import itertools

def calculate\_distance(route, distance\_matrix):

total\_distance = 0

for i in range(len(route) - 1):

total\_distance += distance\_matrix[route[i]][route[i + 1]]

total\_distance += distance\_matrix[route[-1]][route[0]]

return total\_distance

def traveling\_salesman(distance\_matrix):

num\_cities = len(distance\_matrix)

cities = list(range(num\_cities))

all\_routes = itertools.permutations(cities)

min\_distance = float('inf')

best\_route = None

for route in all\_routes:

current\_distance = calculate\_distance(route, distance\_matrix)

if current\_distance < min\_distance:

min\_distance = current\_distance

best\_route = route

return best\_route, min\_distance

if \_\_name\_\_ == "\_\_main\_\_":

distance\_matrix = [

[0, 10, 15, 20],

[10, 0, 35, 25],

[15, 35, 0, 30],

[20, 25, 30, 0]

]

best\_route, min\_distance = traveling\_salesman(distance\_matrix)

print("Best route:", best\_route)

print("Minimum distance:", min\_distance)