

README

hungarian_final.ipynb:

The method `step_1` minimises the rows of the matrix by finding the smallest element in each row and subtracting it from each element in the row.

The method `step_2` minimises the columns by transposing the matrix using the transpose method, applying `step_1` and then transposing it back.

The method `assignment` is the largest method which uses methods `single_value`, `check_route2`, `cross_off_coord` and `main_step` to assign the values in the matrix.

The method `hungarian_algorithm` then combines these steps to perform the complete algorithm on an input matrix. An example matrix has then been hard coded and the hungarian algorithm has been applied to show it's success.

In the last cell of the notebook, I have included the code for the final travelling salesperson solution (which also includes the method `next_minimum`) which uses the hungarian algorithm in a while loop, which cannot run without errors.