

Assignment - 1

Roll no - B16012
B16021

Solving TSP using greedy and 2 opt approach

TSP PROBLEM -

The traveling salesman problem consists of a salesman and a set of cities. The salesman has to visit each one of the cities starting from a certain one (e.g. the hometown) and returning to the same city. The challenge of the problem is that the traveling salesman wants to minimize the total length of the trip.

OUR SOLUTION-

First we use the greedy algorithm to find first tour. Then we use the 2-opt algorithm to get the optimal path.

A TSP tour T is called 2-optimal if there is no 2-adjacent tour to T with lower cost than T.

Look for a 2-adjacent tour with lower cost than the current tour. If one is found, then it replaces the current tour. This continues until there is a 2-optimal tour

PSUEDO CODE-

```
repeat until no improvement is made {
  start_again:
    best_distance = calculateTotalDistance (existing_route)
    for (i = 0; i < number of nodes eligible to be swapped - 1; i++)
    {
      for (k = i + 1; k < number of nodes eligible to be swapped; k++)
      {
        new_route = 2optSwap (existing_route, i, k)
        new_distance = calculateTotalDistance (new_route)
        If (new_distance < best_distance)
        {
          existing_route = new_route
          goto start_again
        }
      }
    }
  }
```

OUR RESULT-

TYPE	OPTIMAL COST
Euclidean 100	1614.27
Euclidean 250	2642.43
Euclidean 500	11766.4
Non Euclidean 100	5391.2
Non Euclidean 250	13031.6
Non Euclidean 500	25664.7

IMPROVMENT CAN BE DONE-

- 3 opt implementation instead of 2-opt
- Simulated Annealing can be done in place of 2-opt to get better optimal path
- Better greedy algorithm in place of our greedy algorithm to obtain better initial tour.

REFERENCE -

<https://www.csd.uoc.gr/~hy583/papers/ch11.pdf>