TRAVELLING SALESMAN PROBLEM

STATEMENT OF THE PROBLEM

To implement the Travelling Salesman Problem (TSP) using different algorithms like Cheapest Link Algorithm and Nearest neighbour heuristic and compare their performance.

EXPLANATION OF THE PROBLEM

The Travelling Salesman Problem describes a salesman who must travel between N cities. The order in which he does so is something he does not care about, as long as he visits each once during his trip, and finishes where he was at first. Each city is connected to other close by cities, or nodes, by airplanes, or by road or railway. Each of those links between the cities has one or more weights (or the cost) attached. The cost describes how "difficult" it is to traverse this edge on the graph, and may be given, for example, by the cost of an airplane ticket or train ticket, or perhaps by the length of the edge, or time required to complete the traversal. The salesman wants to keep both the travel costs, as well as the distance he travels as low as possible.
We will be discussing about two algorithms to solve this problem and we'll compare the performances of both the algorithms.

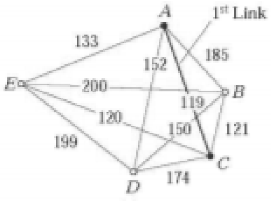
ALGORITHMS TO SOLVE THE PROBLEM

Here we'll be discussing about two algorithms: -

1. Cheapest Link Algorithm.
2. Nearest Neighbour Algorithm.

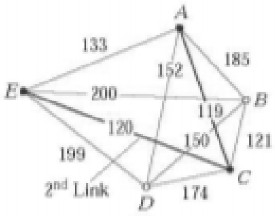
➢ CHEAPEST LINK ALGORITHM

1. Pick the cheapest link (edge with smallest weight) available. Among all the edges of the graph, the "cheapest link" is edge AC, with a cost of $119.



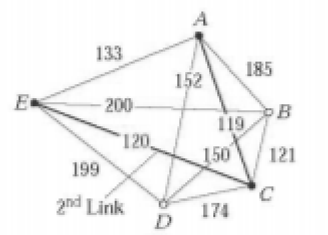
(a)

2. Pick the next cheapest link available and mark it. In this case edge CE with a cost of $120.



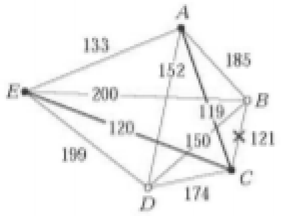
(b)

3. Continue picking and marking the cheapest unmarked link available that does not (a) close a circuit, or (b) create three edges coming out of a single vertex.



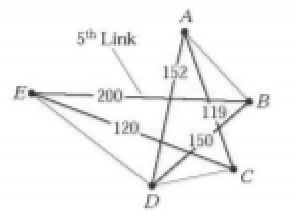
(b)

3.1. The next cheapest link available is edge BC ($121), but we should not choose BC– we would have three edges coming out of vertex C.



(c)

3.2. Approaching like this, ***The Hamilton circuit*** can be described using any vertex as the reference point at last.

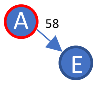


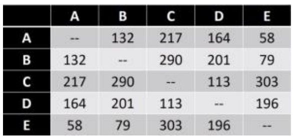
(g)

A Hamiltonian circuit also called Hamilton cycle is a graph cycle(closed loop) through a graph that visits each node exactly once.

➢ NEAREST NEIGHBOUR ALGORITHM

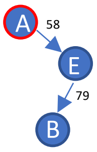
1. Begin at any city and visit nearest city. Here we begin with A and visit nearest city E.

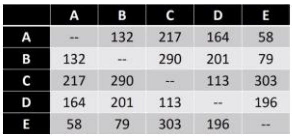




	A	B	C	D	E
A	--	132	217	164	58
B	132	--	290	201	79
C	217	290	--	113	303
D	164	201	113	--	196
E	58	79	303	196	--

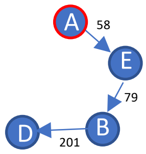
2. Then go to the unvisited city closest to the city most recently visited. Here next closest is E-B.

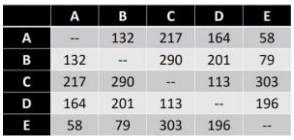




	A	B	C	D	E
A	--	132	217	164	58
B	132	--	290	201	79
C	217	290	--	113	303
D	164	201	113	--	196
E	58	79	303	196	--

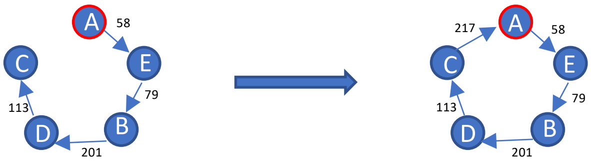
3. Continue in this manner and terminate the condition, if all the cities are visited exactly once. i.e. We get Hamilton circuit.





	A	B	C	D	E
A	--	132	217	164	58
B	132	--	290	201	79
C	217	290	--	113	303
D	164	201	113	--	196
E	58	79	303	196	--

3.1. Continuing step 2.



Hamilton Circuit