

CHAPTER 2

ALI AND HIS FAMILY'S NEW BEGINNING

*Nur Syafiqah binti Mohd Kamarol Hisyam, Nur Qamarina
binti Othman, Aziemah binti Mohd Apan, Nadhirah Humaira
binti Mohd Nasir, Adnin Afifi binti Nawi**

Centre for Diploma Studies, Universiti Tun Hussein Onn
Malaysia, Pagoh Higher Education Hub, 84600 Pagoh,
Johor, MALAYSIA.

*Corresponding Email: adnin@uthm.edu.my

2.0 INTRODUCTION

Ali and his family recently moved to a new town and are eager to familiarize themselves with their new surroundings. Excited about their new home, they decide to spend the day exploring the town together. They plan to visit various landmarks, parks, and shopping areas to get a better sense of the community. Wanting to cover each location only once while minimizing their travel distance, they carefully map out their journey. To make their exploration more efficient, they consider using concepts from graph theory, such as the Minimum Spanning Tree (MST), which helps in finding the shortest path that connects all locations with minimum total distance [1]. MST is widely used in network optimization problems, including transportation and communication networks [2].

To determine the most efficient route, Ali and his family explore two well-known algorithms: Prim's and Kruskal's. Prim's algorithm builds the MST by starting from a single node and expanding by selecting the shortest edge at each step

Graph Theory in Action: Solving Real-Life Problems with Prim's and Kruskal Algorithms

[3]. On the other hand, Kruskal's algorithm sorts all edges by weight and progressively adds them to the tree while avoiding cycles [4]. Both algorithms have been extensively studied for their efficiency in finding optimal paths in various applications, including road network design [5] and urban planning [6]. By applying these algorithms, Ali and his family can determine the most efficient way to explore the town while ensuring a fun and memorable experience.

2.1 METHODOLOGY

To help Ali and his family familiarize themselves with their new town, Graph Online [7] was used to create the undirected, weighted graph shown in Figure 2.1. This graph visually represents the town, with the eight nodes corresponding to eight local landmarks. The weight of each edge indicates the distance in kilometres (km) between those landmarks.

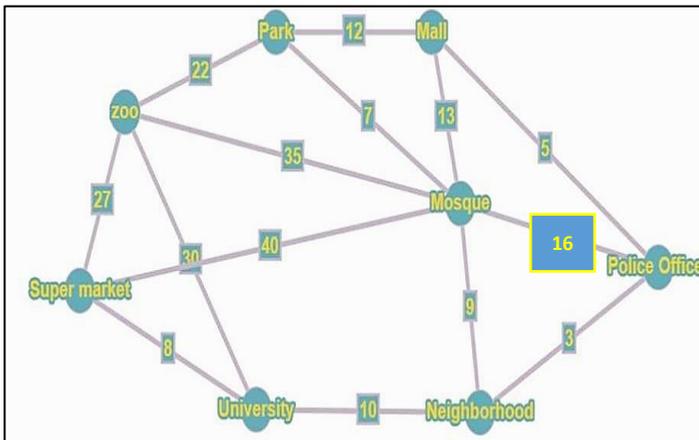


Figure 2.1: An undirected weighted graph representing the maps of the town

Graph Theory in Action: Solving Real-Life Problems with Prim's and Kruskal Algorithms

Here are the abbreviations used for each landmark:

A- mall

B-police office

C-neighbourhood

D-university

E-supermarket

F-zoo

G-park

H-mosque

Figure 2.1 presents a graph comprised of eight vertices and 14 edges. Table 2.1 provides a comprehensive listing of these edges and their associated weights.

Table 2.1: All edges and weight.

No.	Edges	Weight (km)
1	FG	22
2	FH	35
3	FD	30
4	FE	27
5	EH	40
6	ED	8
7	DC	10
8	CH	9
9	CB	3
10	BH	16
11	BA	5
12	HA	13
13	HG	7
14	GA	12
Total weight of all edges		237

As shown in Figure 2.1 and detailed in Table 2.1, the graph consists of 14 edges with a total weight of 237 km. This study

Graph Theory in Action: Solving Real-Life Problems with Prim's and Kruskal Algorithms

aims to minimize the total travel distance to all eight locations, a goal achievable through the identification of a minimum spanning tree. This will be accomplished using both Prim's and Kruskal's algorithms.

2.1.1 PRIM'S ALGORITHM

First, Prim's algorithm will be used to determine the minimum travel distance within the new town. We can begin at any node; for this example, we will start with node E. Initially, only node D connects to E. We then select the edge with the smallest weight connected to either E or D. This process repeats, iteratively adding nodes until all have been traversed, ensuring no cycles are created.

Table 2.2: Calculation by using Prim's algorithm.

Iteration	Edges	Visited vertices	Unvisited vertices	Weight (km)
1	ED	D,E	A,B,C,F,G,H	8
2	DC	C,D,E	A,B,F,G,H	10
3	CB	B,C,D,E	A,F,G,H	3
4	CH	B,C,D,E,H	A,F,G	9
5	HG	B,C,D,E,G,H	A,F	7
6	GF	B,C,D,E,F,G,H	A	22
7	BA	A,B,C,D,E,F,G,H	-	5
Total weight				64

Table 2.2 shows that the shortest network connecting all eight landmarks involves seven edges with a combined length of 64 km.

2.1.2 KRUSKAL ALGORITHM

This subsection demonstrates how to find the minimum travel distance using Kruskal's algorithm. This algorithm starts by

Graph Theory in Action: Solving Real-Life Problems with Prim's and Kruskal Algorithms

selecting the edge with the smallest weight. In this case, that is edge BC with a weight of 3 km. The next shortest edge is AB, at 5 km. This process continues, selecting the next shortest edge and adding it to the growing network, until all nodes are included, while ensuring that no cycles are created.

Table 2.3: Calculation by using Kruskal algorithm

Iteration	Edges	Visited vertices	Unvisited vertices	Weight (km)
1	BC	B,C	A,D,E,F,G,H	3
2	BA	A,B,C	D,E,F,G,H	5
3	HG	A,B,C,G,H	D,E,F	7
4	DE	A,B,C,D,E,G,H	F	8
5	CH	A,B,C,D,E,G,H	F	9
6	CD	A,B,C,D,E,G,H	F	10
7	GF	A,B,C,D,E,F,G,H	-	22
Total weight				64

Based on Table 2.3, the minimum spanning tree connecting all eight landmarks is calculated to be 64 km, comprised of seven edges.

2.2 MINIMUM SPANNING TREE

Figure 2.2 shows the minimum spanning tree (MST) generated by both Prim's and Kruskal's algorithms, which has a total weight of 64 km. This MST adheres to the required criteria: it includes $(n-1)$ edges, where 'n' represents the number of vertices (landmarks), and it minimizes the total edge weight for any spanning tree connecting these locations.

Graph Theory in Action: Solving Real-Life Problems with Prim's and Kruskal Algorithms

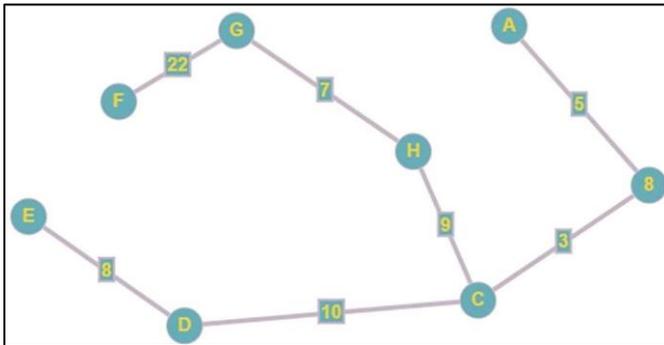


Figure 2.2: Minimal spanning tree for Ali's problem

2.3 CONCLUSION

Using both Prim's and Kruskal's algorithms, the minimum total distance needed to visit all the landmarks in Ali's new town was calculated by constructing a minimum spanning tree. This solution helps Ali and his family determine the most efficient routes to travel between all the locations.

2.4 REFERENCES

- [1] Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms* (3rd ed.). MIT Press.
- [2] Tarjan, R. E. (1983). *Data Structures and Network Algorithms*. SIAM.
- [3] Prim, R. C. (1957). *Shortest connection networks and some generalizations*. Bell System Technical Journal, 36(6), 1389-1401.
- [4] Kruskal, J. B. (1956). *On the shortest spanning subtree of a graph and the traveling salesman problem*. Proceedings of the American Mathematical Society, 7(1), 48-50.

**Graph Theory in Action: Solving Real-Life Problems with Prim's and
Kruskal Algorithms**

- [5] Zhang, H., & Wang, Y. (2015). *Application of minimum spanning tree algorithms in road network optimization*. *Journal of Transport Systems*, 12(3), 223-235.
- [6] Gao, L., Chen, X., & Li, P. (2018). *Urban road network planning using graph theory and optimization algorithms*. *International Journal of Urban Planning*, 25(4), 67-82.
- [7] GraphOnline. (n.d.). *GraphOnline*. Retrieved January 31, 2025, from <https://graphonline.top/en/>