Vocabulary:
- tree;
- spanning tree;
- greedy algorithm.

Kruskal’s algorithm is a greedy algorithm for finding a minimum weight spanning tree in a weighted graph, which proceeds as follows:

Step 1. List all the edges of the graph in the order of increasing weight.
Step 2. Select the first edge from the list.
Step 3. Consider the next edge in the list. If adding it to the already selected edges creates a cycle, then throw it out. Otherwise, add it to the selected edges.
Step 4. Repeat the last step until all vertices are connected.

You can find a TSP route in a weighted graph using the following method:

Step 1. Find a minimum weight spanning tree in the graph.
Step 2. Use the edges in the tree to find a circuit that visits every vertex (you may have to backtrack).
Step 3. Turn the circuit that you found into a cycle by taking shortcuts.

Further reading:


This is an engaging and immensely entertaining account of the history and the current status of the Traveling Salesman Problem. I highly recommend it—especially the first two chapters. Unfortunately, a paper copy is not available at the library right now, but you can use your library access to get an electronic version.
Example of using Kruskal’s algorithm to find a minimum weight spanning tree:

Edges (listed in the order of increasing weight):

2–3 (weight 109); 1–4 (weight 117); 3–4 (weight 127); 2–4 (weight 129); 3–5 (weight 141);
1–5 (weight 143); 1–3 (weight 155); 4–5 (weight 160); 2–5 (weight 171); 1–2 (weight 176).

The first edge in the list is 2–3.
The second edge in the list is 1–4.
The next edge in the list is 3–4.
The next edge in the list is 2–4.
We cannot add it, because then it would complete a cycle (2–3–4–2).
The next edge in the list is 3–5.
The resulting graph is connected, so we have found the desired minimum weight spanning tree. The weight of this tree is

\[109 + 117 + 127 + 141 = 494.\]