Our group initially chose to implement the branch and bound method to find an exact solution to the traveling salesman problem, however this proved to be difficult. After several days working on the implementation, our program still did not find optimal solutions in the allotted time. Once we had exhaustively combed through the code with no fix, we made the decision to scrap the program and start over from scratch using the 2-opt method.

## 2-Opt Solution

### Description

The 2-Opt method is a heuristic algorithm that can be used to solve the traveling salesman problem. While it cannot guarantee optimal results for all cases, it can return near-optimal solutions quickly. Two-opt starts with an initial tour and creates permutations by removing two edges from the tour and reconnecting the resulting paths, then evaluates the resulting distance. [1] If the permutation is an improvement, that permutation is kept as the optimal solution. If not, the swap is undone, and the next swap made. Permutations are repeatedly created and compared, until (in our case) 20 consecutive attempts have been made with no improvements. At this point, the best permutation thus far is considered optimal, and the algorithm terminates.

The time complexity of 2-opt is still debated, however the speed can be affected by the method used to generate the initial tour. This can be done using any algorithm, however we chose to use a greedy method, which makes the time complexity O(n2 ln(n)). [2]

# Pseudocode[3]

TSP\_2Opt

while tries < max tries  
 best distance = tour distance

for i = 0 to (tour size – 1)

for k = i + 1 to tour size

#create new permutation

2\_opt\_swap(i,k)

if new distance < best distance

#reset tries since improved tour found

tries = 0

tour = new tour

best distance = new distance

tries++

2\_opt\_swap(i, k)

take first segment of route (0 to i – 1) and add to the new route in order

take second segment (i to k) and add to new route in reverse order

add remaining portion (k+1 to end) to new route in order

# Bibliography

|  |  |
| --- | --- |
| [1] | Wikipedia, "2-Opt," 2018. [Online]. Availablehttps://en.wikipedia.org/wiki/2-opt. [Accessed 5 June 2018]. |

|  |  |
| --- | --- |
| [2] | Amanur Saiyed, "The Traveling Salesman Problem," 2012. [Online]. Available: http://cs.indstate.edu/~zeeshan/aman.pdf. [Accessed 7 June 2018]. |

|  |  |
| --- | --- |
| [3] | Andy, "C++ Implementation of the 2-opt to the "Att48" Traveling Salesman Problem," 2012. [Online]. Available: http://www.technical-recipes.com/2012/applying-c-implementations-of-2-opt-to-travelling-salesman-problems/. [Accessed 6 June 2018]. |