CODE:

```
import sys

# Define the graph as an adjacency matrix
graph = [
    [0, 10, 15, 20],
    [10, 0, 35, 25],
    [15, 35, 0, 30],
    [20, 25, 30, 0]
]

# Function to find the nearest neighbor
def nearest_neighbor(graph, start):
    num_cities = len(graph)
    visited = [False] * num_cities
    path = [start]
    length = 0
    visited[start] = True

    for _ in range(num_cities - 1):
        min_dist = sys.maxsize
        nearest_city = None

        for city in range(num_cities):
            if not visited[city] and graph[path[-1]][city] < min_dist:
                min_dist = graph[path[-1]][city]
                nearest_city = city

        path.append(nearest_city)
        length += min_dist
        visited[nearest_city] = True

    length += graph[path[-1]][path[0]]  # Return to the starting city
    path.append(path[0])

    return path, length

# Main function
if __name__ == "__main__":
    start_city = 0
    shortest_path, total_cost = nearest_neighbor(graph, start_city)

    print(f"Shortest Path: {shortest_path}")
    print(f"Total Cost: {total_cost}")
```

OUTPUT:

```
PS C:\Users\shrey\OneDrive\Desktop\Projects\micro_project> & C:/Users/shrey/AppData/Local/Programs/Python/Python311/python
.exe c:/Users/shrey/OneDrive/Desktop/programs/Python/AI4.py
Shortest Path: [0, 1, 3, 2, 0]
Total Cost: 80
PS C:\Users\shrey\OneDrive\Desktop\Projects\micro_project>
```