Name: Lea D. Lobanov User-ID: cssg28
Algorithm A: Genetic Algorithm
Algorithm B: Simulated Annealing

Description of enhancement of Algorithm A: Genetic Algorithm

- Addition of elitism parameter; given a value x between 0 and 1, the fittest x proportion of the current generation are automatically carried over to the next generation
- Addition of crossover rate parameter; given a value x between 0 and 1, crossover between 2 parents will occur with probability x. If the crossover probability is not met, the second parent is returned as the child.
- Introduced concept of mating pool; given a value x between 0 and 1, only the fittest x proportion of the current generation participate in mating (parent selection) to produce the next generation
- Modified means of parent selection:
    - 1st parent: fittest parent in the current generation is automatically selected to breed
    - 2nd parent: selected via k-tournament method (vs. roulette method in the basic implementation); k individuals are randomly selected from the current generation and participate in a tournament, in which the fittest (tour with lowest path cost) is selected as the second parent
- Introduced 9 different mutation operators; if the mutation probability is met, the fittest (lowest path cost) of the 9 proposed variations is selected as the mutated child. A varied mutation scheme encourages exploration of global optima and helps overcome premature convergence; mutation operators attributed to https://www.researchgate.net/publication/304623320_Enhancing_genetic_algorithms_using_multi_mutations .
    - Swap 2 cities at randomly specified indices
    - Swap the “worst” gene (max distance to left neighbour) with random city
    - Swap the worst-LR gene (max distance to left/right neighbour) with random city
    - Swap the worst-LR gene with its nearest neighbour to the left, within a specified search radius
    - Swap the worst-LR gene with its nearest neighbour to the left or right, within a specified search radius
    - Swap 2 worst genes in the tour
    - Swap genes local to the worst gene (direct left/right neighbours)
    - Reverse the ordering of a random subset of the tour
    - Naively shuffle the tour
- Prevent early convergence by checking for population uniformity; if all members of the current generation are identical to one another, force a mutation probability of 1 when producing the subsequent generation. Doing so ensures population diversity is maintained, and avoids early convergence at local minima; the best global tour is saved and updated as the algorithm executes, ensuring strong tours are not lost.
- Experimentation with parameter values to improve overall performance; for parameters present in both the basic and extended implementations, the same parameter values are used for consistency

Description of enhancement of Algorithm B: Simulated Annealing

- Introduced max_iterations parameter; algorithm terminates if T is cool enough or # of specified iterations exceeded
- Introduced geometric cooling function (vs. linear in basic implementation); hence the temperature decreases more rapidly at high temperatures, and more slowly as the algorithm approaches cooler temperatures
- Modified successor function; introduced new successor operators to select next tour. The next successor is dynamically selected as the fittest of the generated candidate successors.
    - a) Naively generate a valid, random shuffle of the current tour; b) Swap 2 adjacent cities, a randomly selected city and its right neighbor are chosen; c) Swap 2 adjacent cities, a randomly selected city and its left neighbor are chosen; d) Swap 2 cities at randomly specified indices; e) Insert random city at specified index, shifting remaining cities to the right until original index of random city is met; f) Reverse the ordering of a random subset of the current tour
- Temperature cooling rate dependent on hot/cold breakpoints; above and below specified partitions of T, a different (increased) temperature decrement value is used. This allows for full exploration of local optima at lower temperatures, and wider traversal of the tour set at higher temperatures.
    - An additional temperature decrement parameter, temp_breakpoint, has been added to facilitate this enhancement.
- Modified generation of initial tour; starting from a randomly chosen city, greedily select the initial tour (vs. naïve shuffle of cities in basic implementation)
- Record tour with global minimal path cost, even if probability condition to replace current state with successor is not met
- Experimentation with parameter values to improve overall performance; for parameters present in both the basic and extended implementations, the same parameter values are used for consistency