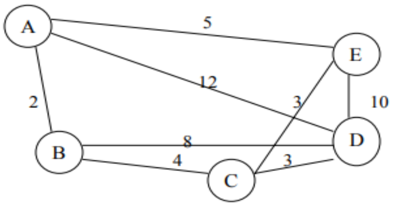
CSE 245: Algorithm

Project: Travelling Salesman Problem

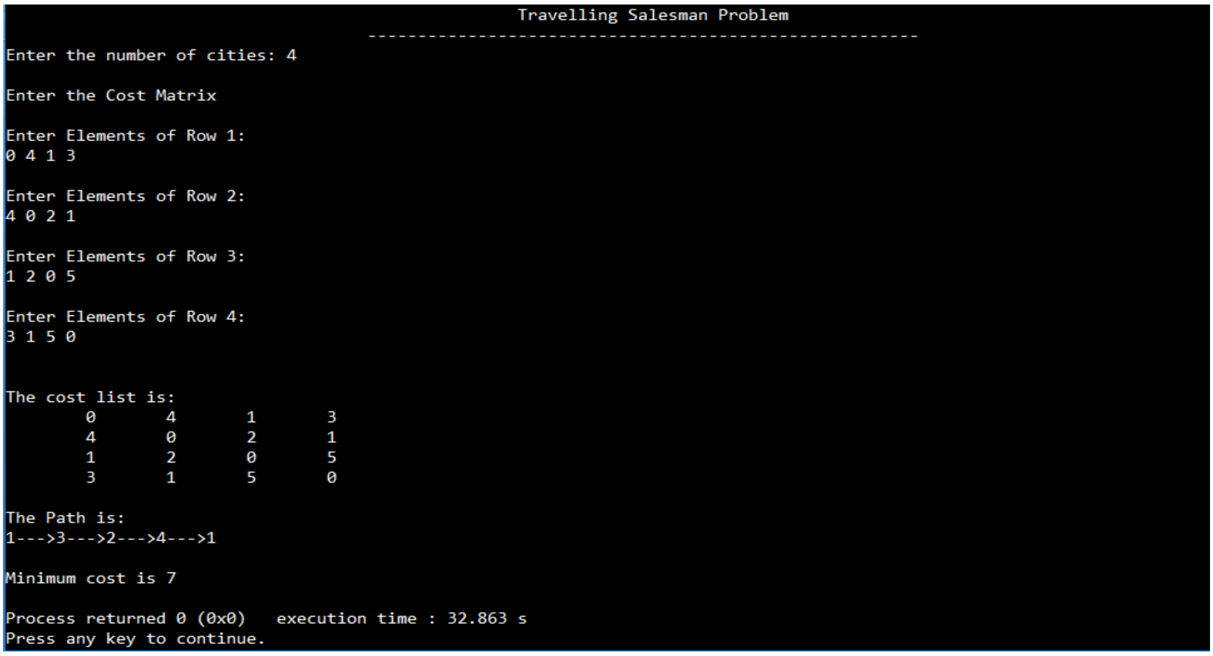
A traveller needs to visit all the cities from a list, where distances between all the cities are known and each city should be visited just once. What is the shortest possible route that he visits each city exactly once and returns to the origin city? Given a set of cities and the cost of travel (or distance) between each possible pair, you have to find the best possible way of visiting all the cities and returning to the starting point that minimizes the travel cost (or travel distance). Consider the following set of cities:



The problem lies in finding a minimal path passing from all vertices once. For example, the path Path1 {A, B, C, D, E, A} and the path Path2 {A, B, C, E, D, A} pass all the vertices but Path1 has a total length of 24 and Path2 has a total length of 31.

In this project, you have to write a conde in C/C++ that solves the travelling salesman problem and shows the best path along with the minimal cost. You should use **Dynamic Programming** or a recursive algorithm based on **Backtracking** approach in this regard as they have the ability to give same result in far fewer attempts than Brute Force method trials and thus handles the complexity better.

Output:



```
                                                Travelling Salesman Problem
                                     -------------------------------------------------------
Enter the number of cities: 4

Enter the Cost Matrix

Enter Elements of Row 1:
0 4 1 3

Enter Elements of Row 2:
4 0 2 1

Enter Elements of Row 3:
1 2 0 5

Enter Elements of Row 4:
3 1 5 0


The cost list is:
        0       4       1       3
        4       0       2       1
        1       2       0       5
        3       1       5       0

The Path is:
1--->3--->2--->4--->1

Minimum cost is 7

Process returned 0 (0x0)   execution time : 32.863 s
Press any key to continue.
```