CHAPTER 4

NETWORK CABLE INSTALLATION IN A UNIVERSITY COMPOUND

*Aqilah binti Mohd Yusof, Elya Sufiyah binti Iskandar, Nur Zawanah binti Zainizam, Nur Qistina binti Nasrudin, Nur Aleaya Firzani binti Khairul Faizi, 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

4.0 INTRODUCTION

A university plans to connect several buildings, including the cafeteria, laboratory, library, and other facilities, using network cables while minimizing the total cable length. This problem can be modelled as a graph, where buildings represent vertices and the possible cable connections between them are edges [1]. To achieve an optimal solution, a spanning tree, which is a subset of the graph that connects all vertices without forming cycles, is required [2]. Specifically, the goal is to find a minimum spanning tree (MST), which ensures that all buildings are connected using the least possible total cable length [3]. Two well-known algorithms for finding the MST are Prim's algorithm and Kruskal's algorithm. Prim's algorithm starts from an arbitrary node and grows the tree by adding the smallest edge that connects a new vertex to the tree [4,5]. On the other hand, Kruskal's algorithm sorts all edges by weight and continuously adds the shortest edge that does not form a cycle until all vertices are connected [6].

Both algorithms are widely used in network design due to their efficiency in minimizing costs while ensuring full connectivity [7].

4.1 METHODOLOGY

A graph of the university compound as illustrated in Figure 4.1 was constructed using Graph Online [8]. This tool was selected for its user-friendly interface, ability to handle edge list input, helpful visualization and editing features, and advanced analytical capabilities, including minimum spanning tree calculations. These features made it well-suited for the graph analysis performed. The resulting graph's ten nodes correspond to the university's ten buildings, with edges representing potential cable routes between buildings. The weights assigned to each edge reflect the distance (in meters) between the connected buildings.

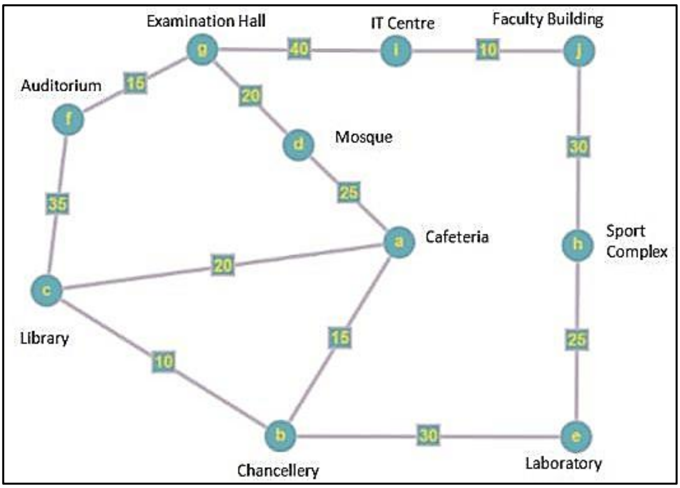

Figure 4.1: An undirected weighted graph representing the university compound

The following list defines the abbreviations used to represent each building.
a-cafeteria
b-chancellery
c-library
d-mosque
e-laboratory
f-auditorium
g-examination hall
h-sport complex
i-IT centre
j-faculty building

The graph in Figure 4.1 consists of ten vertices and 12 edges. Table 4.1 provides a clear listing of these edges and their corresponding weights.

Table 4.1: All edges and weight

No.	Edges	Weight (m)
1	fc	35
2	fg	15
3	gd	20
4	gi	40
5	da	25
6	ab	15
7	ac	20
8	bc	10
9	be	30
10	eh	25
11	hj	30
12	ji	10
Total weight of all edges		275

Based on Table 4.1, the total weight of all 12 edges of the graph in Figure 4.1 is 275 m.

This study seeks to minimize the total cable length needed to connect all ten university buildings, a problem that can be solved by finding a minimum spanning tree. Both Prim's and Kruskal's algorithms will be used to achieve this

4.1.1 PRIM'S ALGORITHM

In this subsection, Prim's Algorithm was used to determine the minimum total edge weight. This algorithm starts with an arbitrary vertex; here, node f was selected. It then iteratively adds the edge with the smallest weight that is connected to the existing tree (initially just node f). So, the edge connecting f to g, which has the smallest weight, is added first. This process continues, always selecting the smallest-weight edge that connects a new vertex to the existing tree, until all vertices are included, and no cycles are formed.

Table 4.2: Calculation by using Prim's algorithm

Iteration	Edges	Visited vertices	Unvisited vertices	Weight (m)
1	fg	f,g	d,a,c,b,e,h,j,i	15
2	gd	f,g,d	a,c,b,e,h,j,i	20
3	da	f,g,d,a	c,b,e,h,j,i	25
4	ab	f,g,d,a,b	c,e,h,j,i	15
5	bc	f,g,d,a,b,c	e,h,j,i	10
6	be	f,g,d,a,b,c,e	h,j,i	30
7	eh	f,g,d,a,b,c,e,h	j,i	25
8	hj	f,g,d,a,b,c,e,h,j	i	30
9	ji	-	-	10
Total weight				180

Based on Table 4.2, the minimum spanning tree constructed using Prim's algorithm connects all ten buildings in the university compound with a total of nine edges and a combined length of 180 m.

4.1.2 KRUSKAL ALGORITHM

This subsection describes the solution using Kruskal's algorithm. This algorithm begins by selecting the edge with the smallest weight; here, that's edge bc at 10 m. Next, edge ji, also at 10 m, is selected. This process of selecting the smallest remaining edge continues until all nodes are connected, ensuring no cycles are created.

Table 4.3: Solution by using Kruskal algorithm

Iteration	Edges	Visited vertices	Unvisited vertices	Weight (m)
1	fg	f,g	d,a,c,b,e,h,j,i	15
2	gd	f,g,d	a,c,b,e,h,j,i	20
3	da	f,g,d,a	c,b,e,h,j,i	25
4	ab	f,g,d,a,b	c,e,h,j,i	15
5	bc	f,g,d,a,b,c	e,h,j,i	10
6	be	f,g,d,a,b,c,e	h,j,i	30
7	eh	f,g,d,a,b,c,e,h	j,i	25
8	hj	f,g,d,a,b,c,e,h,j	i	30
9	ji	-	-	10
Total weight				180

As shown in Table 4.3, Kruskal's algorithm also produced a minimum spanning tree connecting all ten buildings with nine edges and a total length of 180 m.

4.2 MINIMAL SPANNING TREE

The minimum spanning tree (MST) depicted in Figure 4.2, with a total weight of 180 m, was generated by both Prim's and Kruskal's algorithms. This tree successfully connects all nodes without creating any cycles, thus satisfying the definition of an MST.

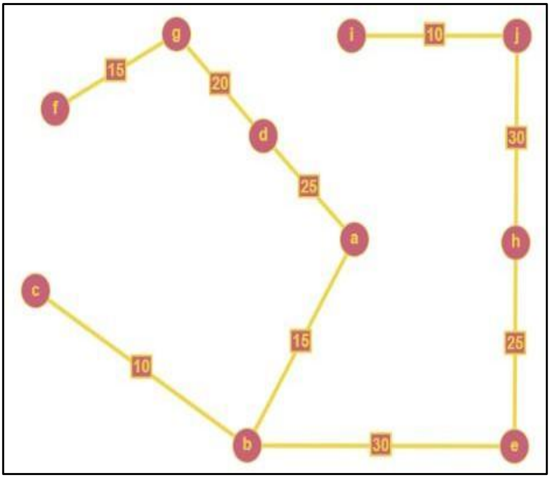

Figure 4.2: Minimal spanning tree for the university's problem

4.3 CONCLUSION

Using Prim's and Kruskal's algorithms, the minimum cable length needed to connect the university buildings was calculated, represented by a minimum spanning tree. This solution allows the university to determine the shortest total cable length required for connecting the necessary buildings within the compound.

4.4 REFERENCES

[1] Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to algorithms* (3rd ed.). MIT Press.

[2] West, D. B. (2001). *Introduction to graph theory* (2nd ed.). Prentice Hall.

[3] Tarjan, R. E. (1983). *Data structures and network algorithms*. SIAM.

[4] Jarník, V. (1930). *O jistém problému minimálním*. Práce moravské přírodovědecké společnosti.

[5] Prim, R. C. (1957). *Shortest connection networks and some generalizations*. Bell System Technical Journal, 36(6), 1389-1401.

[6] 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.

[7] Ahuja, R. K., Magnanti, T. L., & Orlin, J. B. (1993). *Network flows: Theory, algorithms, and applications*. Prentice Hall.

[8] GraphOnline. (n.d.). *GraphOnline*. Retrieved January 28, 2025, from https://graphonline.top/en/