

BAB II TINJAUAN PUSTAKA

A. Penelitian Terkait

Mengenai adanya penelitian – penelitian yang telah dilakukan sebelumnya. Hasil penelitian – penelitian tersebut menjadi referensi pada penelitian ini. Berikut adalah penelitian – penelitian sebelumnya yang terkait dengan penelitian ini.

Tabel 1. Penelitian Terkait

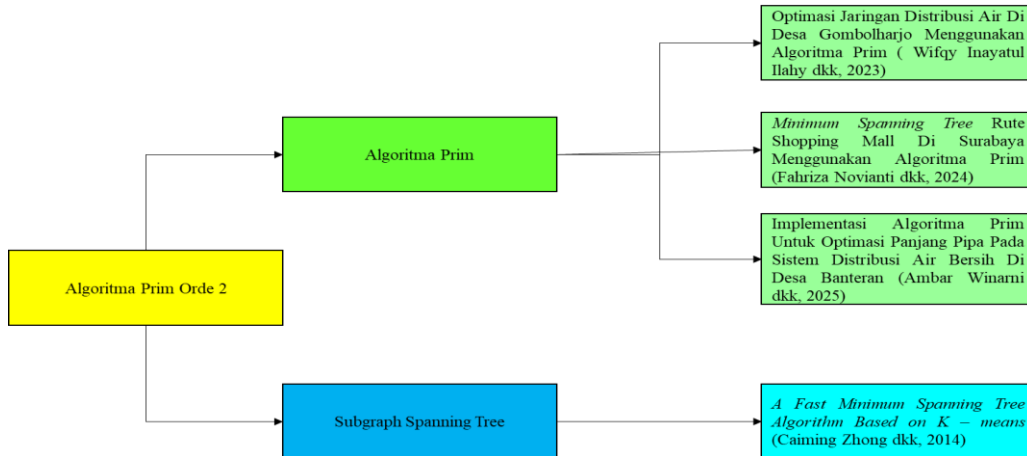
No	Peneliti	Keterangan
1	Rina Filia Sari, Rina Widyasari, Fithria Aidra Marpaung (2023) [9]	Judul : Optimasi Pemasangan Jalur Pipa Bersih Melalui <i>Minimum Spanning Tree</i> dengan Algoritma Prim Metode : Kuantitatif. Hasil : Penelitian ini menghasilkan penghematan biaya dalam pemasangan instalasi air bersih sepanjang 162 meter.
2	Nabila Tasya Amalia, Fahriza Novianti, Yasirah Rezqita Aisyah Yasmin (2024) [10]	Judul : <i>Minimum Spanning Tree</i> Rute Shopping Mall di Surabaya Menggunakan Algoritma Prim Metode : Kuantitatif. Hasil : Penelitian ini menghasilkan rute terpendek pusat perbelanjaan di Surabaya didapatkan rute terpendek dengan percabangan 2 rute.
3	Wifqy Inayatul Ilahy, Mizan Ahmad, Bryan Pudji Hartono (2023) [11]	Judul : Optimasi Jaringan Distribusi Air di Desa Gombolharjo Menggunakan Algoritma Prim Metode : Kuantitatif. Hasil : Penelitian ini menghasilkan jalur distribusi jaringan air bersih diperoleh total <i>vertex</i> sebanyak 60 dan total <i>edge</i> sebanyak 59 dengan panjang jaringan pipa sepanjang 5706 m.

4	Arya Bima Mohammad Heriansyah, Rinaldi Rizwar, Muhamad Rafliansyah, Efrans Christian, Viktor Handrianus Pranatawijaya (2024) [12]	Judul : Penerapan Algoritma Prim dalam Menentukan Rute Prioritas Pengiriman Paket <i>Express</i> Metode : Kuantitatif. Hasil : Penelitian ini menghasilkan rute yang efisien untuk menentukan rute pengiriman pake <i>express</i>.
5	Syamsuddin Mas'ud (2024) [13]	Judul : Penentuan Rute Pendistribusian Gas LPG Menggunakan Algoritma Prim dengan Optimalisasi Melalui Pergantian Sisi Metode : Studi Literatur. Hasil : Penelitian ini menganalisis hasil dari peneliti sebelumnya yang menghasilkan rute optimal menggunakan algoritma Greedy dengan total bobot 13,13 km, sedangkan peneliti menggunakan algoritma Prim menghasilkan rute dengan total bobot 10,91 km.
6	Shafira Zakiatul Faizah, Dwiani Listya Kartika, Ambar Winarni (2025) [14]	Judul : Implementasi Algoritma Prim untuk Optimasi Panjang Pipa pada Sistem Distribusi Air Bersih di Desa Banteran Metode : Kuantitatif. Hasil : Penelitian ini dapat mengatasi masalah pendistribusian air secara efisien, dengan panjang pipa yang semula mencapai 14.890 meter dapat dioptimalkan menjadi 11.020 meter.
7	Moufya Fattan Ghifara, Muhammad Ilham Chasanudin, Azza An Nafisah, Muhammad Gani Rohman (2024) [15]	Judul : Optimalisasi Rute Pendistribusian Buku dari Perpustakaan Daerah Salatiga ke SMA dan SMK Negeri dengan Algoritma Prim Metode : Kuantitatif. Hasil : Penelitian ini menunjukkan keberhasilan dalam mengoptimalkan proses pendistribusian

		buku dari Perpustakaan Daerah kota salatiga ke SMA dan SMK Negeri di Salatiga dengan menerapkan algoritma Prim menghasilkan total bobot 12,6 km.
8	Amalia Ahsanti, Asyanada Insyafilla, Nadhifa Nur Fatimah, Winda Cahya Dwi Wahyuni, Deddy Rahmadi (2025) [16]	Judul : Optimalisasi Jaringan Jalan Antar Kecamatan dengan <i>Minimum Spanning Tree</i> dan Algoritma Prim di Kabupaten Ngawi Metode : Kuantitatif dengan berbantuan alat yaitu <i>software Visual Studio Code</i> dengan bahasa pemrograman Phyton. Hasil : Penelitian ini menghasilkan rute terpendek dari 19 Kecamatan di Kabupaten Ngawi sepanjang 146 km.
9	Zulva Nur Laely, Miftachul Janah, Muhamad Gani Rohman (2025) [17]	Judul : Aplikasi Algoritma Prim untuk Optimasi pada Aliran Sumber Mata Air di Desa Banjarsari Metode : Kuantitatif. Hasil : Algoritma Prim secara signifikan membantu mengoptimalkan jalur distribusi air bersih di Desa Banjarsari dengan menghasilkan jalur terpendek dan biaya minimum.
10	Caiming Zhong, Mikko Malinen, Duoqian Miao, Pasi Franti (2014) [18]	Judul : <i>A Fast Minimum Spanning Tree Algorithm Based on K – means</i> Metode : Kuantitatif. Hasil : Algoritma FMST berbasis K – means menghasilkan <i>Minimum Spanning Tree</i> yang lebih cepat dengan kompleksitas $O(N^{1.5})$ dan tetap memiliki akurasi mendekati MST sebenarnya.

Perbedaan penelitian ini dengan penelitian terdahulu adalah pada penerapan kasus dan algoritma yang digunakan, pada penelitian sebelumnya penerapan kasus yang digunakan menggunakan 1 titik awal, sedangkan pada penelitian ini

menguunakan 2 titik awal. Oleh karena itu, peneliti mengembangkan algoritma Prim menjadi algoritma Prim orde 2 untuk mendapatkan solusi dari kasus pendistribusian dengan 2 titik awal. Khusus untuk penelitian Caiming Zhong dkk perbedaanya terletak pada hasil *Minimum Spanning Tree*, pada penelitian tersebut menghasilkan 1 *Minimum Spanning Tree* sedangkan penelitian ini menghasilkan 2 *Minimum Spanning Tree*.



Gambar 1. *Mind Map*

B. Landasan Teori

1. Teori Dasar Graf

Menurut Chartrand [19] terkait dengan definisi dari beberapa dasar graf, seperti definisi graf, $u - v$ walk, $u - v$ trail, $u - v$ path, cycle, adjacent, incident, dan graf terhubung.

Definisi 2.1 Graf G adalah himpunan tak kosong berhingga $V(G)$ disertai dengan relasi yang *irreflexive* dan *symmetric* R pada V . Dalam graf G , $V(G)$ adalah himpunan *vertex* dan banyaknya *vertex* dalam G dinotasikan orde G atau $|V(G)|$, sedangkan setiap elemen $E(G)$ adalah himpunan *edge* dan banyaknya *edge* dalam G disebut *size* G . Selain itu, graf G didefinisikan pula sebagai himpunan tak kosong V dari objek – objek yang disebut *vertex* bersama dengan himpunan mungkin kosong E dari dua elemen pada V yang disebut *edge*. Graf G dengan himpunan *vertex* V dan himpunan *edge* E biasanya dinotasikan sebagai $G = (V, E)$.

Definisi 2.2 *Vertex* u dan *vertex* v dikatakan *adjacent* di graf G jika $uv \in E(G)$. Apabila suatu *vertex* u *incident* dengan *edge* e dan *vertex* v *incident* dengan *edge* e , jika $e = uv \in E(G)$, maka u dan v adalah *incident* dengan e .

Definisi 2.3 suatu $u - v$ walk pada suatu graf G adalah barisan bergantian dari vertex dan edge G , dimulai dari vertex u dan diakhiri dengan vertex v , dengan setiap edge menghubungkan vertex terhadap barisan tersebut. Suatu $u - v$ trail pada suatu graf adalah lintasan $u - v$ walk yang tidak mengulang edge manapun, sedangkan suatu $u - v$ path adalah lintasan $u - v$ walk yang tidak mengulang vertex manapun.

Definisi 2.4 Graf dikatakan terhubung jika untuk sembarang dua vertex dalam graf G terdapat suatu path yang menghubungkan dua vertex tersebut. Apabila tidak terjadi hal tersebut, maka graf tersebut tidak terhubung.

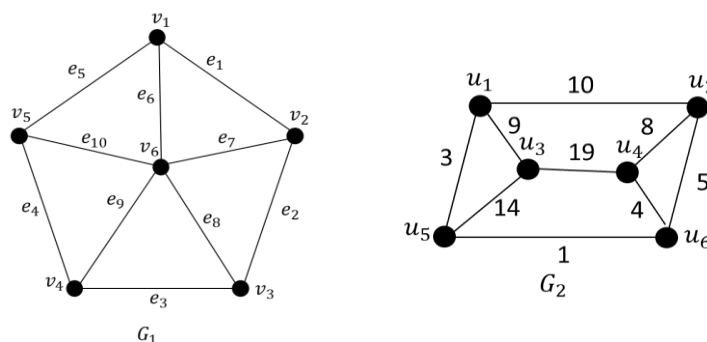
Definisi 2.5 Circuit adalah $u - v$ trail dengan $u = v$ yang mengandung setidaknya tiga edge, dengan kata lain sebuah circuit harus berakhir pada vertex yang sama dengan vertex awal. Sedangkan cycle, yaitu sebuah circuit yang tidak mengulangi setiap vertex (kecuali vertex pertama dan vertex yang terakhir).

Definisi 2.6 Suatu graf G dikatakan berbobot apabila setiap edge-nya diberikan sebuah bobot (nilai).

Definisi 2.7 Jarak $d(u,v)$ dari u ke v adalah panjang $u - v$ path terpendek.

Definisi 2.8 Kardinalitas $|G|$ adalah banyaknya vertex pada graf G .

Berikut ini diberikan contoh graf dari definisi di atas :



Gambar 2. Graf G_1 dan G_2

Gambar 2 graf G_1 merupakan contoh graf terhubung dengan $|V(G)| = 6$ dan $|E(G)| = 10$. Dari graf G_1 didapatkan bahwasannya v_1 adjacent dengan v_2 dan v_5 , e_1 incident dengan vertex v_1 dan v_2 . Contoh $u - v$ walk adalah $v_1, e_1, v_2, e_2, v_3, e_3, v_4$.

Contoh dari $u - v$ trail $v_1, e_1, v_2, e_7, v_6, e_6, v_1$. Contoh dari $u - v$ path adalah $v_1, e_1, v_2, e_7, v_6, e_6, v_1$. Contoh dari circuit $v_1, e_1, v_2, e_2, v_3, e_8, v_6$. Sedangkan pada graf G_2 merupakan contoh dari graf berbobot atau setiap edgenya mempunyai nilai. Contoh dari jarak, misalkan dari $u_1 \rightarrow u_2$ adalah 10.

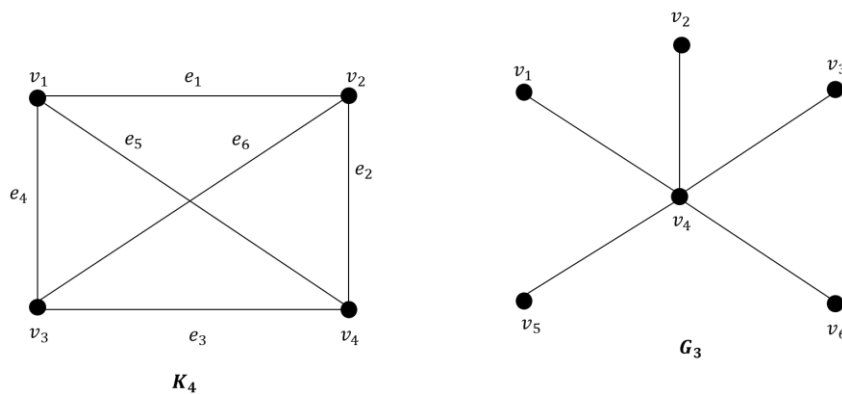
2. Jenis – Jenis Graf

Berikut dijelaskan definisi dari graf lengkap dan graf *tree* menurut G. Chartand, L. Lesniak, dan P. Zhang [3] :

Definisi 2.9 Graf lengkap adalah graf sederhana yang setiap *vertex* mempunyai *edge* ke semua *vertex* lainnya. Notasi dari graf lengkap dengan n buah *vertex* adalah K_n .

Definisi 2.10 Graf *tree* adalah graf terhubung sederhana yang tidak mengandung *circuit*.

Berikut ini merupakan contoh graf lengkap dan graf *tree* :



Gambar 3. Graf Lengkap K_4 dan Graf Pohon G_3

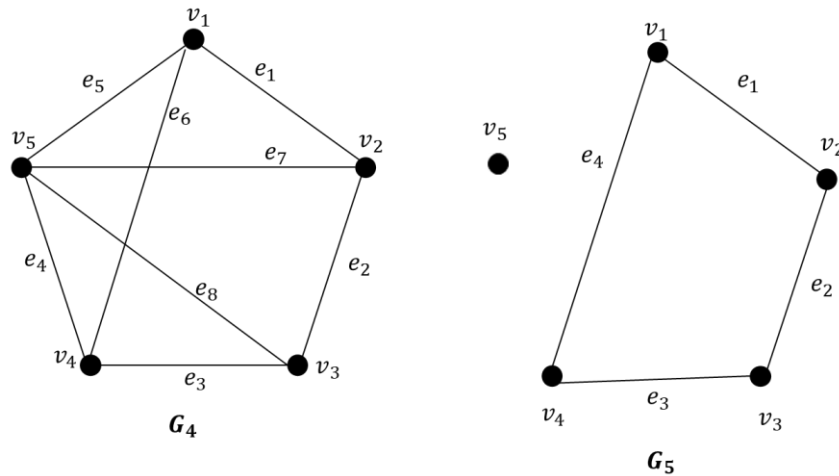
Gambar 3 graf K_4 merupakan contoh dari graf lengkap dengan 4 Vertex, sedangkan graf G_3 adalah contoh dari graf *tree*.

3. Pohon Rentang Minimum (*Minimum Spanning Tree*)

Berikut dijelaskan definisi dari subgraf, graf *tree*, *spanning tree* dan *minimum spanning tree* menurut G. Chartand, L. Lesniak, dan P. Zhang [3] :

Definisi 2.11 Sebuah graf H disebut subgraf dari G jika $V(H) \subseteq V(G)$ dan $E(H) \subseteq E(G)$, yang mana dapat ditulis $H \subseteq G$. Jika $V(H) = V(G)$ maka H adalah spanning subgraf dari G .

Berdasarkan definisi 2.11, berikut diberikan contoh graf yang memenuhi definisi tersebut :

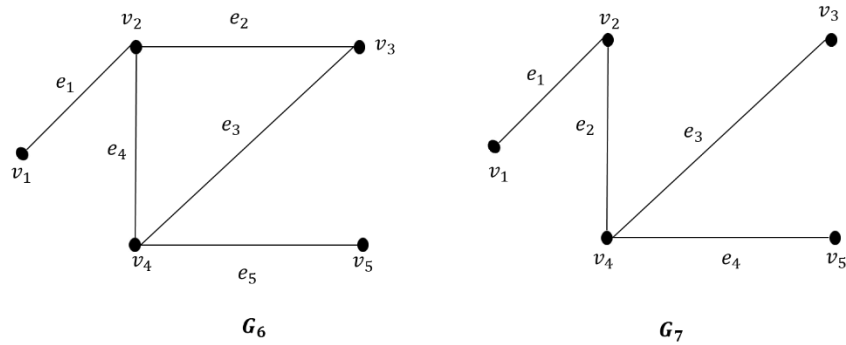


Gambar 4. Graf G_4 dan G_5

Gambar 4 graf G_5 merupakan *Spanning Subgraph* dari G_4 karena $V(H) \subseteq V(G)$.

Definisi 2.12 Sebuah *Spanning Tree* dari graf G adalah *spanning subgraph* dari G yang berupa graf *tree*.

Berdasarkan definisi dari *spanning tree*, berikut ini diberikan contoh graf yang memenuhi definisi tersebut :

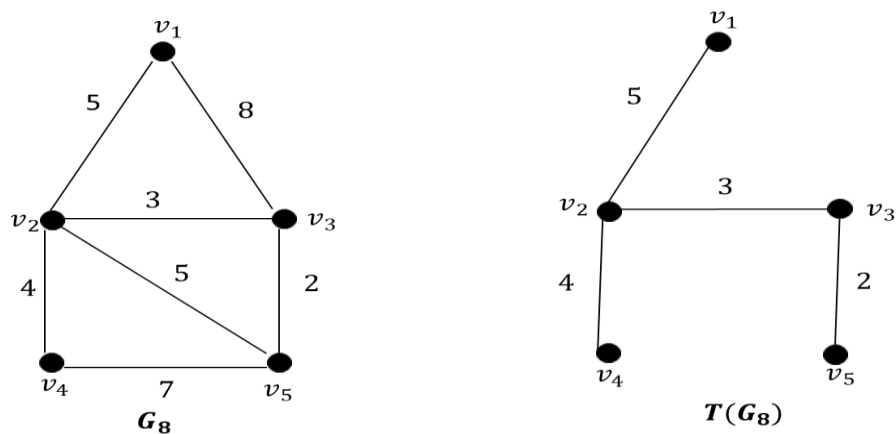


Gambar 5. Graf G_6 dan G_7

Gambar 5 graf G_7 merupakan *spanning tree* dari graf G_6 .

Definisi 2.13 Sebuah *spanning tree* dari graf G yang bobotnya minimum diantara semua *spanning tree* dari graf G disebut *Minimum Spanning Tree*.

Berdasarkan definisi 2.13, berikut diberikan contoh *Minimum Spanning Tree* :



Gambar 6. Graf G_8 dan Graf Pohon $T(G_8)$

Gambar 6 Graf $T(G_8)$ merupakan contoh graf *Minimum Spanning Tree* dari graf G_8 .

4. Algoritma Prim

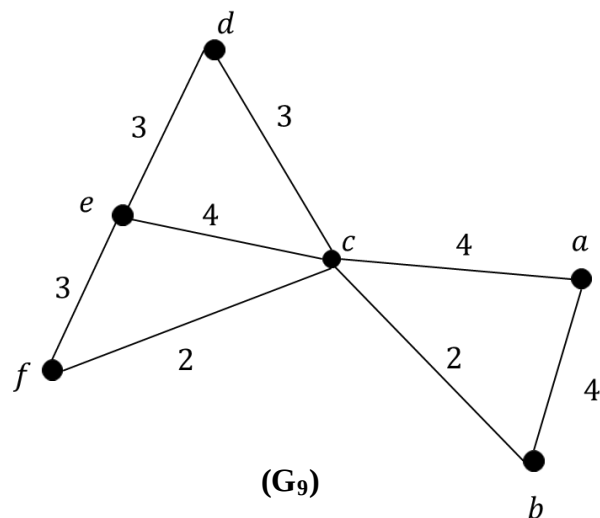
Menurut Chartand [3], graf terhubung berbobot G yang berorde n *Minimum Spanning Tree* T dikonstruksi sebagai berikut. Diambil sembarang *vertex* u dalam graf G , pilih *edge* dengan bobot minimum yang berhubungan dengan *vertex* u

(e_1 dari graf T), untuk *edge* selanjutnya ($e_2, e_3, \dots, e_{n-1}$) dipilih *edge* dengan bobot – bobot minimum tepat satu dari *vertex* yang sudah dipilih.

Proses algoritma prim adalah sebagai berikut :

1. Pilih *vertex* awal $V(G)=V(T(G))$.
2. (i.) Pilih *edge* yang *incident* dengan $V(T(G))$ yang memiliki bobot terkecil, misalkan e_1 yang tidak menimbulkan *cycle*, jika menimbulkan *cycle* maka pilih *edge* yang *incident* dengan bobot terkecil ke- 2.
(ii.) Pilih *vertex* yang *adjacent* dengan $V(T(G))$ dan *incident* dengan e_1 .
3. Ulangi langkah b(i) dan b(ii) hingga diperoleh $E(T(G)) = n-1$.

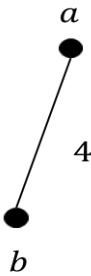
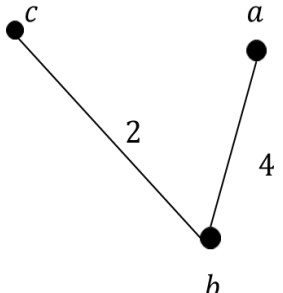
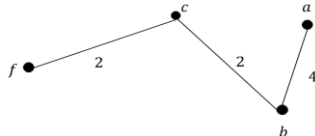
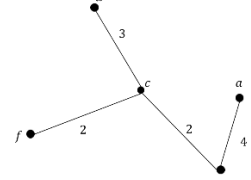
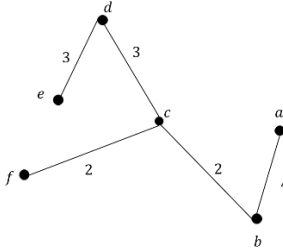
Berikut merupakan contoh graf dengan algoritma Prim :



Gambar 7. Graf G_9

Berikut ini penyelesaian *Minimum Spanning Tree* menggunakan algoritma Prim disajikan dalam Tabel 2.

Tabel 2. Iterasi Algoritma Prim

Iterasi	Vertex	Bobot	Graf
0	a	-	
1	T{a}	0	
	a - b	4	
	a - c	4	
Dipilih	a - b	4	
2	T : {a,b}	0+4=4	
	a - c	4	
	b - c	2	
Dipilih	b - c	2	
3	T : {a, b, c}	4+2=6	
	c - d	3	
	c - e	4	
	c - f	2	
Dipilih	c - f	2	
4	T : {a, b, c, f}	4+2+2=8	
	c - d	3	
	c - e	4	
	f - e	3	
Dipilih	c - d	3	
5	T : {a, b, c, f, d}	4+2+2+3=11	
	c - e	4	
	f - e	3	
	d - e	3	
Dipilih	d - e	3	
6	T : {a, b, c, f, d, e}	4+2+2+3+3=14	