CMPE 160 ASSIGNMENT 3

Migros Delivery Using Ant Colony Optimization

Name: Ahmet Erdem Bulut

Student ID: 2022400093

Advantages of Ant Colony Optimization:

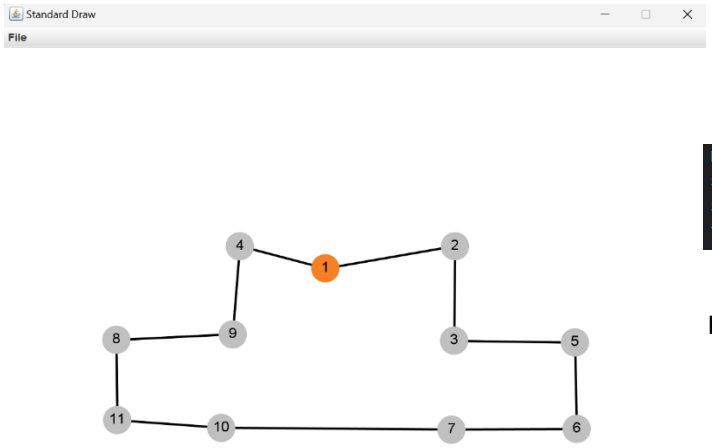
- It is effective in finding good solutions for problems where the search space is complex. In this kind of search space, it will take significantly longer to calculate all permutations and find the shortest path. Hence, ACO becomes more useful due to its fast and near-optimal solutions.
- The pheromone updating mechanism allows for a form of positive feedback, where good solutions found by one ant can guide others toward these paths, which will increase the probability of finding optimal or near-optimal solutions over time.
- It is robust to changes in the problem environment or initial conditions, which makes it suitable for dynamic problems where changes might occur during the execution of algorithm.
- It is flexible and can be adapted to many different optimization problems, such as traveling salesman problem, which this assignment is in a way, scheduling, routing, and many more tasks.

Disadvantages of Ant Colony Optimization:

- It can converge prematurely to suboptimal solutions when there are not enough exploration mechanisms.
- It requires careful tuning of parameters, and its performance is highly sensitive to these settings. Incorrect parameter choices can lead to poor solutions.
- As the problem size increases, the pheromone matrix and the number of required iterations to find a satisfactory solution can scale significantly, which might make it computationally expensive for large problems.

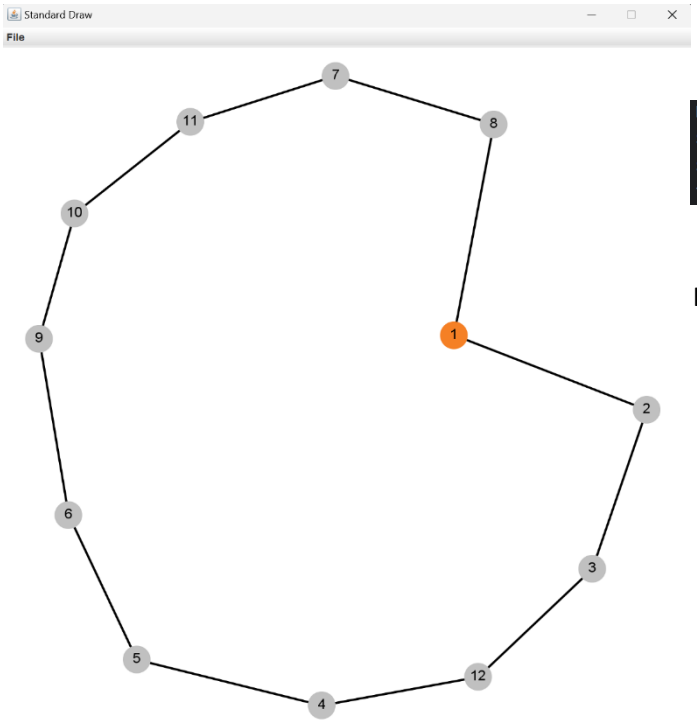
In the following pages, the outputs of the input files for both the ant colony and the brute-force method will be provided.

Brute-Force Outputs



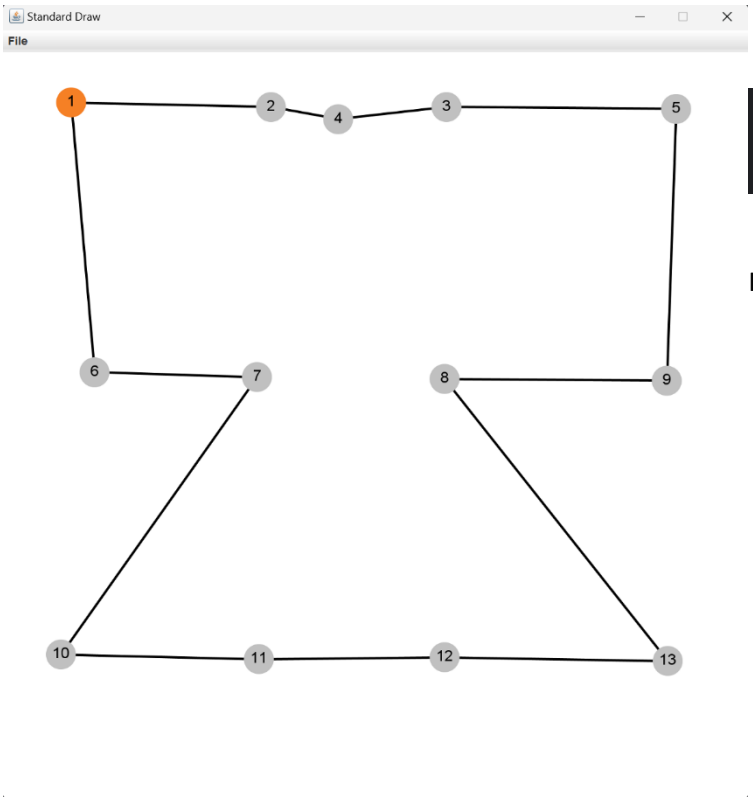

```
Method: Brute-Force Method
Shortest Distance: 1,79529
Shortest path: [1, 4, 9, 8, 11, 10, 7, 6, 5, 3, 2, 1]
Time it takes to find the shortest path: 0,91 seconds
```

Input1



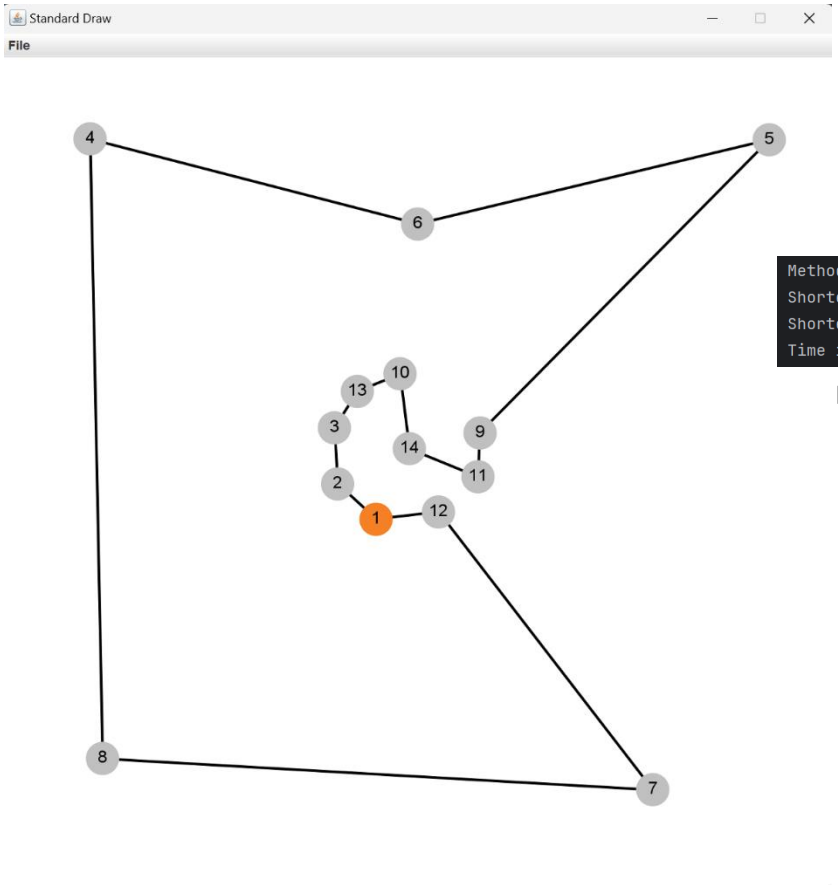

```
Method: Brute-Force Method
Shortest Distance: 2,93588
Shortest path: [1, 8, 7, 11, 10, 9, 6, 5, 4, 12, 3, 2, 1]
Time it takes to find the shortest path: 3,41 seconds
```

Input2




```
Method: Brute-Force Method
Shortest Distance: 3,80292
Shortest path: [1, 2, 4, 3, 5, 9, 8, 13, 12, 11, 10, 7, 6, 1]
Time it takes to find the shortest path: 37,47 seconds
```

Input3

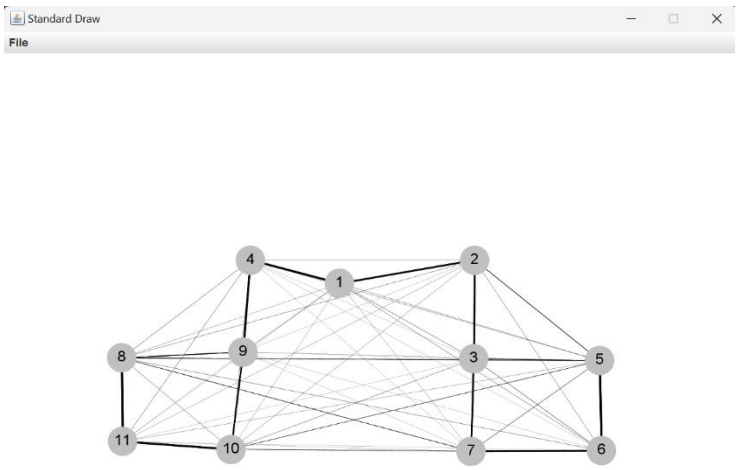



```
Method: Brute-Force Method
Shortest Distance: 3,71091
Shortest path: [1, 2, 3, 13, 10, 14, 11, 9, 5, 6, 4, 8, 7, 12, 1]
Time it takes to find the shortest path: 550,67 seconds
```

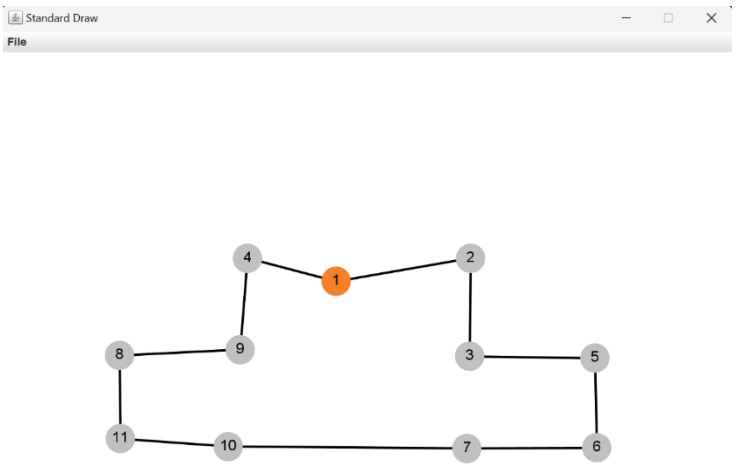
Input4

It takes too long to compute input5 with brute force.

Ant Colony Outputs

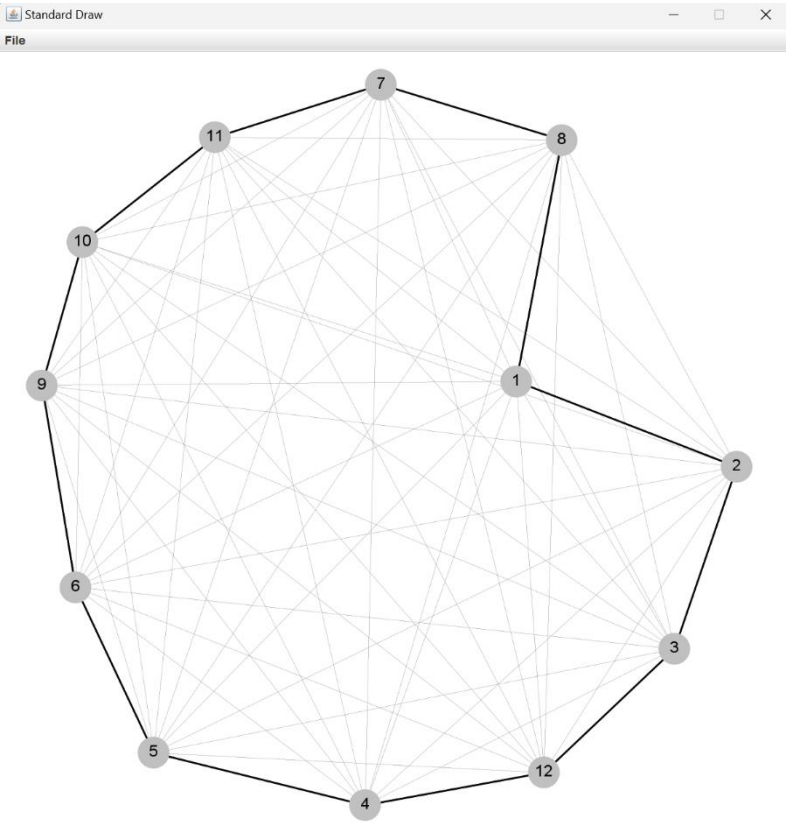


Pheromone Intensities for Input 1

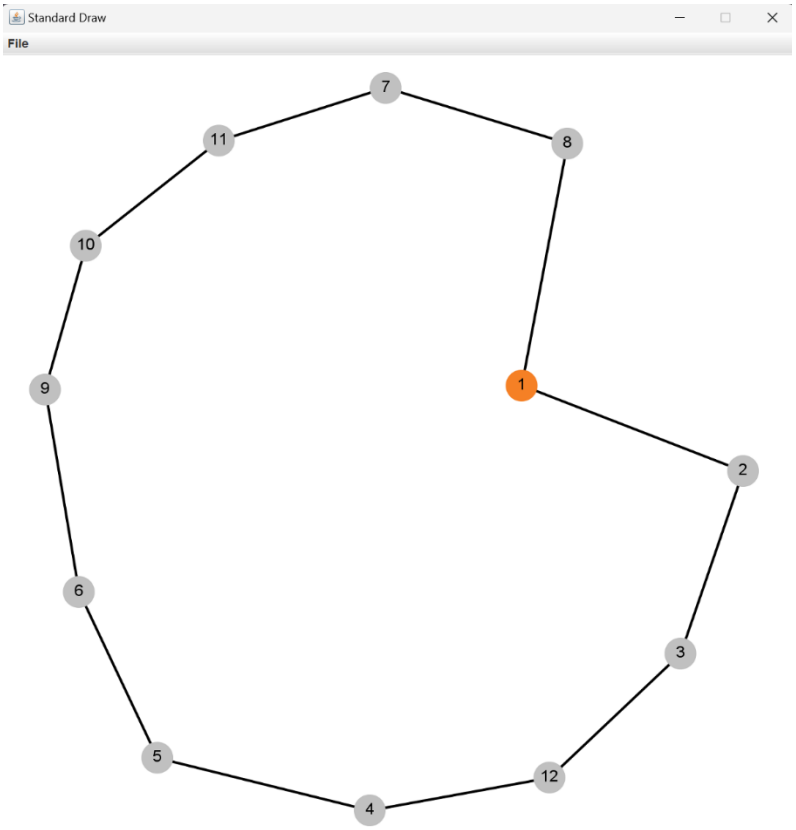


Shortest Path for Input 1

```
Method: Ant Colony Optimization
Shortest Distance: 2,93588
Shortest path: [1, 8, 7, 11, 10, 9, 6, 5, 4, 12, 3, 2, 1]
Time it takes to find the shortest path: 0,57 seconds
```

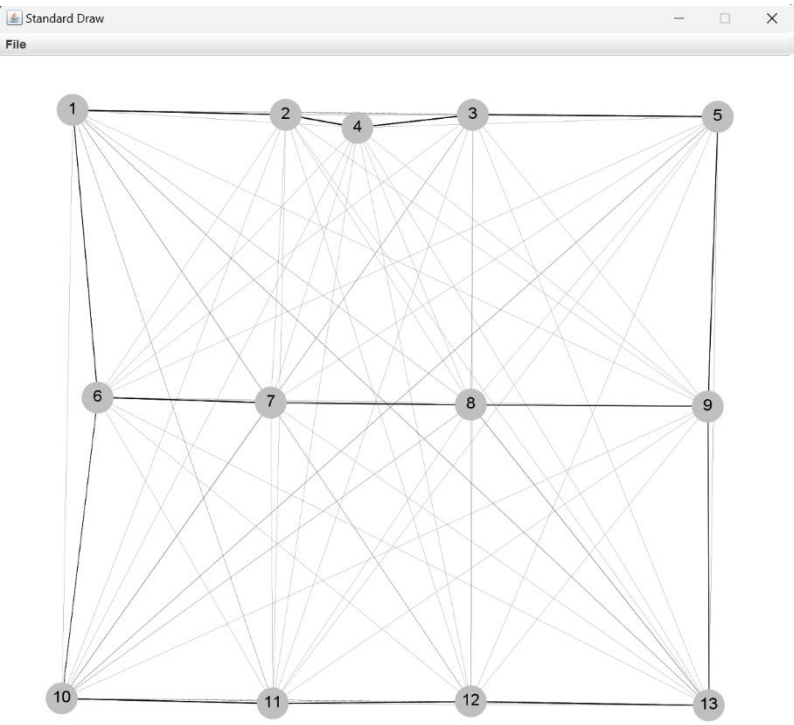


Pheromone Intensities for Input 2

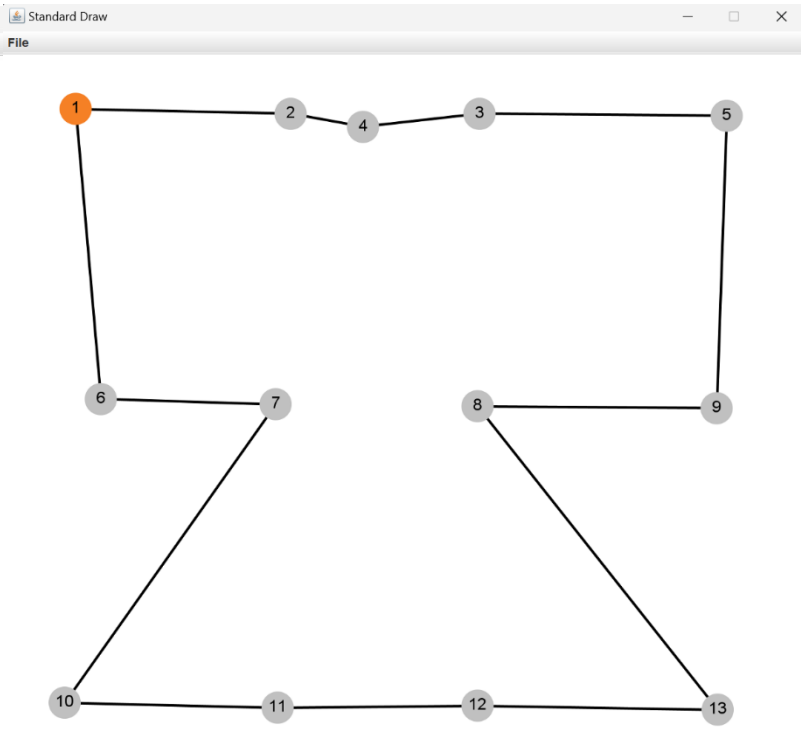


Shortest Path for Input 2

```
Method: Ant Colony Optimization
Shortest Distance: 2,93588
Shortest path: [1, 8, 7, 11, 10, 9, 6, 5, 4, 12, 3, 2, 1]
Time it takes to find the shortest path: 0,57 seconds
```

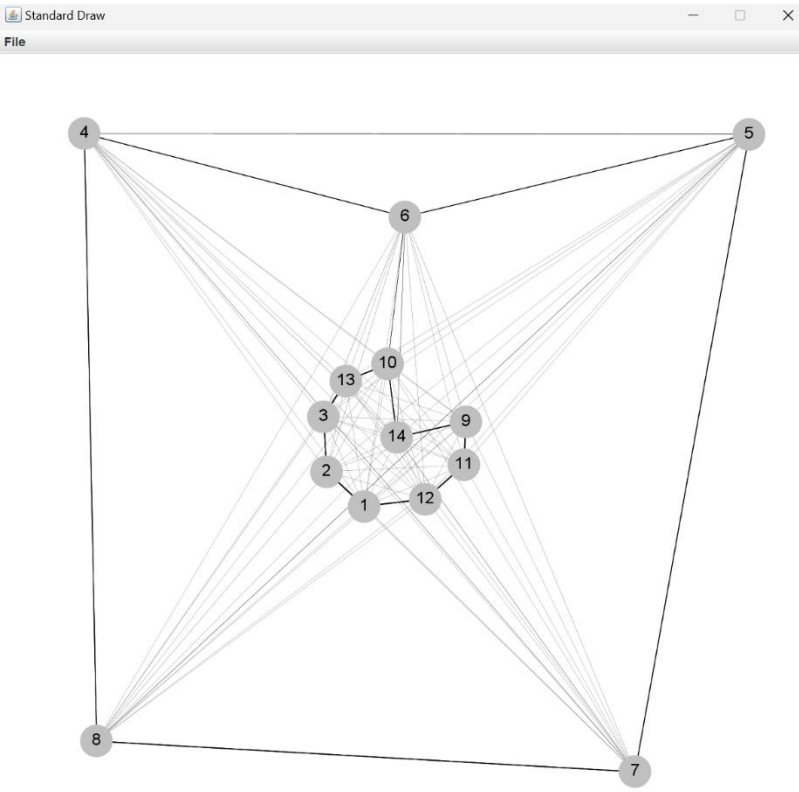


Pheromone Intensities for Input 3

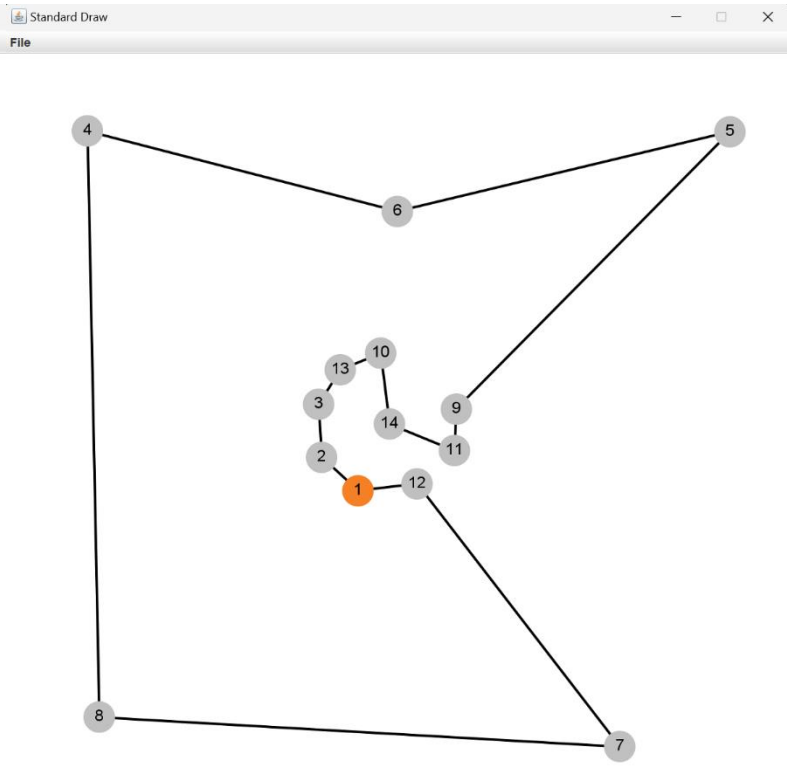


Shortest Path for Input 3

```
Method: Ant Colony Optimization
Shortest Distance: 3,80292
Shortest path: [1, 2, 4, 3, 5, 9, 8, 13, 12, 11, 10, 7, 6, 1]
Time it takes to find the shortest path: 0,52 seconds
```

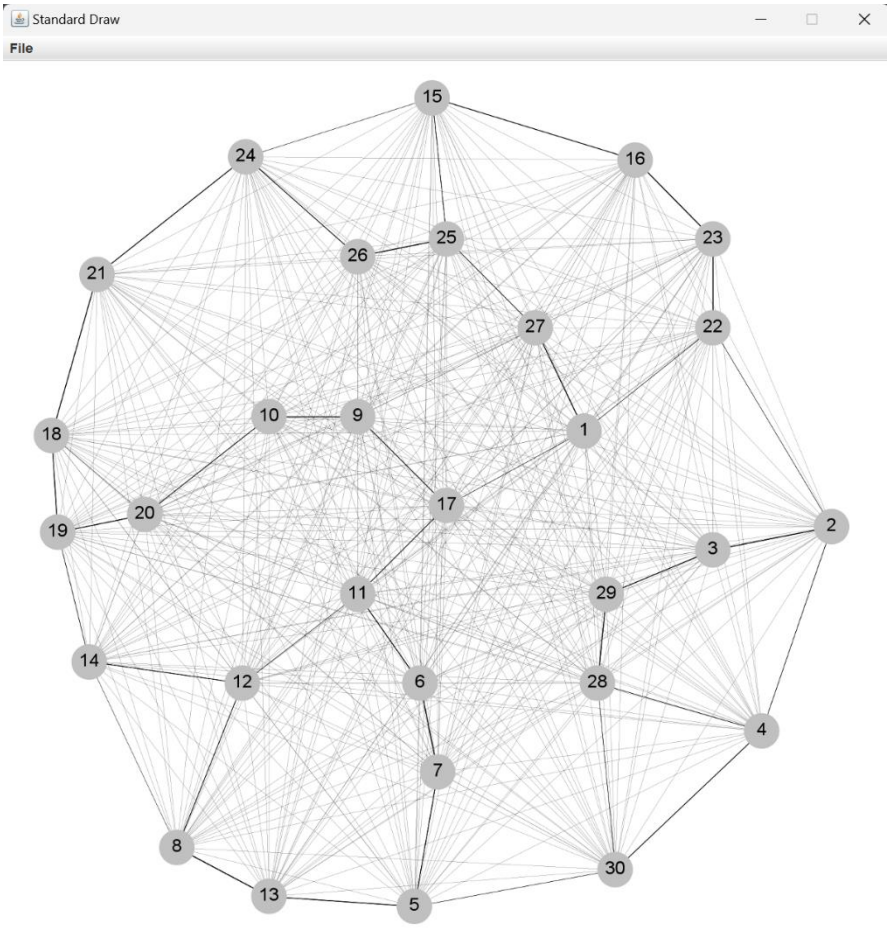


Pheromone Intensities for Input 4

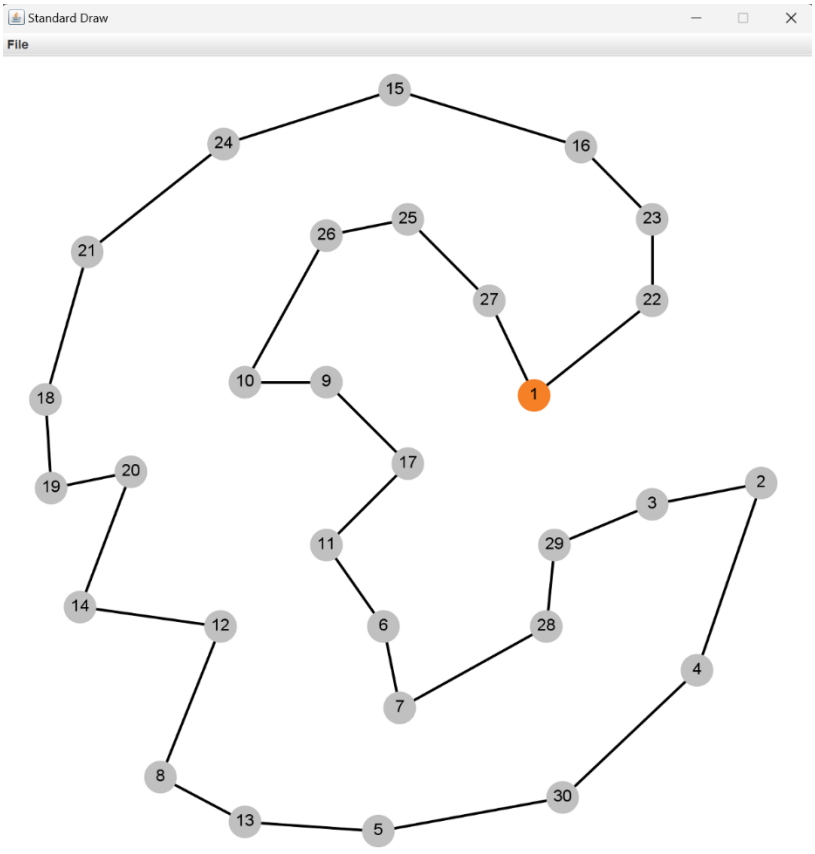


Shortest Path for Input 4

```
Method: Ant Colony Optimization
Shortest Distance: 3,71091
Shortest path: [1, 2, 3, 13, 10, 14, 11, 9, 5, 6, 4, 8, 7, 12, 1]
Time it takes to find the shortest path: 0,51 seconds
```



Pheromone Intensities for Input 5



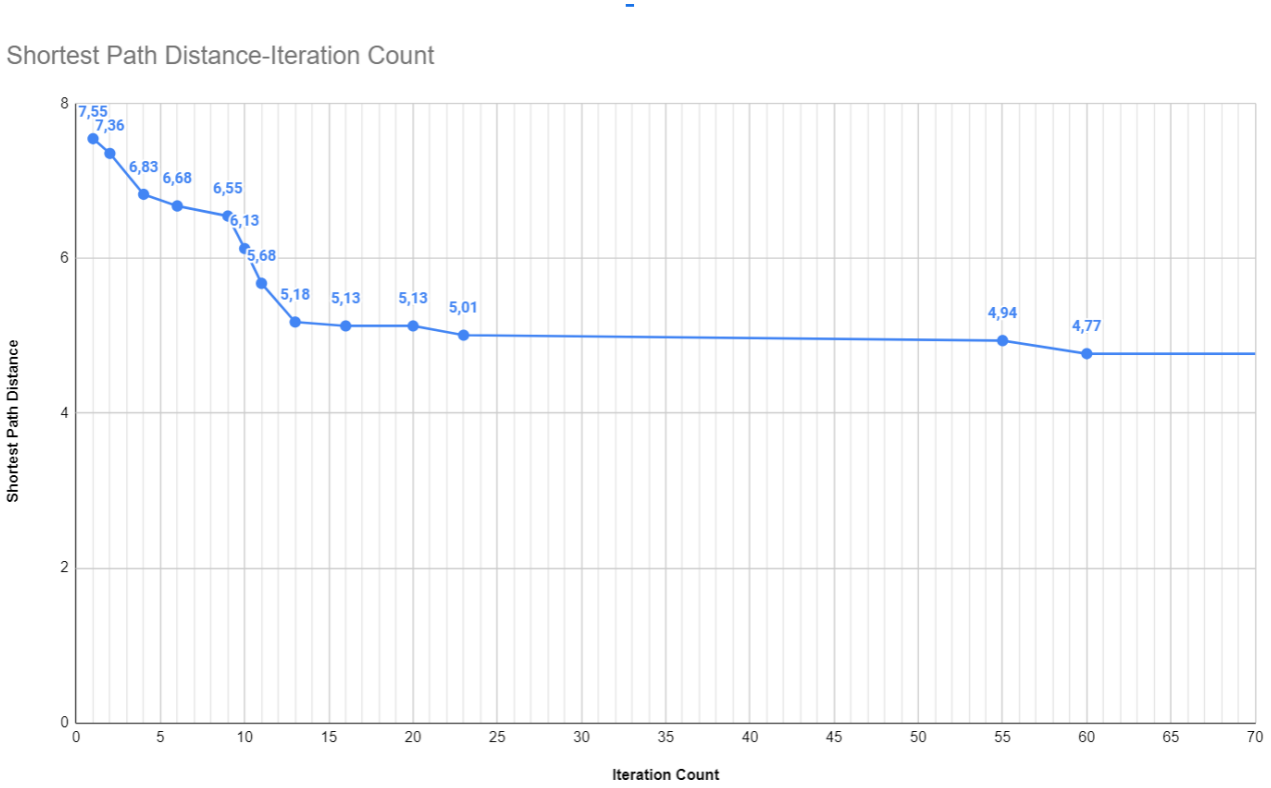
Shortest Path for Input 5

```
Method: Ant Colony Optimization
Shortest Distance: 4,77101
Shortest path: [1, 27, 25, 26, 10, 9, 17, 11, 6, 7, 28, 29, 3, 2, 4, 30, 5, 13, 8, 12, 14, 20, 19, 18, 21, 24, 15, 16, 23, 22, 1]
Time it takes to find the shortest path: 0,88 seconds
```

Input File	Number of Houses + Migros	Brute-Force Time (seconds)	Ant Colony Time (seconds)	Speed Up Factor
Input1	11	0.91 (Distance: 1.79529)	0.57 (Distance: 1.79529)	1.6
Input2	12	3.41 (Distance: 2.93588)	0.57 (Distance: 2.93588)	6
Input3	13	37.47 (Distance: 3.80292)	0.52 (Distance: 3.80292)	72
Input4	14	550.67 (Distance: 3.71091)	0.51 (Distance: 3.71091)	1080
Input5	30	Too long	0.88 (Distance: 4,77101)	Too much

```
private static final double ALPHA = 0.9;
1 usage
private static final double BETA = 2.3;
1 usage
private static final double DEGRADATION = 0.5;
1 usage
private static final double Q = 0.0001;
1 usage
private static final double INIT_PHEROMONE = 0.1;
1 usage
private static final double NUMBER_OF_ITERATIONS = 150;
1 usage
private static final double ANT_PER_ITERATION = 100;
```

I got near-optimal values for input 5 and optimal values for the other input files with these parameters.



Best distance vs Iteration graph for input 5.

Sources:

1. https://www.geeksforgeeks.org/introduction-to-ant-colony-optimization/
2. https://www.youtube.com/watch?v=u7bQomllcJw&t=825s