Nearest-neighbor algorithm, using a table

1. Find the abbreviation for the current city on the diagonal in the table.
2. Find the smallest number in that row and column that has not been marked out.
3. Draw the corresponding edge on the map.
4. List the distance (from the current city to the new city) on the map.
5. Go back to the abbreviation (for the current city) on the diagonal in the table, and mark out everything in that row and column.
6. Now let the new city be called the current city.
7. Go to step (1).
8. When you finish a Hamilton circuit, add up all of the distances listed on the map.

Cheapest-link algorithm, using a table

1. Find the smallest number that is listed in the table and has not been circled or marked out.
2. See if drawing the corresponding edge on the map would create a subcircuit/loop.
   
   (2a) If it would,...
   Mark out the number.
   Go to step (1).
   
   (2b) If it would not,...
   Draw the corresponding edge on the map.
   Circle the distance in the table.
   List the distance on the map.
   If any city now has 2 edges connected to it,
   find its abbreviation on the diagonal in the table, and
   mark out everything in that row and column.
   Go to step (1).

3. When you finish a Hamilton circuit, add up all of the distances listed on the map.