def nearest\_neighbor(distances):

num\_cities = len(distances)

visited = [False] \* num\_cities

tour = [0] # Start from the first city

visited[0] = True

while len(tour) < num\_cities:

current\_city = tour[-1]

nearest\_dist = float('inf')

nearest\_city = None

for next\_city in range(num\_cities):

if not visited[next\_city] and distances[current\_city][next\_city] != 0:

dist = distances[current\_city][next\_city]

if dist < nearest\_dist:

nearest\_dist = dist

nearest\_city = next\_city

if nearest\_city is not None:

tour.append(nearest\_city)

visited[nearest\_city] = True

else:

break

tour.append(0) # Return to the starting city

return tour

def calculate\_total\_distance(tour, distances):

total\_distance = 0

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

city1 = tour[i]

city2 = tour[i + 1]

total\_distance += distances[city1][city2]

return total\_distance

# Example distances matrix

distances = [

[0, 10, 15, 20],

[10, 0, 35, 25],

[15, 25, 0, 30],

[20, 25, 30, 0]

]

tour = nearest\_neighbor(distances)

total\_distance = calculate\_total\_distance(tour, distances)

print("Nearest Neighbor Tour:", tour)

print("Total Distance:", total\_distance)