9.Write the python to implement Travelling Salesman Problem

import itertools

def tsp\_brute\_force(graph, start):

n = len(graph)

vertices = list(range(n))

vertices.remove(start)

min\_path = float('inf')

best\_order = []

for perm in itertools.permutations(vertices):

current\_cost = 0

k = start

for j in perm:

current\_cost += graph[k][j]

k = j

current\_cost += graph[k][start] # return to start

if current\_cost < min\_path:

min\_path = current\_cost

best\_order = [start] + list(perm) + [start]

return min\_path, best\_order

# Example graph (cost matrix)

graph = [

[0, 10, 15, 20],

[10, 0, 35, 25],

[15, 35, 0, 30],

[20, 25, 30, 0]

]

cost, path = tsp\_brute\_force(graph, 0)

print("Minimum Cost:", cost)

print("Path:", path)

OUTPUT:

