The following heuristics are used by the program to generate a tour for the Travelling Salesman Problem:

1. Nearest Neighbour Heuristic

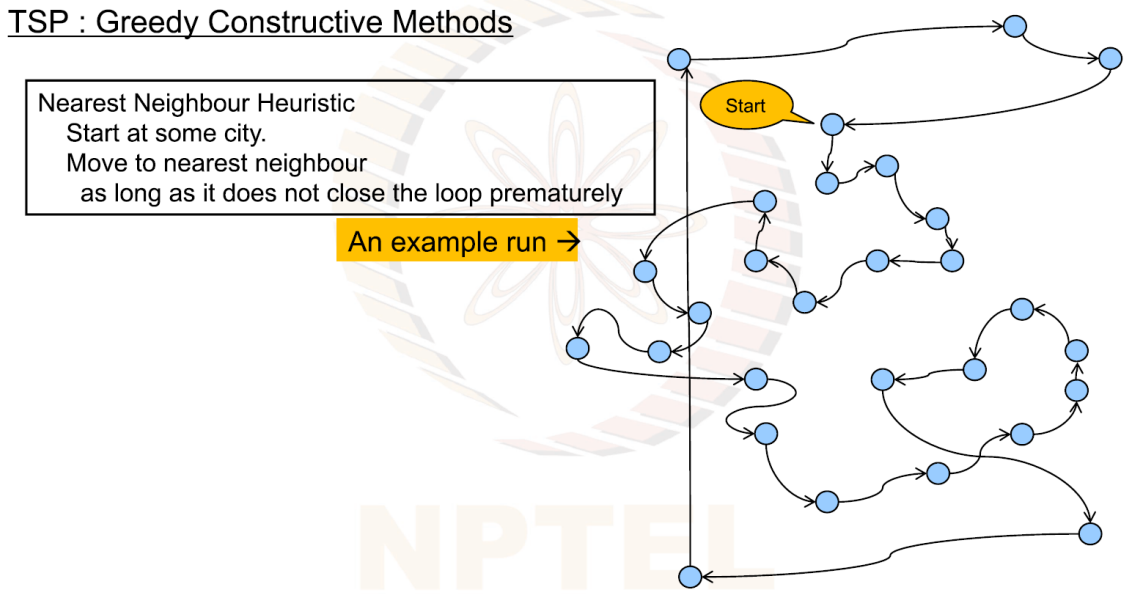

Figure 1: Nearest Neighbour Heuristic

2. Variation of Nearest Neighbour Heuristic

Figure 2: Nearest Neighbour Heuristic on both sides

3. Greedy Algorithm

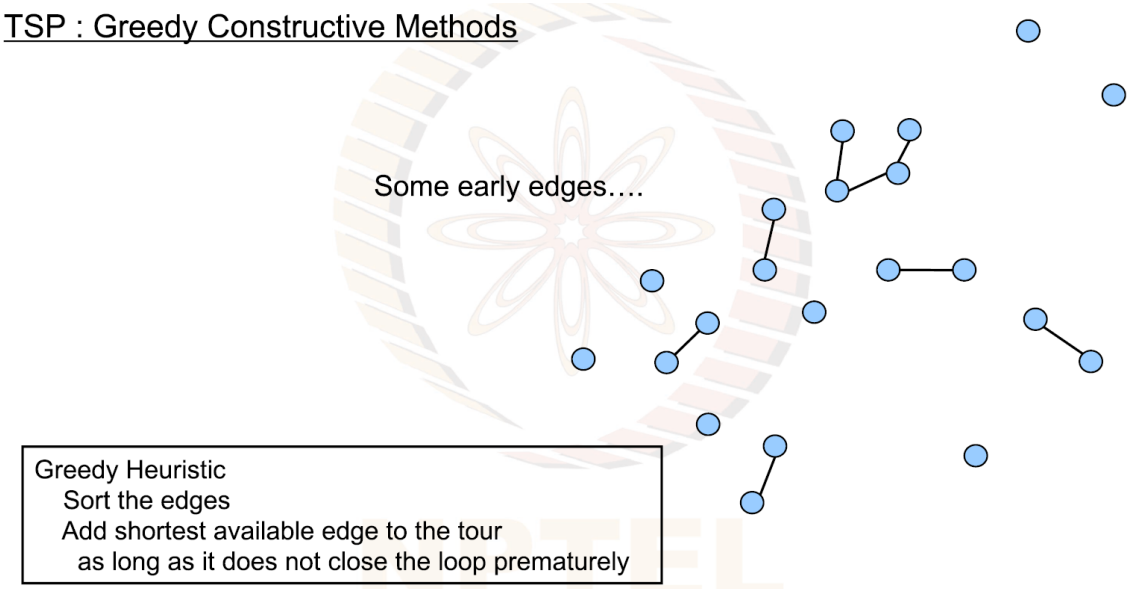

Figure 3: Greedy Algorithm

4. Saving Heuristic

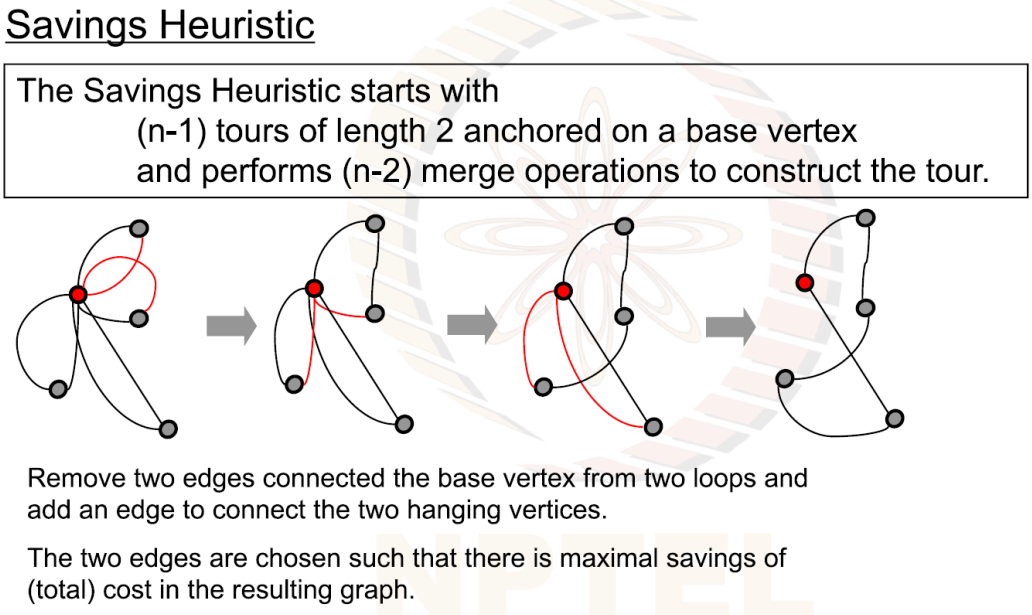

Figure 4: Saving Heuristic

Variation of Nearest Neighbour heuristic where edges can be added from both sides seems to be perform best for the given examples(euclidean100.txt,non euclidean100.txt).

Initially, I started off with the Nearest Neighbour heuristic and extended to allow edges to be added from either sides. In both of these algorithms, I consider all the n vertices as starting vertex and create a tour and choose the best out of these.

Then, I went ahead implementing the Greedy heuristic but it didn't improve on the tours generated by the nearest neighbour heuristic.

Finally, I tried implementing the Saving Heuristic and ran into small implementation issues and wasn't able to complete the implementation.