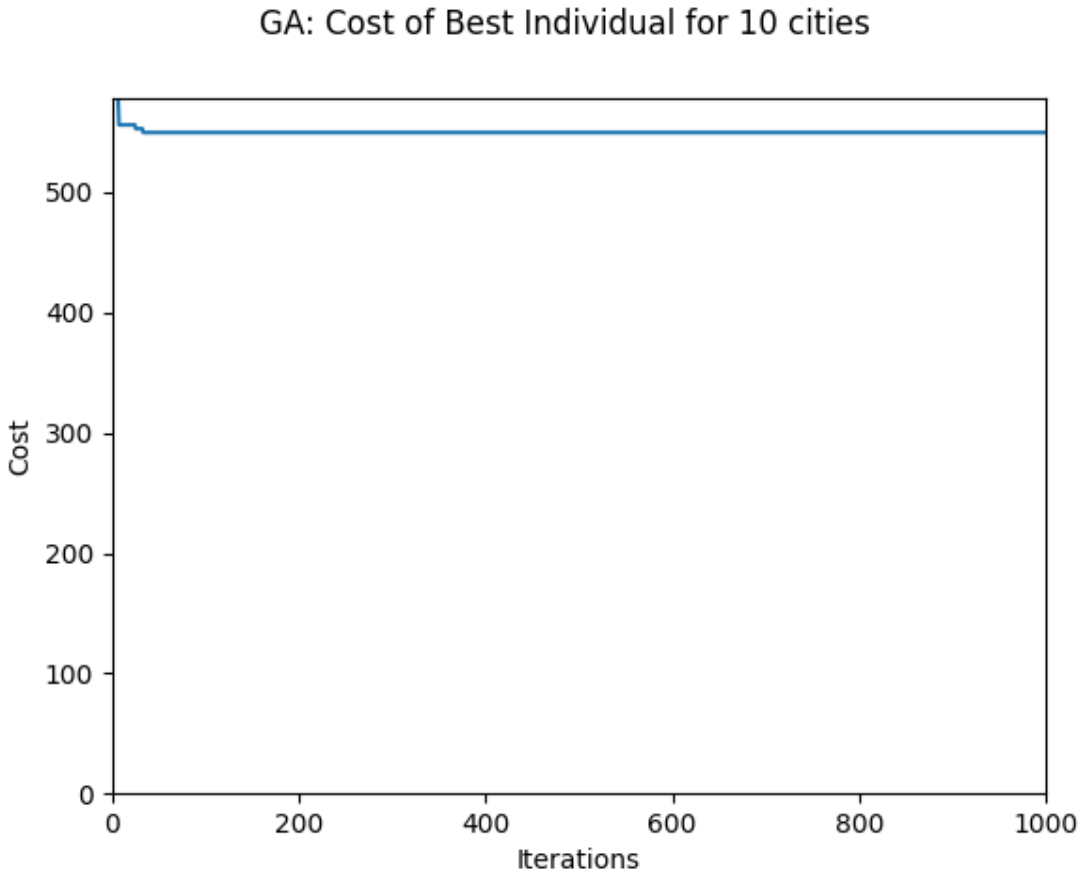
Implementation of Standard Genetic Algorithm to solve the Traveling Salesman Problem

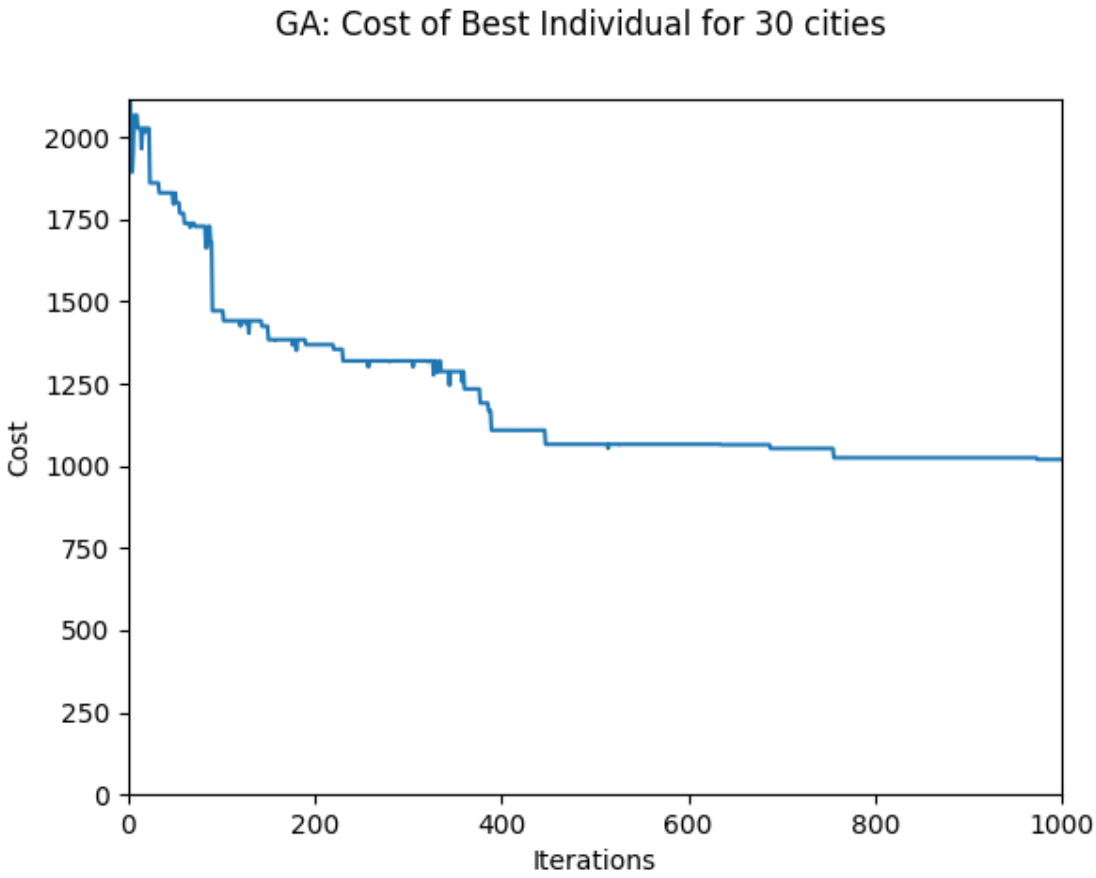
Result on 10 cities




```
0: (123,108)
3: (123,99)
7: (116,26)
4: (144,197)
1: (129,179)
2: (84,147)
6: (54,119)
9: (30,143)
8: (14,77)
5: (23,5)
0: (123,108)
```

Best tour cost: 549.7103044520712
Iterations: 1000
Population size: 200
Mutation rate: 0.01

Result on 30 cities

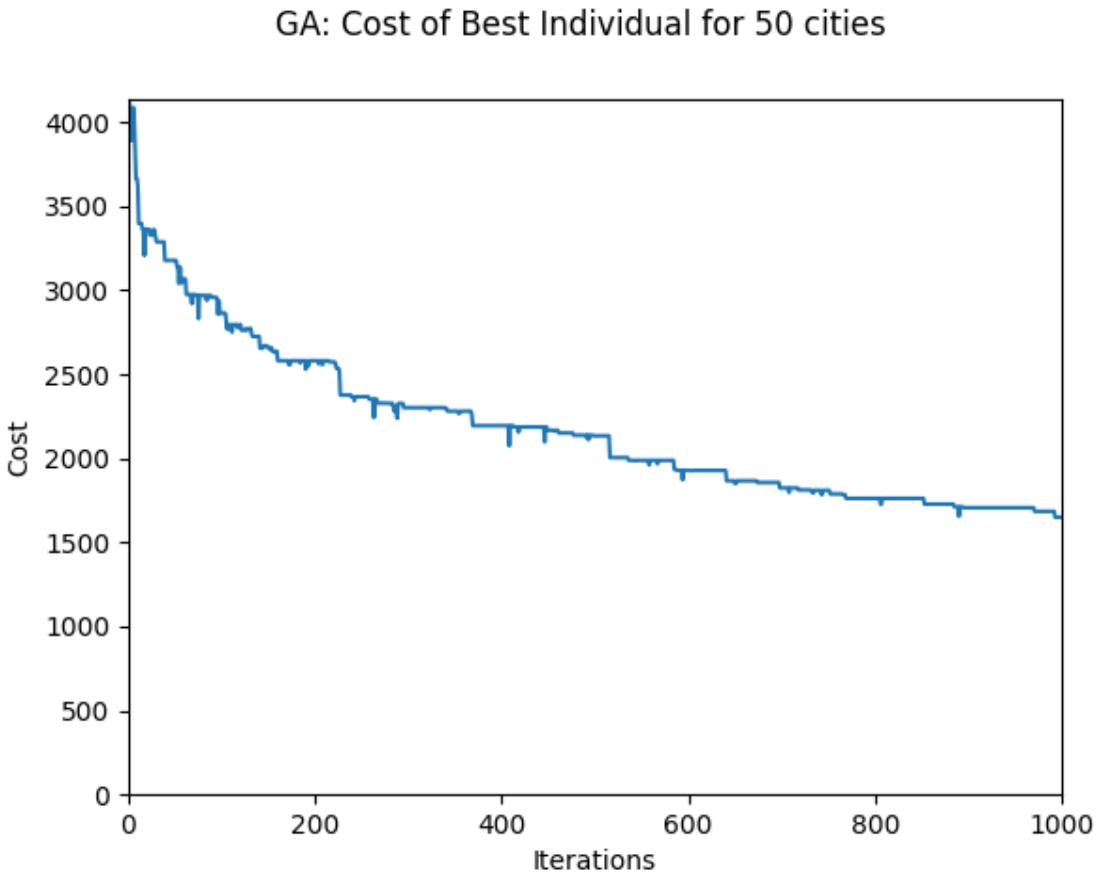



```
0: (159,198)
26: (155,187)
24: (153,175)
28: (164,156)
4: (108,183)
2: (105,198)
29: (88,179)
22: (84,155)
8: (72,160)
11: (68,154)
23: (40,160)
10: (2,103)
18: (94,111)
6: (158,110)
14: (173,104)
13: (165,81)
27: (96,93)
7: (74,85)
3: (46,60)
19: (14,53)
9: (2,13)
1: (40,17)
15: (64,5)
25: (91,48)
```

17: (123,67)
5: (173,32)
12: (195,4)
21: (171,1)
20: (163,11)
16: (123,6)
0: (159,198)

Best tour cost: 1019.8094354781998
Iterations: 1000
Population size: 200
Mutation rate: 0.01

Result on 50 cities




```
0: (80,119)
16: (82,122)
18: (86,123)
33: (94,149)
25: (95,159)
15: (80,162)
49: (50,199)
31: (34,189)
28: (37,151)
47: (33,118)
37: (52,107)
29: (48,83)
34: (65,55)
2: (57,37)
43: (170,7)
22: (135,22)
17: (124,30)
42: (132,57)
9: (155,71)
12: (143,84)
4: (177,92)
3: (193,118)
24: (141,137)
20: (139,149)
```

26: (196,192)
45: (191,147)
27: (164,112)
41: (160,107)
32: (145,93)
5: (145,86)
30: (98,123)
7: (120,85)
40: (70,39)
10: (59,34)
8: (91,13)
36: (65,3)
21: (37,0)
23: (35,42)
6: (11,19)
46: (7,37)
35: (18,77)
48: (37,106)
19: (18,69)
44: (47,46)
14: (59,56)
13: (92,88)
11: (62,153)
1: (47,189)
39: (127,181)
38: (124,185)
0: (80,119)

Best tour cost: 1650.1485130790684
Iterations: 1000
Population size: 200
Mutation rate: 0.01