Optimización de flujo en redes

1455175: Angel Moreno

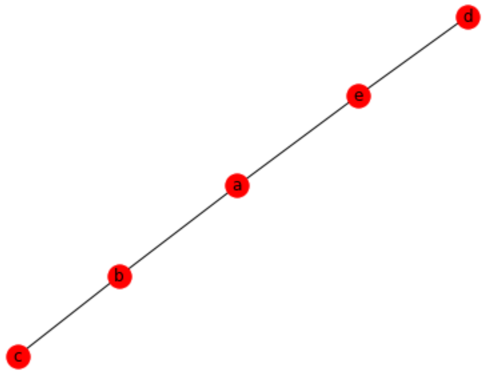
Tarea #1

12 de febrero de 2019

1. Grafo simple no dirigido acíclico

Este primer grafo es el mas sencillo, como ejemplo aplicado se utiliza para la representación de ubicaciones de localidades ó ciudades para tener representación gráfica.

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

nodos = [”a”, ”b”, ”c”, ”d”, ”e”]
vertices = [(”a”, ”b”), (”b”, ”c”), (”a”, ”e”), (”d”, ”e”)]

G = nx.Graph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)

nx.draw(G, with_labels = True)
plt.show()
```

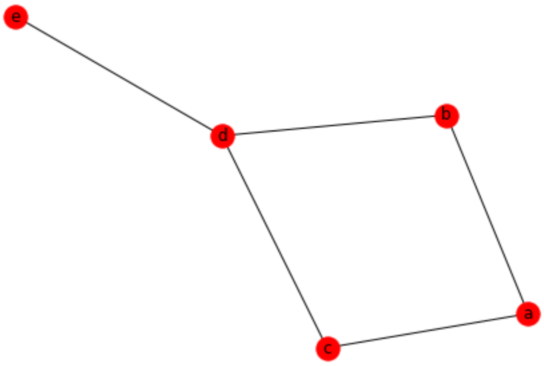


2. Grafo simple no dirigido cíclico

El metro o sistema de transporte es un ejemplo aplicado.

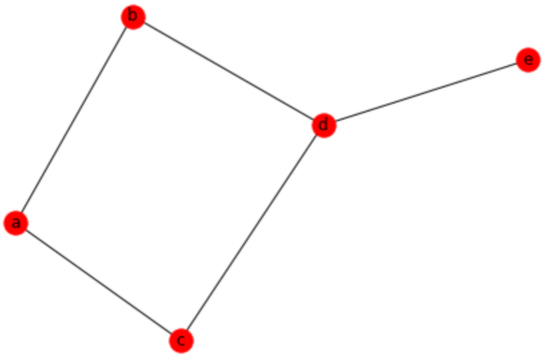
```
import networkx as nx
import matplotlib.pyplot as plt

nodos = [”a”, ”b”, ”c”, ”d”, ”e”]
vertices = [(”a”, ”b”), (”b”, ”d”), (”a”, ”c”), (”c”, ”d”), (”d”, ”e”)]

```

```
G = nx.Graph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)

nx.draw(G, with_labels = True)
plt.show()
```



3. Grafo simple no dirigido reflexivo

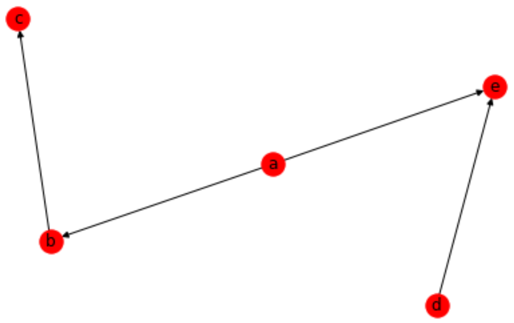
```
import networkx as nx
import matplotlib.pyplot as plt

nodos = [”a”, ”b”, ”c”, ”d”, ”e”]
vertices = [(”a”, ”b”), (”b”, ”d”), (”a”, ”c”), (”c”, ”d”), (”d”, ”e”)]
reflexivo = [(”a”, ”a”), (”b”, ”b”), (”c”, ”c”), (”d”, ”d”), (”e”, ”e”)]

G = nx.Graph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)
G.add_edges_from(reflexivo)

nx.draw(G, with_labels = True)
plt.show()
```



4. Grafo simple dirigido acíclico

Las tuberías de agua y drenaje se puede representar ya que tienen un flujo hacia una dirección.

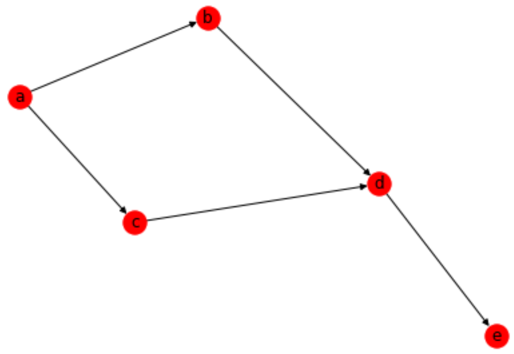
```python
import networkx as nx
import matplotlib.pyplot as plt

nodos = ["a", "b", "c", "d", "e"]
vertices = [("a", "b"), ("b", "c"), ("a", "e"), ("d", "e")]

G = nx.DiGraph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)

nx.draw(G, with_labels = True)
plt.show()
```



5. Grafo simple dirigido cíclico

Como ejemplo aplicado se toma la pista de carreras de vehículos donde tiene un circuito.

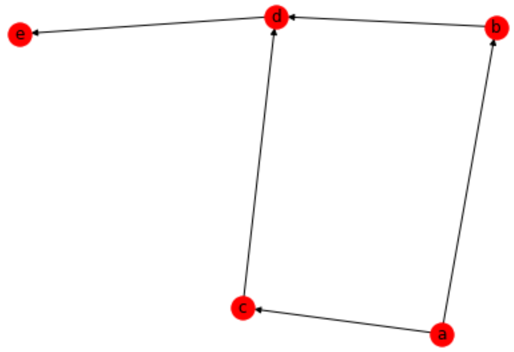
```python
import networkx as nx
import matplotlib.pyplot as plt

nodos = ["a", "b", "c", "d", "e"]
vertices = [("a", "b"), ("b", "d"), ("a", "c"), ("c", "d"), ("d", "e")]

G = nx.DiGraph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)

nx.draw(G, with_labels = True)
plt.show()
```



6. Grafo simple dirigido reflexivo

Las tuberías de petroleo, donde la reflexividad de los nodos es donde el petroleo se puede quedar como un punto de extracción.

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

nodos = ["a", "b", "c", "d", "e"]
vertices = [("a", "b"), ("b", "d"), ("a", "c"), ("c", "d"), ("d", "e")]
reflexivo = [("a", "a"), ("b", "b"), ("c", "c"), ("d", "d"), ("e", "e")]

G = nx.MutiGraph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)
G.add_edges_from(reflexivo)

nx.draw(G, with_labels = True)
plt.show()
```

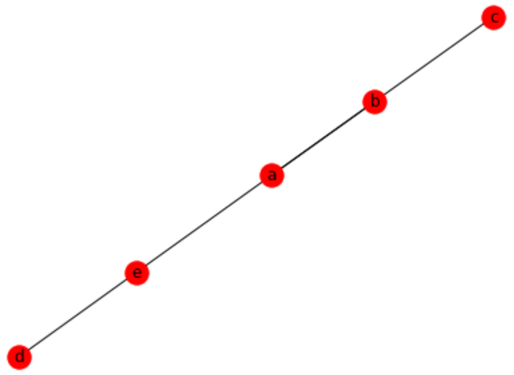


7. Multigrafo no dirigido acíclico

La red total de las tuberías de agua de drenaje representa un ejemplo para esta sección.

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

nodos = ["a", "b", "c", "d", "e"]
aristas = [("a", "b"), ("b", "a"), ("b", "c"), ("a", "e"), ("d", "e")]
```

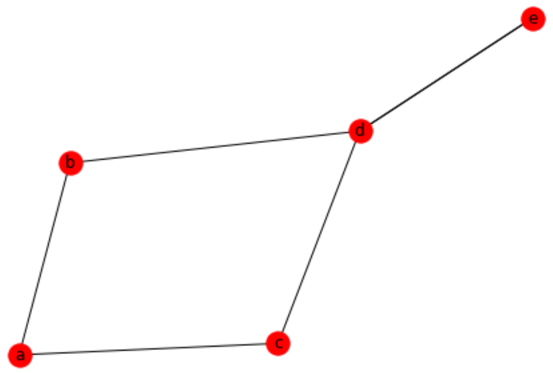
```python

G = nx.MultiDiGraph()
G.add_nodes_from(nodos)
G.add_edges_from(aristas)

nx.draw(G, with_labels = True)
plt.show()
```



8. Multigrafo no dirigido cíclico

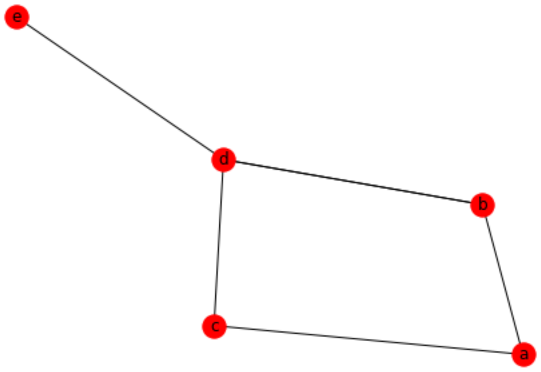
La red de autopistas de una ciudad.

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

nodos = ["a", "b", "c", "d", "e"]
vertices = [("a", "b"), ("b", "d"), ("a", "c"), ("c", "d"), ("d", "e"), ("e", "d")]

G = nx.MultiGraph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)

nx.draw(G, with_labels = True)
plt.show()
```



9. Multigrafo no dirigido reflexivo

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

nodos = ["a", "b", "c", "d", "e"]
vertices = [("a", "b"), ("b", "d"), ("a", "c"), ("c", "d"), ("d", "e"), ("d", "b
    ")]
reflexivo = [("a", "a"), ("b", "b"), ("c", "c"), ("d", "d"), ("e", "e")]

G = nx.MultiGraph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)
G.add_edges_from(reflexivo)

nx.draw(G, with_labels = True)
plt.show()
```

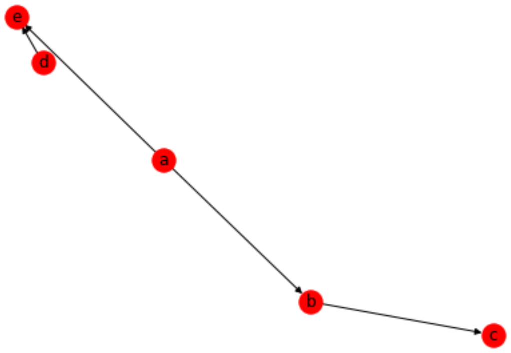


10. Multigrafo dirigido acíclico

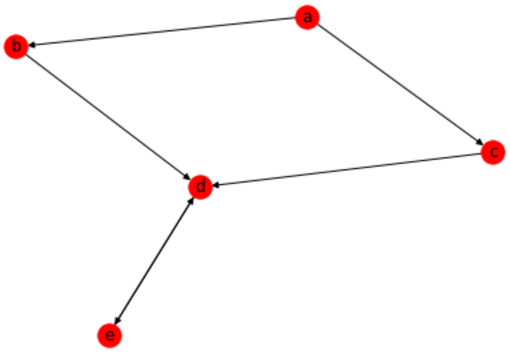
```
import networkx as nx
import matplotlib.pyplot as plt

nodos = ["a", "b", "c", "d", "e"]
aristas = [("a", "b"), ("b", "c"), ("a", "e"), ("d", "e")]

G = nx.MultiDiGraph()
G.add_nodes_from(nodos)
G.add_edges_from(aristas)

nx.draw(G, with_labels = True)
plt.show()
```

11. Multigrafo dirigido cíclico

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

nodos = ["a", "b", "c", "d", "e"]
vertices = [("a", "b"), ("b", "d"), ("a", "c"), ("c", "d"), ("d", "e"), ("e", "d
    ")]

```

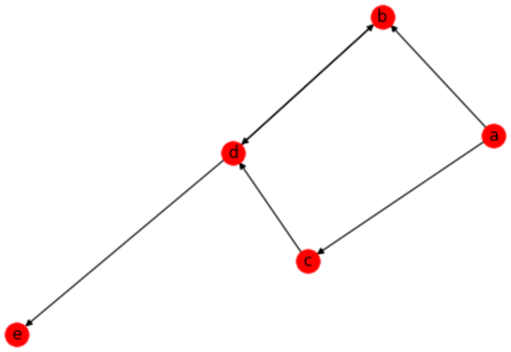


```
G = nx.MultiDiGraph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)

nx.draw(G, with_labels = True)
plt.show()
```



12. Multigrafo dirigido reflexivo

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

nodos = ["a", "b", "c", "d", "e"]
vertices = [("a", "b"), ("b", "d"), ("a", "c"), ("c", "d"), ("d", "e"), ("d", "b
    ")]
reflexivo = [("a", "a"), ("b", "b"), ("c", "c"), ("d", "d"), ("e", "e")]

G = nx.MultiDiGraph()
G.add_nodes_from(nodos)
G.add_edges_from(vertices)
G.add_edges_from(reflexivo)

nx.draw(G, with_labels = True)
plt.show()
```



b
a
d
c
e