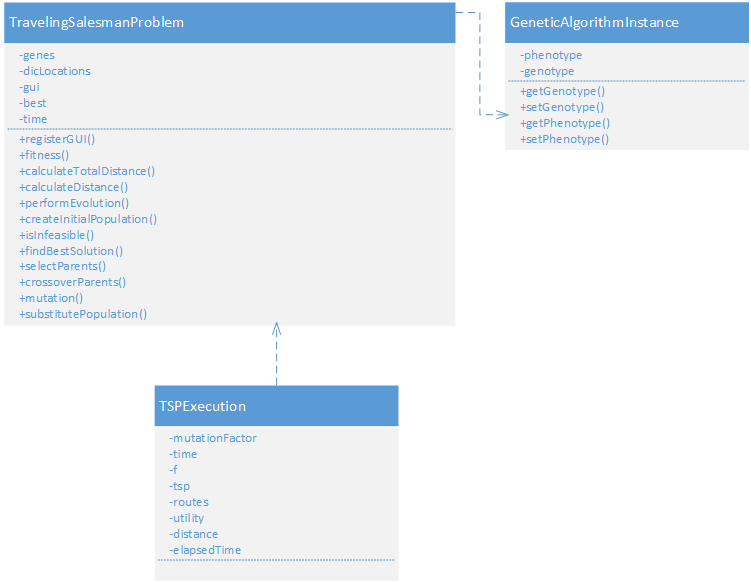
# UML diagram :



# GA Steps :

Every step is same as the original implementation, except Selection, Crossover, and Substitute step. These 3 steps has been changed based on greedy approach.

- Selection step

* The original implementation uses rank based. This approach allows the inferior to have more chance to survive, but that is not what I want. I aim to greedy approach. Therefore I use a combination of roulette wheel based and deterministic sampling, “so that relevant organisms with higher fitness are selected and survived.” [1]
* Roulette wheel based :
* Deterministic sampling : Si = round(Pi \* sizeofpopulation) + 1

- Crossover step

* Order Crossover : “Alleles from parent1 that fall between the two cross points are copied into the same positions of the offspring. The remaining allele order is determined by parent2. Nonduplicative alleles are copied from parent2 to the offspring beginning at the position following the second cross point. Both the parent2 and the offspring are traversed circularly from that point. A copy of the parent′s next nonduplicative allele is placed in the next available child position.” [1]

- Substitution step

* The original implementation discard old generation even if their fitness are good enough.
* My approach combines old and new generations by selecting best fitness from both of them.

# Experiments :

Every experiment is run for 10 times and the smallest and highest value of distance is selected.

Input data is taken from provided CSV dataset which have 40 cities.

Mutation parameter (offspring = 60, population = 100):

|  |  |  |
| --- | --- | --- |
| Mutation | Minimal Distance | Maximum Distance |
| 0.25% | 12619.2425758 | 14050.8069414 |
| 1% | 12210.6803957 | 13785.9139034 |
| 30% | 12929.3802487 | 13984.7079393 |
| 70% | 12577.4887739 | 13976.3265466 |
| 100% | 13369.6976291 | 13819.7170364 |

1% of mutation gives the best(minimum and maximum) distance.

Offspring size (population = 100, mutation = 1%):

|  |  |  |
| --- | --- | --- |
| Offspring | Minimal Distance | Maximum Distance |
| 5 | 13642.776786 | 14850.2225133 |
| 10 | 13422.7516814 | 14260.9191256 |
| 20 | 13840.5198416 | 14078.5605443 |
| 30 | 13421.2267589 | 14446.3945299 |
| 40 | 13595.0267683 | 14247.1541378 |
| 60 | 10390.46171 | 12512.1599165 |
| 70 | 13242.4413978 | 14115.5981283 |
| 80 | 13161.063378 | 13565.9550383 |
| 90 | 13259.5190319 | 14276.6327415 |
| 100 | 11283.6491708 | 12740.5217276 |

Offspring size of 60 (60% of population size) gives the best distance.

Population size (offspring = 60%, mutation = 1%):

|  |  |  |
| --- | --- | --- |
| Population | Minimal Distance | Maximum Distance |
| 100 | 10390.46171 | 12512.1599165 |
| 300 | 12738.9050549 | 13655.5758149 |
| 700 | 13481.9613056 | 14121.5591979 |
| 1000 | 13063.2540302 | 14135.3893195 |

Population size of 100 gives the best distance.

# References :

[1] Buthainah Fahran Al-Dulaimi, and Hamza A. Ali, “Enhanced Traveling Salesman Problem Solving by Genetic Algorithm Technique (TSPGA)”, World Academy of Science, Engineering and Technology, 2008