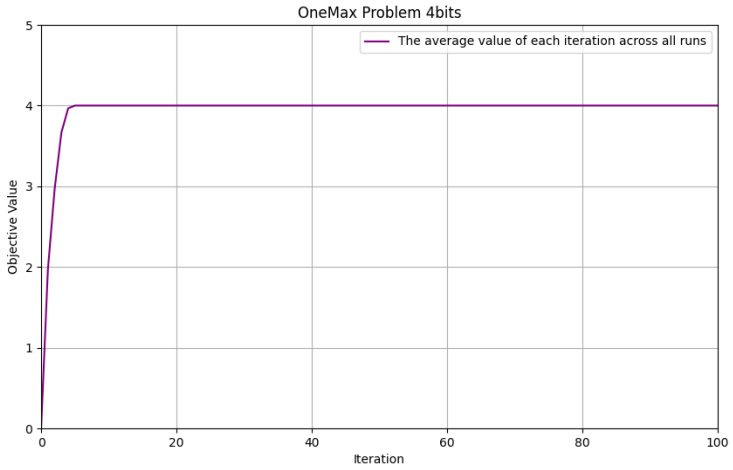
(1) Tabu Search

A. OneMax Problem

I. 4bits, 30runs, 5000evaluations

```
allen@DESKTOP-TKIN1M5:/mnt/c/Users/User/Desktop/Lab/Week4$ ./main 4 30 5000 OneMax
```

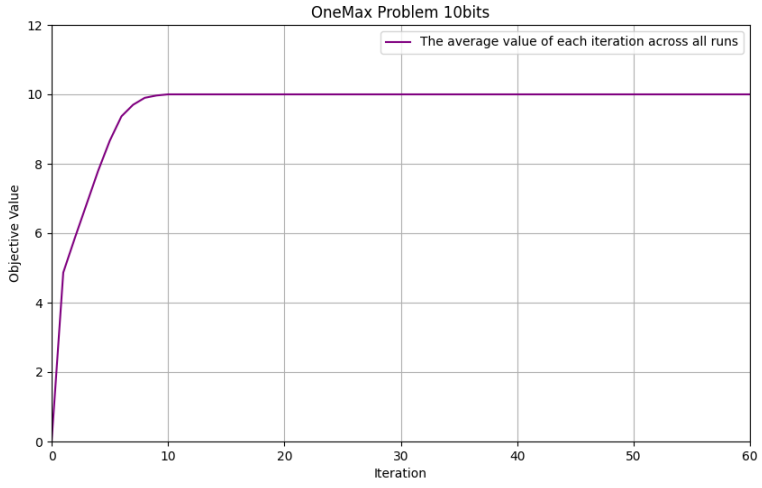
```
Run: 30, NFES: 50, Best Value: 4, Best Solution: 1111
Run: 30, NFES: 50, Best Value: 4, Best Solution: 1111
Run: 30, NFES: 50, Best Value: 4, Best Solution: 1111
```



II. 10bits, 30runs, 5000evaluations

```
allen@DESKTOP-TKIN1M5:/mnt/c/Users/User/Desktop/Lab/Week4$ ./main 10 30 5000 OneMax
```

```
Run: 30, NFES: 49, Best Value: 10, Best Solution: 1111111111
Run: 30, NFES: 50, Best Value: 10, Best Solution: 1111111111
Run: 30, NFES: 51, Best Value: 10, Best Solution: 1111111111
```

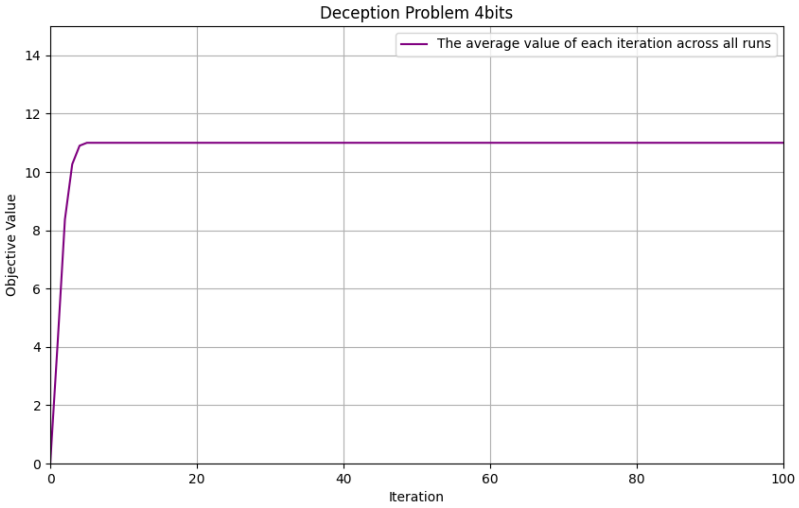


B. Deception Problem

I. 4bits, 30runs, 5000evaluations

```
allen@DESKTOP-TKIN1M5:/mnt/c/Users/User/Desktop/Lab/Week4$ ./main 4 30 5000 Deception
```

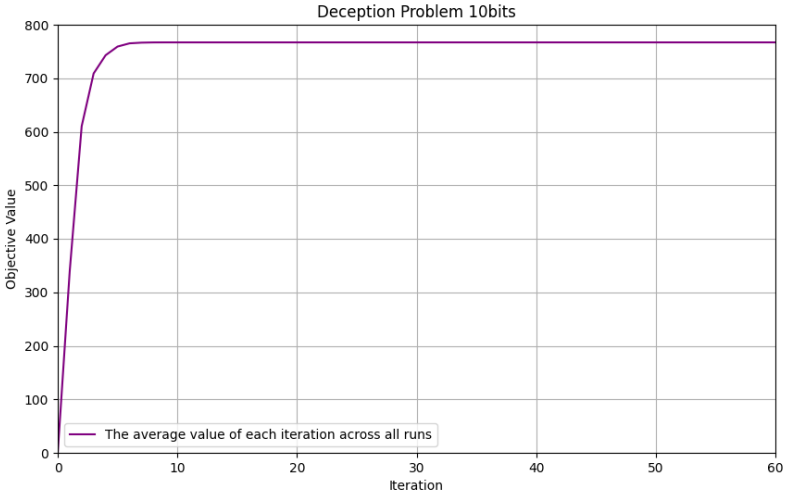
```
Run: 30, NFES: 50, Best Value: 11, Best Solution: 1111
Run: 30, NFES: 50, Best Value: 11, Best Solution: 1111
Run: 30, NFES: 50, Best Value: 11, Best Solution: 1111
```



II. 10bits, 30runs, 5000evaluations

```
allen@DESKTOP-TKIN1M5:/mnt/c/Users/User/Desktop/Lab/Week4$ ./main 10 30 5000 Deception
```

```
Run: 30, NFES: 49, Best Value: 767, Best Solution: 1111111111
Run: 30, NFES: 50, Best Value: 767, Best Solution: 1111111111
Run: 30, NFES: 50, Best Value: 767, Best Solution: 1111111111
```



(2)Ant Colony Optimization

1 run

200 iterations

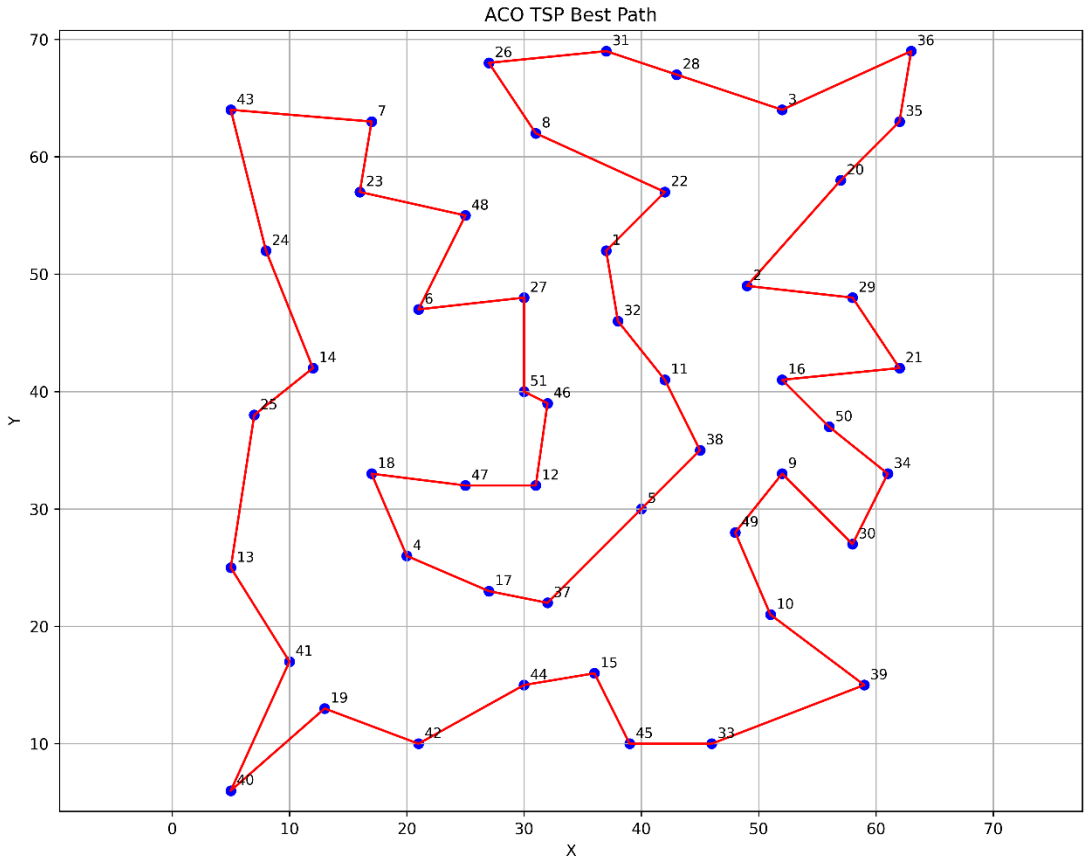
100 ants

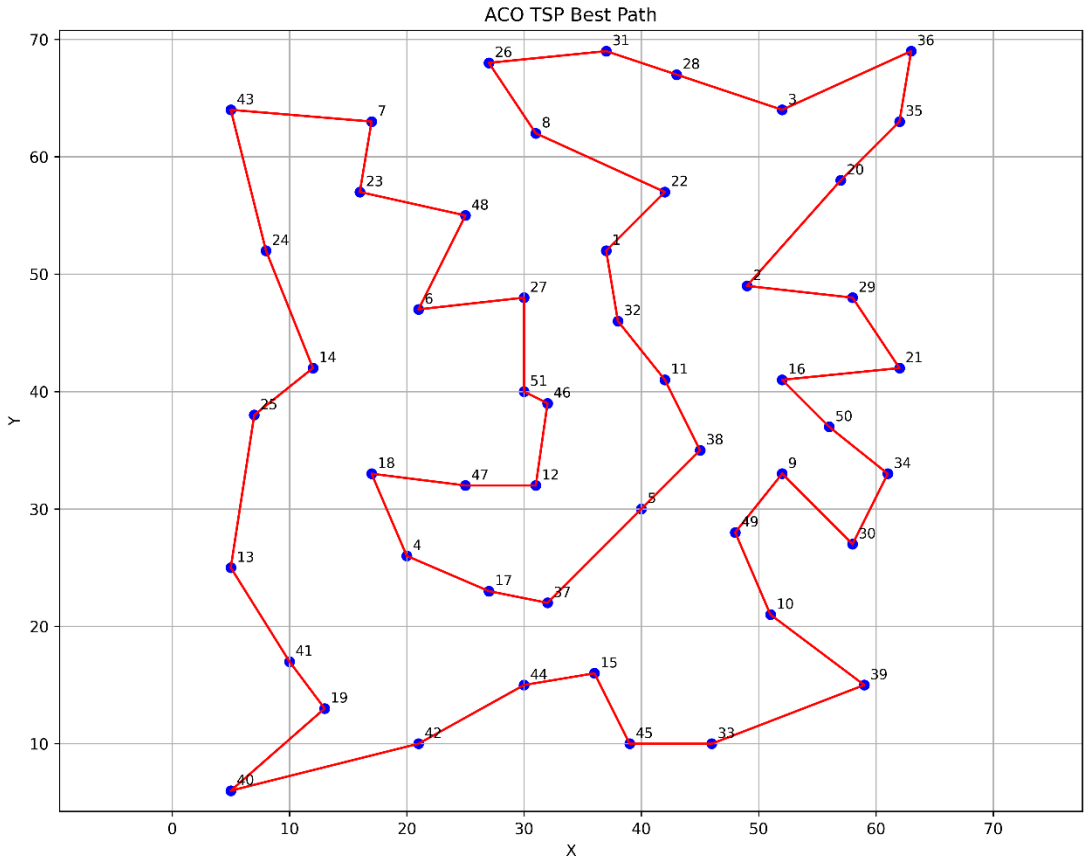
Alpha = 2.3, beta = 4.5, evaporation rate = 0.05, Q = 1, default pheromone on path = 0.5, τ_min = 0.3, τ_max = 5

```
allen@DESKTOP-TKIN1M5:/mnt/c/Users/User/Desktop/Lab/Week4/Ant Colony Optimization$ make
g++ -std=c++17 -Wall -O2 -c main.cpp -o main.o
g++ -std=c++17 -Wall -O2 -c Alg.cpp -o Alg.o
g++ -std=c++17 -Wall -O2 -o main main.o Alg.o
```

```
allen@DESKTOP-TKIN1M5:/mnt/c/Users/User/Desktop/Lab/Week4/Ant Colony Optimization$ ./main 1 200 100 2.3 4.5 0.05 1
```

```
Iteration 198 Best length: 426
Iteration 199 Best length: 426
Iteration 200 Best length: 426
Final best tour length: 426
Tour: 9 49 10 39 33 45 15 44 42 40 19 41 13 25 14 24 43 7 23 48 6 27 51 46 12 47 18 4 17 37 5 38 11 32 1 22 8 26 31 28 3 36 35 20 2 29 21 16 50 34 30
```





此兩張不同路徑花費距離均為 426，差別在左下角。

Discussion:

根據網路上找的資料，426 的確為 eil51 之最佳解，只是我發現長度 426 不是只有唯一的一條路徑，我一開始在算兩點距離時沒有四捨五入，後來把不同 426 之路徑算法改成不四捨五入長度大概 429.多，但我在跑沒有四捨五入的時候基本上都有 428.多的，將 428.多的幾條路徑換成要四捨五入的話，他們的長度會變 427，但基於 TSPLIB 裡定義距離要用四捨五入，所以本次作業我最後是用四捨五入算距離。

```
xd = x[i] - x[j];
yd = y[i] - y[j];
dij = nint( sqrt( xd*xd + yd*yd) );
```

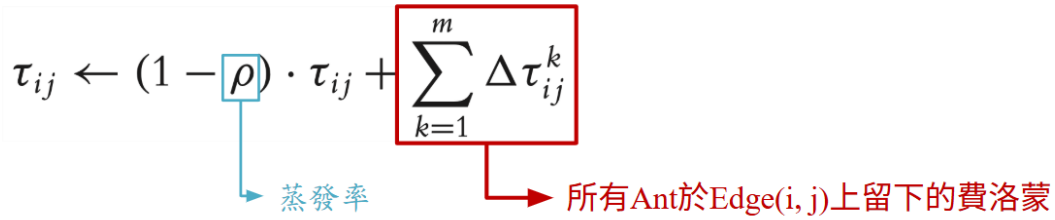
Method:

1. Construct Ant Solutions:

   螞蟻選擇下一步採用的是輪盤法

   突變交換兩點(可選可不選)，本次作業產生的結果我沒選
2. Apply Local Search:

   採用 2-OPT
3. Update Pheromones:



   除了原本基礎的公式外，我還加入以下其他方法：

   (a) 設定路徑上費洛蒙上下限

   (b) 獎勵最優之路徑額外費洛蒙

   (c) 每循環 N 次（例如 10），就對非最優路徑的費洛蒙進行重置。

   參考來源：ACO 優化策略

Conclusion:

上述參數基本上在 200 次迭代內都能找到最優解 426 或次優解 427。