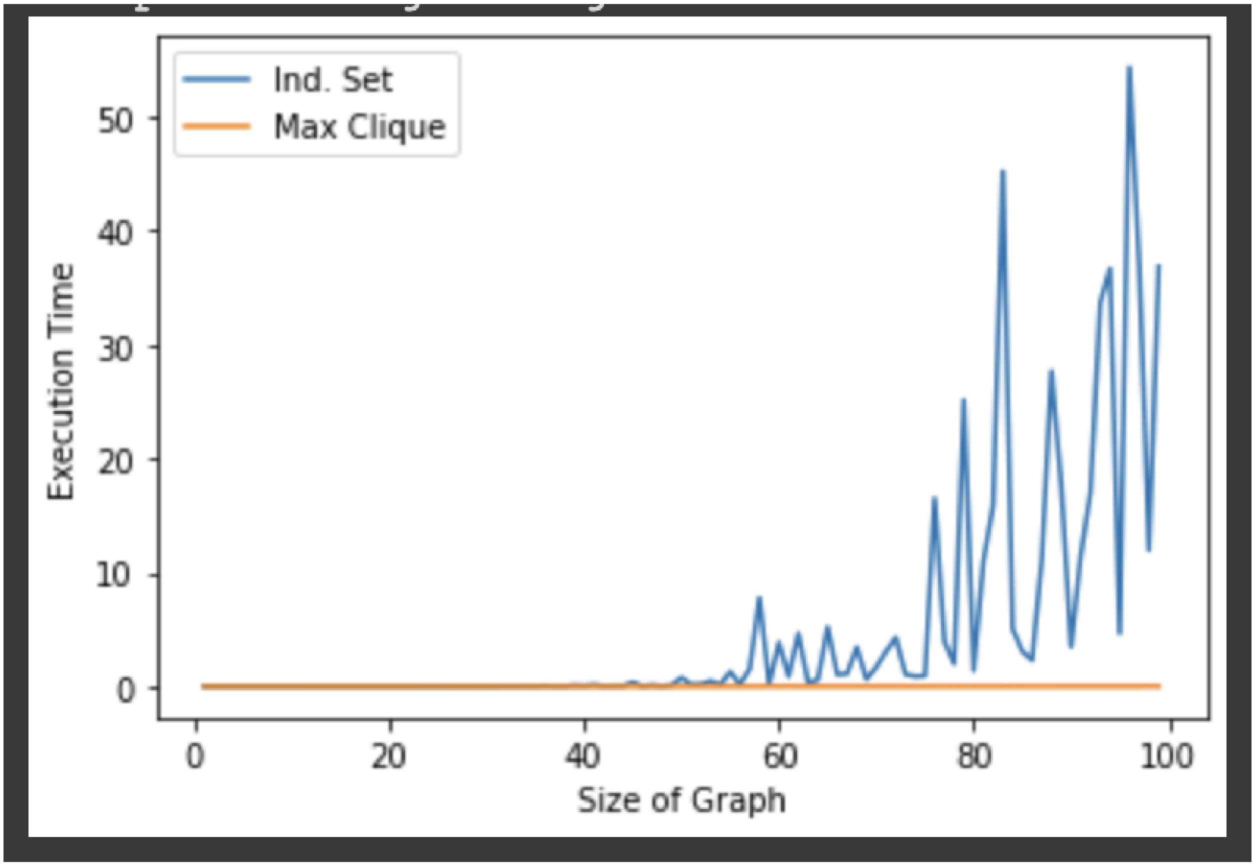
Problem 1

```
Original Array:
[62, 31, 84, 96, 19, 47]
Table for the counts:
[3, 1, 4, 5, 0, 2]
Sorted Array:
[19, 31, 47, 62, 84, 96]


Original Array:
[1, 4, 1, 2, 7, 5, 2]
Table for the counts:
[0, 4, 1, 2, 6, 5, 3]
Sorted Array:
[1, 1, 2, 2, 4, 5, 7]


Original Array:
[5, 2, 9, 5, 2, 3, 5]
Table for the counts:
[3, 0, 6, 4, 1, 2, 5]
Sorted Array:
[2, 2, 3, 5, 5, 5, 9]
```

Problem 2




```
Max ind. set:
1 2 4 6 7
Max clique is:
7 8 9
```

```
Max ind. set:
1 2 7 9
Max clique is:
2 3 5 6
```

The algorithm for the maximum independent set is a recursive algorithm. There are two situations, either the current node will be a part of the maximum independent set or it won't. In both cases, recursive functions are called. Thus, the time complexity of the algorithm for the maximum independent set is O(2^n).

The algorithm for the maximum clique has three embedded for loops which can go on till n statements.Hence the theoretical time complexity of the algorithm is O(n^3).

Problem 3

The Greedy approach is having time complexity O(n^2logn) as after considering every node, the algorithm checks whether the cycle has been completed or not. For the nearest neighbor heuristic algorithm, the given time complexity is O(n^2) due to the 2 embedded loops.

As one can see in the results of the test cases, we come across the same minimum cost for both of the algorithms. Thus the ratio between the weight of the tour by B / the weight of the tour by A is 1. It can also be observed that the nearest neighbor heuristic is in a way, another greedy algorithm for the Traveling Salesperson Problem.

```
The minimum cost is:  275
The minimum cost is:  80
The total cost is: 275
The total cost is: 80
```