

## Statistics

Input	Algorithm	Average Time (ms)	Distance
dj38	Nearest Neighbor	0.09	9,745
	GreedyTSP	1	7,019
	Optimal		6,656
qa194	Nearest Neighbor	1.2	11,640
	GreedyTSP	30	11,521
	Optimal		9,352
zi929	Nearest Neighbor	18	113,926
	GreedyTSP	427	109,377
	Optimal		95,345
ja9847	Nearest Neighbor	825	665,821
	GreedyTSP	73081	604,185
	Optimal		491,924

Looking at the table I can see that **Nearest Neighbor** total distances seem in practice to be in the range of 1.2 - 1.5 times greater than the optimal tour. However this probably cannot be guaranteed as it is a heuristic algorithm and there might be some inputs that can throw it off. I recall in class we discussed that Worst case NN, if not constrained by the triangle inequality, can return the worst possible tour.

As for **GreedyTSP**, the total distances improved noticeably from NN. In practice my implementation of GreedyTSP outputted distances in the range of 1.06 - 1.23 times greater than optimal. However, GreedyTSP was also noticeably slower than NN. This was especially noticeable when using it on the Japan data set where it took over a minute to execute.