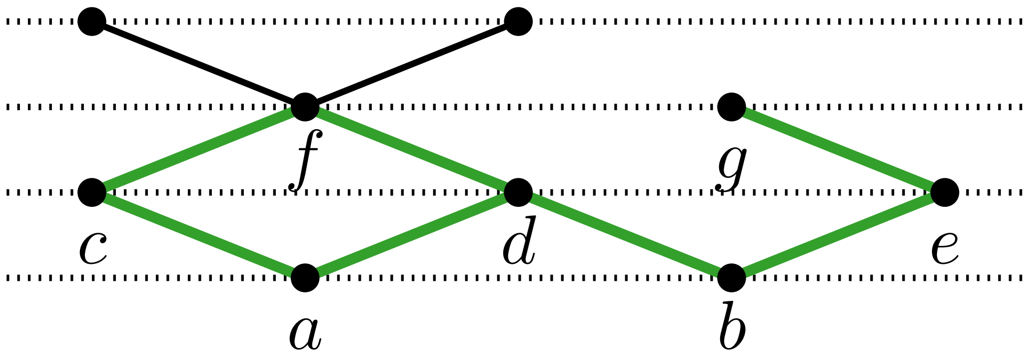
1-Page Queue Layouts

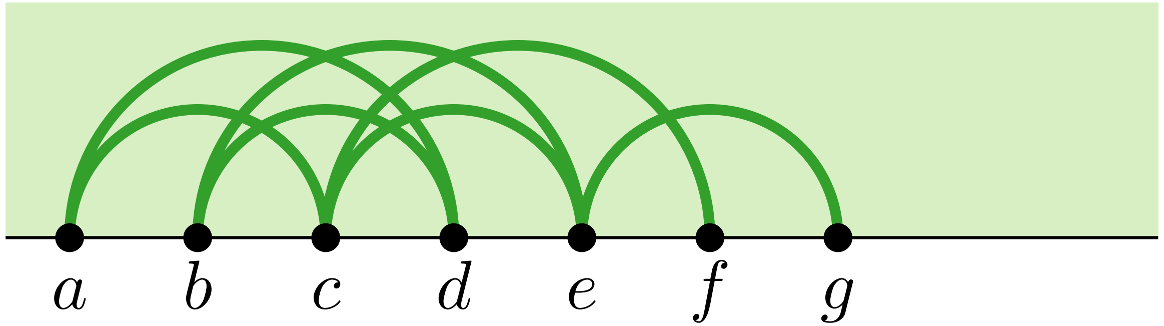
Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\text{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.





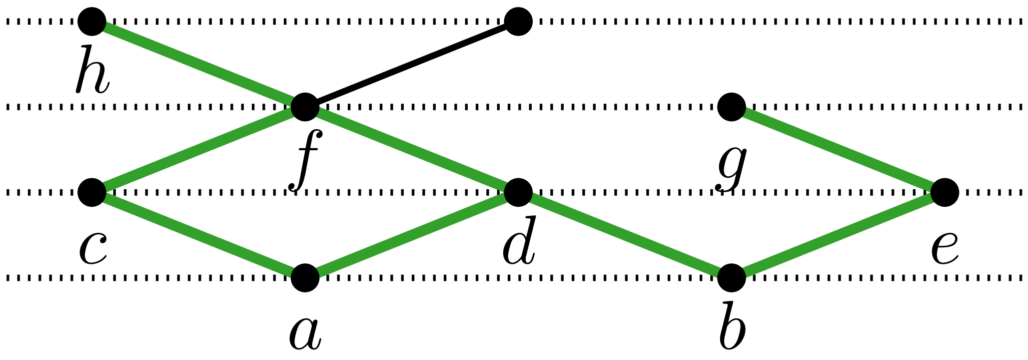
1-Page Queue Layouts

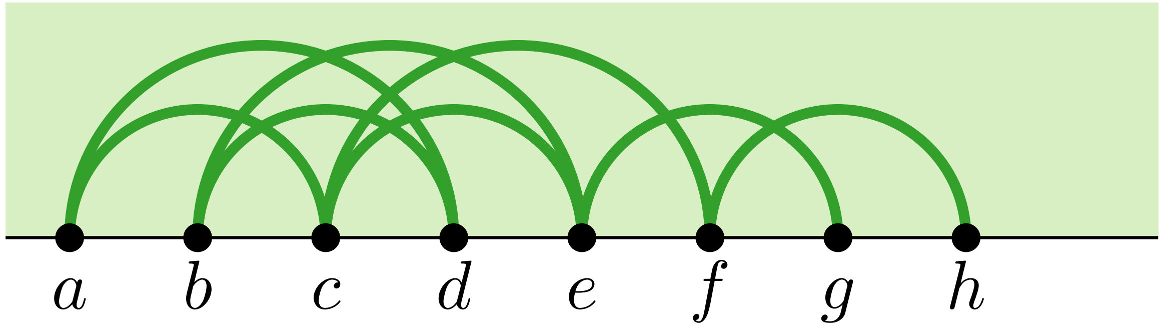
Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\mathrm{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.





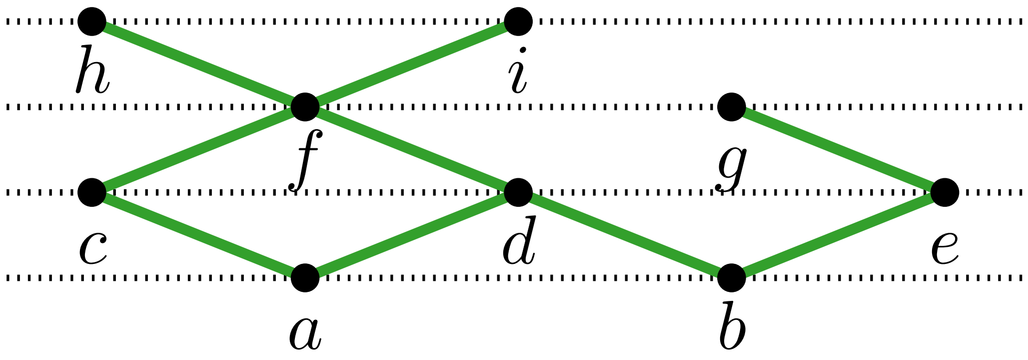
1-Page Queue Layouts

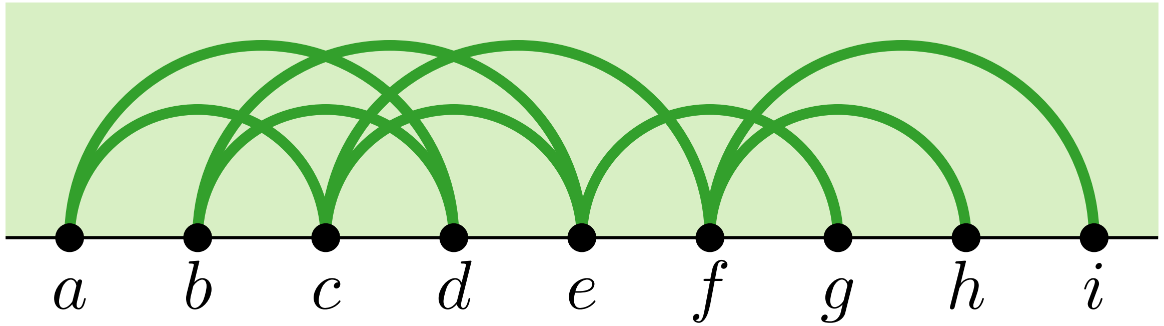
Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\text{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.





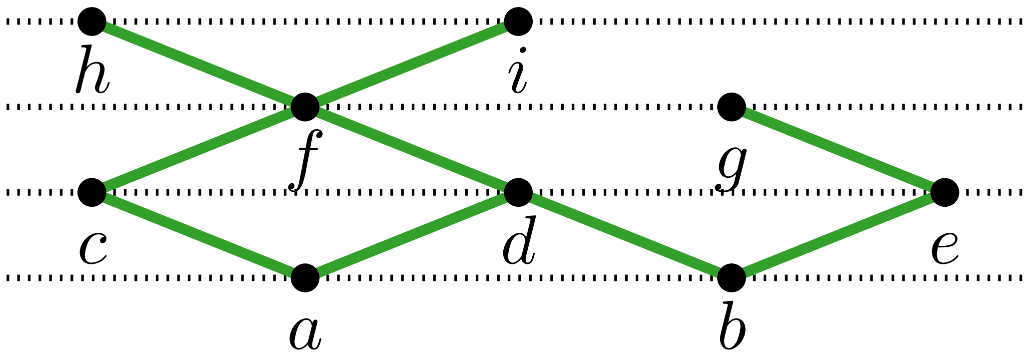
1-Page Queue Layouts

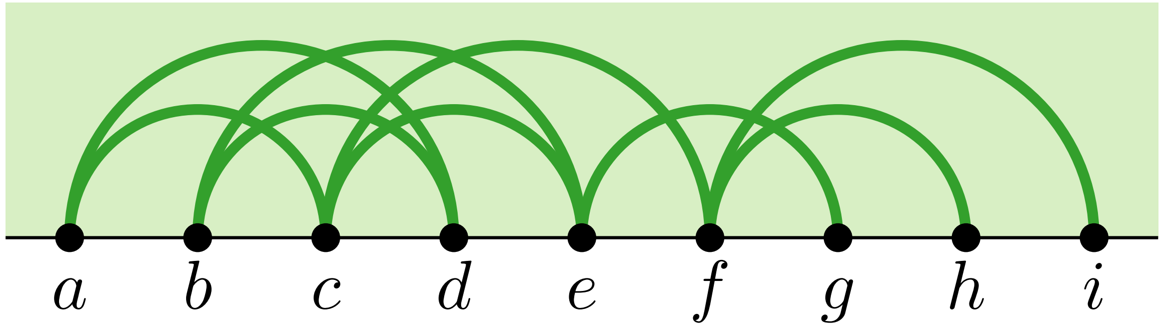
Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\mathrm{qn}(G) = 1$.

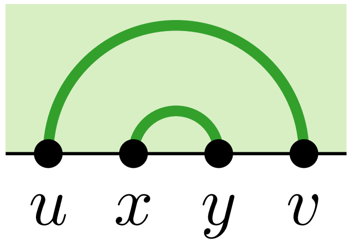
A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.
- If there was a nesting uv above xy,



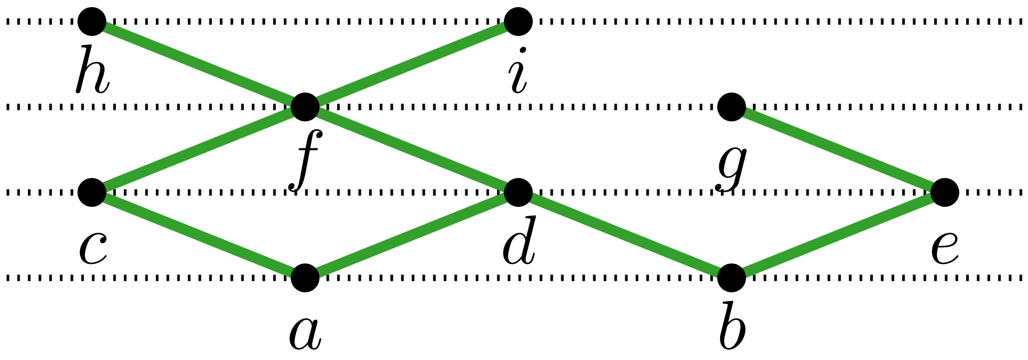


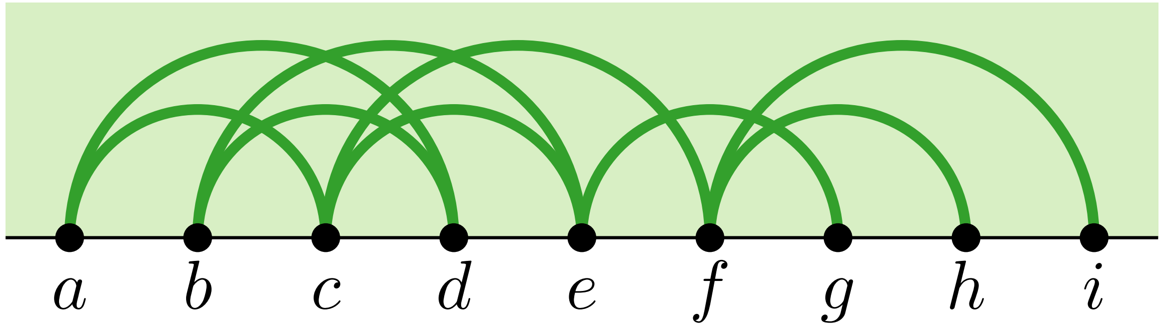


1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\mathrm{qn}(G) = 1$.

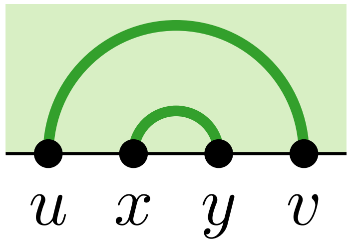
A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

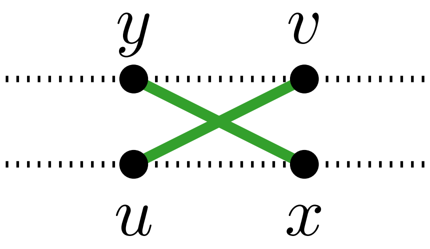




Proof.

- Take a leveled-planar drawing, order the vertices from bottom to top and left to right; this yields a queue layout.
- If there was a nesting uv above xy, u would be to the left of x on one level, and y would be to the left of v on the level above; this would not be planar. □

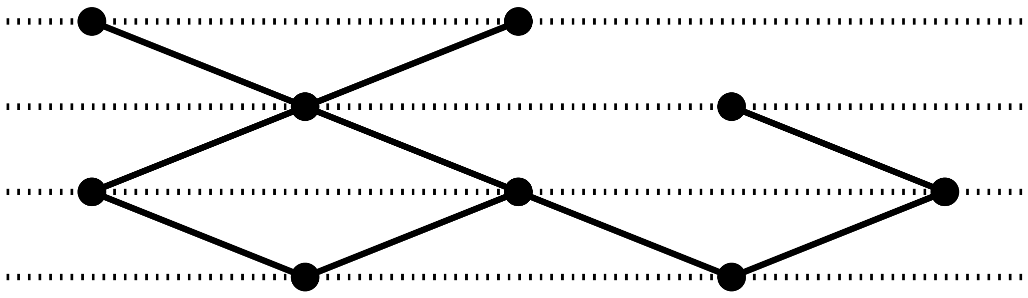




1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\text{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.

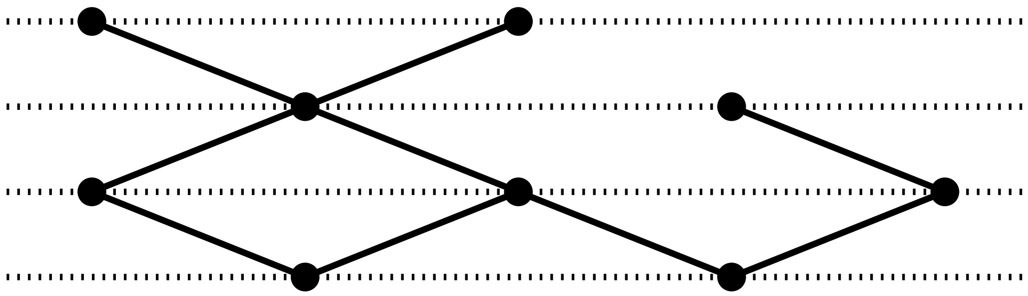


Theorem. [Heath & Rosenberg 1992]
For a graph G holds:
$\text{qn}(G) = 1 \Leftrightarrow G$ is arched leveled-planar.

1-Page Queue Layouts

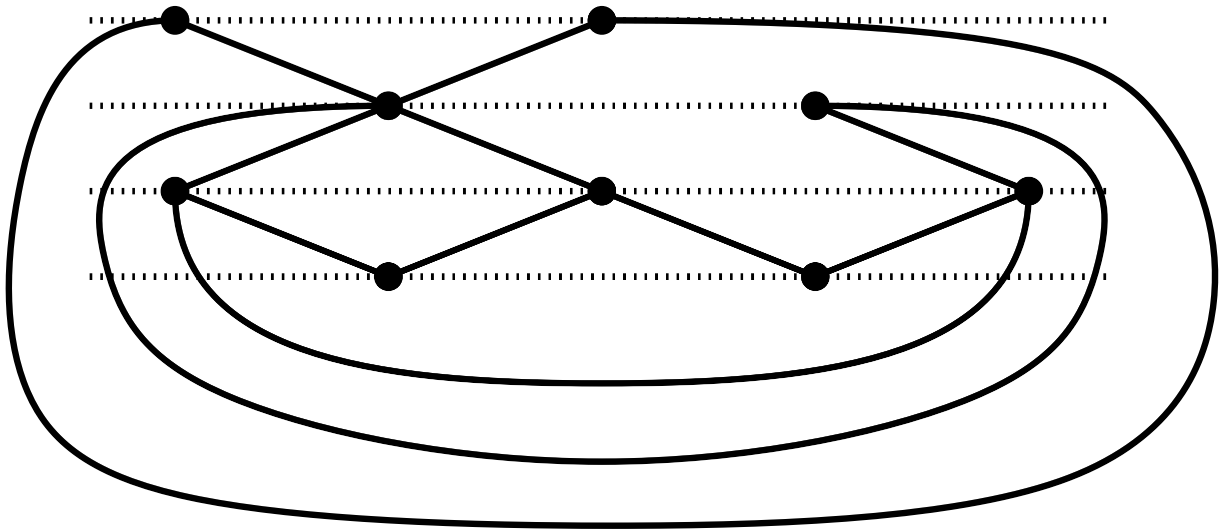
Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\text{qn}(G) = 1$.

A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.



Theorem. [Heath & Rosenberg 1992]
For a graph G holds:
$\text{qn}(G) = 1 \Leftrightarrow G$ is arched leveled-planar.

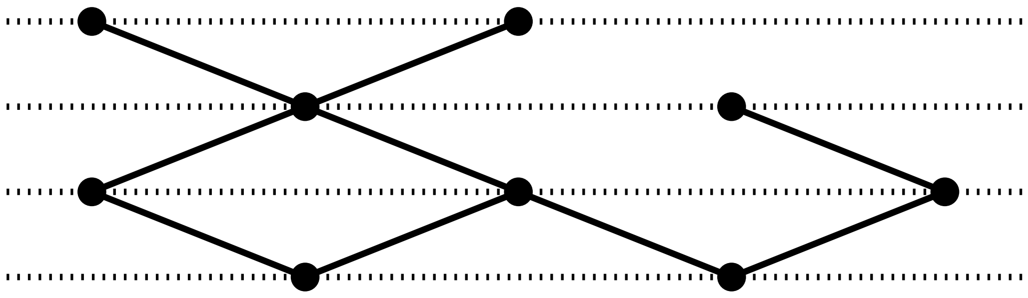
A graph is **arched leveled-planar** if it has a leveled-planar drawing where additionally vertices on the same level may be connected by edges that enclose all lower levels.



1-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992]
For every leveled-planar graph G, $\text{qn}(G) = 1$.

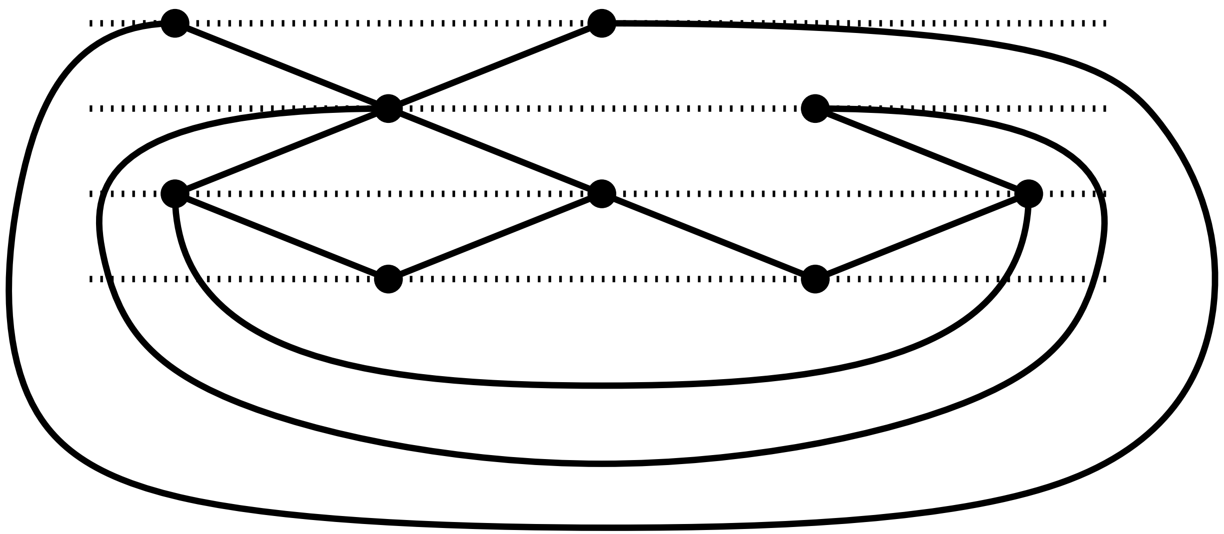
A graph is **leveled-planar** if it has a planar drawing where all vertices are arranged on horizontal lines (**levels**) and edges only connect vertices of adjacent levels.



Theorem. [Heath & Rosenberg 1992]
For a graph G holds:
$\text{qn}(G) = 1 \Leftrightarrow G$ is arched leveled-planar.

Proof. $\rightarrow$ *Exercise!*

A graph is **arched leveled-planar** if it has a leveled-planar drawing where additionally vertices on the same level may be connected by edges that enclose all lower levels.



2-Page and 3-Page Queue Layouts

Theorem. [Heath & Rosenberg 1992, Rengarajan & Veni Madhavan 1995.]

For every outerplanar graph G, $\mathrm{qn}(G) \leq 2$.

Theorem. [Rengarajan & Veni Madhavan 1996.]

For every series-parallel graph G, $\mathrm{qn}(G) \leq 3$.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, $\mathrm{qn}(G) \in \mathcal{O}(\sqrt{n})$.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\sqrt{n})$.

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that,
for every planar graph G, $\text{qn}(G) \leq C$.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, qn$(G) \in \mathcal{O}(\sqrt{n})$.

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that,
for every planar graph G, qn$(G) \leq C$.

Conjecture 2. [Pemmaraju 1992, Heath & Rosenberg 2011]
For $n \to \infty$, there are n-vertex planar graphs such that qn$(G) \to \infty$. (No bounding constant)

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42 log(n) $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that,
for every planar graph G, qn(G) $\leq C$.

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, qn(G) $\in \mathcal{O}(\sqrt{n})$.

Conjecture 2. [Pemmaraju 1992, Heath & Rosenberg 2011]
For $n \to \infty$, there are n-vertex planar graphs such that qn(G) $\to \infty$. (No bounding constant)

Theorem. [Di Battista, Frati & Pach 2013]
For every planar graph G, qn(G) $\in \mathcal{O}(\log^2 n)$.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that,
for every planar graph G, $\text{qn}(G) \leq C$.

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\sqrt{n})$.

Conjecture 2. [Pemmaraju 1992, Heath & Rosenberg 2011]
For $n \to \infty$, there are n-vertex planar graphs such that $\text{qn}(G) \to \infty$. (No bounding constant)

Theorem. [Di Battista, Frati & Pach 2013]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\log^2 n)$.

Theorem. [Dujmović 2015]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\log n)$.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2   3   4   7   9   42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that,
for every planar graph G, $\text{qn}(G) \leq C$.

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\sqrt{n})$.

~~**Conjecture 2.** [Pemmaraju 1992, Heath & Rosenberg 2011]
For $n \to \infty$, there are n-vertex planar graphs such that $\text{qn}(G) \to \infty$. (No bounding constant)~~

Theorem. [Di Battista, Frati & Pach 2013]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\log^2 n)$.

Theorem. [Dujmović 2015]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\log n)$.

Theorem. [Dujmović, Joret, Micek, Morin, Ueckerdt & Wood 2020]
For every planar graph G, $\text{qn}(G) \leq 49$.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\sqrt{n})$.

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that,
for every planar graph G, $\text{qn}(G) \leq C$.

~~**Conjecture 2.** [Pemmaraju 1992, Heath & Rosenberg 2011]
For $n \to \infty$, there are n-vertex planar graphs such that $\text{qn}(G) \to \infty$. (No bounding constant)~~

Theorem. [Di Battista, Frati & Pach 2013]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\log^2 n)$.

Theorem. [Dujmović 2015]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\log n)$.

Theorem. [Dujmović, Joret, Micek, Morin, Ueckerdt & Wood 2020]
For every planar graph G, $\text{qn}(G) \leq 49$.

Theorem. [Bekos, Gronemann & Raftopoulou 2021]
For every planar graph G, $\text{qn}(G) \leq 42$.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that,
for every planar graph G, $\text{qn}(G) \leq C$.

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\sqrt{n})$.

~~**Conjecture 2.** [Pemmaraju 1992, Heath & Rosenberg 2011]
For $n \to \infty$, there are n-vertex planar graphs such that $\text{qn}(G) \to \infty$. (No bounding constant)~~

Theorem. [Di Battista, Frati & Pach 2013]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\log^2 n)$.

Theorem. [Dujmović 2015]
For every planar graph G, $\text{qn}(G) \in \mathcal{O}(\log n)$.

Theorem. [Dujmović, Joret, Micek, Morin, Ueckerdt & Wood 2020]
For every planar graph G, $\text{qn}(G) \leq 49$.

Theorem. [Bekos, Gronemann & Raftopoulou 2021]
For every planar graph G, $\text{qn}(G) \leq 42$.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42* $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that,
for every planar graph G, $\mathrm{qn}(G) \leq C$.

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, $\mathrm{qn}(G) \in \mathcal{O}(\sqrt{n})$.

~~**Conjecture 2.** [Pemmaraju 1992, Heath & Rosenberg 2011]
For $n \to \infty$, there are n-vertex planar graphs such that $\mathrm{qn}(G) \to \infty$. (No bounding constant)~~

Theorem. [Di Battista, Frati & Pach 2013]
For every planar graph G, $\mathrm{qn}(G) \in \mathcal{O}(\log^2 n)$.

Theorem. [Dujmović 2015]
For every planar graph G, $\mathrm{qn}(G) \in \mathcal{O}(\log n)$.

Theorem. [Dujmović, Joret, Micek, Morin, Ueckerdt & Wood 2020]
For every planar graph G, $\mathrm{qn}(G) \leq 49$.

Theorem. [Bekos, Gronemann & Raftopoulou 2021]
For every planar graph G, $\mathrm{qn}(G) \leq 42$.

*This is just an upper bound; the tight bound may be lower.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42* $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, qn$(G) \in \mathcal{O}(\sqrt{n})$.

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that,
for every planar graph G, qn$(G) \leq C$.

~~**Conjecture 2.** [Pemmaraju 1992, Heath & Rosenberg 2011]
For $n \to \infty$, there are n-vertex planar graphs such that qn$(G) \to \infty$. (No bounding constant)~~

Theorem. [Di Battista, Frati & Pach 2013]
For every planar graph G, qn$(G) \in \mathcal{O}(\log^2 n)$.

Theorem. [Dujmović 2015]
For every planar graph G, qn$(G) \in \mathcal{O}(\log n)$.

Theorem. [Dujmović, Joret, Micek, Morin, Ueckerdt & Wood 2020]
For every planar graph G, qn$(G) \leq 49$.

Theorem. [Bekos, Gronemann & Raftopoulou 2021]
For every planar graph G, qn$(G) \leq 42$.

Theorem. [Alam, Bekos, Gronemann, Kaufmann & Pupyrev 2020]
There is a planar graph G with qn$(G) \geq 4$.

*This is just an upper bound; the tight bound may be lower.

Queue Layouts of Planar Graphs

We have seen planar graphs have **stack** number at most 4. What is the max. **queue** number?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Conjecture 1. [Heath, Leighton & Rosenberg 1991]
There is a constant C such that, for every planar graph G, qn$(G) \leq C$.

Theorem. [Heath, Leighton & Rosenberg 1991]
For every planar graph G, qn$(G) \in \mathcal{O}(\sqrt{n})$.

~~**Conjecture 2.** [Pemmaraju 1992, Heath & Rosenberg 2011]
For $n \to \infty$, there are n-vertex planar graphs such that qn$(G) \to \infty$. (No bounding constant)~~

Theorem. [Di Battista, Frati & Pach 2013]
For every planar graph G, qn$(G) \in \mathcal{O}(\log^2 n)$.

Theorem. [Dujmović 2015]
For every planar graph G, qn$(G) \in \mathcal{O}(\log n)$.

Theorem. [Dujmović, Joret, Micek, Morin, Ueckerdt & Wood 2020]
For every planar graph G, qn$(G) \leq 49$.

Theorem. [Bekos, Gronemann & Raftopoulou 2021]
For every planar graph G, qn$(G) \leq 42$.

Theorem. [Alam, Bekos, Gronemann, Kaufmann & Pupyrev 2020]
There is a planar graph G with qn$(G) \geq 4$.

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

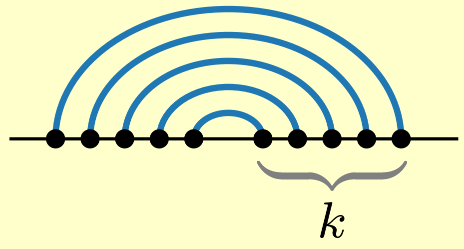
Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine.
Such a layout can be found in $\mathcal{O}(|E(G)| \log \log n)$ time.

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine.
Such a layout can be found in $\mathcal{O}(|E(G)| \log \log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.

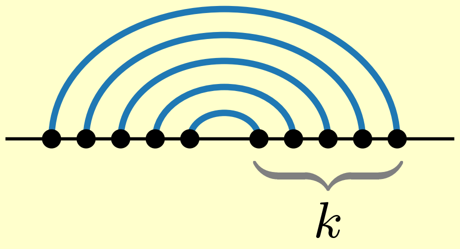


Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine. Such a layout can be found in $\mathcal{O}(|E(G)| \log \log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



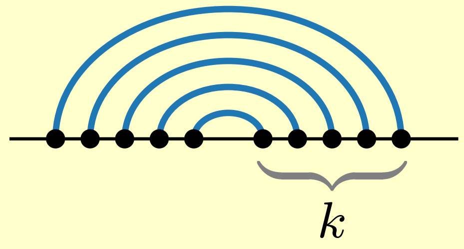
Proof Idea.

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine. Such a layout can be found in $\mathcal{O}(|E(G)| \log\log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Proof Idea.

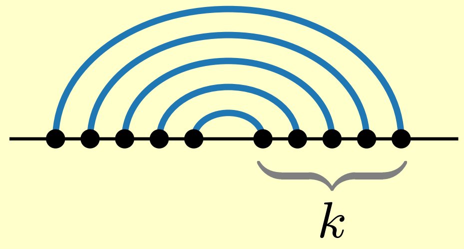
- For each edge e of G: if e is the outermost edge of an i-rainbow but of no $(i+1)$-rainbow, assign e to the i-th queue.

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine.
Such a layout can be found in $\mathcal{O}(|E(G)| \log \log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Proof Idea.

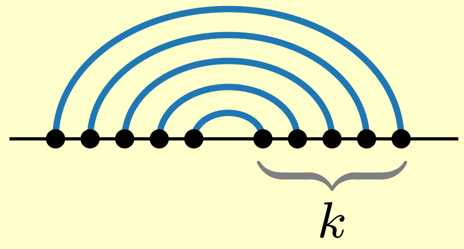
- For each edge e of G: if e is the outermost edge of an i-rainbow but of no $(i+1)$-rainbow, assign e to the i-th queue.
- Suppose in one queue, there is a nesting uv above xy.

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine. Such a layout can be found in $\mathcal{O}(|E(G)| \log\log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Proof Idea.

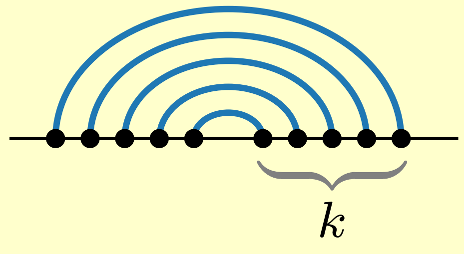
- For each edge e of G: if e is the outermost edge of an i-rainbow but of no $(i+1)$-rainbow, assign e to the i-th queue.
- Suppose in one queue, there is a nesting uv above xy.
- Both uv and xy are topmost edges of i-rainbows R_{uv} and R_{xy} but of no $(i+1)$-rainbows.

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine.
Such a layout can be found in $\mathcal{O}(|E(G)| \log\log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Proof Idea.

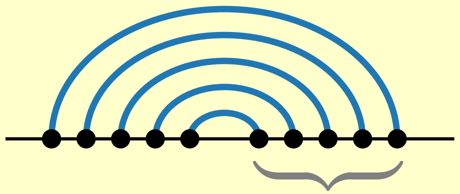
- For each edge e of G: if e is the outermost edge of an i-rainbow but of no $(i+1)$-rainbow, assign e to the i-th queue.
- Suppose in one queue, there is a nesting uv above xy.
- Both uv and xy are topmost edges of i-rainbows R_{uv} and R_{xy} but of no $(i+1)$-rainbows.
- Consider rainbow $R_{xy} \cup \{uv\}$.

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine.
Such a layout can be found in $\mathcal{O}(|E(G)| \log\log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Proof Idea.

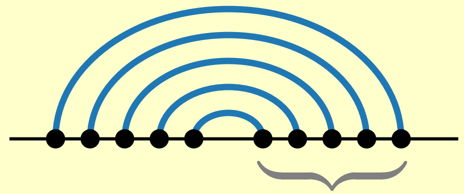
- For each edge e of G: if e is the outermost edge of an i-rainbow but of no $(i+1)$-rainbow, assign e to the i-th queue.
- Suppose in one queue, there is a nesting uv above xy.
- Both uv and xy are topmost edges of i-rainbows R_{uv} and R_{xy} but of no $(i+1)$-rainbows.
- Consider rainbow $R_{xy} \cup \{uv\}$. $\Rightarrow$ uv is the topmost edge of an $(i+1)$-rainbow. ↯

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine.
Such a layout can be found in $\mathcal{O}(|E(G)| \log \log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Proof Idea.

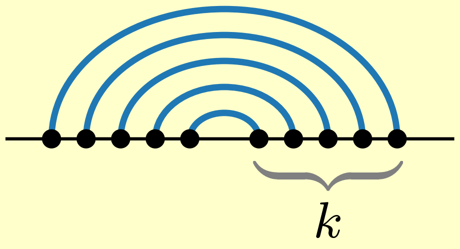
- For each edge e of G: if e is the outermost edge of an i-rainbow but of no $(i+1)$-rainbow, assign e to the i-th queue.
- Suppose in one queue, there is a nesting uv above xy.
- Both uv and xy are topmost edges of i-rainbows R_{uv} and R_{xy} but of no $(i+1)$-rainbows.
- Consider rainbow $R_{xy} \cup \{uv\}$. $\Rightarrow$ uv is the topmost edge of an $(i+1)$-rainbow. ↯
- For the running time, see the implementation described by [Heath & Rosenberg 1992]. □

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine. Such a layout can be found in $\mathcal{O}(|E(G)| \log \log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



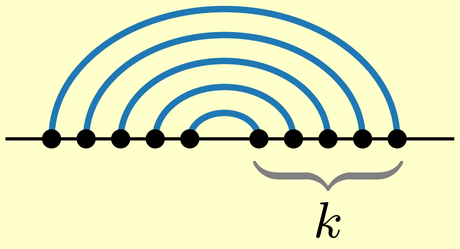
Is there a symmetric argument for k-twists in stack layouts (with fixed vertex order)?

Queue Layouts with Fixed Vertex Order

If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine.
Such a layout can be found in $\mathcal{O}(|E(G)| \log \log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



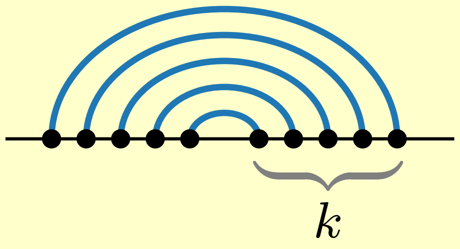
Is there a symmetric argument for k-twists in stack layouts (with fixed vertex order)? No!

Queue Layouts with Fixed Vertex Order

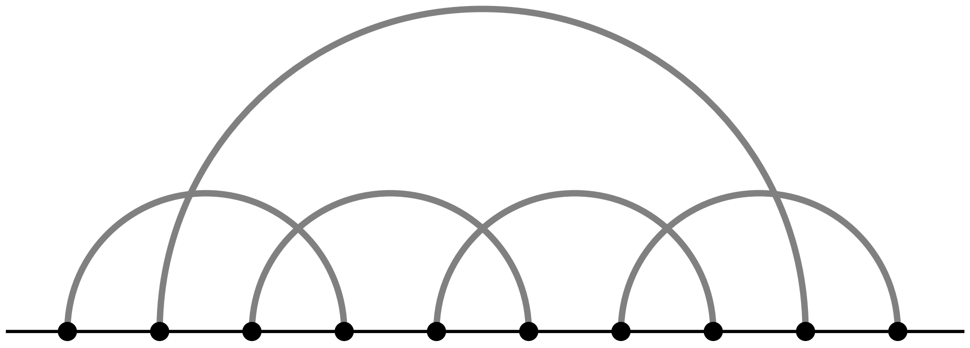
If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine.
Such a layout can be found in $\mathcal{O}(|E(G)| \log \log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Is there a symmetric argument for k-twists in stack layouts (with fixed vertex order)? No!



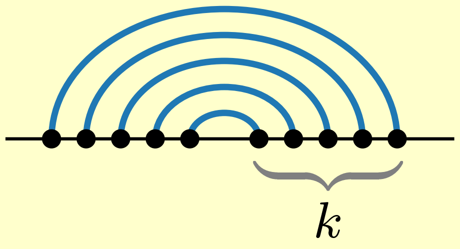
size of largest twist: 2

Queue Layouts with Fixed Vertex Order

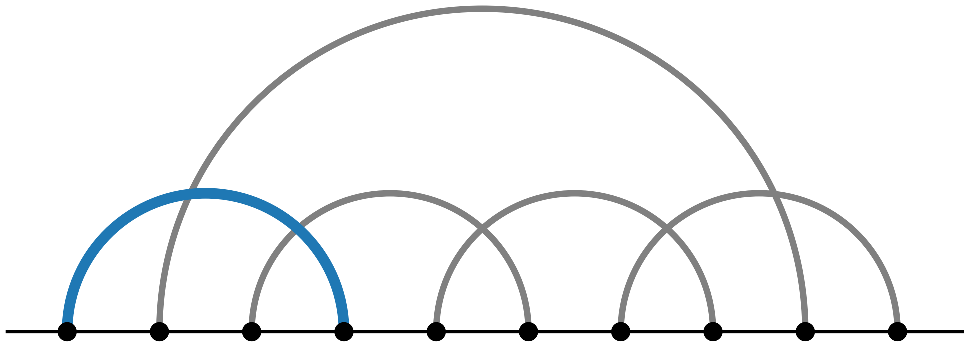
If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine. Such a layout can be found in $\mathcal{O}(|E(G)| \log\log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Is there a symmetric argument for k-twists in stack layouts (with fixed vertex order)? No!



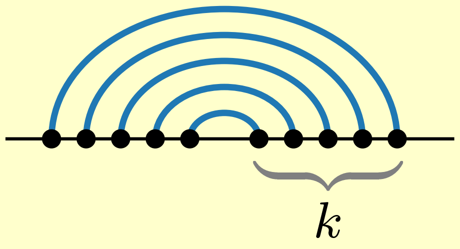
size of largest twist: 2

Queue Layouts with Fixed Vertex Order

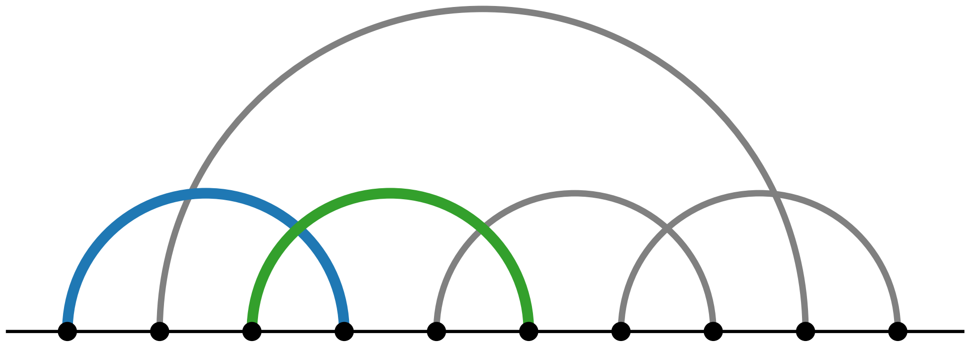
If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine. Such a layout can be found in $\mathcal{O}(|E(G)| \log\log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Is there a symmetric argument for k-twists in stack layouts (with fixed vertex order)? No!



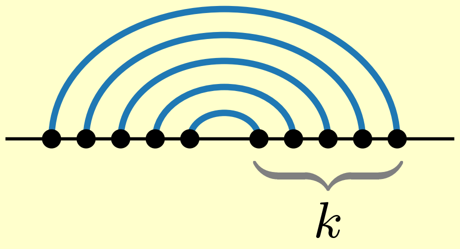
size of largest twist: 2

Queue Layouts with Fixed Vertex Order

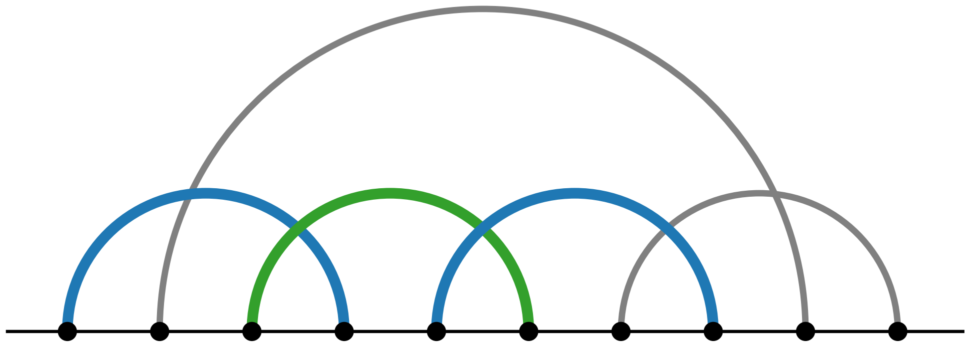
If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine. Such a layout can be found in $\mathcal{O}(|E(G)| \log\log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Is there a symmetric argument for k-twists in stack layouts (with fixed vertex order)? No!



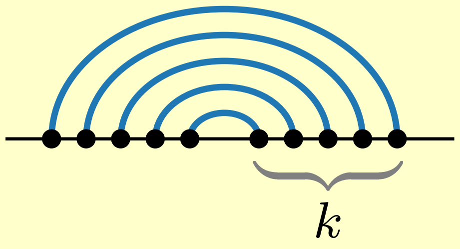
size of largest twist: 2

Queue Layouts with Fixed Vertex Order

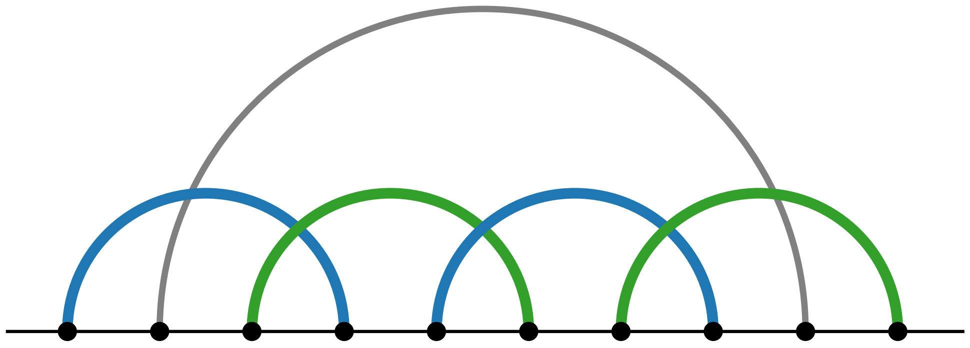
If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine. Such a layout can be found in $\mathcal{O}(|E(G)| \log\log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Is there a symmetric argument for k-twists in stack layouts (with fixed vertex order)? No!



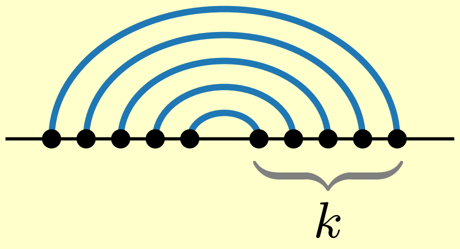
size of largest twist: 2

Queue Layouts with Fixed Vertex Order

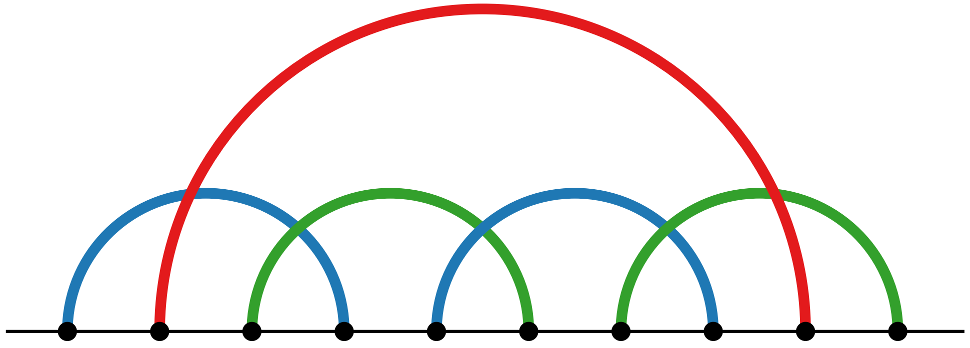
If we fix the order of the vertices on the spine, how many queues do we need?

Lemma 1. [Heath & Rosenberg 1992]
For a graph G, let an order $\prec$ of $V(G)$ be given. If k is the size of a largest rainbow in G under vertex order $\prec$, then there is a k-page queue layout of G with vertex order $\prec$ on the spine.
Such a layout can be found in $\mathcal{O}(|E(G)| \log \log n)$ time.

A **rainbow** of size k (or k-**rainbow**) is a set of k pairwise nesting edges.



Is there a symmetric argument for k-twists in stack layouts (with fixed vertex order)? No!



size of largest twist: 2

number of stacks needed: 3

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath & Rosenberg 1992]
For any $n \in \mathbb{N}$, $\text{qn}(K_n) = \lfloor n/2 \rfloor$.

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath & Rosenberg 1992]
For any $n \in \mathbb{N}$, $\text{qn}(K_n) = \lfloor n/2 \rfloor$.

Proof Sketch.

Assume that n is even (the case for odd n is similar).

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2   3   4   7   9   42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath & Rosenberg 1992]
For any $n \in \mathbb{N}$, $\mathrm{qn}(K_n) = \lfloor n/2 \rfloor$.

Proof Sketch.

Assume that n is even (the case for odd n is similar).

We first show that $\mathrm{qn}(K_n) \geq n/2$.

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath & Rosenberg 1992]
For any $n \in \mathbb{N}$, $\text{qn}(K_n) = \lfloor n/2 \rfloor$.

Proof Sketch.

Assume that n is even (the case for odd n is similar).

We first show that $\text{qn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \ldots, v_n$ accordingly.

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath & Rosenberg 1992]
For any $n \in \mathbb{N}$, $\text{qn}(K_n) = \lfloor n/2 \rfloor$.

Proof Sketch.

Assume that n is even (the case for odd n is similar).

We first show that $\text{qn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \ldots, v_n$ accordingly.
- Consider the set of edges $\{v_i v_{n+1-i} \mid i \in \{1, \ldots, n/2\}\}$.

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath & Rosenberg 1992]
For any $n \in \mathbb{N}$, $\text{qn}(K_n) = \lfloor n/2 \rfloor$.

Proof Sketch.

Assume that n is even (the case for odd n is similar).

We first show that $\text{qn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \ldots, v_n$ accordingly.
- Consider the set of edges $\{v_i v_{n+1-i} \mid i \in \{1, \ldots, n/2\}\}$.
- These are $n/2$ pairwise nesting edges (a $n/2$-rainbow). Each edge needs a separate queue.

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath & Rosenberg 1992]
For any $n \in \mathbb{N}$, $\text{qn}(K_n) = \lfloor n/2 \rfloor$.

Proof Sketch.

Assume that n is even (the case for odd n is similar).

We first show that $\text{qn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \dots, v_n$ accordingly.
- Consider the set of edges $\{v_i v_{n+1-i} \mid i \in \{1, \dots, n/2\}\}$.
- These are $n/2$ pairwise nesting edges (a $n/2$-rainbow). Each edge needs a separate queue.

Now we show that $\text{qn}(K_n) \leq n/2$.

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath & Rosenberg 1992]
For any $n \in \mathbb{N}$, $\text{qn}(K_n) = \lfloor n/2 \rfloor$.

Proof Sketch.

Assume that n is even (the case for odd n is similar).

We first show that $\text{qn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \ldots, v_n$ accordingly.
- Consider the set of edges $\{v_i v_{n+1-i} \mid i \in \{1, \ldots, n/2\}\}$.
- These are $n/2$ pairwise nesting edges (a $n/2$-rainbow). Each edge needs a separate queue.

Now we show that $\text{qn}(K_n) \leq n/2$.

- With n vertices, no rainbow can have size larger than $n/2$.

Queue Layouts of Complete Graphs

The stack number can be linear in n. What about the the queue number of K_n?

2 3 4 7 9 42 $\log(n)$ $\sqrt{n}$ $n^{2/3}$ $n/8$ $n/4$ $n/2$

Theorem. [Heath & Rosenberg 1992]
For any $n \in \mathbb{N}$, $\text{qn}(K_n) = \lfloor n/2 \rfloor$.

Proof Sketch.

Assume that n is even (the case for odd n is similar).

We first show that $\text{qn}(K_n) \geq n/2$.

- Consider any order $\prec$ of the vertices on the spine and name them $v_1, \dots, v_n$ accordingly.
- Consider the set of edges $\{v_i v_{n+1-i} \mid i \in \{1, \dots, n/2\}\}$.
- These are $n/2$ pairwise nesting edges (a $n/2$-rainbow). Each edge needs a separate queue.

Now we show that $\text{qn}(K_n) \leq n/2$.

- With n vertices, no rainbow can have size larger than $n/2$.
- Then, $\text{qn}(K_n) \leq n/2$ follows directly from Lemma 1. $\square$

Complexity of Determining the Stack Number

Complexity of Determining the Stack Number

Theorem. [Chung, Leighton & Rosenberg 1987]
Deciding whether a graph G has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 2$.

Complexity of Determining the Stack Number

Theorem. [Chung, Leighton & Rosenberg 1987]
Deciding whether a graph G has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 2$.

Proof Sketch.

Complexity of Determining the Stack Number

Theorem. [Chung, Leighton & Rosenberg 1987]
Deciding whether a graph G has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 2$.

Proof Sketch.

- If we have a k-page stack layout given, we can verify its correctness in polynomial time. This shows containment in NP.

Complexity of Determining the Stack Number

Theorem. [Chung, Leighton & Rosenberg 1987]
Deciding whether a graph G has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 2$.

Proof Sketch.

- If we have a k-page stack layout given, we can verify its correctness in polynomial time. This shows containment in NP.
- The problem of finding a Hamiltonian cycle in a planar graph is NP-complete.

Complexity of Determining the Stack Number

Theorem. [Chung, Leighton & Rosenberg 1987]
Deciding whether a graph G has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 2$.

Proof Sketch.

- If we have a k-page stack layout given, we can verify its correctness in polynomial time. This shows containment in NP.
- The problem of finding a Hamiltonian cycle in a planar graph is NP-complete.
- By the characterization of graphs with stack number 2, finding a Hamiltonian cycle in a planar graph is equivalent to deciding whether $\mathrm{sn}(G) \leq 2$. □

Complexity of Determining the Stack Number

Theorem. [Chung, Leighton & Rosenberg 1987]
Deciding whether a graph G has stack number sn$(G) \leq k$ is NP-complete for $k \geq 2$.

Proof Sketch.

- If we have a k-page stack layout given, we can verify its correctness in polynomial time. This shows containment in NP.
- The problem of finding a Hamiltonian cycle in a planar graph is NP-complete.
- By the characterization of graphs with stack number 2, finding a Hamiltonian cycle in a planar graph is equivalent to deciding whether sn$(G) \leq 2$. $\square$

The difficult part in the Hamiltonian-cycle problem is to find a permutation of the vertices.

Complexity of Determining the Stack Number

Theorem. [Chung, Leighton & Rosenberg 1987]
Deciding whether a graph G has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 2$.

Proof Sketch.

- If we have a k-page stack layout given, we can verify its correctness in polynomial time. This shows containment in NP.
- The problem of finding a Hamiltonian cycle in a planar graph is NP-complete.
- By the characterization of graphs with stack number 2, finding a Hamiltonian cycle in a planar graph is equivalent to deciding whether $\mathrm{sn}(G) \leq 2$. □

The difficult part in the Hamiltonian-cycle problem is to find a permutation of the vertices.

So, is determining the stack number easier if the order of the vertices on the spine is given?

Complexity of Determining the Stack Number

Theorem. [Unger 1988, Masuda, Nakajima, Kashiwabara & Fujisawa 1990]
Deciding whether a graph G given with an order of the vertices on the spine has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 4$.

Complexity of Determining the Stack Number

Theorem. [Unger 1988, Masuda, Nakajima, Kashiwabara & Fujisawa 1990]
Deciding whether a graph G given with an order of the vertices on the spine has stack number sn$(G) \leq k$ is NP-complete for $k \geq 4$.

Proof Sketch.

Complexity of Determining the Stack Number

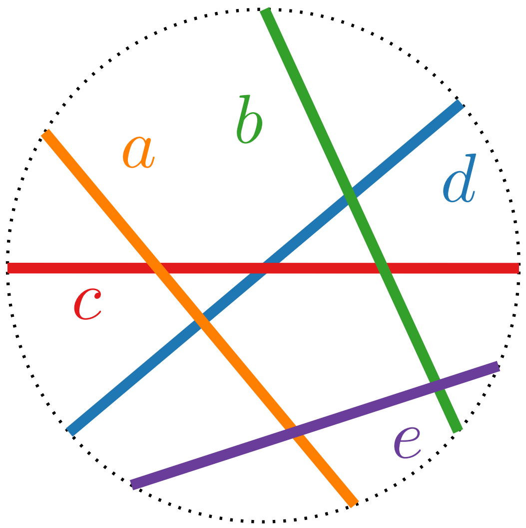
Theorem. [Unger 1988, Masuda, Nakajima, Kashiwabara & Fujisawa 1990]
Deciding whether a graph G given with an order of the vertices on the spine has stack number $\text{sn}(G) \leq k$ is NP-complete for $k \geq 4$.

Proof Sketch.

- An intersection graph of chords of a circle is called **circle graph**.

Complexity of Determining the Stack Number

Theorem. [Unger 1988, Masuda, Nakajima, Kashiwabara & Fujisawa 1990]
Deciding whether a graph G given with an order of the vertices on the spine has stack number sn$(G) \leq k$ is NP-complete for $k \geq 4$.



Proof Sketch.

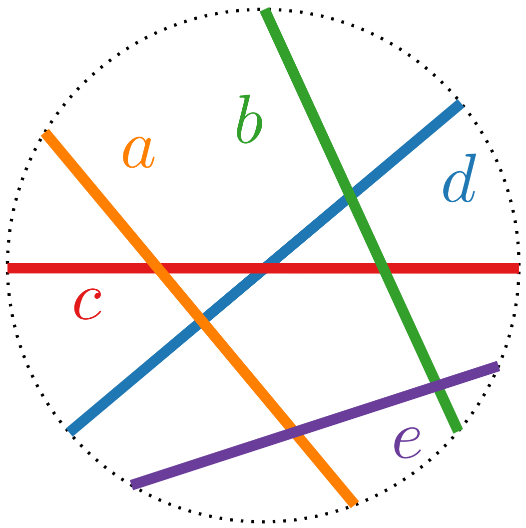
- An intersection graph of chords of a circle is called **circle graph**.

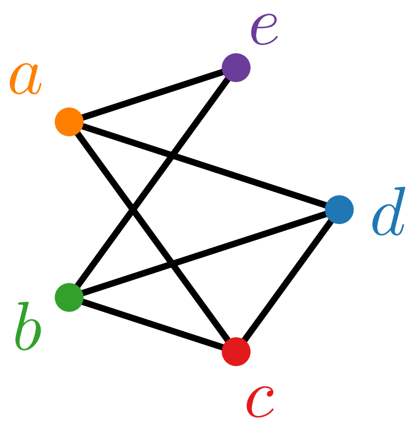
Complexity of Determining the Stack Number

Theorem. [Unger 1988, Masuda, Nakajima, Kashiwabara & Fujisawa 1990]
Deciding whether a graph G given with an order of the vertices on the spine has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 4$.

Proof Sketch.

- An intersection graph of chords of a circle is called **circle graph**.



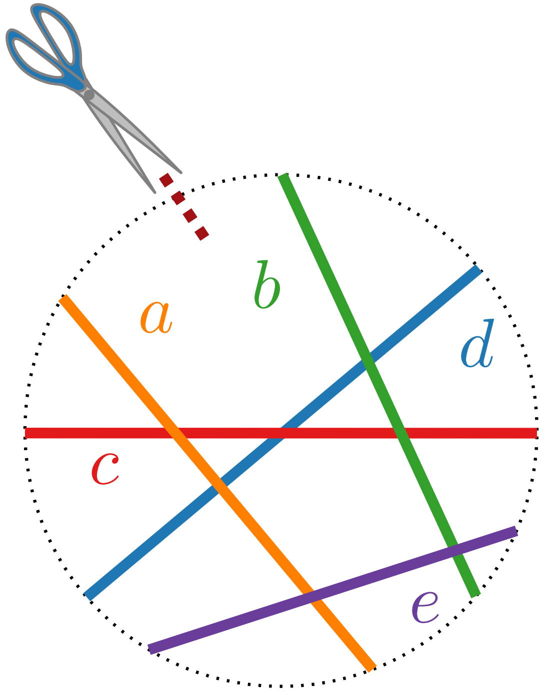


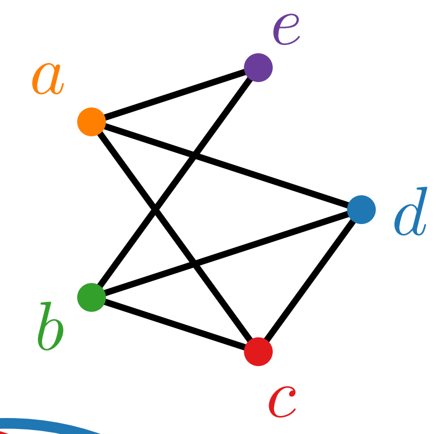
Complexity of Determining the Stack Number

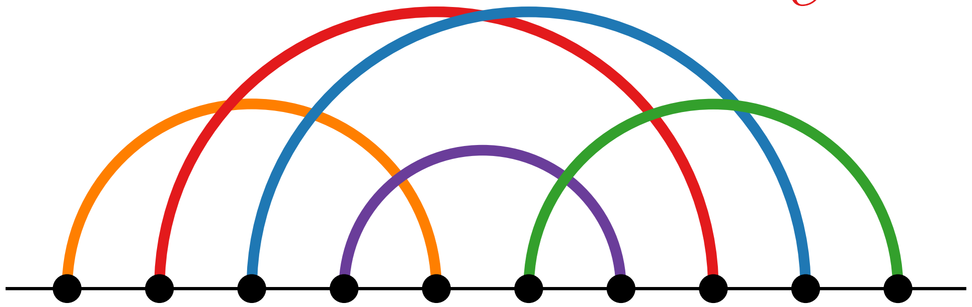
Theorem. [Unger 1988, Masuda, Nakajima, Kashiwabara & Fujisawa 1990]
Deciding whether a graph G given with an order of the vertices on the spine has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 4$.

Proof Sketch.

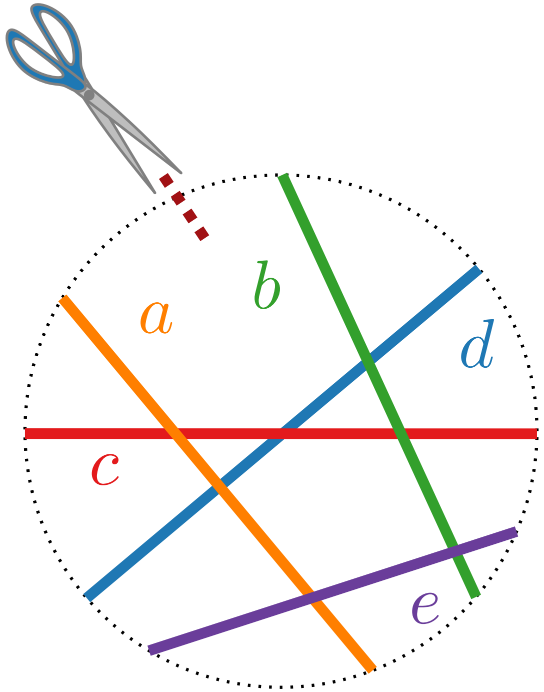
- An intersection graph of chords of a circle is called **circle graph**.
- The intersection representation of a circle graph can be seen as a linear layout (chords $\leftrightarrow$ edges, endpoints on the circle $\leftrightarrow$ vertices).







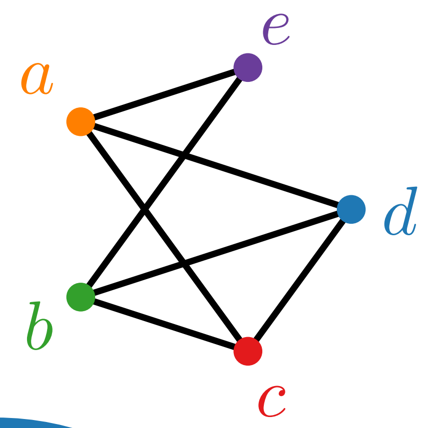
Complexity of Determining the Stack Number

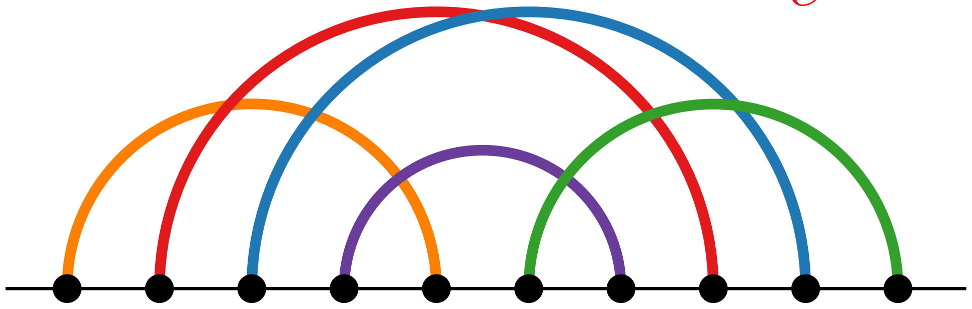


Theorem. [Unger 1988, Masuda, Nakajima, Kashiwabara & Fujisawa 1990]
Deciding whether a graph G given with an order of the vertices on the spine has stack number $\text{sn}(G) \leq k$ is NP-complete for $k \geq 4$.

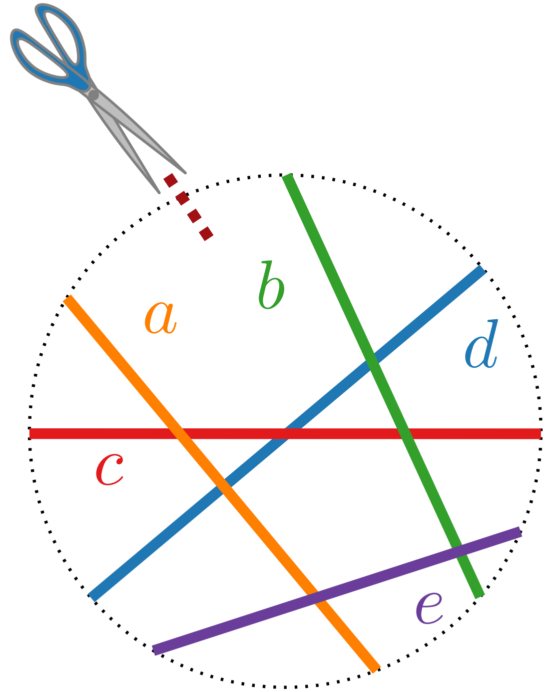
Proof Sketch.

- An intersection graph of chords of a circle is called **circle graph**.
- The intersection representation of a circle graph can be seen as a linear layout (chords $\leftrightarrow$ edges, endpoints on the circle $\leftrightarrow$ vertices).
- The circle graph is the conflict graph for the stack assignment (two edge can go to the same stack if and only if they don't share an edge in the circle graph).





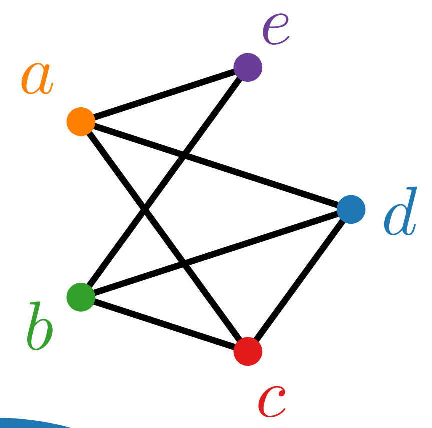
Complexity of Determining the Stack Number

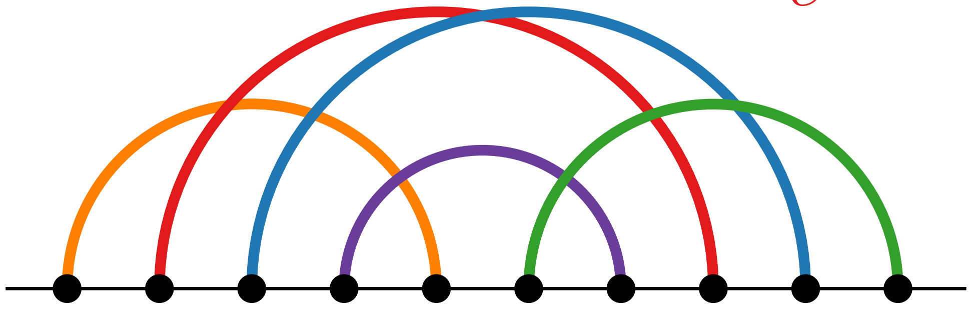


Theorem. [Unger 1988, Masuda, Nakajima, Kashiwabara & Fujisawa 1990]
Deciding whether a graph G given with an order of the vertices on the spine has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 4$.

Proof Sketch.

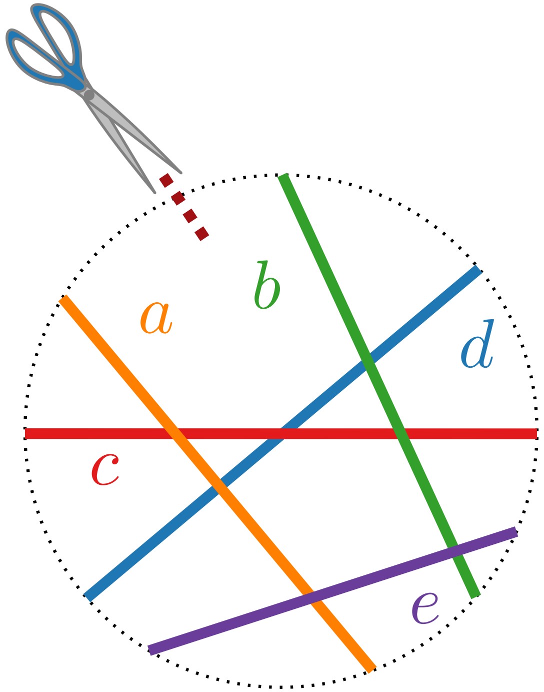
- An intersection graph of chords of a circle is called **circle graph**.
- The intersection representation of a circle graph can be seen as a linear layout (chords $\leftrightarrow$ edges, endpoints on the circle $\leftrightarrow$ vertices).
- The circle graph is the conflict graph for the stack assignment (two edge can go to the same stack if and only if they don't share an edge in the circle graph).
- Coloring the circle graph with k colors is equivalent to assigning the edges to k stacks.





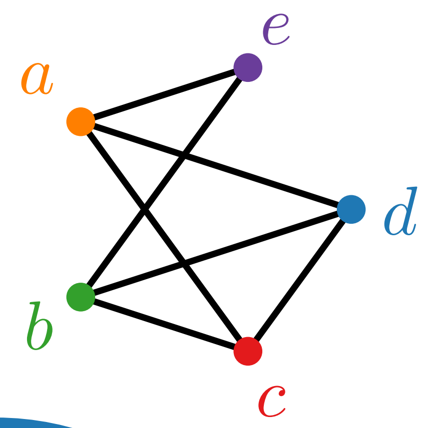
Complexity of Determining the Stack Number

Theorem. [Unger 1988, Masuda, Nakajima, Kashiwabara & Fujisawa 1990]
Deciding whether a graph G given with an order of the vertices on the spine has stack number $\mathrm{sn}(G) \leq k$ is NP-complete for $k \geq 4$.

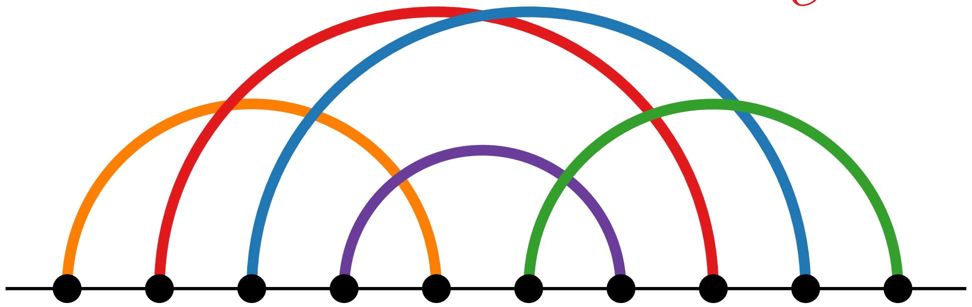


Proof Sketch.

- An intersection graph of chords of a circle is called **circle graph**.
- The intersection representation of a circle graph can be seen as a linear layout (chords $\leftrightarrow$ edges, endpoints on the circle $\leftrightarrow$ vertices).



- The circle graph is the conflict graph for the stack assignment (two edge can go to the same stack if and only if they don't share an edge in the circle graph).
- Coloring the circle graph with k colors is equivalent to assigning the edges to k stacks.



- Coloring circle graphs is NP-complete for $k \geq 4$ colors. □

Complexity of Determining the Queue Number

Complexity of Determining the Queue Number

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G has queue number $\mathrm{qn}(G) \leq k$ is NP-complete for $k \geq 1$.

Complexity of Determining the Queue Number

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G has queue number $\text{qn}(G) \leq k$ is NP-complete for $k \geq 1$.

Proof Sketch.

Complexity of Determining the Queue Number

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G has queue number $\text{qn}(G) \leq k$ is NP-complete for $k \geq 1$.

Proof Sketch.

- Deciding whether a given graph is arched leveled-planar is NP-complete.
  Hence, deciding whether $\text{qn}(G) = 1$ is NP-hard. $\square$

Complexity of Determining the Queue Number

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G has queue number $\text{qn}(G) \leq k$ is NP-complete for $k \geq 1$.

Proof Sketch.

- Deciding whether a given graph is arched leveled-planar is NP-complete.
  Hence, deciding whether $\text{qn}(G) = 1$ is NP-hard. □

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G given with an order of the vertices on the spine has queue number $\text{qn}(G) \leq k$ is

Complexity of Determining the Queue Number

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G has queue number $\text{qn}(G) \leq k$ is NP-complete for $k \geq 1$.

Proof Sketch.

- Deciding whether a given graph is arched leveled-planar is NP-complete.
  Hence, deciding whether $\text{qn}(G) = 1$ is NP-hard. □

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G given with an order of the vertices on the spine has queue number $\text{qn}(G) \leq k$ is polynomial-time solvable.

Complexity of Determining the Queue Number

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G has queue number $\mathrm{qn}(G) \leq k$ is NP-complete for $k \geq 1$.

Proof Sketch.

- Deciding whether a given graph is arched leveled-planar is NP-complete.
  Hence, deciding whether $\mathrm{qn}(G) = 1$ is NP-hard. □

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G given with an order of the vertices on the spine has queue number $\mathrm{qn}(G) \leq k$ is polynomial-time solvable.

Proof Sketch.

Complexity of Determining the Queue Number

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G has queue number $\mathrm{qn}(G) \leq k$ is NP-complete for $k \geq 1$.

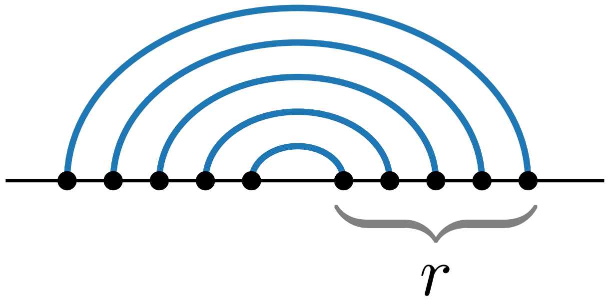
Proof Sketch.

- Deciding whether a given graph is arched leveled-planar is NP-complete.
  Hence, deciding whether $\mathrm{qn}(G) = 1$ is NP-hard. □

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G given with an order of the vertices on the spine has queue number $\mathrm{qn}(G) \leq k$ is polynomial-time solvable.

Proof Sketch.

- Determine the size r of the largest rainbow in polynomial time.



Complexity of Determining the Queue Number

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G has queue number $\mathrm{qn}(G) \leq k$ is NP-complete for $k \geq 1$.

Proof Sketch.

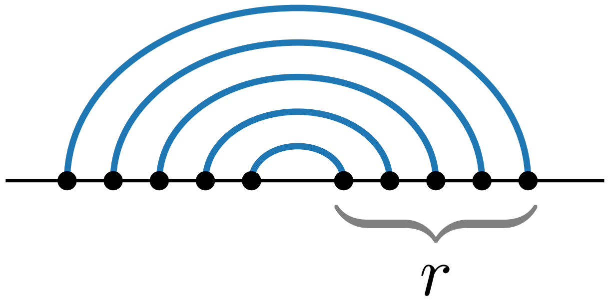
- Deciding whether a given graph is arched leveled-planar is NP-complete.
  Hence, deciding whether $\mathrm{qn}(G) = 1$ is NP-hard. $\square$

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G given with an order of the vertices on the spine has queue number $\mathrm{qn}(G) \leq k$ is polynomial-time solvable.

Proof Sketch.

Details in exercise!

- Determine the size r of the largest rainbow in polynomial time.



Complexity of Determining the Queue Number

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G has queue number $\text{qn}(G) \leq k$ is NP-complete for $k \geq 1$.

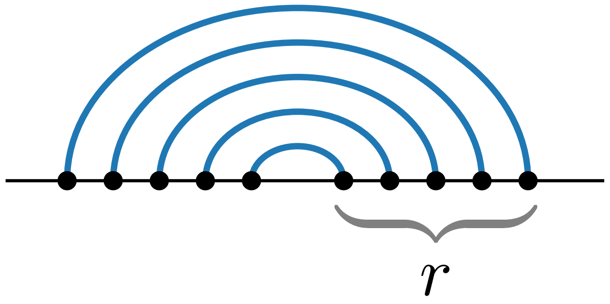
Proof Sketch.

- Deciding whether a given graph is arched leveled-planar is NP-complete.
  Hence, deciding whether $\text{qn}(G) = 1$ is NP-hard. □

Theorem. [Heath & Rosenberg 1992]
Deciding whether a graph G given with an order of the vertices on the spine has queue number $\text{qn}(G) \leq k$ is polynomial-time solvable.

Proof Sketch.

Details in exercise!



- Determine the size r of the largest rainbow in polynomial time.
- If $r \leq k$, then there is k-page queue layout due to Lemma 1. □

Discussion

- There are surprisingly many applications of stack and queue layouts, e.g., in computational biology (RNA folding), VLSI design, traffic control, . . .

Discussion

- There are surprisingly many applications of stack and queue layouts, e.g., in computational biology (RNA folding), VLSI design, traffic control, ...
- By the book-embedding paradigm, **page number** and **book thickness** are alternative terms for *stack number*.

Discussion

- There are surprisingly many applications of stack and queue layouts, e.g., in computational biology (RNA folding), VLSI design, traffic control, . . .
- By the book-embedding paradigm, **page number** and **book thickness** are alternative terms for *stack number*.
- There are many more variants, e.g., for fixed vertex order, directed graphs, using other data structures, . . .

Literature

Sources for the overview:

- [Ueckerdt 2022] Invited Talk on WG 2022: *Stack and queue layouts of planar graphs.*
- [Pupyrev 2024] Website on Linear Layouts:
  `https://spupyrev.github.io/linearlayouts.html`

Some of the referenced papers:

- [Bernhart & Kainen 1979] *The book thickness of a graph.*
- [Yannakakis 1986] *Embedding planar graphs in four pages.*
- [Heath & Rosenberg 1992] *Laying out graphs using queues.*
- [Bekos, Kaufmann, Klute, Pupyrev, Raftopoulou & Ueckerdt 2020]
  Four pages are indeed necessary for planar graphs.
- [Dujmović, Joret, Micek, Morin, Ueckerdt & Wood 2020]
  Planar graphs have bounded queue-number.
- [Bekos, Gronemann & Raftopoulou 2021]
  An improved upper bound on the queue number of planar graphs.