♠ > Gallery > Graph > Minimum Spanning Tree



Go to the end to download the full example code

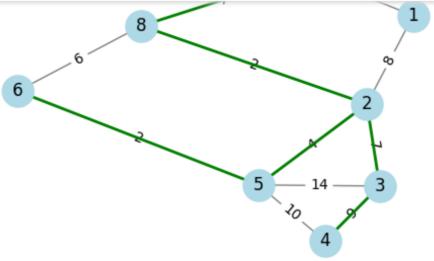
Minimum Spanning Tree

A minimum spanning tree (MST) is a subset of edges in a weighted, connected graph that connects all vertices together with the minimum possible total edge weight. The minimum spanning tree function is used to compare the original graph with its MST.









```
import networkx as nx
import matplotlib.pyplot as plt

# Create a graph
G = nx.Graph()
G.add_edges_from(
    [
            (0, 1, {"weight": 4}),
            (0, 7, {"weight": 8}),
            (1, 7, {"weight": 11}),
```

Skip to main content

```
(2, 5, {"weight": 4}),
        (2, 3, {"weight": 7}),
        (3, 4, {"weight": 9}),
        (3, 5, {"weight": 14}),
        (4, 5, {"weight": 10}),
        (5, 6, {"weight": 2}),
        (6, 8, {"weight": 6}),
        (7, 8, {"weight": 7}),
# Find the minimum spanning tree
T = nx.minimum spanning tree(G)
# Visualize the graph and the minimum spanning tree
pos = nx.spring layout(G)
nx.draw networkx nodes(G, pos, node color="lightblue", node size=500)
nx.draw networkx edges(G, pos, edge color="grey")
nx.draw networkx labels(G, pos, font size=12, font family="sans-serif")
nx.draw networkx edge labels(
    G, pos, edge labels={(u, v): d["weight"] for u, v, d in G.edges(data=True)}
nx.draw networkx edges(T, pos, edge color="green", width=2)
```

© Copyright 2004-2023, NetworkX Developers.

Created using Sphinx 6.1.3.

Built with the PyData Sphinx Theme 0.13.3.