Hubert Radom 145445
Radosław Bodus 145457

github: https://github.com/HubertRadom/EvolutionaryComputation/tree/main/lab2

Problem description:
We are given three columns of integers with a row for each node. The first two columns contain x and y coordinates of the node positions in a plane. The third column contains node costs. The goal is to select exactly 50% of the nodes (if the number of nodes is odd we round the number of nodes to be selected up) and form a Hamiltonian cycle (closed path) through this set of nodes such that the sum of the total length of the path plus the total cost of the selected nodes is minimized. The distances between nodes are calculated as Euclidean distances rounded mathematically to integer values. The distance matrix should be calculated just after reading an instance and then only the distance matrix (no nodes coordinates) should be accessed by optimization methods to allow instances defined only by distance matrices.

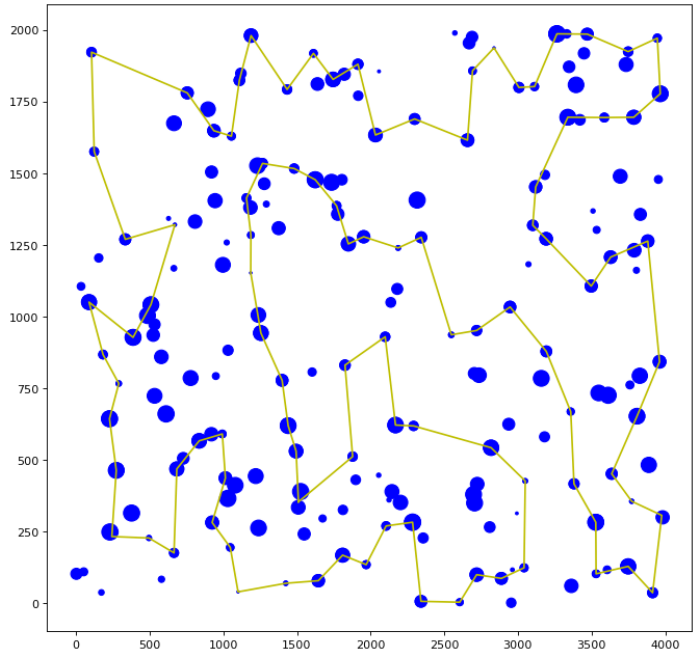
Pseudocode:

1. Create a greedy cycle for the initial node using NN heuristic
2. While cycle length is less than floor(all nodes / 2) do:
    2.1 calculate change in distance that would occur when inserting a new node into the cycle
    2.2 calculate k-regret for each possible new node using previously calculated change in distance
    2.3 determine which node to add (one with the greatest regret) and in what place (best possible place for this node to be added i.e. smallest cost)
    2.4 add the new node into the cycle

TESTS:

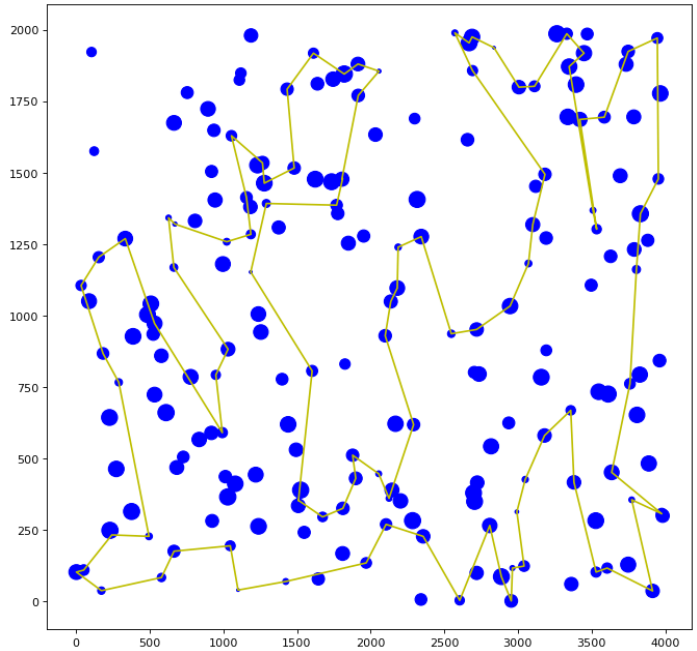
Set A

Unweighted



MINIMUM: 104829
MAXIMUM: 124764
AVERAGE: 116240.25

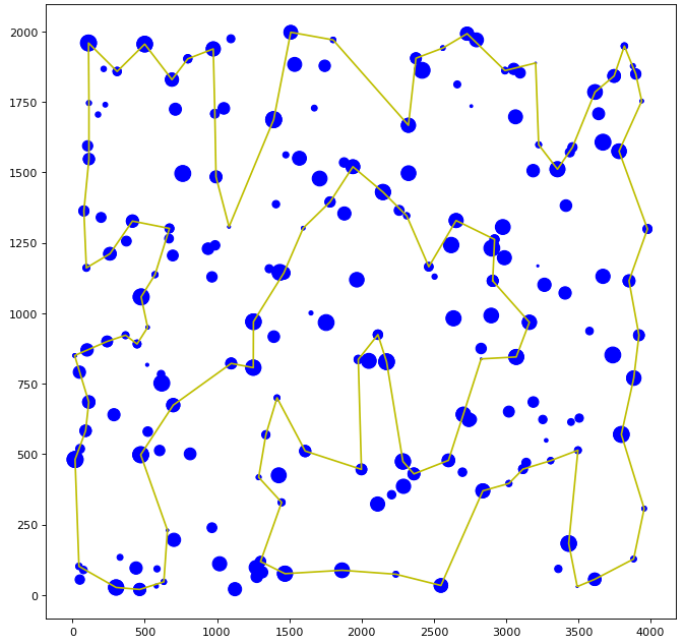
Weighted



MINIMUM: 75052
MAXIMUM: 78929
AVERAGE: 76698.215

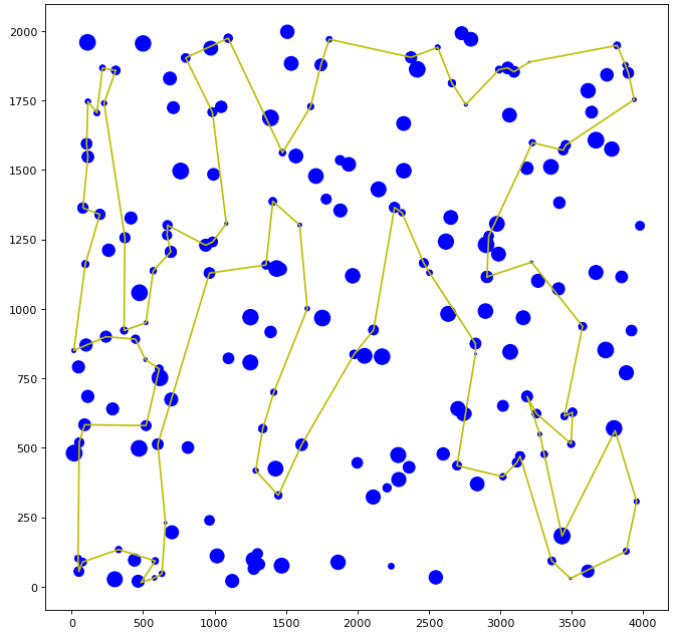
Set B

Unweighted



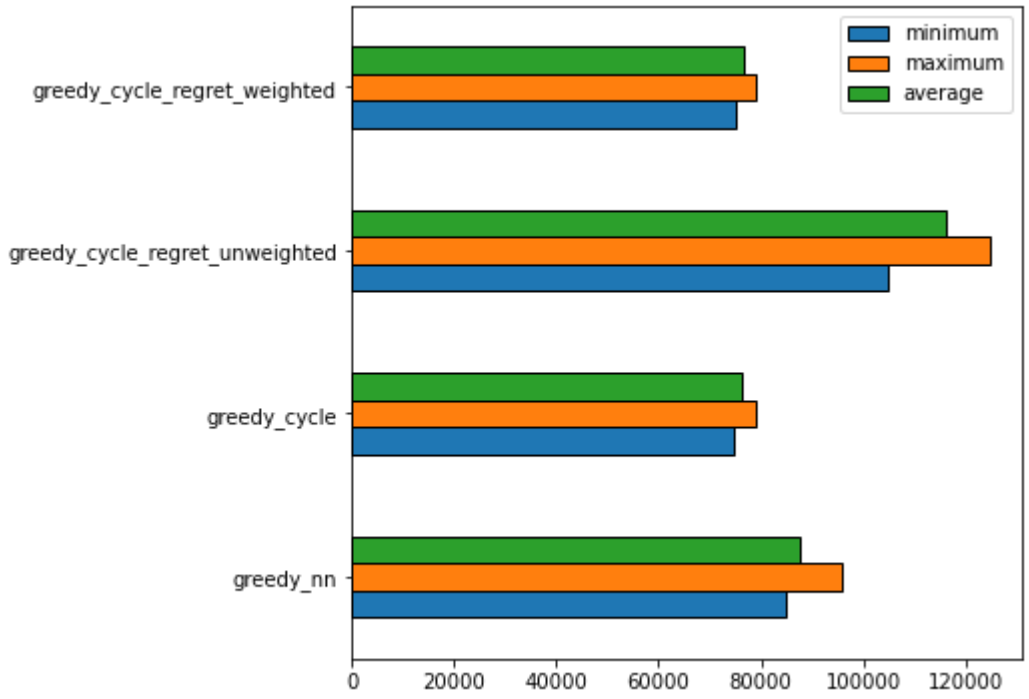
MINIMUM: 109774
MAXIMUM: 128550
AVERAGE: 118806.91

Weighted

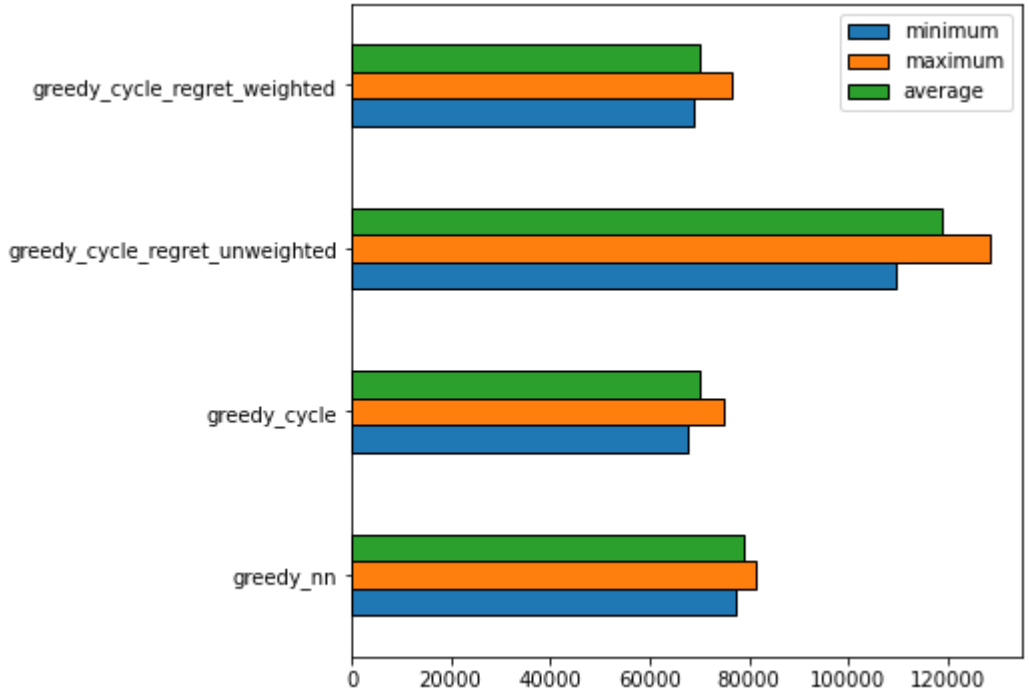


MINIMUM: 68743
MAXIMUM: 76640
AVERAGE: 70150.105

Comparison of different methods for set A:



Comparison of different methods for set B:



Time Complexity of different methods:

Greedy NN - ~0.015 seconds
Greedy Cycle - ~0.8 seconds
Greedy Cycle with Regret - ~5.89 seconds

Conclusions:

Greedy Cycle with Regret without any weights performs far worse than the previous two methods but when weights are added then it performs on par with a base Greedy Cycle, though the time complexity is an order of magnitude bigger.