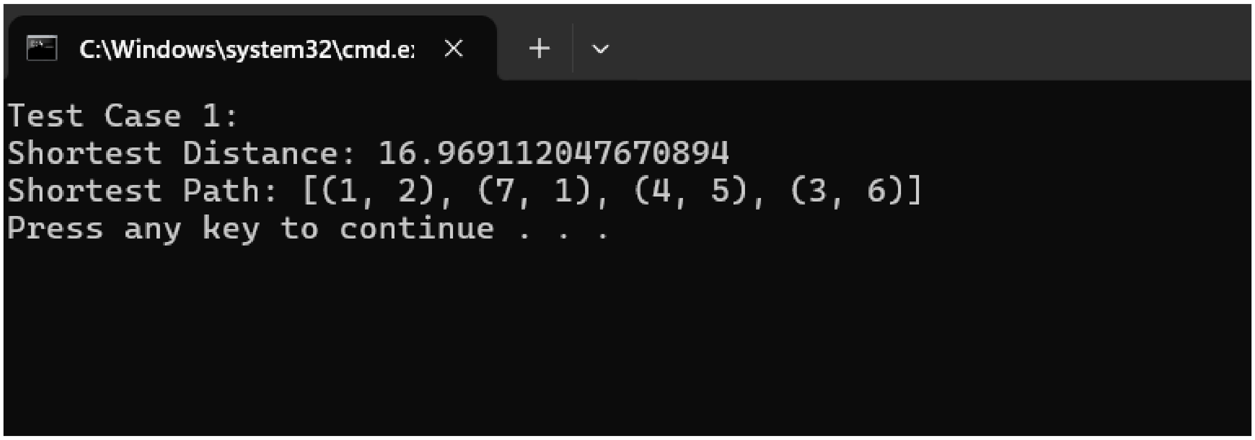
142) Travelling sales man

CODE:

```
import itertools
import math

def distance(city1, city2):
    return math.sqrt((city1[0] - city2[0])**2 + (city1[1] - city2[1])**2)

def tsp(cities):
    shortest_distance = float('inf')
    shortest_path = None

    for perm in itertools.permutations(cities[1:]):
        perm = [cities[0]] + list(perm)
        total_distance = sum(distance(perm[i], perm[i+1]) for i in range(len(perm) -
1))
        total_distance += distance(perm[-1], perm[0])

        if total_distance < shortest_distance:
            shortest_distance = total_distance
            shortest_path = perm

    return shortest_distance, shortest_path

# Test Cases
cities1 = [(1, 2), (4, 5), (7, 1), (3, 6)]
cities2 = [(2, 4), (8, 1), (1, 7), (6, 3), (5, 9)]

shortest_distance1, shortest_path1 = tsp(cities1)
shortest_distance2, shortest_path2 = tsp(cities2)

print("Test Case 1:")
print("Shortest Distance:", shortest_distance1)
print("Shortest Path:", shortest_path1)
```

OUTPUT:



```
Test Case 1:
Shortest Distance: 16.969112047670894
Shortest Path: [(1, 2), (7, 1), (4, 5), (3, 6)]
Press any key to continue . . .
```

TIME COMPLEXITY : O(n-1)!