**Assignment 3: Optimization of City Transportation Network**

**Minimum Spanning Tree Analysis Report**

Student: Galidullayeva Aliya, SE-2407

**1. Summary of Input Data and Algorithm Results**

**Input Data Overview**

The analysis was performed on two transportation network graphs:

Graph 1:  
- Vertices: 5 (A, B, C, D, E)  
- Edges: 7  
- Graph Density: 0.7 (dense graph)

Graph 2:  
- Vertices: 4 (A, B, C, D)  
- Edges: 5  
- Graph Density: 0.83 (very dense graph)

**Algorithm Results**

**Graph 1 Results**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | MST Cost | Operations Count | Execution Time (ms) | MST Edges |
| Prim's | 16 | 64 | 1.06 | A-C(3), B-C(2), B-D(5), D-E(6) |
| Kruskal's | 16 | 190 | 0.63 | B-C(2), A-C(3), B-D(5), D-E(6) |

Graph 2 Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | MST Cost | Operations Count | Execution Time (ms) | MST Edges |
| Prim's | 6 | 46 | 0.01 | A-B(1), B-C(2), C-D(3) |
| Kruskal's | 6 | 79 | 0.01 | A-B(1), B-C(2), C-D(3  ) |

Key Observations

- Both algorithms found identical MST costs for both graphs  
- Kruskal's algorithm used more operations but was faster  
- Prim's algorithm used fewer operations but had higher overhead

2. Comparison Between Prim's and Kruskal's Algorithms

Performance Analysis  
  
Prim's: More efficient in operations (up to 66% fewer)  
Kruskal's: Faster execution (up to 40% faster)  
  
Prim's uses more memory due to adjacency lists and queues.  
Kruskal's is simpler and memory-light using sorting and Union-Find.  
  
Strengths/Weaknesses summarized as:  
- Prim's: Best for dense graphs, but higher memory use  
- Kruskal's: Simpler, better for sparse graphs

3. Conclusions and Recommendations

Choose Prim's for dense graphs; Kruskal's for sparse.  
Graph density heavily impacts performance balance.  
Prim's: Better for dense networks like computer topologies.  
Kruskal's: Better for sparse systems like transport or social networks.

4. References

1. Cormen et al. (2009). Introduction to Algorithms, MIT Press.  
2. Sedgewick & Wayne (2011). Algorithms, Addison-Wesley.  
3. Weiss (2012). Data Structures and Algorithm Analysis in Java, Pearson.

Appendix: Implementation Details

Prim's: Uses PriorityQueue, HashSet, etc.  
Kruskal's: Uses ArrayList, sorting, Union-Find.  
  
Complexities:  
- Prim's: O(E log V)  
- Kruskal's: O(E log E)  
- Union-Find: O(α(V))