Description

This algorithm is using the greedy heuristic approach. Meaning it is only an approximation of, and not the absolute optimal result.

The first TSP function receives a list of cities from the input in main, and marks them “mustVisit,” and a separate list for “visited” is created. A variable tourLength is created and set to zero. A while loop runs as long as mustVisit contains any cities. Inside the while loop, there is a for loop that runs over the visited array. Then, another for loop inside that one which iterates over the mustVisit array. The “distance” variable is set to the distance between the city at visited[i] and mustVisit[j]. When the nearest city in the mustVisit array is found, it is popped from that array and added to the visited array. The tourLength is incremented in this case.

The second algorithm is faster but less accurate.

Pseudo Code

Best Tours for Three Example Instances

Best Tours for the Competition Test Instances