# Importando las bibliotecas necesarias

import networkx as nx

import matplotlib.pyplot as plt

# Creando una figura para visualizar el grafo

fig = plt.figure(figsize=(12, 12))

ax = plt.subplot(111)

# Definiendo nodos (variables)

Ola = "Ola"

La\_Pintada = "La\_Pintada"

Nata = "Nata"

Anton = "Anton"

Aguadulce = "Aguadulce"

Penonome = "Penonome"

# Añadiendo nodos al grafo

G = nx.Graph()

G.add\_node(Ola)

G.add\_node(La\_Pintada)

G.add\_node(Nata)

G.add\_node(Anton)

G.add\_node(Aguadulce)

G.add\_node(Penonome)

# Definiendo conexiones y pesos entre nodos

G.add\_edge(Ola, La\_Pintada, weight=18)

G.add\_edge(Ola, Anton, weight=22)

G.add\_edge(Nata, Aguadulce, weight=52)

G.add\_edge(Aguadulce, Penonome, weight=38)

G.add\_edge(La\_Pintada, Nata, weight=36)

G.add\_edge(La\_Pintada, Penonome, weight=19)

G.add\_edge(Anton, Penonome, weight=65)

# Configurando el diseño y la visualización del grafo

posiciones = nx.spring\_layout(G)

nx.draw(G, posiciones, node\_size=1300, node\_color='red', font\_size=10, font\_weight='bold', with\_labels=True)

# Atributos del grafo y etiquetas de las aristas (pesos)

pesos = nx.get\_edge\_attributes(G, "weight")

nx.draw\_networkx\_edge\_labels(G, posiciones, edge\_labels=pesos)

# Solicitando las rutas de inicio y destino al usuario

punto\_partida = input("Ingrese el punto de partida: ")

punto\_destino = input("Ingrese el punto de destino: ")

# Aplicando el algoritmo de Dijkstra para encontrar la ruta más corta y su distancia

ruta\_mas\_corta = nx.shortest\_path(G, punto\_partida, punto\_destino, weight='weight')

distancia\_total = nx.shortest\_path\_length(G, punto\_partida, punto\_destino, weight='weight')

# Configurando el título del grafo con la información de la ruta más corta

titulo = f"La ruta más corta es: {ruta\_mas\_corta} y tiene una distancia de {distancia\_total} km"

ax.set\_title(titulo, fontsize=10)

# Guardando el grafo como PNG

plt.tight\_layout()

plt.savefig("Grafo.png", format="PNG")

# Mostrando el grafo visualmente

plt.show()