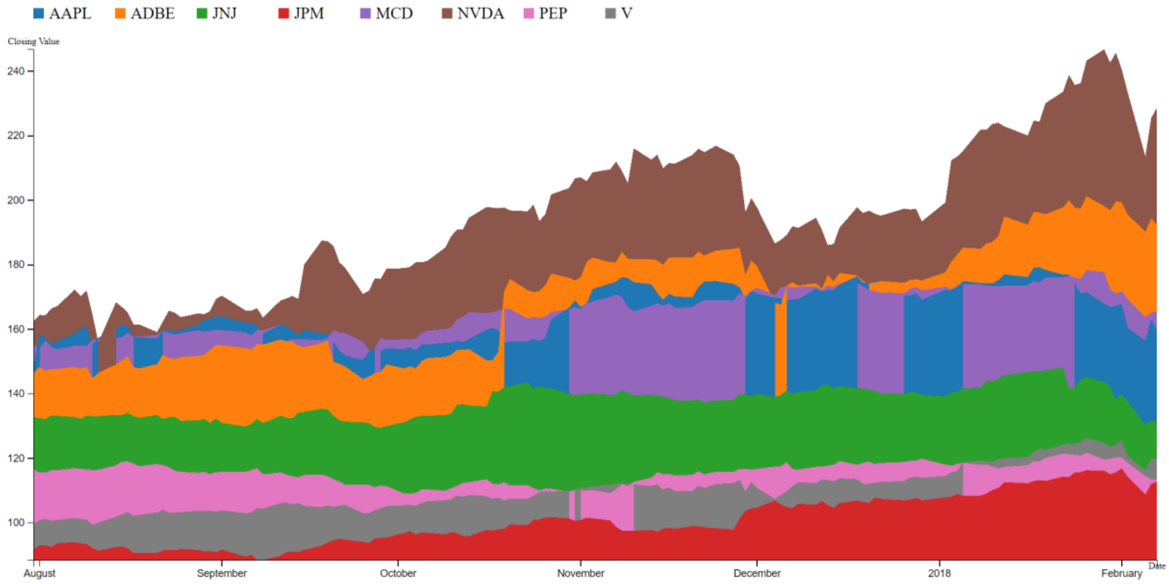
Exercise 10: Braided Graph (20 points)

Due: 17.07.2023 8AM

Contributor 1: Anurag Pacholi
Contributor 2: Mohammad Nomaan Husain

Task 1: (12 points)

Braided Graph

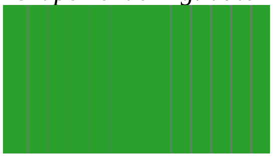


For this exercise, your task is to create a braided graph of the appended *stonks.js* dataset as shown in the figure above. You may use your own previously written code as orientation.

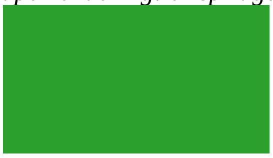
Mind the following hints as orientation:
When building the visualization, it makes sense to group the data by its days. For each day, create a *g* element. Draw a respective *path* element between the current and the next timestamp for each stock within each created *g*. Think of plotting the higher value first to make the smaller elements visible.

Additional Hint: To avoid vertical rendering lines between the paths, set the css attribute *shape-rendering*. (*https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/shape-rendering*)

shape-rendering: auto



shape-rendering: crispEdges



Task 2: (8 points)

2a) Name and explain two advantages of node-link diagrams for representing relational data.

1. Node-link diagrams allow for exploratory analysis and discovery of hidden patterns or insights within the relational data.
2. Node-link diagrams can also be enhanced with interactive features, such as tooltips or animations, to provide additional details or reveal dynamic aspects of the relationships.

2b) Explain the overplotting problem in node-link diagrams. Illustrate with an example.

The overplotting problem in node-link diagrams occurs when multiple nodes or links overlap or cluster together, making it difficult to distinguish individual elements or perceive the underlying relationships.
Example:
Imagine a node-link diagram representing a railway's network, where nodes represent stops/stations and links represent their connections. Each node is labeled with the stops' names, and the links indicate their relationships. As the railway network grows and more connections are added, the diagram becomes crowded.

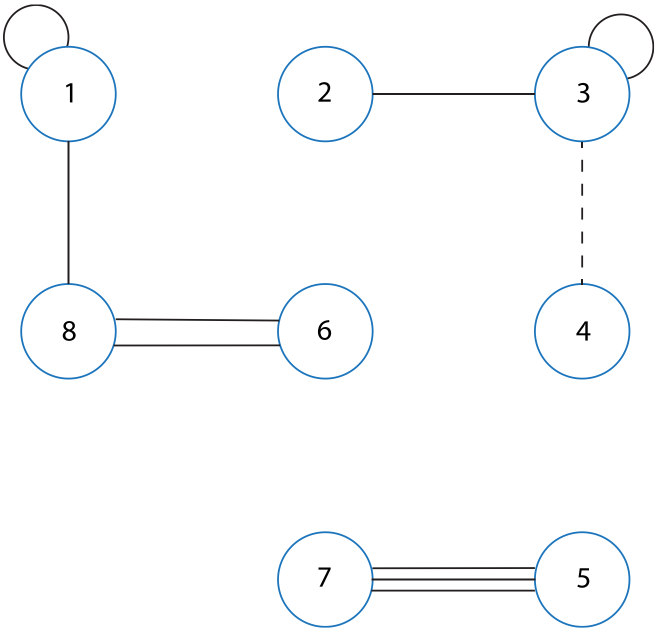
Hint: For the illustration, can include an image based on your own past work or from a third-party source (remember to add a reference, e.g., a URL when taking an example from a different source).

2c) Give an example of how you can use a node-link diagram to show the hierarchical relationship in data. Name two disadvantages associated with your example.

An example of using a node-link diagram to show hierarchical relationships in data is to represent the taxonomy or classification of species in biology where each node represents a species, and the links depict the hierarchical relationships between species at different taxonomic levels, such as kingdom, phylum, class, order, family, genus, and species. The diagram showcases the branching structure of the taxonomic hierarchy.
Disadvantages associated with this example of a hierarchical node-link diagram are:

1. Limited Screen Space: When representing a comprehensive taxonomy with numerous species and taxonomic levels, the diagram can become very large and may require significant screen space to display properly.
2. Information Overload: Taxonomy encompasses a vast number of species and taxonomic levels, and representing them all in a single node-link diagram may result in information overload.

2d) Define: "adjacency matrix" in your own words. Draw the adjacency matrix of the following graph:



An adjacency matrix is a square matrix representation of a graph or network, where each cell in the matrix represents the connection or relationship between two vertices or nodes.

	1	2	3	4	5	6	7	8
1	1							1
2			1					
3		1	1					
4								
5							3	
6								2
7					3			
8	1					2		

Hint: You can include a photo of your hand-drawn solution. Please make sure, that it’s easily readable.

Submission: Zipped folder including all necessary files to display the visualizations. Please state both your names on the submission.