Name: Aleksandar Yordanov User-name: jvgw34
Algorithm A: Genetic Algorithm
Algorithm B: Ant Colony Optimisation
Description of enhancement of Algorithm A:

*1st Enhancement: Generation of an elite population + a greedy tour – picking the fittest n individuals from a population of 10 * n for the initial population with a greedy tour so the algorithm can evolve off the elite population.*

2nd Enhancement: Partially mapped crossover – This crossover algorithm runs much faster than classic crossover. Allowing me more iterations in the time given, it also provides a better way to transplant genetic information than classical crossover.

3rd Enhancement: Improved mutation strategy – Picking 2 random indices and reversing the contents between those indices instead of just flipping 2 random positions. – This allows for more varied mutations and easier escape of local minima.

4th Enhancement: 2-opt: Searching through the tour and finding sub-tours that have crossover and reordering them such that they do not.

5th Enhancement: Time based iterations instead of fixed iteration. This allows to balance execution time with population size much easier and keep the execution time under a minute for the final submission.

6th Enhancement: Optimised roulette wheel approach. Keeping probabilities bound from 0 to probability_sum to avoid unnecessary calculations.

Description of enhancement of Algorithm B:

1st Enhancement: Rank-Based ant system: Choosing the ants with the shortest tours to lay the most pheromone allows for quicker convergence to minima.

2nd Enhancement: Avoidant ants: Ants that prefer heuristic desirability and have a chance to "reroll" a path. I.e., They will choose another city excluding the previously selected one if certain conditions are met. This encourages exploration and allows the ant colony system to escape local minima easier.

3rd Enhancement: Scout ants: I add ants that explore around the current best solution and randomly pick different cities to travel to within those paths, which is done by mutating city locations/sub-paths, allowing for some further exploration.

4th Enhancement: 2-opt on only select best ranked tours. This maintains efficiency while quickly converging on an optimised path.

5th Enhancement: Experimentation with non-standard values for initial pheromone, rho, w, alpha and beta to work with my other enhancements. Found ideal values through trial and error for my enhanced ant colony algorithm, which differ from the recommended values.

6th Enhancement: Time based iterations instead of fixed iteration. This allows to balance execution time with population size much easier and keep the execution time under a minute for the final submission.

7th Enhancement: Optimised roulette wheel approach. Keeping probabilities bound from 0 to probability_sum to avoid unnecessary calculations.

DESCRIPTION OF ALGORITHM ONLY IF THE ALGORITHM IS NOT COVERED IN LECTURES

Description of *non-standard* Algorithm A:

Describe any non-standard algorithms you have implemented that ***have not been covered in lectures*** *(otherwise these boxes should be blank) You need to convince me that your implementation is indeed that of the named algorithm and you need to* ***provide a full reference to the source for your algorithm****. You should* ***include a pseudocode description****. You can vary the sizes of these boxes but do not change the font (Calabri), font size (11), the paragraph properties (single space) or the header and footer and everything should fit onto one side of A4. (You can delete these instructions.)*

Remember: You need my express permission to implement a non-standard algorithm!

Description of *non-standard* Algorithm B:

Type here, as above.