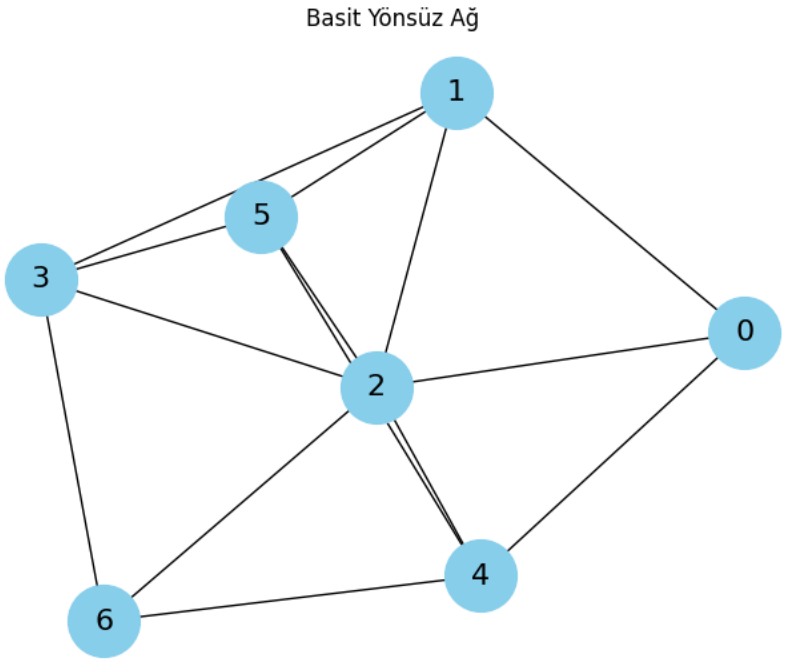
Karmaşık Ağlar

DR.ÖĞR.ÜYESİ EMRAH ÖZKAYNAK

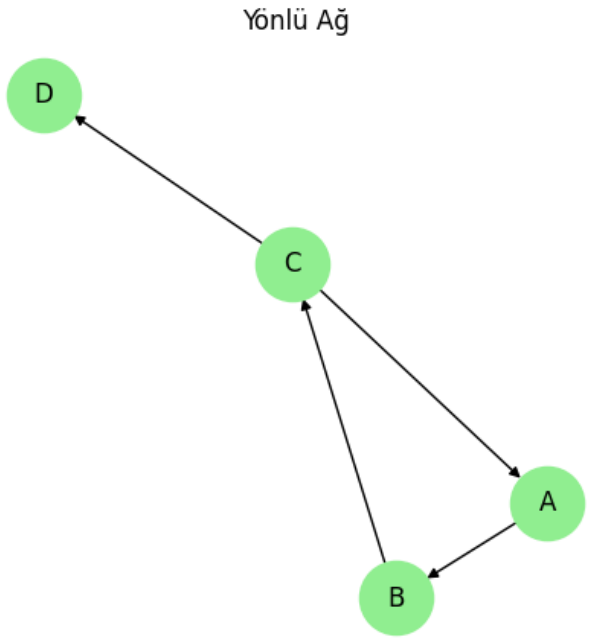
Karmaşık Ağlar




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

# Boş bir yönsüz ağ oluştur
G = nx.Graph()

# Düğümleri ekle
G.add_nodes_from(["0", "1", "2", "3","4", "5", "6"])


# Kenarları ekle
G.add_edges_from([("0", "1"), ("0", "2"), ("0", "4"),
                  ("1", "2"),("1", "3"),("1", "5"),
                  ("2", "3"),("2", "4"),("2", "5"),("2", "6"),
                  ("3", "5"),("3", "6"),
                  ("4", "5"),("4", "6")])


# Basit Bir Yönsüz Ağ
nx.draw(G, with_labels=True, node_color='skyblue', node_size=1500, font_size=16)
plt.title("Basit Yönsüz Ağ")
plt.show()
print(G.degree("3"))
```

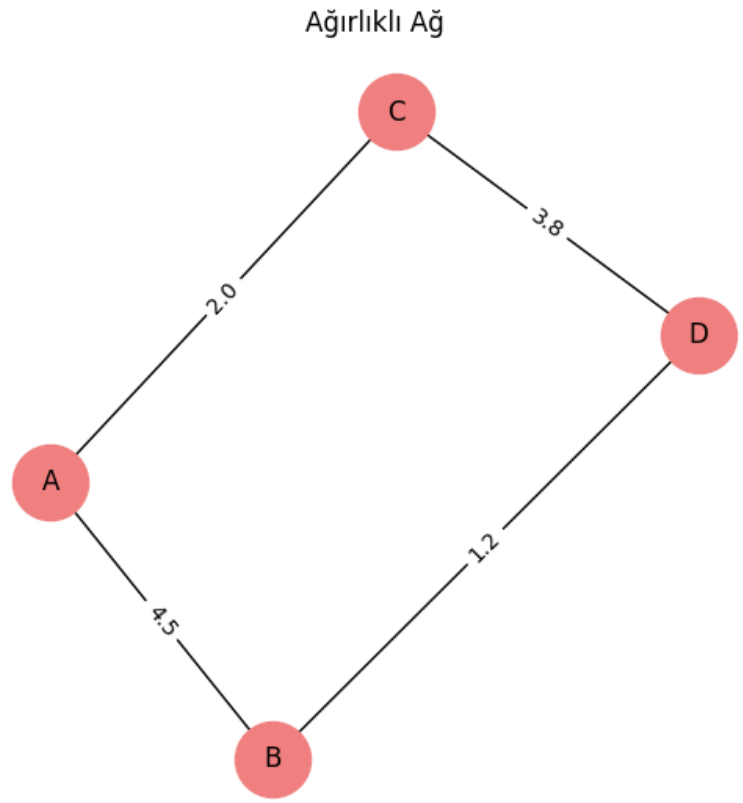
Karmaşık Ağlar

```
# Basit Bir Yönlü Ağ
DG = nx.DiGraph()  # Directed graph


DG.add_edges_from([("A", "B"), ("B", "C"),
("C", "A"), ("C", "D")])


plt.figure(figsize=(4, 4))
nx.draw(DG, with_labels=True,
node_color='lightgreen', node_size=1200,
arrows=True)
plt.title("Yönlü Ağ")
plt.show()
```



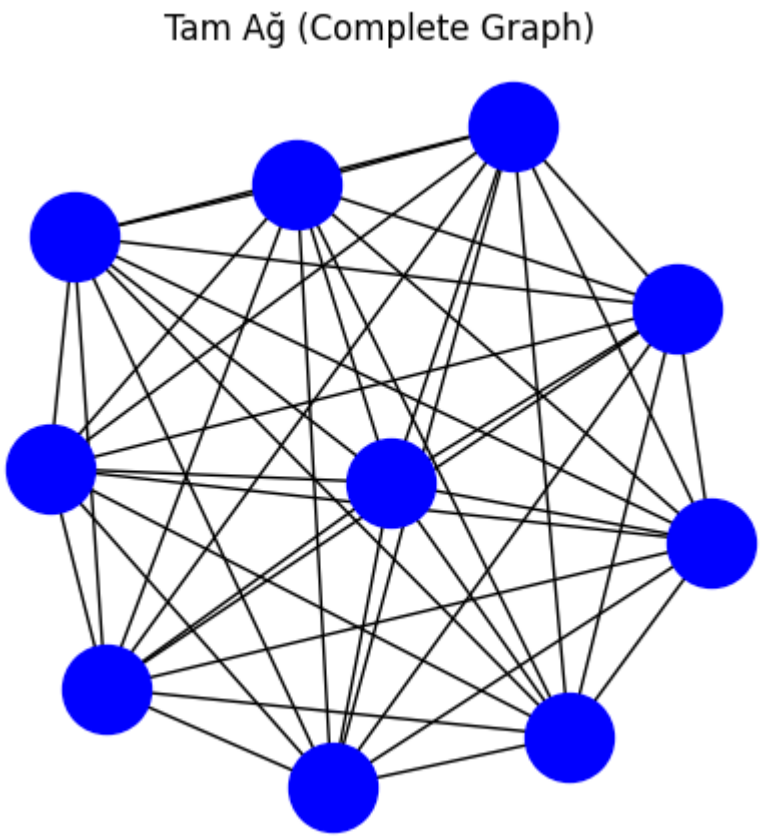
Karmaşık Ağlar

```
# Basit Bir Ağırlıklı  Ağ
WG = nx.Graph()


# (düğüm1, düğüm2, ağırlık)
WG.add_weighted_edges_from([
    ("A", "B", 4.5),
    ("A", "C", 2.0),
    ("B", "D", 1.2),
    ("C", "D", 3.8)
])


pos = nx.spring_layout(WG)  # Konum hesapla
weights = nx.get_edge_attributes(WG, 'weight')


plt.figure(figsize=(5, 5))
nx.draw(WG, pos, with_labels=True, node_color='lightcoral',
node_size=1200)
nx.draw_networkx_edge_labels(WG, pos, edge_labels=weights)
plt.title("Ağırlıklı Ağ")
plt.show()
```



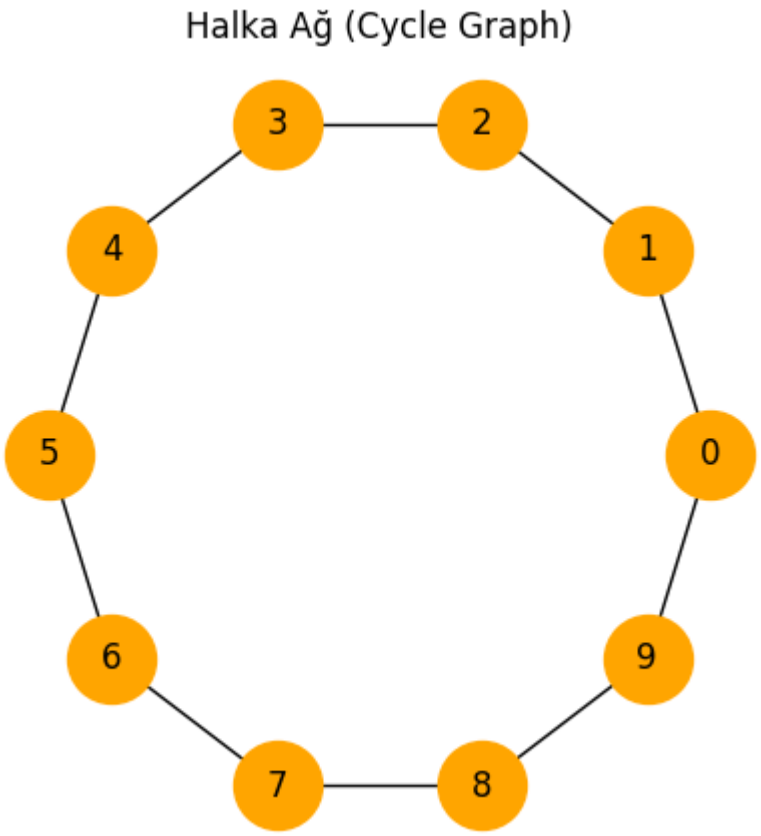
Karmaşık Ağlar

```
# 10 düğümlü tam ağ
KG = nx.complete_graph(10)


plt.figure(figsize=(4, 4))
nx.draw(KG, with_labels=False,
node_color='blue', node_size=1000)
plt.title("Tam Ağ (Complete Graph)")
plt.show()
```



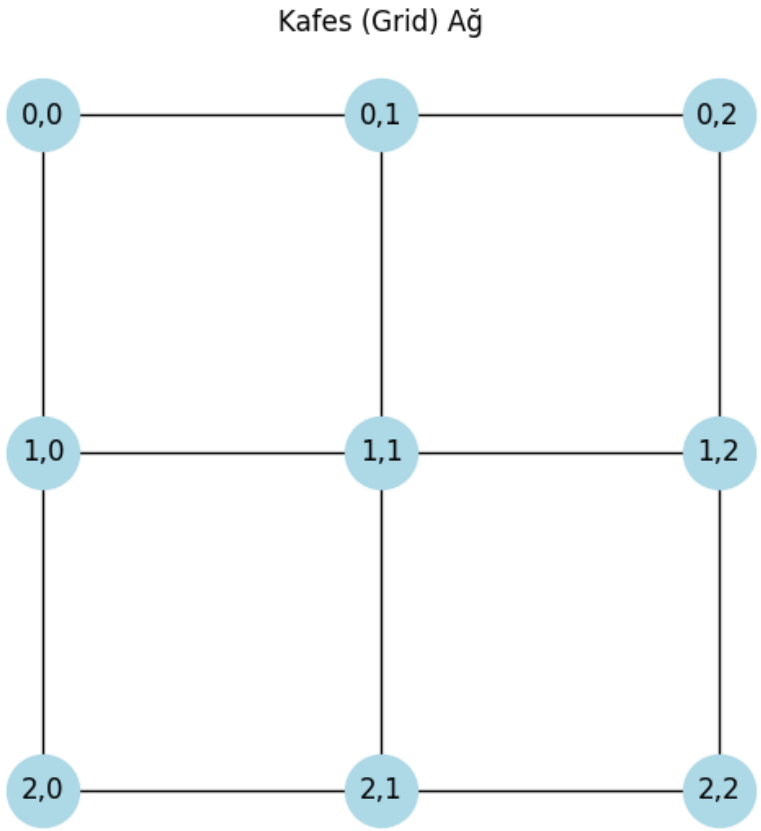
Karmaşık Ağlar

```
# Ring Ağ

RG = nx.cycle_graph(10)


plt.figure(figsize=(4, 4))
nx.draw_circular(RG, with_labels=True,
node_color='orange', node_size=1000)
plt.title("Halka Ağ (Cycle Graph)")
plt.show()
```



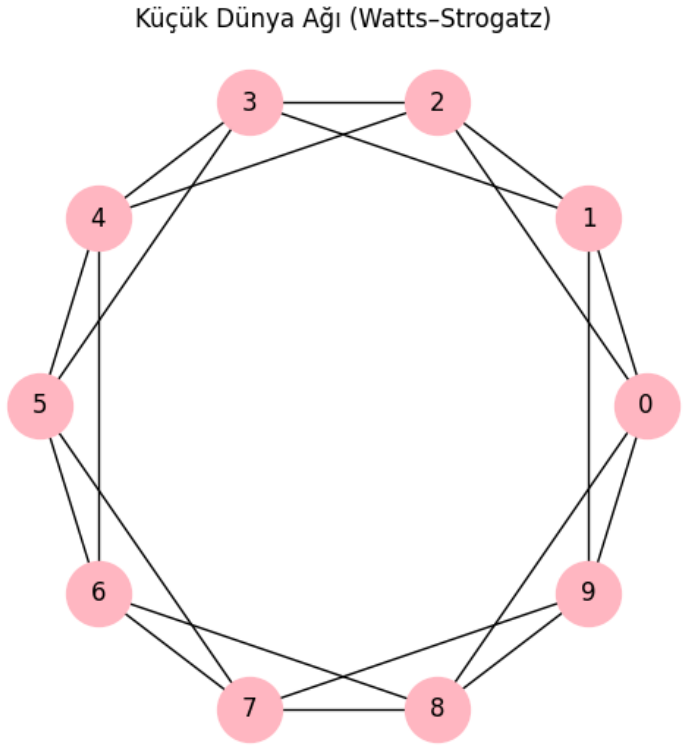
Karmaşık Ağlar

```
# Kafes (Grid / Lattice) Ağ
LG = nx.grid_2d_graph(3, 3)  # 3x3 kafes


pos = {(x, y): (y, -x) for x, y in
LG.nodes()}  # konumları düzenle
labels = {(x, y): f"{x},{y}" for x, y in
LG.nodes()}


plt.figure(figsize=(5, 5))
nx.draw(LG, pos, labels=labels,
node_color='lightblue', node_size=1000)
plt.title("Kafes (Grid) Ağ")
plt.show()
```



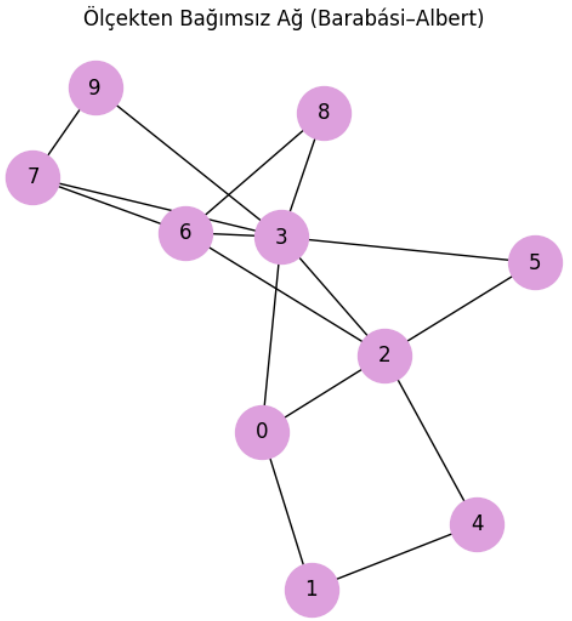
Karmaşık Ağlar

```
1# Küçük Dünya (Small-World) Ağı - Watts-Strogatz Modeli
SW = nx.watts_strogatz_graph(n=10, k=4, p=0.0)


plt.figure(figsize=(5, 5))
nx.draw_circular(SW, with_labels=True, node_color='lightpink', node_size=1000)
plt.title("Küçük Dünya Ağı (Watts-Strogatz)")
plt.show()
```



Karmaşık Ağlar

```
# Ölçekten Bağımsız (Scale-Free) Ağ - Barabási-Albert Modeli
SF = nx.barabasi_albert_graph(n=10, m=2)


plt.figure(figsize=(5, 5))
nx.draw(SF, with_labels=True, node_color='plum', node_size=1000)
plt.title("Ölçekten Bağımsız Ağ (Barabási-Albert)")
plt.show()
```



Karmaşık Ağlar

```
# Rastgele Ağ - Erdos-Rényi Modeli
ER = nx.erdos_renyi_graph(n=10, p=0.5)


plt.figure(figsize=(5, 5))
nx.draw(ER, with_labels=True,
node_color='lightgray', node_size=1000)
plt.title("Rastgele Ağ (Erdős-Rényi)")
plt.show()
```