

Traveling Salesperson Problem Solver

Command Line Arguments

Example of how to execute this program on the command line:

```
./tsp -m OPTTSP < sample.txt
```

The executable tsp takes the following command line options:

-m [arg]

The -m flag is required and must be followed by either MST, OPTTSP, or FASTTSP which indicates what algorithm the tsp executable runs:

- **MST** Minimum Spanning Tree, which is a tree that connects all the vertices together using the edges with the smallest possible total length. This program uses Kruskal's algorithm to calculate the MST.
- **OPTTSP** Optimal Traveling Salesperson Problem. This program uses a branch and bound algorithm with pruning to find the shortest possible hamiltonian circuit (shortest path between the cities visiting each city only once.) Note that the Traveling Salesperson Problem is NP Complete, so inputs of over 27 vertices begin to take a long time.
- **FASTTSP** Fast Traveling Salesperson Problem. This algorithm finds a relatively short path between all the cities. Specifically, it uses a variation of the farthest insertion heuristic to solve the traveling salesperson problem fast for a large number of input vertices.
- **ANTCOL** Ant Colony Optimization Method. The current version is a variation of the algorithm published in 2008 by Jinhui Yang, Xiaohu Shia, Maurizio Marchese, and Yanchun Liang. <https://www.sciencedirect.com/science/article/pii/S1002007108002736>

Input/Output

Input:

- The total number of vertices, followed by
- A list of their coordinates on an XY plane. Input is separated by new lines and spaces in the way specified in the example below, adding an extra new line at the end of a txt file.

Output for MST:

- The total weight of all the edges followed by
- A list of all the edges, where edges are listed as pairs of vertices. The numbering of the vertices corresponds to the order in which they are listed in input.

Important Note: this particular MST implementation lists the following maps as being invalid: Maps that don't have indices on either the negative X axis or the negative Y axis. One coordinate in these maps must either be of the form (0, a) or of the form (a, 0) , where a is some negative number.

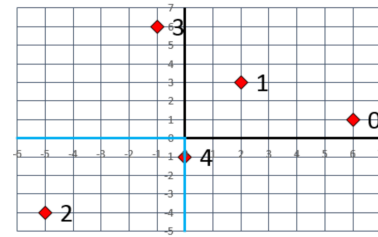
Output for OPTTSP, FASTTSP, and ANTCOL:

- The total distance of the tour, followed by
- An ordered list of the nodes visited: Their numbering corresponds to the order in which they are listed in input.

Example

Lets use the input in the sample.txt file, The coordinates of the vertices are listed below, numbered 0 through 4, visualized in fig 1:

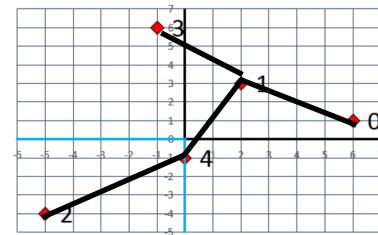
```
5
6 1
2 3
-5 -4
-1 6
0 -1
```



MST would produce the following output, where the edges listed not as coordinates, but as pairs of vertices, visualized in fig 2:

are

```
19.02
0 1
2 4
1 3
1 4
```

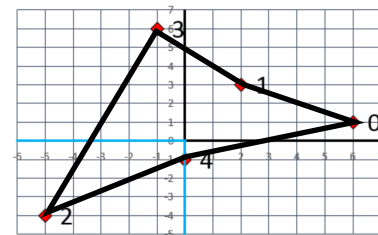


FASTTSP or OPTTSP would produce the either of the following outputs, visualized in fig 3:

```
31.64
0 4 2 3 1
```

Or

```
31.64
0 1 3 2 4
```



You can run some sample input files like sample.txt (a short sample) and large.txt (a "large" sample with 26 nodes for OPTTSP)