Kruskal's Algorithm

Vansh Jain 2020A7PS0079U

Introduction

- Kruskal's algorithm is a greedy algorithm used to find the Minimum Spanning Tree (MST) of a weighted, undirected graph. The MST of a graph is the tree that spans all the vertices of the graph, has the minimum total weight, and does not contain any cycles.
- The algorithm works as follows:
  - Sort all the edges in the graph in increasing order of their weight.
  - Initialize an empty set to hold the edges of the MST.
  - For each edge in the sorted list of edges, if adding it to the MST would not create a cycle, add it to the MST. To check if adding the edge creates a cycle, we can use a disjoint-set data structure.
  - Continue this process until there are V-1 edges in the MST, where V is the number of vertices in the graph.
  - Return the set of edges in the MST.

Ayegba, P.O., Okeyinka, A.E., Adebiyi, M., Asani, E.O., Ayoola, J.A. and Ben, G.C., Comparative Performance Analysis of Kruskal and Prim MST Algorithms.

- Objective: To compare the computational speed complexities of Kruskal's and Prim's algorithms for obtaining the minimal spanning tree of a graph.
- Conclusion: Kruskal's algorithm outperformed Prim's algorithm in terms of computational speed for road network datasets of varying sizes.

P. Ayegba, J. Ayoola, E. Asani and A. Okeyinka, "A Comparative Study Of Minimal Spanning Tree Algorithms" 2020

- Objective: To implement and compare the efficiency of Kruskal's and Prim's algorithms for acquiring the minimum spanning tree of a graph, specifically the road network of the 36 states in Nigeria.
- Conclusion: The current study found Kruskal's algorithm to be better for a sparse graph, but a conclusive comparison with Prim's algorithm requires investigation on dense graphs.

B F Melnikov and Y Y Terentyeva "Building communication networks: on the application of the Kruskal's algorithm in the problems of large dimensions" 2021

- Objective: Develop topology for ultra-large communication networks with minimum sum of weights and explore heuristic algorithms as direct Kruskal's algorithm takes too long to operate.

=

- Conclusion: The paper presents two specific variants of Kruskal's algorithm and two heuristics, with one heuristic considered acceptable based on practical results. These approaches can be applied to more complex problems beyond the starting problem.

Y. Zuo "Target Detection System of Agricultural Economic Output Efficiency Based on Kruskal Algorithm "2022

- Objective: To design an agricultural economic output efficiency target detection system using the Kruskal algorithm and compare its stability rate with other algorithms.

- Conclusion: The Kruskal algorithm demonstrated greater stability in the target detection system of agricultural economic output efficiency, making it more applicable and effective than other algorithms.

D. Rachmawati, Herriyance and F. Y. Putra Pakpahan "Comparative Analysis of the Kruskal and Boruvka Algorithms in Solving Minimum Spanning Tree on Complete Graph,"2020

- Objective: To build a system using Kruskal and Boruvka algorithms to find the optimal path that connects all points in a complete graph with the lowest optimization value.

- Conclusion: The Kruskal algorithm is more effective than the Boruvka algorithm in finding the minimum spanning tree in a complete graph with 15 nodes and 105 sides.

Practical Application

- Logistics and delivery: Companies that need to deliver goods or services to multiple locations can use Kruskal's algorithm to find the most efficient route for their delivery vehicles.

- Image segmentation: Used to partition an image into regions of similar features.Kruskal's algorithm can be used to segment images into distinct regions based on the similarity of pixels.

- Spanning tree protocol: Used in computer networking to prevent loops in a network topology. Helps to optimize the use of resources and minimize costs.

Practical Application

- Circuit board design: Kruskal's algorithm can be used to find the minimum spanning tree that connects all components on a circuit board, which can help optimize the design and reduce production costs.

- Vehicle routing: Kruskal's algorithm can be used to optimize vehicle routing problems, such as those faced by delivery companies and public transportation systems, to minimize travel time and fuel costs.

- DNA sequencing: In bioinformatics, Kruskal's algorithm can be used to construct a phylogenetic tree that shows the evolutionary relationships between different organisms based on their DNA sequences.

Algorithm

```
algorithm Kruskal(G) is
    F:= ∅
    for each v ∈ G.V do
        MAKE-SET(v)
    for each (u, v) in G.E ordered by weight(u, v), increasing do
        if FIND-SET(u) ≠ FIND-SET(v) then
            F:= F ∪ {(u, v)} ∪ {(v, u)}
            UNION(FIND-SET(u), FIND-SET(v))
    return F
```

1. Initialize an empty set F to represent the MST of the graph.
2. Create a disjoint set (or forest) for each vertex in the graph using the MAKE-SET operation.
3. Sort the edges of the graph in non-decreasing order of weight.
4. For each edge (u, v) in the sorted order: If FIND-SET(u) is not equal to FIND-SET(v), add the edge to F and merge the two sets using UNION(FIND-SET(u), FIND-SET(v)).
5. Return the set F as the MST of the graph.

Time Complexity Analysis

- E is the number of edges.
- V is the number of vertices in the graph.

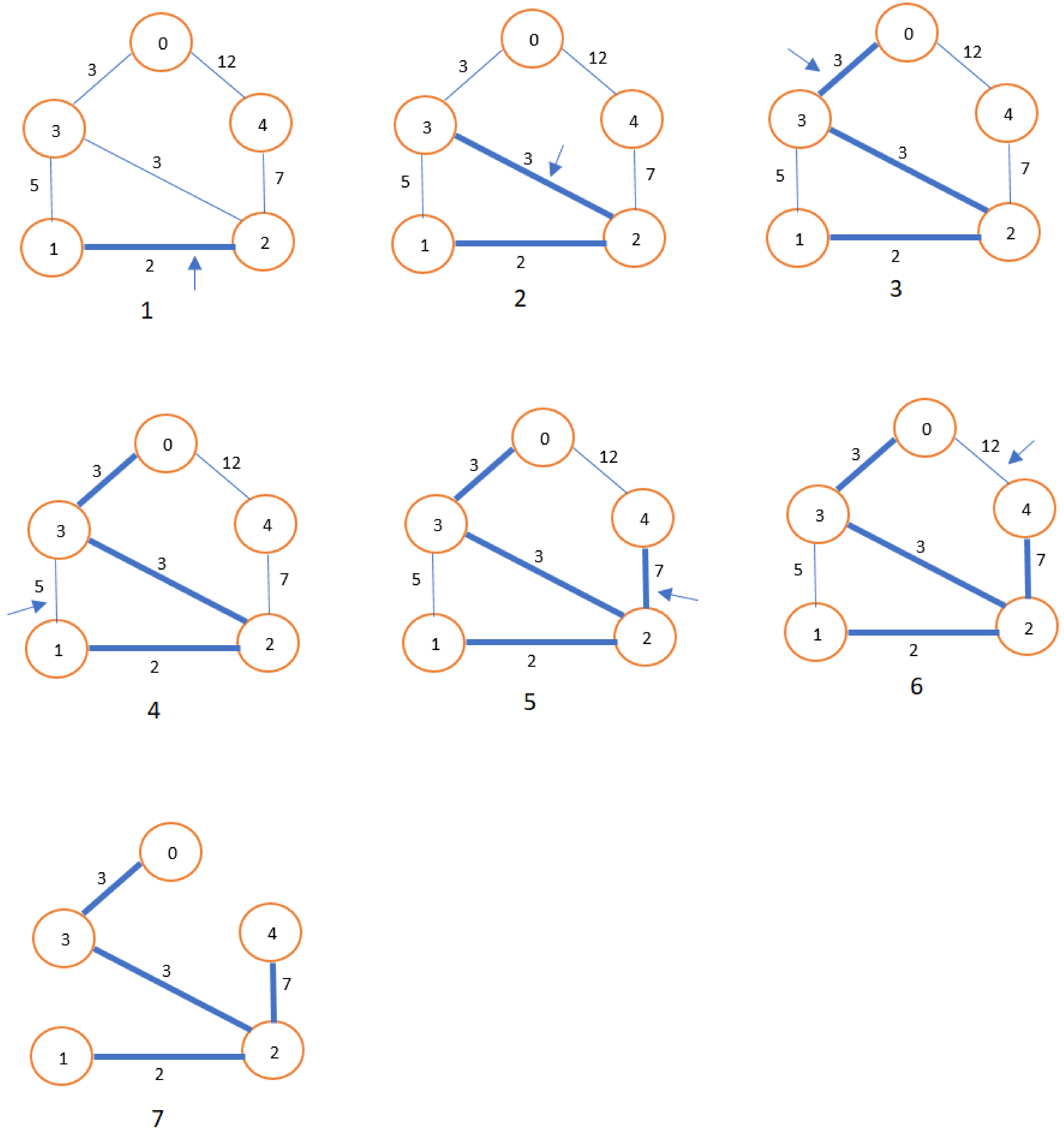
Best Case
- O(E(log(E))

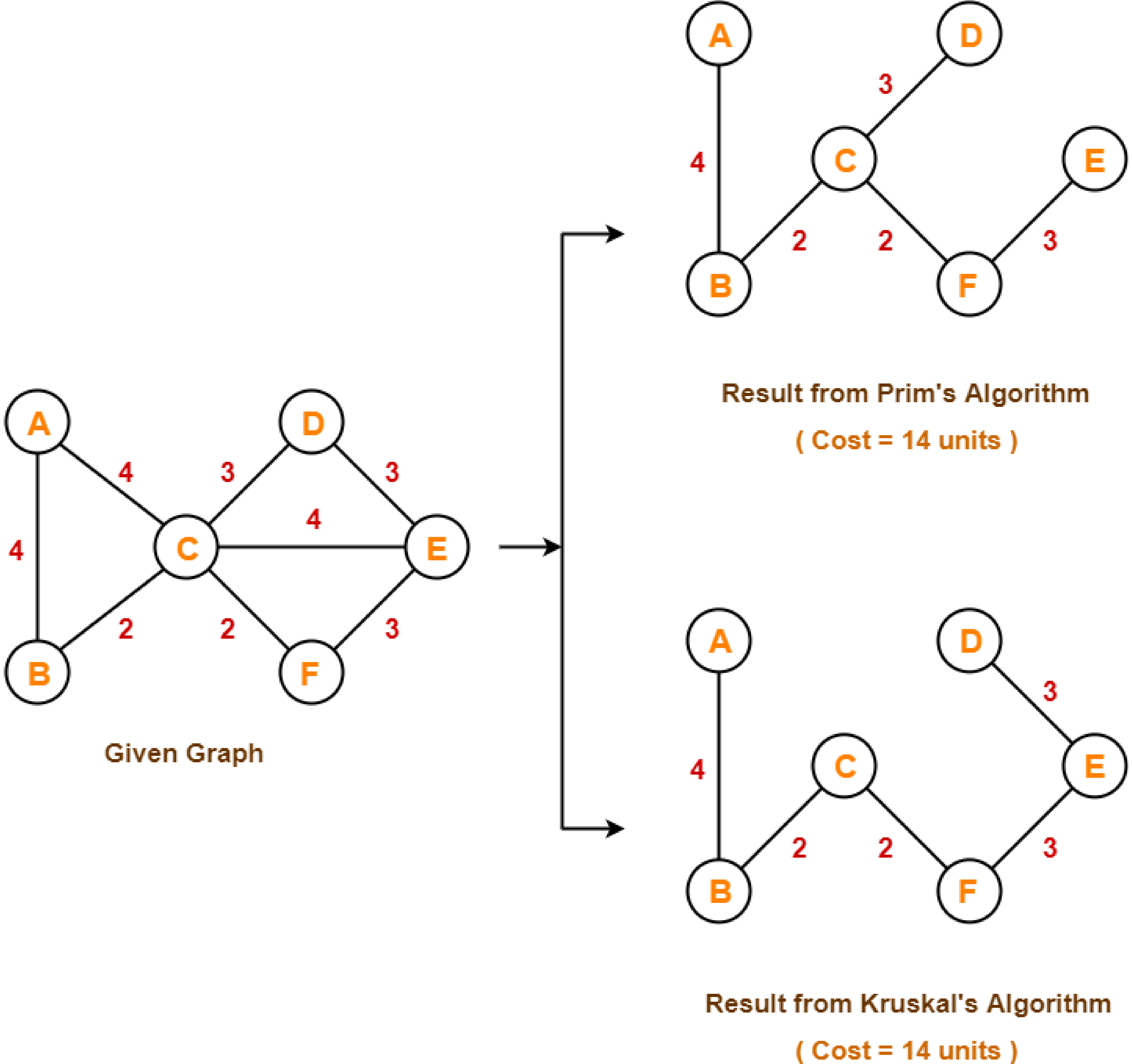
Average case
- O(E(log(E))

Worst Case
- O(E(log(E))

Kruskal's algorithm has a time complexity that is dominated by the sorting operation and is proportional to the number of edges in the graph.

- We can see edge (1, 3) and (0, 4) is not added in minimum spanning tree because if we include any one of these, tree will form cycles which can not be true in case of a tree.
- Or we can understand it by algorithm since vertex 1 and 3 belongs to same set so it won't be added in results and same is for 0 and 4.
- End up with a minimum spanning tree with a total cost of 11 (1 + 2 + 3 + 5).





Given Graph

Result from Prim's Algorithm

(Cost = 14 units)

Result from Kruskal's Algorithm

(Cost = 14 units)

- If all the edge weights are not distinct, then both the algorithms may not always produce the same MST.
- However, cost of both the MSTs would always be same in both the cases.

Comparative Table

	Time Complexity	Space Complexity	Description	Advantages	Disadvatages
Kruskal's Algorithm	O (E log V)	O (V + E)	A greedy algorithm for finding a minimum spanning tree in a weighted graph by iteratively adding the next cheapest edge that does not create a cycle.	Works well for sparse graphs.	Can be slower for dense graphs.
Prism's Algorthim	O ((V + E) log V)	O (V + E)	A greedy algorithm for finding a minimum spanning tree in a weighted graph by iteratively adding the next cheapest edge that connects a vertex in the current tree to a vertex outside of the tree.	Works well for dense graphs.	Can be slower for sparse graphs.

References

- Ayegba, P.O., Okeyinka, A.E., Adebiyi, M., Asani, E.O., Ayoola, J.A. and Ben, G.C., Comparative Performance Analysis of Kruskal and Prim MST Algorithms.
- P. Ayegba, J. Ayoola, E. Asani and A. Okeyinka, "A Comparative Study Of Minimal Spanning Tree Algorithms," 2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS), Ayobo, Nigeria, 2020, pp. 1-4, doi: 10.1109/ICMCECS47690.2020.240900.
- B F Melnikov and Y Y Terentyeva 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1047 012089
- Y. Zuo, "Target Detection System of Agricultural Economic Output Efficiency Based on Kruskal Algorithm," 2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC), Tumkur, Karnataka, India, 2022, pp. 1-5, doi: 10.1109/ICMNWC56175.2022.10031889.
- D. Rachmawati, Herriyance and F. Y. Putra Pakpahan, "Comparative Analysis of the Kruskal and Boruvka Algorithms in Solving Minimum Spanning Tree on Complete Graph," 2020 International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA), Medan, Indonesia, 2020, pp. 55-62, doi: 10.1109/DATABIA50434.2020.9190504.