

**RENCANA INDUK PENGOPTIMALAN JARINGAN PIPA  
DISTRIBUSI PERUSAHAAN AIR MINUM (PDAM)  
DI AURDURI DENGAN METODE  
ALGORITMA KRUSKAL**

SKRIPSI



**BARON AQYS AL-RASYID ARITONANG  
F1C217010**

**PROGRAM STUDI MATEMATIKA  
JURUSAN MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS JAMBI  
2021**

## **SURAT PERNYATAAN**

Dengan ini saya menyatakan bahwa skripsi ini benar-benar karya sendiri. Sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan atau kutipan dengan mengikuti tata penulisan karya ilmiah yang telah lazim.

Tanda tangan yang tertera dalam halaman pengesahan adalah asli. Jika tidak asli, saya siap menerima sanksi sesuai dengan peraturan yang berlaku.

Jambi, 21 Juni 2021

Yang menyatakan

BARON A. A. ARITONANG

F1C217010

## RINGKASAN

Jaringan pipa distribusi perusahaan daerah air minum (PDAM) di Aurduri selama ini masih berpatokan pada konsep yang sangat sederhana, seperti tidak adanya perhitungan khusus untuk mengoptimalkan pembangunan jaringan pipa. Dibutuhkan solusi pengoptimalan dalam merencanakan konsep pembangunan jaringan pipa distribusi air minum di Aurduri untuk mengefisiensi dana yang dikeluarkan. Serta menyesuaikan dengan program jangka menengah RI-SPAM (Rencana Induk Sistem Penyediaan Air Minum) adalah pembangunan Intake dermaga di IPA (Instalasi Pengolahan Air) Aurduri berkapasitas (1000-1500) liter/detik untuk mengantisipasi peningkatan kapasitas IPA Aurduri dari 300 liter/detik menjadi 1800 liter/detik dalam memenuhi kebutuhan air minum rata-rata pada tahun 2034 sebesar 3.702 liter/detik. Pengoptimalan jarak jaringan pipa dapat dilakukan dengan pencarian pohon merentang minimum. Pohon merentang minimum yaitu menentukan sisi-sisi yang menghubungkan titik-titik yang ada pada jaringan hingga yang diperoleh merupakan panjang sisi total yang minimum. Pada penelitian ini dilakukan pencarian pohon merentang minimum dengan memodelkan jaringan pipa PDAM di Aurduri ke dalam bentuk graf. Dalam pencarian pohon merentang minimum terdapat beberapa algoritma yang dapat digunakan seperti Algoritma Kruskal, Warshall dan Dijkstra. Karena pada jaringan pipa termasuk aplikasi dari graf berbobot dan tidak berarah maka algoritma yang tepat digunakan yaitu Algoritma Kruskal. Konsep awal yang digunakan Algoritma Kruskal dalam menentukan pohon Merentang minimum atau *Minimum Spanning Tree* (MST) adalah dengan cara memilih sisi dari graf secara berurutan berdasarkan besarnya bobot graf tersebut, dari bobot kecil ke bobot terbesar. Berdasarkan hasil dan pembahasan pada penelitian ini bahwa Algoritma Kruskal dapat digunakan dalam pencarian Pohon Merentang Minimum pada jaringan pipa PDAM di Perumahan Aurduri. Dengan menggunakan data yang ada, diperoleh banyaknya sisi pohon merentang minimum adalah 33 sisi dengan jumlah iterasi sebanyak 34 iterasi. Sisi yang awalnya berjumlah 34 dihapus satu sisi, yaitu sisi ( $V_{17}$ - $V_{27}$ ) sepanjang 193 meter. Maka diperoleh panjang pipa primer menggunakan Algoritma Kruskal adalah sepanjang 3728 meter. Sedangkan panjang jaringan pipa primer PDAM yang dihitung sebelum menggunakan Algoritma Kruskal pada Perumahan Aurduri adalah sepanjang 3921 meter.

## **SUMMARY**

*So far, the distribution pipeline network of the regional drinking water company (PDAM) in Aurduri is still based on a very simple concept, such as the absence of special calculations to optimize the construction of the pipeline network. Optimization solutions are needed in planning the concept of building a drinking water distribution pipeline network in Aurduri to streamline the funds spent. As well as adjusting to the RI-SPAM (Water Supply System Master Plan) medium-term program is the construction of a wharf intake at the Aurduri IPA (Water Treatment Plant) with a capacity (1000-1500) liters / second to anticipate an increase in Aurduri IPA capacity from 300 liters / second to 1800 liters / second in meeting water needs. drink an average in 2034 of 3,702 liters / second. Optimizing the distance of the pipeline network can be done by searching for the minimum spanning tree. The minimum spanning tree is determining the sides that connect the points on the network so that the minimum total side length is obtained. In this study, a minimum spanning tree search was carried out by modeling the PDAM pipeline network in Aurduri into a graph. In the search for minimum spanning trees, there are several algorithms that can be used, such as the Kruskal, Warshall and Dijkstra algorithms. Because the pipeline network includes applications of weighted and undirected graphs, the appropriate algorithm is used, namely the Kruskal algorithm. The initial concept used by Kruskal's Algorithm in determining the Minimum Spanning Tree (MST) is by selecting the sides of the graph sequentially based on the weight of the graph, from small to largest. Based on the results and discussion in this study, the Kruskal Algorithm can be used in the search for Minimum Spanning Trees in PDAM pipelines in Aurduri Housing. By using the existing data, the minimum number of sides of a tree that spans is 33 with a total of 34 iterations. The sides that originally numbered 34 were removed from one side, namely the 193 meters (V17-V27) side. Then the primary pipe length is obtained using Kruskal's Algorithm is 3728 meters long. Meanwhile, the length of the primary PDAM pipeline network calculated before using the Kruskal Algorithm in Aurduri Housing is 3921 meters long.*

**RENCANA INDUK PENGOPTIMALAN JARINGAN PIPA  
DISTRIBUSI PERUSAHAAN AIR MINUM (PDAM)  
DI AURDURI DENGAN METODE  
ALGORITMA KRUSKAL**

**S K R I P S I**

Diajukan sebagai salah satu syarat untuk memperoleh  
Gelar Sarjana pada Program Studi Matematika



**BARON AQYS AL-RASYID ARITONANG  
F1C217010**

**PROGRAM STUDI MATEMATIKA  
JURUSAN MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS JAMBI**

**2021**

## LEMBAR PENGESAHAN

Skripsi dengan Judul **RENCANA INDUK PENGOPTIMALAN JARINGAN PIPA DISTRIBUSI PERUSAHAAN AIR MINUM (PDAM) DI AURDURI DENGAN METODE ALGORITMA KRUSKAL** yang disusun oleh **BARON AQYS AL-RASYID ARITONANG, NIM: F1C217010** telah dipertahankan di depan tim penguji pada tanggal MEI 2021 dan dinyatakan lulus.

Susunan Tim Penguji:

Ketua : Drs. Sufri, M.Si.  
Anggota : 1. Gusmi Kholijah, S.Si., M.Si.  
2. Niken Rarasati, S.Si., M.Si.  
3. Drs. Wardi Syafmen, M.Si.  
4. Gusmanely Z, S.Pd., M.Si.

Disetujui:

Pembimbing Utama

Pembimbing Pendamping

Drs. Wardi Syafmen, M.Si.  
NIP. 196202071992031002

Gusmanely Z, S.Pd., M.Si.  
NIK. 201509072033

Diketahui:

Dekan Fakultas Sains dan  
Teknologi

Ketua Jurusan Matematika dan  
Ilmu Pengetahuan Alam

Prof. Damris M, M.Sc., Ph.D  
NIP. 196605191991121001

Dr. Madyawati Latief, SP. M.Si.  
NIP. 197206241999032001

## RIWAYAT HIDUP



Baron Aqys Al-rasyid Aritonang lahir di Tanjung Pinang Kepulauan Riau, pada tanggal 29 Agustus 1999. Penulis merupakan anak pertama dari tiga bersaudara dari pasangan Bapak Herry J. Aritonang dan Ibu Jumharni. Jalur pendidikan formal yang pernah ditempuh penulis adalah:

1. SD Negeri 14 Binaan Bukit Bestari
2. SD Negeri 76/IX Mendalo Darat
3. SMP Negeri 7 Muaro Jambi
4. SMA Negeri 11 Muaro Jambi
5. Pada tahun 2017, penulis diterima di Perguruan Tinggi Negeri Universitas Jambi, Program S1 dan tercatat sebagai mahasiswa Program Studi Matematika Fakultas Sains dan Teknologi melalui jalur SBMPTN.

Selama menempuh pendidikan di jenjang S1 penulis cukup aktif dalam bidang akademik maupun organisasi kampus. Salah satunya dengan memenangkan Juara 2 Onmipa Matematika Tingkat Fakultas tahun 2019, Juara 3 Onmipa Matematika Tingkat Universitas Jambi tahun 2019, Juara 1 Onmipa Matematika Tingkat Fakultas tahun 2020, Juara 4 Onmipa Matematika Tingkat Universitas tahun 2020. Penulis juga pernah memenangkan Juara 2 Lomba Debat Bahasa Indonesia tingkat Fakultas tahun 2019 dan Juara Terbaik 2 kepenulisan fiksi se-Indonesia yang diadakan oleh penerbit One Peach Media. Bukan hanya itu, penulis berhasil meraih juara 1 untuk lomba menulis senandika se-Indonesia yang diadakan oleh penerbit Maple Media. Penulis baru saja melahirkan buku pertamanya Desember 2020 yang berjudul Sebelum Waktu Memakan Habis Kita. Penulis aktif di organisasi kampus salah satunya dengan menjadi Ketua Himatika 2019-2020. Penulis mengikuti kegiatan Magang di Kantor Cabang Sutomo PT. Bank Pembangunan Daerah Jambi . Selain itu, penulis juga aktif dalam kegiatan seminar-seminar baik tingkat jurusan regional maupun kampus.

## PRAKATA

Puji dan syukur kepada Allah SWT. karena berkat rahmat dan karunia-Nya penulis dapat menyelesaikan skripsi yang berjudul “**RENCANA INDUK PENGOPTIMALAN JARINGAN PIPA DISTRIBUSI PERUSAHAAN AIR MINUM (PDAM) DI AURDURI DENGAN METODE ALGORITMA KRUSKAL**”. Shalawat beriring salam tidak lupa penulis haturkan kepada Nabi Muhammad SAW.

Skripsi ini disusun dan diajukan sebagai salah satu syarat untuk memperoleh Gelar Sarjana Program Studi Matematika Fakultas Sains dan Teknologi Universitas Jambi. Dalam penyusunan skripsi ini, tidak sedikit hambatan dan halangan yang penulis hadapi. Akan tetapi, dengan adanya semangat dan bantuan dari berbagai pihak, Penulis mampu menyelesaikan skripsi ini. Untuk itu, Penulis ingin mengucapkan terima kasih kepada:

1. Allah SWT. Yang telah memberikan karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini.
2. Terima kasih kepada Kedua orang tuaku tercinta yaitu bapak Herry J. Aritonang dan ibu Jumharni serta kedua adikku yaitu Franco Al-rasyid Aritonang dan Aisyah S. O. A. B. Aritonang yang selalu memberikan do'a dan dukungan serta motivasi kepada penulis. Terima kasih kepada seluruh keluargaku.
3. Prof. Drs. Damris M, M.Sc., Ph.D selaku Dekan Fakultas Sains dan Teknologi Universitas Jambi.
4. Gusmi Kholijah, S.Si., M.Si selaku Ketua Program Studi Matematika Fakultas Sains dan Teknologi Universitas Jambi.
5. Drs. Wardi Syafmen, M.Si. dan Gusmanely Z, S.Pd., M.Si. selaku pembimbing skripsi.
6. Drs. Sufri, M.Si., Gusmi Kholijah, S.Si., M.Si. dan Niken Rarasati, S.Si., M.Si. selaku tim penguji.
7. Gusmanely Z, S.Pd., M.Si. selaku pembimbing akademik penulis.
8. Untuk semua para dosenku tercinta, terima kasih.
9. Terima Kasih Kepada Infinity Squad, Himatika, Fst dan Unja.
10. Teman-teman seperjuangan di Program Studi Matematika Angkatan 2017.
11. Kakak tingkat 2015, kakak tingkat 2016, adik tingkat 2018 dan adik tingkat 2019 yang telah membantu dalam penyusunan laporan magang.
12. Serta semua pihak yang telah membantu dan tidak bisa disebutkan satu persatu.

Penulis menyadari skripsi ini masih banyak terdapat kesalahan dan kekurangan. Penulis berharap, semoga skripsi ini dapat bermanfaat bagi kita semua di masa yang akan datang. Penulis juga mengharapkan saran dan kritik yang dapat membangun penyempurnaan skripsi ini. Sekian dan terima kasih.

Jambi, Juni 2021

BARON A. A. ARITONANG  
F1C217010

## DAFTAR ISI

LEMBAR PENGESAHAN.....	i
RIWAYAT HIDUP.....	i
PRAKATA.....	ii
DAFTAR ISI .....	iv
DAFTAR TABEL .....	vi
DAFTAR GAMBAR.....	vii
I. PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Identifikasi dan Perumusan Masalah.....	4
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian .....	4
1.5 Manfaat Penelitian.....	4
II. TINJAUAN PUSTAKA.....	5
2.1 Teori Graf.....	5
2.2 Graf dan Komponen Graf .....	5
2.3 Representasi Graf .....	9
2.4 Pohon ( <i>Tree</i> ) .....	11
2.5 Pohon Merentang.....	12
2.6 Pohon Merentang Minimum ( <i>Minimum Spanning Tree / MST</i> ) .	12
2.7 Algoritma Kruskal.....	13
2.8 Sistem Jaringan Pipa Air Bersih .....	16
2.9 Jaringan Pipa PDAM Aurduri .....	18
III. METODOLOGI PENELITIAN .....	19
3.1 Jenis dan Sumber Data .....	19
3.2 Metode Penelitian .....	19
3.3 Alur Penelitian.....	20
IV. HASIL DAN PEMBAHASAN.....	21
4.1 Pengumpulan Data.....	21
4.2 Perhitungan Data .....	23
V. KESIMPULAN DAN SARAN.....	47
5.1 Kesimpulan .....	47

5.2	Saran.....	47
	DAFTAR PUSTAKA.....	48

**DAFTAR TABEL**

Tabel	Halaman
1. Tabel 4.1 Data Panjang jaringan pipa primer PDAM Perumahan Aurduri.....	16
2. Tabel 4.2 Urutan sisi graf dari bobot terkecil hingga terbesar.....	18

## DAFTAR GAMBAR

Gambar	Halaman
2.1 Contoh Graf .....	5
2.2 Graf Tak Berarah .....	6
2.3 Graf Berarah.....	6
2.4 Graf Sederhana.....	7
2.5 Graf Ganda .....	7
2.6 Graf Semu .....	8
2.7 Graf Siklus atau Sirkuit .....	8
2.8 Graf Berbobot .....	9
2.9 Graf Matriks Bertetanggaan.....	10
2.10 Graf Matriks Bersisian.....	10
2.11 Graf Pohon.....	12
2.12 Graf Berbobot yang memiliki sirkuit .....	14
2.13-2.17 Contoh langkah pencarian Algoritma Kruskal.....	14-15
2.18 Graf lengkap yang memiliki banyak sirkuit.....	16
3.1 Diagram Alir.....	20
4.1 Denah Lokasi Perumahan.....	21
4.2 Jaringan pipa primer dalam bentuk graf.....	23
4.3-4.36 Iterasi 1-34.....	25-45
4.37 Jaringan pipa yang telah terbentuk Pohon Merentang Minimum.....	45
4.38 Jaringan pipa sebelum dan setelah perhitungan.....	46

# I. PENDAHULUAN

## 1.1 Latar Belakang

Berdasarkan situs resmi yang dikelola oleh Kementerian Pekerjaan Umum dan Perumahan Rakyat, dengan memperhitungkan arahan perkembangan kota yang diwujudkan dalam pemanfaatan ruang berdasarkan struktur dan pola peruntukan lahannya, maka penyusunan Rencana Induk Sistem Penyediaan Air Minum (RI-SPAM) harus searah dalam program dan kegiatannya. Berdasarkan analisis arah perkembangan kota dan areal lahan terbangun, struktur dan pola ruang kota, kondisi debit dan jaringan pipa distribusi, jumlah sebaran pelanggan air minum PDAM, jumlah sebaran dan tingkat kepadatan penduduk, kondisi topografi kota dan kebijakan dan peraturan terkait. Berdasarkan Skenario Rencana Induk Sistem Penyediaan Air Minum (RI-SPAM) Kota Jambi yang dikelola oleh PDAM Tirta Mayang untuk periode mendesak, maka ada tiga program utama atau tiga tahap. Salah satu tahap dalam Skenario Rencana Induk Sistem Penyediaan Air Minum (RI-SPAM) Kota Jambi adalah program jangka menengah.

Program jangka menengah RI-SPAM adalah pembangunan Intake dermaga di IPA (Instalasi Pengolahan Air) Aurduri berkapasitas (1000-1500) liter/detik untuk mengantisipasi peningkatan kapasitas IPA Aurduri dari 300 liter/detik menjadi 1800 liter/detik dalam memenuhi kebutuhan air minum rata-rata pada tahun 2034 sebesar 3.702 liter/detik. Pengembangan IPA Aurduri ini dipilih selain mengantisipasi peningkatan kebutuhan air minum di wilayah barat yang berbatasan dengan Kabupaten Muaro Jambi yang perkembangan kotanya sangat cepat, juga luas lahannya masih sangat mencukupi dibandingkan dengan lahan IPA Broni dan IPA Benteng yang sudah tidak memungkinkan dikembangkan produksinya dalam mengantisipasi kebutuhan jangka panjang tahun 2034. Pembangunan IPA *compact* lengkap dengan pompa air baku dan pompa distribusi sampai tahun 2024 minimal 500 liter/detik untuk memenuhi kebutuhan air minum rata-rata sebesar  $(Q) = 2.324$  liter/detik. Pembangunan jaringan pipa distribusi utama (JDU) Lingkar Barat, Lingkar Timur dan Lingkar Selatan minimal berdiameter 300 mm (12 inch) untuk mengantisipasi pengembangan di Wilayah Barat, Tengah dan Timur serta membangun jaringan perpipaan tertutup (*loop*), pada tahun 2021-2022.

Jaringan pipa distribusi perusahaan daerah air minum (PDAM) di Aurduri selama ini masih berpatokan pada konsep yang sangat sederhana,

seperti tidak adanya perhitungan khusus untuk mengoptimalkan pembangunan jaringan pipa. Dibutuhkan solusi pengoptimalan dalam merencanakan konsep pembangunan jaringan pipa distribusi air minum di Aurduri untuk mengefisiensi dana yang dikeluarkan dan mengantisipasi peningkatan kebutuhan air minum. Pengoptimalan jarak jaringan pipa dapat dilakukan dengan pencarian pohon merentang minimum. Pohon merentang minimum yaitu menentukan sisi-sisi yang menghubungkan titik-titik yang ada pada jaringan hingga yang diperoleh merupakan panjang sisi total yang minimum.

Pada penelitian ini dilakukan pencarian pohon merentang minimum dengan memodelkan jaringan pipa PDAM Aurduri ke dalam bentuk graf. Dalam pencarian pohon merentang minimum terdapat beberapa algoritma yang dapat digunakan seperti Algoritma Kruskal, Algoritma Prim dan Algoritma Sollin. Masing-masing algoritma tersebut memiliki aturan yang berbeda dalam menentukan Pohon Merentang Minimum. Jaringan pipa termasuk aplikasi dari graf berbobot dan tidak berarah sehingga ketiga algoritma tersebut bisa digunakan. Namun, penulis akan memfokuskan penelitian terhadap Algoritma Kruskal dikarenakan Algoritma Kruskal memiliki kelebihan salah satunya sangat cocok untuk graf jaringan pipa PDAM Aurduri yang memiliki banyak simpul tetapi tidak terlalu banyak sisi. Menurut Nugraha (2011), konsep awal yang digunakan Algoritma Kruskal dalam menentukan pohon merentang minimum atau *Minimum Spanning Tree* (MST) adalah dengan cara memilih sisi dari graf secara berurutan berdasarkan besarnya bobot graf tersebut, dari bobot kecil ke bobot terbesar.

Menurut Mohamad et al (2019), dalam graf untuk menemukan pohon merentang minimum untuk graf yang terhubung dan berbobot. Algoritma Kruskal selalu memproses suatu tepi yang memiliki bobot terkecil. Algoritma ini dijalankan dengan mempertimbangkan tepi terbesar saat mencari tepi node dalam graf yang telah ditaruh dalam pohon merentang. Jika batas tepi dianggap akan berintegrasi (dengan salah satu titik di pohon merentang) atau integrasi titik dalam pohon merentang (satu titiknya tidak berada dalam pohon merentang), maka batas tepi dan titik akhir termasuk dalam pohon merentang. Mempertimbangkan salah satu batas tepi, algoritma akan melanjutkan dengan mempertimbangkan bobot batas tepi berikutnya yang lebih besar.

Adapun penelitian yang relevan pada kasus *Minimum Spanning Tree* pernah dilakukan oleh Abrori dan Ubaidillah (2014), mengenai pengujian optimisasi jaringan kabel fiber optic di Universitas Islam Indonesia menggunakan *Minimum Spanning Tree*. Pada penelitian tersebut penulis

melakukan pencarian MST pada jaringan kabel dengan menggunakan berbagai algoritma. Hasil yang didapatkan dari penelitian tersebut ialah pada graf kampus UII menghasilkan pohon merentang minimum yang sama.

Penelitian lain yang relevan dengan Algoritma Kruskal adalah penelitian yang dilakukan Azizatul Muallimah dan Aris Fanani (2020), mengenai penggunaan Algoritma Kruskal dalam jaringan pipa pendistribusian Air Perusahaan Daerah Air Minum (PDAM) Tirta Dharma Lamongan. Pada penelitian tersebut dilakukan pencarian pohon merentang minimum jaringan pipa PDAM Tirta Dharma Lamongan dengan menggunakan Algoritma Kruskal. Hasil yang diperoleh pada Penelitian tersebut adalah jaringan pipa yang mempunyai jarak terpendek. Selisih jarak jaringan pipa primer yang terpasang dengan pohon merentang minimum jaringan pipa primer adalah sebesar 14.243,6 meter.

Dengan adanya penelitian tersebut, penulis tertarik untuk menggunakan Algoritma Kruskal untuk mencari pohon merentang minimum berdasarkan jumlah bobot yang dihasilkan. Hal ini karena Algoritma Kruskal merupakan salah satu Algoritma terbaik dalam kasus pencarian pohon merentang minimum. Serta Algoritma Kruskal sangat tepat untuk dipakai saat graf mempunyai jumlah sisi sedikit, tetapi memiliki banyak simpul. Sebab orientasi Algoritma Kruskal berdasarkan pada urutan bobot sisi, bukan berdasarkan simpul. Dalam hal pendistribusian air, jaringan pipa yang optimal sangat diperlukan. Karena setiap pelanggan yang membutuhkan air dapat terlayani dengan baik, tetapi dengan biaya pembangunan dan perawatan pipa saluran air minimal. Pada penelitian ini, penulis menggunakan data panjang jaringan pipa PDAM Tirta Mayang di Aurduri. Dengan menggunakan data jaringan pipa tersebut, penulis ingin melakukan pencarian MST pada jaringan pipa PDAM Tirta Mayang di Aurduri. Pencarian MST tersebut diterapkan untuk mengoptimalkan penggunaan jaringan pipa pada PDAM Tirta Mayang di Aurduri berdasarkan penggunaan banyaknya bobot panjang pipa yang digunakan. Hal ini dilakukan agar dapat meminimalkan biaya yang akan dikeluarkan.

Dengan menggunakan data jaringan pipa pada PDAM Tirta Mayang di Aurduri untuk rencana induk sistem penyediaan air minum sebagai antisipasi peningkatan jumlah kebutuhan air minum di Aurduri. Penulis memutuskan untuk mengajukan penelitian ini dengan judul **“Rencana Induk Pengoptimalan Jaringan Pipa Distribusi Perusahaan Daerah Air Minum (PDAM) di Aurduri dengan Metode Algoritma Kruskal”**.

## **1.2 Identifikasi dan Perumusan Masalah**

Adapun identifikasi dan perumusan masalah pada penelitian ini adalah bagaimana pencarian pohon merentang minimum dengan menggunakan Algoritma Kruskal untuk mengoptimalkan jaringan pipa primer PDAM di Aurduri?

## **1.3 Batasan Masalah**

Adapun batasan masalah pada penelitian ini ialah sebagai berikut:

1. Graf yang digunakan merupakan graf dari bobot panjang jaringan pipa primer PDAM Tirta Mayang di Aurduri.
2. Pencarian *Minimum Spanning Tree* dilakukan berdasarkan jumlah bobot yang dihasilkan.
3. Pencarian *Minimum Spanning Tree* dilakukan menggunakan Algoritma Kruskal dilihat dari jarak antar persimpangan jalan utama.

## **1.4 Tujuan Penelitian**

Adapun tujuan dari dilakukannya penelitian ini adalah mampu menemukan pohon merentang minimum dengan menggunakan Algoritma Kruskal untuk mengoptimalkan jaringan pipa primer PDAM di Aurduri.

## **1.5 Manfaat Penelitian**

Adapun manfaat yang diperoleh dari penelitian ini ialah sebagai berikut:

1. Manfaat bagi penulis  
Sebagai tambahan informasi dan wawasan pengetahuan tentang teori graf, khususnya tentang pohon merentang minimum dan Algoritma Kruskal serta implementasinya di kehidupan nyata.
2. Manfaat bagi PDAM  
Dapat dijadikan sebagai bahan pertimbangan pengambilan keputusan dalam pembaharuan jaringan pipa PDAM yang lebih optimal.

## II. TINJAUAN PUSTAKA

### 2.1 Teori Graf

Teori graf adalah ilmu yang mempelajari tentang graf struktur matematika. Aplikasi dari teori graf sangat luas dan dapat digunakan dalam berbagai disiplin ilmu maupun dalam kehidupan sehari-hari. Dalam implementasinya teori ini sering dimanfaatkan di dalam bidang informatika (penerapan graf pada jaringan), kimia (memodelkan senyawa dalam bentuk graf) dan kelistrikan (jaringan listrik). Pemanfaatan teori graf yang sangat umum digunakan pada pencarian pohon merentang minimum (*Minimum Spanning Tree*), *Travelling Salesman Problem* (TSP) dan *coloring graph* (Monifani dkk, 2014).

Graf dimanfaatkan untuk mempresentasikan objek-objek diskrit dan relasi antar objek-objek. Graf  $G$  diartikan sebagai pasangan himpunan  $(V, E)$ , ditulis dengan notasi  $G = (V, E)$ . Dalam hal ini,  $V$  adalah himpunan tidak kosong dari simpul-simpul (vertex atau node) digambarkan dalam titik-titik, dan  $E$  merupakan himpunan sisi-sisi (*edges*) digambarkan dalam garis-garis yang menghubungkan sepasang simpul (Munir, 2005).

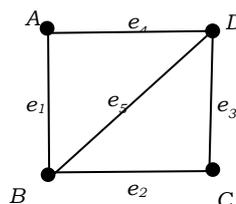
### 2.2 Graf dan Komponen Graf

Graf  $G$  adalah pasangan himpunan  $(V, E)$ , atau biasa dinotasikan dengan  $G = (V, E)$ , dengan

- $V$  adalah himpunan tak kosong titik-titik/ simpul / verteks/ node, dan
- $E$  adalah himpunan garis / rusuk / sisi / edge yang menghubungkan sepasang simpul.

Contoh :

Graf  $G_1$  dengan  $V = \{A, B, C, D\}$  dan  $E = \{e_1, e_2, e_3, e_4, e_5\}$



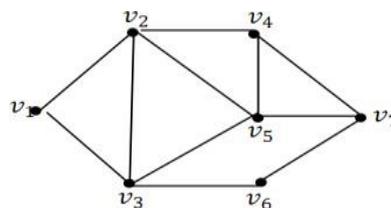
**Gambar 2.1** Contoh graf  $G_1$

Contoh Graf  $G_1$  Pada graf  $G_1$ , sisi  $e_1$  boleh juga dituliskan dengan sisi  $AB$  atau  $BA$ . Jika  $e_1 = AB = BA$ , maka verteks  $A$  dan  $B$  dikatakan bertetangga dalam graf  $G_1$ , atau sisi  $e_1$  menghubungkan verteks  $A$  dan  $B$ . Selanjutnya verteks  $A$  dan sisi  $e_1$  dikatakan bersisian. Verteks  $B$  dan sisi  $e$  juga dikatakan bersisian.

**Definisi 2.1 (Munir, 2016)** Graf  $G$  diartikan sebagai pasangan himpunan  $(V,E)$ , ditulis dengan notasi  $G = (V,E)$ , yang dalam hal ini  $V$  adalah himpunan tidak kosong dari simpul-simpul dan  $E$  adalah himpunan sisi yang menghubungkan sepasang simpul.

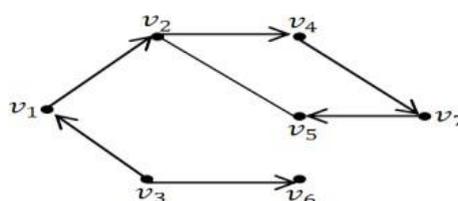
Arti tersebut mengutarakan bahwa  $V$  tidak boleh kosong, sedangkan  $E$  boleh kosong. Jadi, sebuah graf dimungkinkan tidak mempunyai sisi satu buah pun, tetapi simpulnya harus ada, minimal satu. Graf dapat dikelompokkan menjadi beberapa jenis tergantung pada sudut pandang pengelompokkannya. Berdasarkan orientasi arah pada suatu graf, maka graf digolongkan menjadi dua jenis, yaitu graf tak berarah (undirected graph) dan graf berarah (*directed graph*).

Menurut Ramadhan (2017), Graf yang sisinya tak mempunyai orientasi arah disebut dengan graf tak berarah, urutan pasangan simpul yang dihubungkan oleh sisi tidak diperhatikan. Jadi  $i,j = (j,i)$  adalah sisi yang sama. Sisi pada graf ini dinamakan *edge*. Adapun contoh dari graf tak berarah dapat ditunjukkan pada gambar berikut:



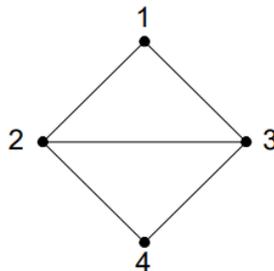
**Gambar 2.2** Graf Tak Berarah

Sedangkan menurut (Anggraeni, 2015), Graf yang setiap sisinya diberikan orientasi arah dinamakan graf berarah. Sisi berarah pada graf ini dinamakan *arc*. Pada graf ini belum tentu  $i,j = (j,i)$  bisa saja  $i,j \neq (j,i)$ . Adapun contoh dari graf berarah dapat ditunjukkan pada gambar berikut:



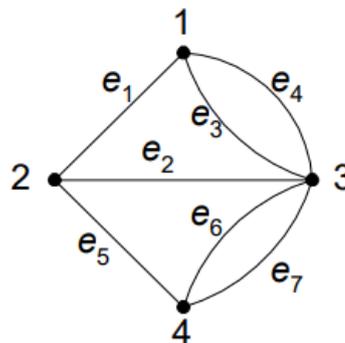
**Gambar 2.3** Graf Berarah

Menurut Anggraeni (2015), berdasarkan ada atau tidaknya gelang atau sisi ganda pada suatu graf, dapat digolongkan menjadi dua jenis yaitu graf sederhana dan graf tak sederhana. Graf sederhana adalah graf yang tidak memiliki sisi kalang maupun sisi ganda di dalamnya. Pada graf sederhana, sisi merupakan pasangan tak-terurut yang artinya jika menuliskan sisi  $(u,v)$  akan sama saja dengan sisi  $(v,u)$ . Adapun contoh dari graf sederhana dapat ditunjukkan pada gambar berikut:

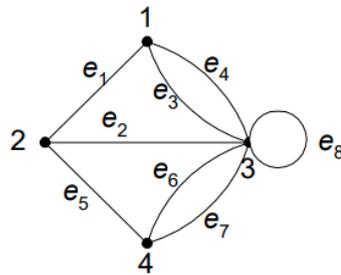
**Gambar 2.4** Graf sederhana

Menurut Anggraeni (2015), Graf yang mengandung sisi ganda atau gelang dinamakan graf tak sederhana. Ada dua jenis graf tak-sederhana, yaitu graf ganda dan graf semu. Adapun pengertian dan contoh dari graf ganda dan graf semu ialah dapat ditunjukkan sebagai berikut:

Graf ganda ialah graf tak sederhana yang mengandung sisi ganda. Di bawah ini merupakan bentuk dari graf ganda.

**Gambar 2.5** Graf Ganda

Graf semu ialah graf tak sederhana yang mengandung *loop*. Di bawah ini merupakan bentuk dari graf semu.



**Gambar 2.6** Graf semu

Ada beberapa terminologi dari teori graf yang digunakan untuk menjelaskan apa yang dilihat ketika melihat suatu graf. Graf dapat dilihat dari komponen-komponen penyusunnya, yang terdiri dari :

1. Titik (*Verteks*)

**Definisi 2.2 (Prasetyo, 2013)** Titik (*Verteks*) yang disimbolkan dengan  $v$  adalah himpunan titik yang terbatas dan tidak kosong. Jumlah titik pada graf dapat dinyatakan dengan  $n = |v|$ .

2. Sisi (*Edge*)

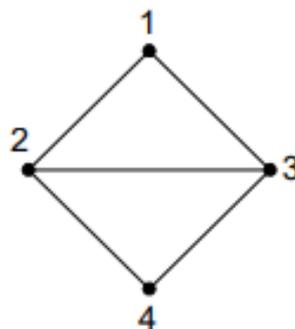
**Definisi 2.3 (Prasetyo, 2013)** Sisi (*edge*) yang disimbolkan dengan  $e$  adalah himpunan sisi yang menghubungkan sepasang titik.

3. Derajat (*Degree*)

**Definisi 2.4 (Munir, 2016)** Derajat yang disimbolkan dengan  $d(v)$  suatu simpul pada graf tak berarah adalah jumlah sisi yang bersisian dengan simpul tersebut.

4. Siklus atau Sirkuit

**Definisi 2.5 (Munir, 2016)** Siklus atau sirkuit adalah lintasan yang berawal dan berakhir pada simpul yang sama.

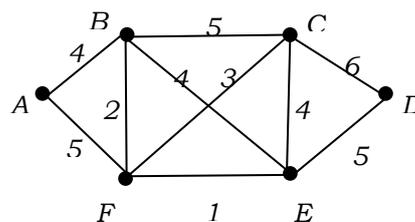


**Gambar 2.7** Graf Siklus atau Sirkuit

Tinjau graf pada **Gambar 2.7** : 1-2-3-1 adalah sebuah sirkuit. Panjang sirkuit adalah jumlah sisi dalam sirkuit tersebut. Sirkuit 1-2-3-1 pada graf tersebut memiliki panjang 3.

## 5. Graf Bobot

**Definisi 2.6 (Munir, 2016)** Graf berbobot adalah graf yang setiap sisinya diberi sebuah harga atau bobot. Bobot pada setiap sisi dapat berbeda-beda bergantung pada masalah yang dimodelkan dengan graf.



**Gambar 2.8** Graf Berbobot

Pada **Gambar 2.8** merupakan graf berbobot sebab pada setiap sisi ada bobotnya. Misal sisi  $(a, b)$  berbobot 4, sisi  $(a, f)$  berbobot 5, dan seterusnya setiap sisi memiliki bobot masing-masing.

## 2.3 Representasi Graf

Menurut Munir (2003), terdapat tiga representasi graf yaitu:

### 1. Matriks Ketetanggaan (*adjacency matrix*)

Misalkan  $G = (V, E)$  graf sederhana dimana  $|V| = n$ ,  $n > 1$ . Maka, matriks ketetanggaan  $A$  dari  $G$  adalah matriks  $n \times n$ . Dimisalkan  $a_{ij}$  merupakan indeks unsur pada matriks tersebut, maka:

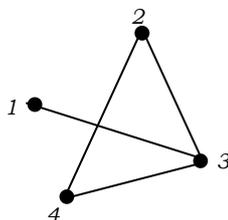
$$A = [a_{ij}],$$

$[a_{ij}]$  menjadi 1 bila simpul  $i$  dan  $j$  bertetangga

$[a_{ij}]$  menjadi 0 bila simpul  $i$  dan  $j$  tidak bertetangga

Jumlah elemen matriks bertetanggaan untuk graf dengan  $n$  simpul adalah  $n^2$ . Jika tiap elemen membutuhkan ruang memori sebesar  $p$ , maka ruang memori yang diperlukan seluruhnya adalah  $pn^2$ .

Keuntungan representasi dengan matriks ketetangaan adalah kita dapat mengakses elemen matriksnya langsung dari indeks. Selain itu, kita juga dapat menentukan dengan langsung apakah simpul  $i$  dan simpul  $j$  bertetangga. Pada graf berbobot,  $a_{ij}$  menyatakan bobot tiap sisi yang menghubungkan simpul  $i$  dengan simpul  $j$ . Bila tidak ada sisi dari simpul  $i$  ke simpul  $j$  atau dari simpul  $j$  ke simpul  $i$ , maka,  $a_{ij}$  diberi nilai tak berhingga.



**Gambar 2.9** Graf Matriks Ketetangaan

Bentuk matriks ketetangaan dari graf pada **Gambar 2.9** adalah

$$\begin{array}{c}
 \\
 1 \\
 2 \\
 3 \\
 4
 \end{array}
 \begin{array}{c}
 1 \quad 2 \quad 3 \quad 4 \\
 \left| \begin{array}{cccc}
 0 & 0 & 1 & 0 \\
 0 & 0 & 1 & 1 \\
 1 & 1 & 0 & 1 \\
 0 & 1 & 1 & 0
 \end{array} \right|
 \end{array}$$

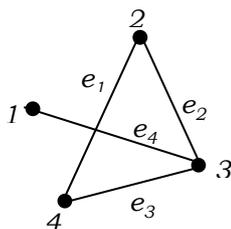
## 2. Matriks Bersisian (*incidency matrix*)

Matriks bersisian menyatakan kebersisian simpul dengan sisi. Misalkan  $G = (V, E)$  adalah graf dengan  $n$  simpul dan  $m$  sisi, maka matriks kebersisian  $A$  dari  $G$  adalah matriks berukuran  $m \times n$ . Dimisalkan  $a_{ij}$  merupakan indeks unsur pada matriks tersebut, maka:

$$A = [a_{ij}],$$

$[a_{ij}]$  menjadi 1 bila simpul  $i$  dan sisi  $j$  bersisian

$[a_{ij}]$  menjadi 0 bila simpul  $i$  dan sisi  $j$  tidak bersisian



**Gambar 2.10** Graf Matriks Bersisian

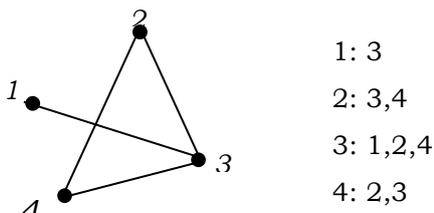
Bentuk matriks bersisian dari graf pada **Gambar 2.10** adalah

$$\begin{array}{c|cccc}
 & e_1 & e_2 & e_3 & e_4 \\
 \hline
 1 & 0 & 0 & 0 & 1 \\
 2 & 1 & 1 & 0 & 0 \\
 3 & 0 & 1 & 1 & 1 \\
 4 & 1 & 0 & 1 & 0
 \end{array}$$

### 3. Senarai Ketetangaan (*adjacency list*)

Matriks ketetangaan memiliki kelemahan apabila graf memiliki jumlah sisi yang relatif sedikit sehingga graf sebagian besar berisi bilangan 0. Hal ini merupakan pemborosan terhadap memori, karena banyak menyimpan bilangan 0 yang seharusnya tidak perlu disimpan. Untuk kepentingan efisiensi ruang, maka tiap baris matriks tersebut digantikan senarai yang hanya berisikan *vertex-vertex* dalam *adjacency set*  $V_x$  dari setiap *vertex*  $x$ .

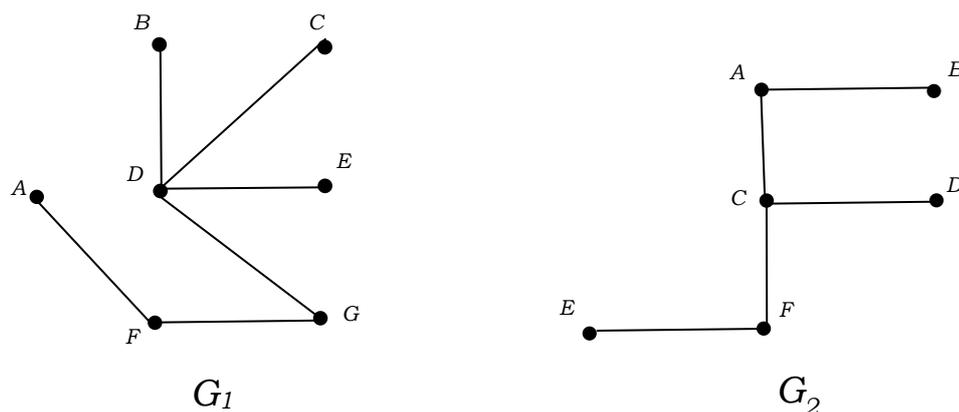
Bentuk senarai ketetangaan berdasarkan graf pada **Gambar 2.9** adalah



## 2.4 Pohon (*Tree*)

Pohon merupakan salah satu bentuk khusus dari suatu graf. Konsep pohon pernah diterapkan pada tahun 1870-an oleh Matematikawan Inggris yang bernama Arthur Cayley dalam penghitungan molekul kimia. Karya yang lebih baru membuktikan bahwa pohon digunakan di banyak bidang, mulai dari linguistik sampai komputer.

**Definisi 2.7 (Pratama Dkk, 2013)** Pohon adalah suatu graf terhubung yang tidak mempunyai subgraf yang memuat siklus.



**Gambar 2.11** Graf Pohon

Pada **Gambar 2.11** dapat dijelaskan bahwa  $G_1$  dan  $G_2$  merupakan contoh-contoh graf pohon karena graf-graf tersebut saling terhubung tetapi tidak memiliki sirkuit. Menurut Wayangkau (2015), sifat-sifat pohon ialah sebagai berikut: Misal  $G = (V,E)$  adalah graf tak berarah sederhana dan jumlah simpulnya  $n$  buah, maka graf  $G$  adalah pohon. Setiap pasang simpul dalam graf  $G$  terhubung dengan lintasan tunggal, graf  $G$  tidak mengandung sirkuit dan memiliki  $m = n - 1$  buah sisi, graf  $G$  tidak mengandung sirkuit dan penambahan satu sisi pada graf akan membuat hanya satu sirkuit dan graf  $G$  terhubung dan semua sisinya adalah jembatan. Seperti  $G_1$  pada **Gambar 2.8** merupakan graf tak berarah sederhana, jumlah simpulnya 6 dan jumlah sisinya 5 serta tidak mengandung sirkuit.

## 2.5 Pohon Merentang

**Definisi 2.8 (Ramadhan, 2017)** Misalkan  $G = (V,E)$  adalah graf tak berarah terhubung yang memiliki beberapa sirkuit.  $G$  dapat diubah menjadi pohon  $T = (V_1, E_1)$  dengan cara menghilangkan sirkuit-sirkuit yang sebelumnya terhubung. Caranya adalah mula-mula pilih sebuah sirkuit, kemudian hapus sebuah sisi dari sirkuit tersebut. Lakukan proses tersebut sampai semua sirkuit pada  $G$  hilang dan menjadi sebuah pohon  $T$  yang dinamakan pohon Merentang.

## 2.6 Pohon Merentang Minimum (*Minimum Spanning Tree / MST*)

**Definisi 2.9 (Ramadhan, 2017)** Pohon merentang minimum adalah pohon merentang dari suatu graf berbobot  $G$  yang memiliki bobot minimum di antara semua pohon merentang yang dapat dibentuk di graf  $G$ .

Permasalahan pohon merentang minimum serupa dengan masalah untuk mencari lintasan terpendek dengan metode djikstra (shortest path).

Terdapat perbedaan antara Pohon Merentang Minimum dengan pencarian lintasan terpendek (*shortest path*), perbedaannya terletak pada jarak yang akan ditempuh dan jumlah titik atau simpul yang terhubung. Pada Pohon Merentang Minimum semua simpul yang ada harus saling terhubung dengan menghasilkan jarak terdekat maksimal pada suatu graf, tetapi tak boleh terbentuk sirkuit pada graf tersebut. Sementara pada pencarian Lintasan terpendek (*shortest path*) tidak harus menghubungkan semua titik atau simpul yang ada untuk mendapatkan jalur terpedek dari titik atau simpul awal ke simpul tujuan (Ismail dan Setiadi, 2014).

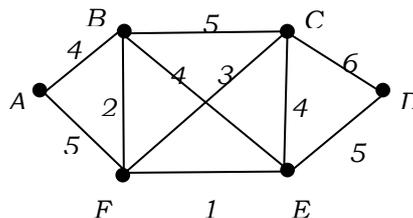
Adapun syarat suatu graf yang dapat dicari pohon merentang minimumnya ialah di antaranya graf tersebut harus terhubung, ruasnya punya bobot/nilai dan graf tersebut tidak berarah. Ada dua algoritma yang dapat membentuk pohon merentang minimum. Salah satu algoritma yang dapat membentuk pohon merentang minimum adalah Algoritma Kruskal.

## **2.7 Algoritma Kruskal**

Algoritma Kruskal merupakan salah satu algoritma dalam teori graf untuk menyelesaikan persoalan pohon merentang minimum. Algoritma Kruskal ditemukan pada tahun 1956 oleh seorang ilmuwan matematika, statistika, komputer dan psikometrika Joseph yaitu Bernard Kruskal, Jr yang berasal dari Amerika. Dia adalah seorang mahasiswa di Universitas Chicago mendapatkan gelar sarjana sains dalam matematika pada tahun 1948 dan master sains dalam matematika pada tahun berikutnya 1949. Dalam statistik, karya Kruskal yang paling berpengaruh adalah kontribusi untuk perumusan penskalaan multidimensi. Dalam ilmu komputer, karyanya yang paling terkenal adalah Algoritma Kruskal untuk menghitung Pohon Merentang Minimum dari grafik berbobot. Algoritma pertama-tama mengurutkan tepi berdasarkan bobot dan kemudian melanjutkan melalui daftar terurut dengan menambahkan tepi baru tidak membuat siklus. Pohon rentang mninimal memiliki aplikasi untuk kontruksi dan penetapan harga jaringan komunikasi. Dalam kombinatorik, ia dikenal dengan torema pohon Kruskal (1960), yang juga menarik dari perspektif logika matematika karena hanya dapat dibuktikan secara nonkonstruktif. Kruskal juga menerapkan karyanya dalam linguistik, dalam studi leksikostatistik eksperimental bahasa Indo-Eropa, bersama dengan ahli bahasa Isidore Dyen dan Paul Black. Database mereka masih banyak digunakan. Dasar pembentukan Algoritma Kruskal berasal dari analogi *growing forest*. *Growing forest* maksudnya adalah untuk membentuk pohon merentang minimum  $T$  dari graf  $G$  adalah dengan cara mengambil satu persatu sisi dari graf  $G$  dan

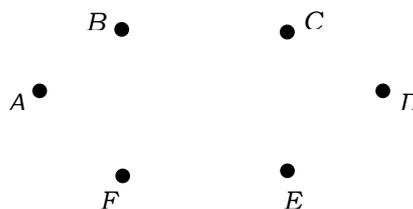
memasukkannya dalam pohon yang telah terbentuk sebelumnya. Seiring dengan berjalannya iterasi pada setiap sisi maka *forest* akan memiliki pohon yang semakin sedikit. Oleh sebab itu analogi ini disebut *growing forest*. Algoritma Kruskal akan terus menambahkan sisi-sisi ke dalam hutan sesuai hingga akhirnya tidak akan ada lagi *forest*, melainkan hanyalah sebuah pohon merentang minimum (Wattimena dan Lawatama, 2013).

Menurut D. K. Dwiyanto dan S. Nurhayati (2014), Algoritma Kruskal adalah sebuah algoritma dalam teori graf yang mencari sebuah *Minimum Spanning Tree* untuk sebuah graf berbobot yang terhubung. Ini berarti mencari subset dari sisi yang membentuk sebuah Tree yang menampung setiap verteks, dimana total bobot dari semua sisi dalam Tree adalah minimum. Pada Algoritma Kruskal, sisi (*edge*) dari Graf diurut terlebih dahulu berdasarkan bobotnya dari kecil ke besar. Sisi yang dimasukkan ke dalam himpunan T adalah sisi graph G yang sedemikian sehingga T adalah Tree (pohon). Sisi dari Graph G ditambahkan ke T jika ia tidak membentuk sirkuit. Kelebihan Algoritma Kruskal sangat cocok digunakan saat graf memiliki sisi berjumlah sedikit namun memiliki sangat banyak simpul, karena orientasi kerja algoritma ini adalah berdasarkan urutan bobot sisi bukan simpul. Kompetensi-kompetensi dinyatakan dalam node dan sisi (*edge*) adalah jarak antar kompetensi. Langkah-langkah Algoritma Kruskal adalah sebagai berikut:



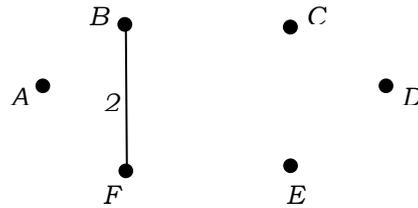
**Gambar 2.12** Graf T adalah Graf berbobot yang terhubung (masih memiliki sirkuit).

1. T masih kosong



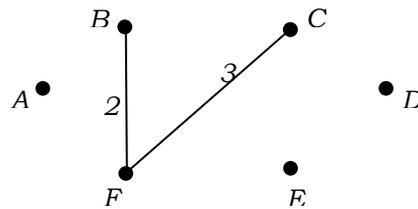
**Gambar 2.13** T yang masih berupa simpul atau masih kosong

2. Pilih sisi dengan bobot minimum. Dipilih sisi  $(B, F)$  dengan bobot 2.



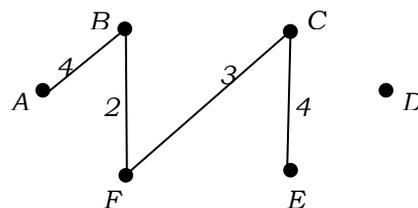
**Gambar 2.14** Sisi  $(B, F)$  pada  $T$  dipilih

3. Pilih sisi dengan bobot minimum berikutnya yang tidak membentuk sirkuit di  $T$ , tambahkan sisi ke  $T$ . Dipilih sisi  $(C, F)$  dengan bobot 3.



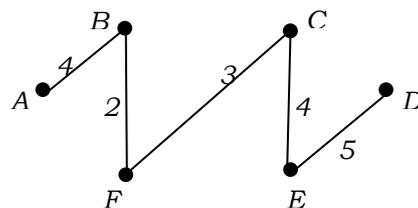
**Gambar 2.15** Sisi  $(C, F)$  pada  $T$  dipilih

4. Ulangi langkah 3 sebanyak  $(n - 2)$  kali. Dipilih sisi  $(A, B)$  dan sisi  $(C, E)$  dengan bobot yang sama yaitu 4.



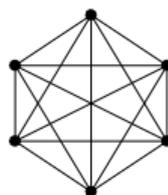
**Gambar 2.16** Sisi  $(A, B)$  dan  $(C, E)$  pada  $T$  dipilih

5. Total langkah  $(n-1)$  kali. Lalu dipilih sisi  $(E, D)$  dengan bobot 2. Maka semua simpul sudah terpenuhi, terbentuklah pohon merentang minimum dengan Algoritma Kruskal.



**Gambar 2.17** Sisi  $(D, E)$  pada  $T$  dipilih dan didapatlah pohon merentang minimum dari graf  $T$

Menurut Wisra, Yuliani dan Marwan (2017), Algoritma Kruskal memiliki kelebihan dibanding algoritma lain. Algoritma Kruskal sangat cocok untuk digunakan saat graf memiliki jumlah sisi sedikit, tetapi memiliki banyak simpul. Karena orientasi cara kerja Algoritma Kruskal berdasarkan pada urutan bobot sisi, tidak berdasarkan simpul. Sementara kekurangannya terletak pada kurang cocoknya Algoritma Kruskal diterapkan saat graf lengkap atau mendekati lengkap. Graf lengkap adalah graf sederhana yang setiap simpulnya mempunyai sisi ke semua simpul lainnya. Graf lengkap memiliki banyak sirkuit dan ini cukup menyulitkan untuk perhitungan menggunakan Algoritma Kruskal. Karena algoritma ini memberatkan pada pencarian sisi, di mana sisi-sisi ini harus diurutkan dan hal ini memakan cukup waktu.



**Gambar 2.18** Contoh graf lengkap (memiliki banyak sirkuit).

Menurut Latifah (2014), penelitian dengan menerapkan Algoritma Kruskal sudah cukup banyak dimanfaatkan oleh masyarakat, terutama di beberapa industri yang ada di Indonesia. Salah satu yang terbesar adalah Perusahaan Daerah Air Minum (PDAM) di Kota Semarang. Latifah meneliti dengan menggunakan program aplikasi Matlab untuk memecahkan permasalahan tersebut, untuk itu diperlukan rencana yang tepat untuk membuat jalur perpipaan untuk menghemat biaya. Dengan demikian diperlukan adanya suatu alat, teknik maupun metode praktis, efektif dan efisien. Dalam penelitiannya itu ia menerapkan Algoritma Kruskal untuk mengoptimalkan jaringan pipa di Perumahan Ratulagi Regency.

## 2.8 Sistem Jaringan Pipa Air Bersih

Menurut Joko (2010), definisi Sistem Jaringan Pipa Transmisi Air Bersih adalah sistem pengaliran air sebelum masuk ke bangunan pengolahan. Pengaliran dapat dilakukan dengan menggunakan pompa maupun dilakukan secara gravitasi.

Air yang dihasilkan dari IPA dapat ditampung dalam reservoir air yang berfungsi untuk menjaga kesetimbangan antara produksi dengan kebutuhan. Reservoir air dibangun di dalam tanah atau dalam bentuk menara air yang

umumnya untuk mengantisipasi kebutuhan puncak di daerah distribusi. (Peraturan Menteri Pekerjaan Umum No 18 tahun 2007. Penyelenggaraan Pengembangan Sistem Penyediaan Air Minum).

Perencanaan SPAM unit distribusi dapat berupa jaringan perpipaan yang membentuk jaringan tertutup (*loop*), sistem jaringan distribusi bercabang (*deadend distribution system*), atau kombinasi dari kedua sistem tersebut (*grade system*). Bentuk jaringan distribusi ditentukan oleh kondisi topografi, lokasi reservoir, luas wilayah pelayanan, jumlah pelanggan dan jaringan jalan dimana pipa akan dipasang. (Peraturan Menteri Pekerjaan Umum No 18 tahun 2007. Penyelenggaraan Pengembangan Sistem Penyediaan Air Minum).

Menurut peraturan menteri Pekerjaan umum Penyelenggaraan Pengembangan Sistem Penyediaan Air Minum (2010 : 54) Ketentuan-ketentuan yang harus dipenuhi dalam perancangan denah (*lay-out*) sistem distribusi adalah sebagai berikut:

1. Denah (*lay-out*) sistem distribusi ditentukan berdasarkan keadaan topografi wilayah pelayanan dan lokasi instalasi pengolahan air.
2. Tipe sistem distribusi ditentukan berdasarkan keadaan topografi wilayah pelayanan.
3. Jika keadaan topografi tidak memungkinkan untuk sistem gravitasi selurunya, diusulkan kombinasi sistem gravitasi dan pompa. Jika semua wilayah pelayanan relative datar, dapat digunakan sistem perpompaan langsung, kombinasi dengan menara air, atau penambahan pompa penguat (*booster pump*).
4. Jika terdapat perbedaan elevasi wilayah pelayanan terlalu besar atau lebih dari 40 m, wilayah pelayanan dibagi menjadi beberapa zone sedemikian rupa sehingga memenuhi persyaratan tekanan minum.

Jaringan Distribusi adalah jaringan pipa yang digunakan untuk mengalirkan air dari reservoir ke tempat pemakaian (konsumen). Jaringan distribusi diperlukan untuk mengalirkan dan membagikan air kepada konsumen pada daerah pelayanan (Khanif dan Uszhani, 2010).

Sistem Pipa Distribusi terbagi dua yaitu;

1. Sistem Cabang Sistem bercabang adalah sistem jaringan pipa induk yang berbentuk cabang, sehingga terdapat satu arah aliran dari pipa induk ke pipa cabang sekunder, kemudian seterusnya ke pipa cabang tersier. Kelemahan dari pipa ini adalah pada ujungnya bertumpuk kotoran yang dapat menutup pipa sehingga distribusi terhenti (Khanif dan Uszhani, 2010).

2. Sistem *Loop* Sistem *Loop* adalah sistem jaringan pipa induk yang melingkar dan tertutup sehingga terdapat arah bolak balik. Pada sistem ini pipa utama/induk dibuat melingkar. Dibandingkan dengan sistem cabang, sistem ini lebih baik karena sirkulasi air lebih baik dan bilamana ada kerusakan pada saat perabakaan distribusi air tidak terhenti (Khanif dan Uszhani, 2010).

## **2.9 Jaringan Pipa PDAM Aurduri**

Berdasarkan Dokumen RPI2-JM Kota Jambi Tahun 2016-2020, jaringan pipa PDAM Aurduri merupakan bagian dari PDAM Tirta Mayang Kota Jambi. Pada tahun 1997-1998, proyek kerja sama atau kemitraan PDAM dengan pihak swasta mengembangkan SPAM di wilayah barat Kota Jambi, khususnya Kecamatan Telanaipura dan Kota Baru dengan membangun IPA Aurduri kapasitas 100 liter/detik dan jaringan pipa induk distribusi.

IPA Aurduri merupakan IPA milik swasta yang dibangun berdasarkan kerja sama model BOT (Build, Operate, Transfer) dengan masa konsesi selama 15 tahun. Selama masa konsensi, PDAM membeli air terolah dengan pembelian model air curah. Untuk merealisasikan target MDG's 2015 dengan cakupan pelayanan wilayah perkotaan sebesar minimal 78%, PDAM merencanakan untuk melanjutkan program kerja salah satunya yaitu menambah kapasitas Aurduri dari 100 liter/detik menjadi 200 liter/detik yang akan menambah pelanggan sebanyak 6000 SR. Harga pipa pun bervariasi. Untuk pipa air jenis PVC-O yang biasa dijadikan untuk saluran air bersih, air limbah dan saluran air hujan memiliki harga berbeda tergantung diameternya. Pipa PVC-O berdiameter 16" memiliki harga Rp.1.400.000/meter.

Seperti pada jaringan pipa PDAM di tempat lain pada umumnya, jaringan pipa PDAM Aurduri juga merupakan pengaplikasian dari pohon. Dengan memodelkannya ke dalam bentuk graf, cukup memiliki banyak titik dan sisinya tidak terlalu banyak. Maka dapat ditentukan pohon merentang minimumnya dari setiap jaringan pipa PDAM di Aurduri dengan menggunakan Algoritma Kruskal.

### III. METODOLOGI PENELITIAN

#### 3.1 Jenis dan Sumber Data

Jenis data yang digunakan dalam penelitian ini berupa data sekunder, yaitu data dari jaringan pipa PDAM di Perumahan Aurduri. Data akan diminta dari Perusahaan Daerah Air Minum (PDAM) Tirta Mayang Kota Jambi Jl. Letkol Jl. Slamet Riyadi, Solok Sipin, Kec. Telanai Pura, Kota Jambi, Jambi 36121. Prosedur pengambilan data juga dibantu dengan menggunakan *Google Maps*.

#### 3.2 Metode Penelitian

Adapun langkah-langkah penelitian yang dilakukan adalah sebagai berikut :

1. Identifikasi Masalah

Pada penelitian ini dilakukan pengoptimalan pada jaringan pipa PDAM yang ada di Aurduri demi terjalannya rencana induk sistem penyediaan air bersih. Pengoptimalan jaringan pipa dilakukan pencarian pohon merentang minimum atau mencari bobot paling minimum dari banyaknya jaringan yang dapat dibentuk. Untuk mencari pohon merentang minimum menggunakan metode Algoritma Kruskal.

2. Pengumpulan Data

Tahapan ini dilakukan dengan melakukan pengambilan data di Perusahaan Daerah Air Minum (PDAM) Tirta Mayang Kota Jambi Jl. Letkol Jl. Slamet Riyadi, Solok Sipin, Kec. Telanai Pura, Kota Jambi, Jambi 36121. Pengumpulan data juga dapat dilakukan dengan mengambil data dari *google maps*.

3. Membuat Graf Awal

Tahapan ini dilakukan dengan membentuk graf awal dari jaringan pipa yang nantinya dicari pohon merentang minimumnya. Pada penelitian ini graf awal dibentuk dengan menentukan titik-titik dan sisi-sisi berdasarkan peta perumahan Aurduri hingga membentuk graf berbobot.

4. Pencarian Pohon Merentang Minimum Menggunakan Algoritma Kruskal

Setelah dibentuk graf berbobot, dijadikan graf kosong, urutkan sisi yang terkecil sampai terbesar, pilih sisi terkecil tetapi tidak boleh terbentuk sirkuit. Lakukan hingga pohon merentang minimum terbentuk.

5. Ditemukan pohon merentang minimum.

Setelah dianalisis menggunakan Algoritma Kruskal, yaitu pencarian sisi-

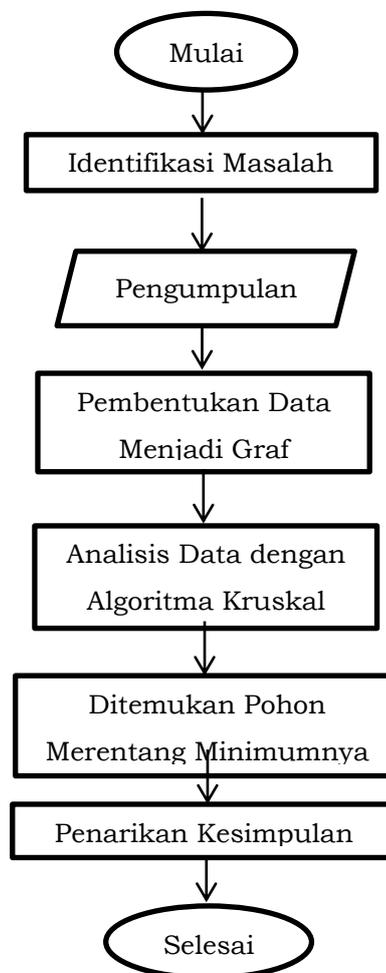
sisi terpendek pada setiap simpul dapat ditemukan pohon merentang minimumnya.

6. Penarikan Kesimpulan.

Kesimpulan dapat ditarik setelah didapat perbedaan dari jumlah bobot awal jaringan sebelum dan sesudah dilakukan analisis. Bila bobot jaringan awal lebih panjang, setelah dilakukan analisis menjadi lebih pendek. Maka pencarian pohon merentang minimum dengan Algoritma Kruskal berhasil.

### 3.3 Alur Penelitian

Adapun langkah-langkah yang dilakukan dalam menyelesaikan penelitian ini, secara skematik dapat dilihat pada **Gambar 3.1** berikut :

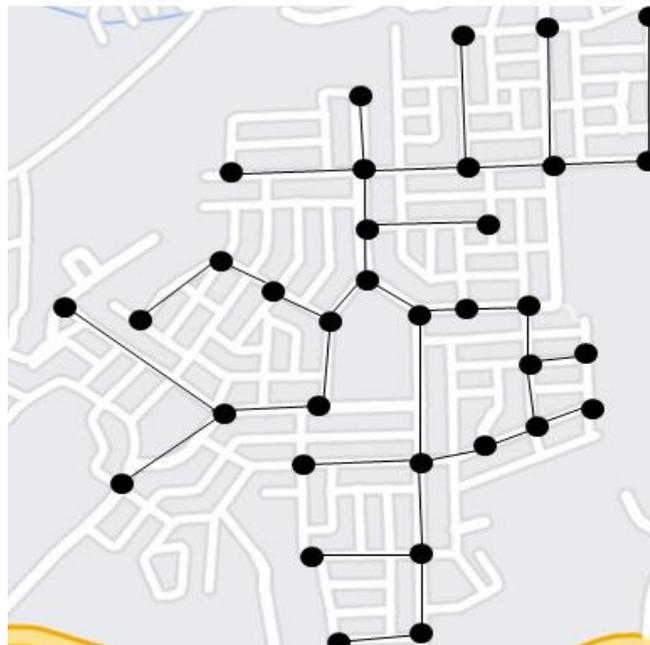


**Gambar 3.1** Diagram Alir

## IV. HASIL DAN PEMBAHASAN

### 4.1 Pengumpulan Data

Jaringan pipa PDAM yang diteliti adalah jaringan pipa PDAM di Perumahan Aurduri Indah yang terletak di Kelurahan Penyengat Rendah Kecamatan Telanaipura Kota Jambi. Data diperoleh dari pencarian dan penelitian secara langsung. Prosedur pengambilan data dibantu dengan menggunakan *Google Maps*. *Google Maps* digunakan sebagai salah satu sumber data yang mengacu pada koefisien yang cukup efektif dari segi jarak. Setiap simpul atau titik pada denah mewakili satu persimpangan di jalan utama. Dikarenakan pada perhitungan ini dicari panjang atau bobot jaringan pipa primer. Berikut ini merupakan denah dari Perumahan Aurduri Indah.



**Gambar 4.1** Denah Lokasi Perumahan Aurduri yang telah diberi simpul

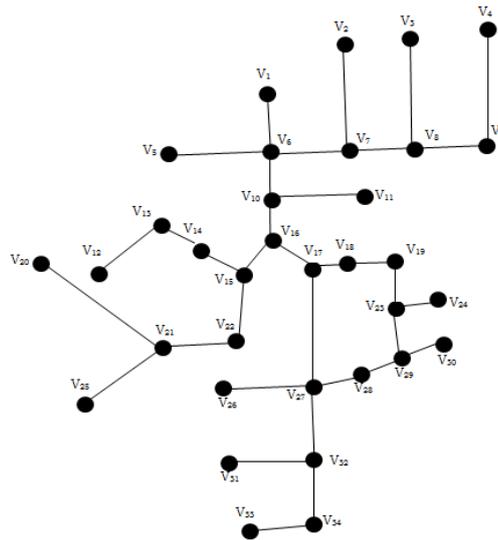
Sumber : *Google Maps*

Adapun dari denah lokasi di atas, dapat dibentuk data berupa bobot panjang jaringan pipa primer PDAM di Perumahan Aurduri sebagai berikut:

**Tabel 4.1** Data Panjang jaringan pipa primer PDAM Perumahan Aurduri

No.	Sisi	Panjang (m)
1	V <sub>1</sub> -V <sub>6</sub>	80
2	V <sub>2</sub> -V <sub>7</sub>	176
3	V <sub>3</sub> -V <sub>8</sub>	188
4	V <sub>4</sub> -V <sub>9</sub>	193
5	V <sub>5</sub> -V <sub>6</sub>	176
6	V <sub>6</sub> -V <sub>7</sub>	152
7	V <sub>6</sub> -V <sub>10</sub>	71
8	V <sub>7</sub> -V <sub>8</sub>	158
9	V <sub>8</sub> -V <sub>9</sub>	160
10	V <sub>10</sub> -V <sub>11</sub>	136
11	V <sub>10</sub> -V <sub>16</sub>	56
12	V <sub>12</sub> -V <sub>13</sub>	129
13	V <sub>13</sub> -V <sub>14</sub>	100
14	V <sub>14</sub> -V <sub>15</sub>	71
15	V <sub>15</sub> -V <sub>16</sub>	61
16	V <sub>15</sub> -V <sub>22</sub>	105
17	V <sub>16</sub> -V <sub>17</sub>	55
18	V <sub>17</sub> -V <sub>18</sub>	60
19	V <sub>17</sub> -V <sub>27</sub>	193
20	V <sub>18</sub> -V <sub>19</sub>	70
21	V <sub>19</sub> -V <sub>23</sub>	72
22	V <sub>20</sub> -V <sub>21</sub>	240
23	V <sub>21</sub> -V <sub>22</sub>	118
24	V <sub>21</sub> -V <sub>25</sub>	133
25	V <sub>23</sub> -V <sub>24</sub>	78
26	V <sub>23</sub> -V <sub>29</sub>	72
27	V <sub>26</sub> -V <sub>27</sub>	81
28	V <sub>27</sub> -V <sub>28</sub>	75
29	V <sub>27</sub> -V <sub>32</sub>	105
30	V <sub>28</sub> -V <sub>29</sub>	72
31	V <sub>29</sub> -V <sub>30</sub>	100
32	V <sub>31</sub> -V <sub>32</sub>	132
33	V <sub>32</sub> -V <sub>34</sub>	125
34	V <sub>33</sub> -V <sub>34</sub>	128

Dari data tersebut digambarkan sebuah graf terhubung sebagai berikut:



**Gambar 4.2** Jaringan pipa primer Perumahan Aurduri dalam bentuk graf sebelum dicari pohon Merentang minimum

Sisi pada tabel data tersebut menunjukkan hubungan antara satu titik ke titik lain. Sedangkan panjang pada tabel tersebut menunjukkan panjangnya jaringan pipa primer yang digunakan dalam satuan meter.

#### 4.2 Perhitungan Data

Tahapan perhitungan data ini dengan melakukan perhitungan dalam mencari pohon merentang minimum dengan menggunakan Algoritma Kruskal. Berikut ini langkah-langkah yang dilakukan untuk mencari pohon merentang minimum menggunakan Algoritma Kruskal:

1. Urutkan sisi graf pada tabel 4.1 dari bobot yang terkecil hingga yang terbesar, dapat dilihat pada tabel berikut:

**Tabel 4.2** Urutan sisi graf dari bobot terkecil hingga terbesar

No.	Sisi	Panjang (m)
1	$V_{16}-V_{17}$	55
2	$V_{10}-V_{16}$	56
3	$V_{10}-V_{18}$	60
4	$V_{15}-V_{16}$	61
5	$V_{18}-V_{19}$	70
6	$V_6-V_{10}$	71
7	$V_{14}-V_{15}$	71
8	$V_{19}-V_{23}$	72

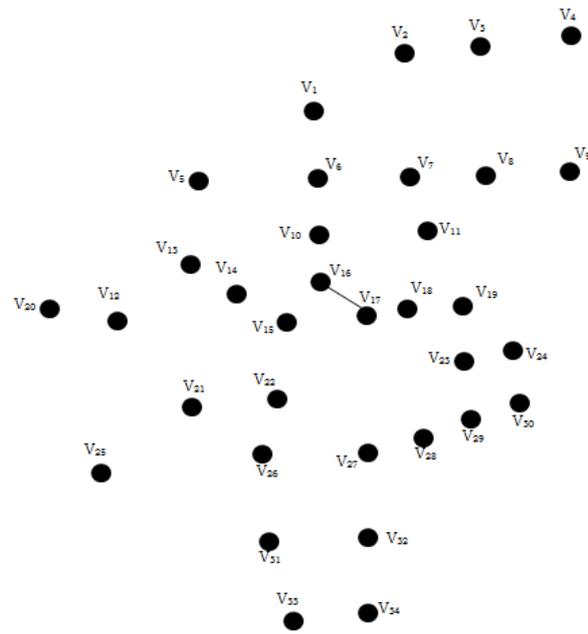
9	$V_{23}-V_{29}$	72
10	$V_{28}-V_{29}$	72
11	$V_{27}-V_{28}$	75
12	$V_{23}-V_{24}$	78
13	$V_1-V_6$	80
14	$V_{26}-V_{27}$	81
15	$V_{13}-V_{14}$	100
16	$V_{29}-V_{30}$	100
17	$V_{27}-V_{32}$	105
18	$V_{15}-V_{22}$	105
19	$V_{21}-V_{22}$	118
20	$V_{32}-V_{34}$	125
21	$V_{33}-V_{34}$	128
22	$V_{12}-V_{13}$	129
23	$V_{31}-V_{32}$	132
24	$V_{21}-V_{25}$	135
25	$V_{10}-V_{11}$	136
26	$V_6-V_7$	152
27	$V_7-V_8$	158
28	$V_8-V_9$	160
29	$V_5-V_6$	176
30	$V_2-V_7$	176
31	$V_3-V_8$	188
32	$V_4-V_9$	193
33	$V_{17}-V_{27}$	193
34	$V_{20}-V_{21}$	240

2. Pilih sisi yang mempunyai panjang minimum tetapi tidak membentuk sirkuit di T. Tambahkan sisi ke dalam T.

3. Ulangi langkah 2 sampai pohon merentang minimum terbentuk.

1. Iterasi 1

Sisi  $V_{16}-V_{17}$  dengan panjang 55 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

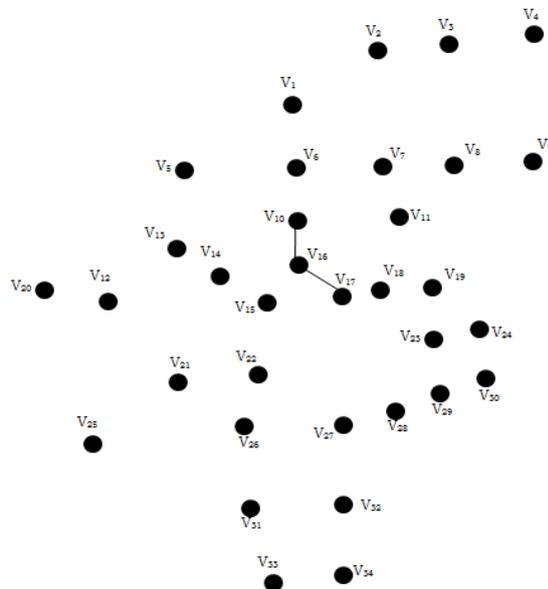


**Gambar 4.3** Iterasi 1 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

2. Iterasi 2

Sisi  $V_{10}$ - $V_{16}$  dengan panjang 56 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

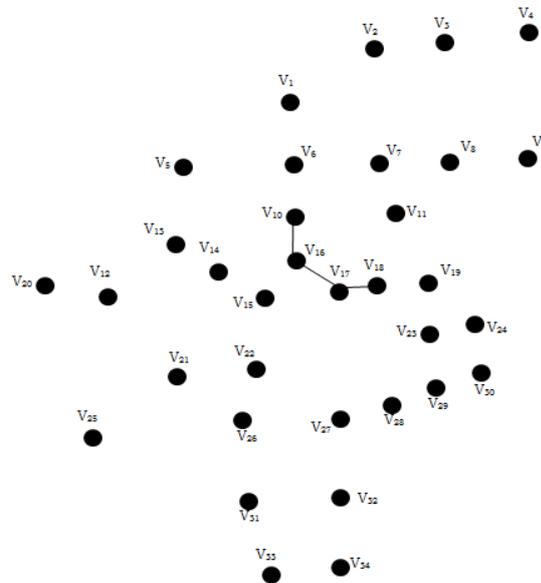


**Gambar 4.4** Iterasi 2 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

3. Iterasi 3

Sisi  $V_{17}$ - $V_{18}$  dengan panjang 60 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

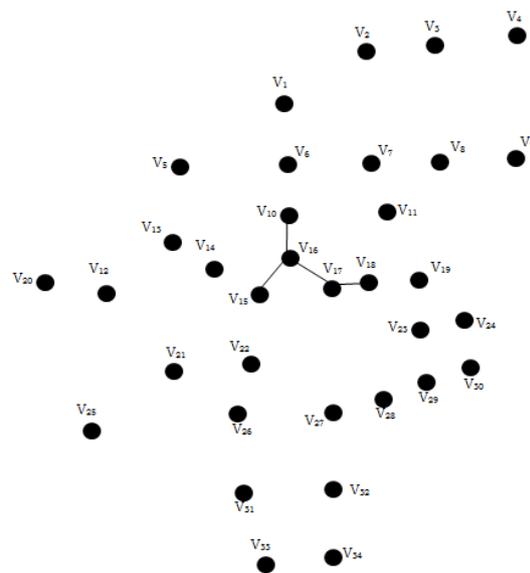


**Gambar 4.5** Iterasi 3 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

4. Iterasi 4

Sisi  $V_{15}$ - $V_{16}$  dengan panjang 61 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

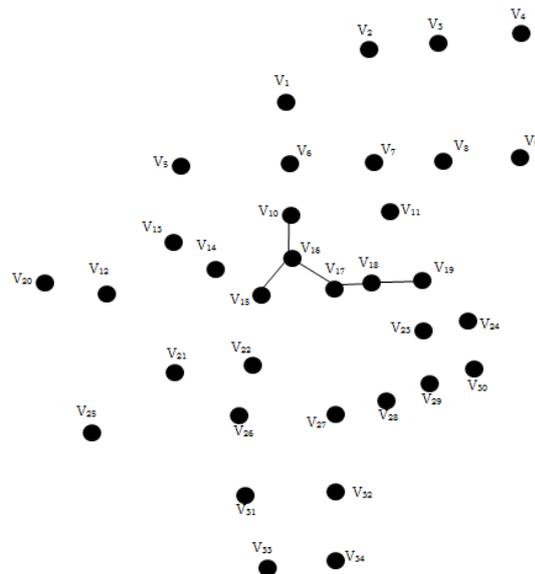


**Gambar 4.6** Iterasi 4 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

5. Iterasi 5

Sisi  $V_{18}$ - $V_{19}$  dengan panjang 70 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

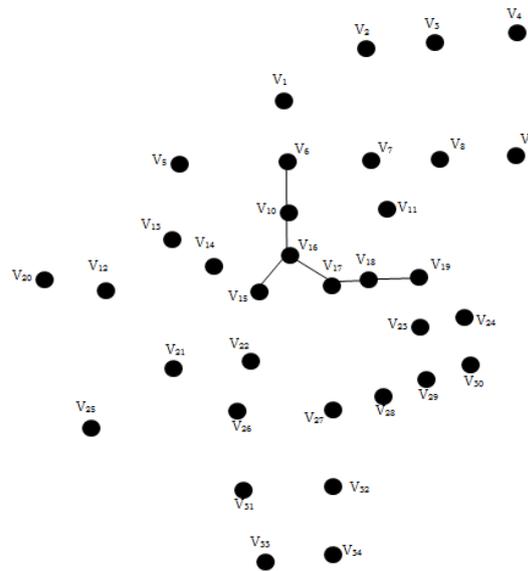


**Gambar 4.7** Iterasi 5 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 6. Iterasi 6

Sisi  $V_6$ - $V_{10}$  dengan panjang 71 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

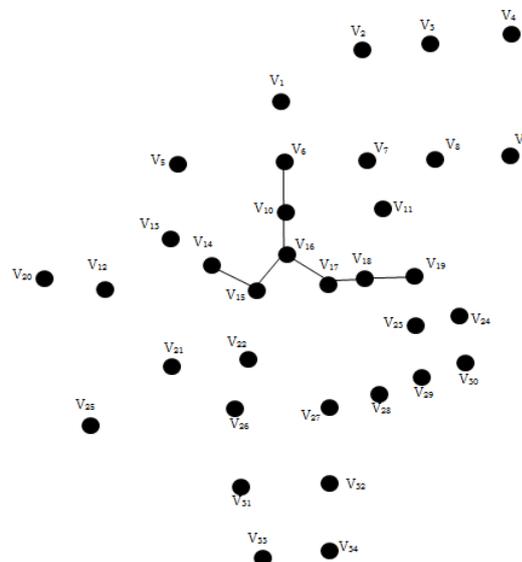


**Gambar 4.8** Iterasi 6 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 7. Iterasi 7

Sisi  $V_{14}$ - $V_{15}$  dengan panjang 71 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

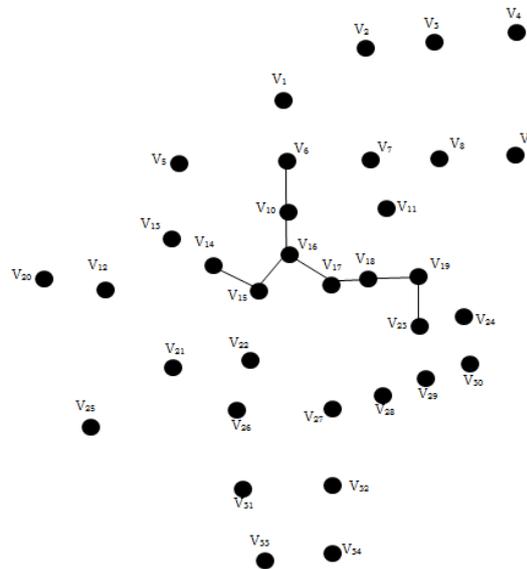


**Gambar 4.9** Iterasi 7 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

8. Iterasi 8

Sisi  $V_{19}$ - $V_{23}$  dengan panjang 72 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

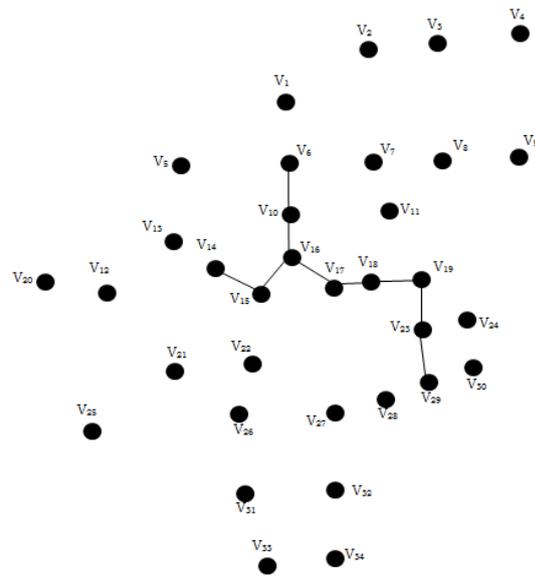


**Gambar 4.10** Iterasi 8 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

9. Iterasi 9

Sisi  $V_{23}$ - $V_{29}$  dengan panjang 72 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

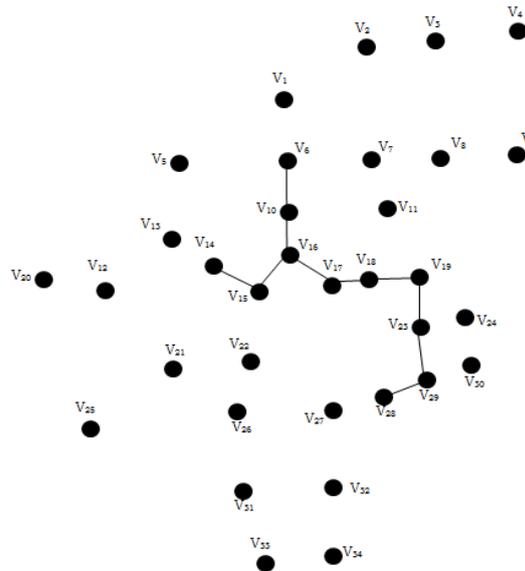


**Gambar 4.11** Iterasi 9 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

#### 10. Iterasi 10

Sisi  $V_{28}$ - $V_{29}$  dengan panjang 72 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

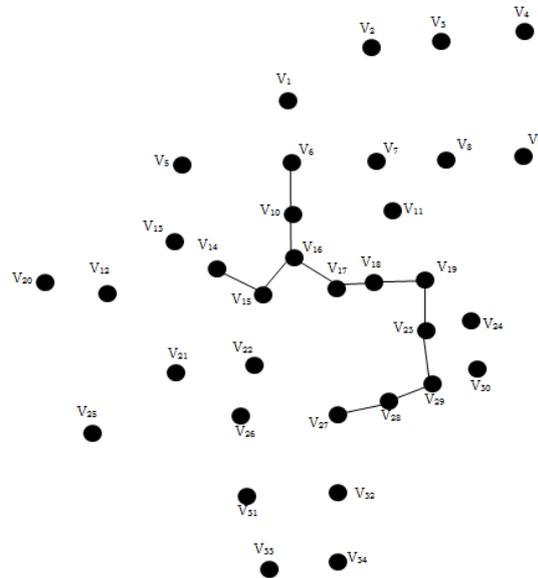


**Gambar 4.12** Iterasi 10 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 11. Iterasi 11

Sisi  $V_{27}$ - $V_{28}$  dengan panjang 75 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

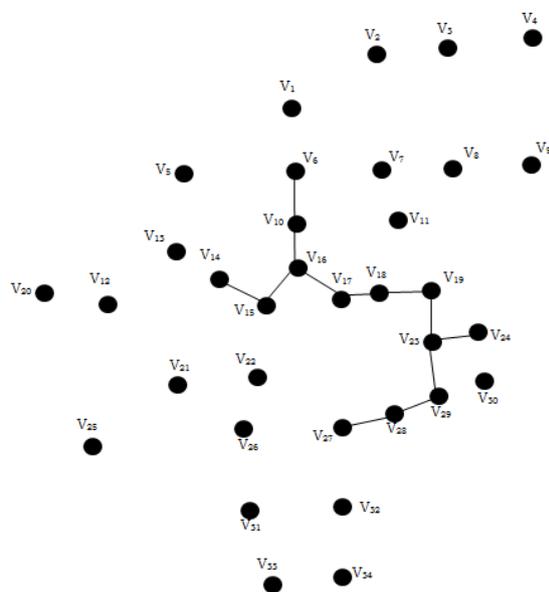


**Gambar 4.13** Iterasi 11 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 12. Iterasi 12

Sisi  $V_{23}$ - $V_{24}$  dengan panjang 78 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

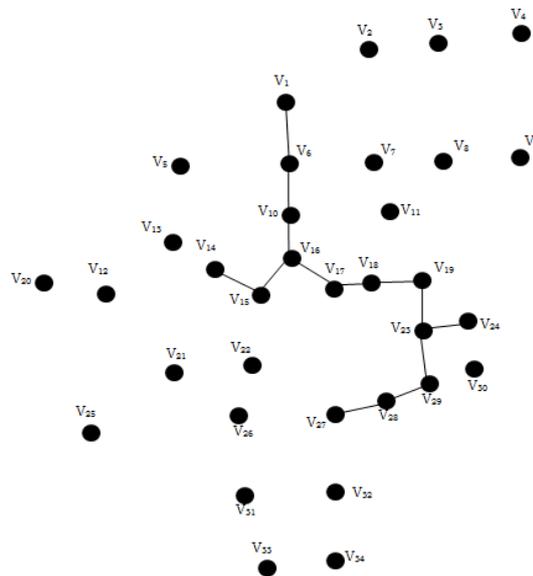


**Gambar 4.14** Iterasi 12 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

13. Iterasi 13

Sisi  $V_1$ - $V_6$  dengan panjang 80 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

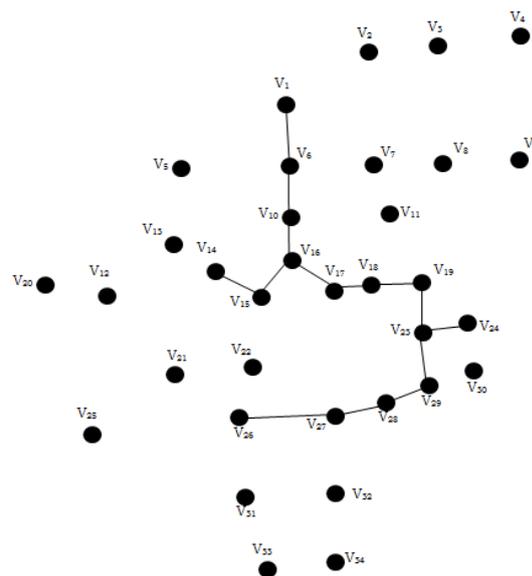


**Gambar 4.15** Iterasi 13 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

14. Iterasi 14

Sisi  $V_{26}$ - $V_{27}$  dengan panjang 81 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

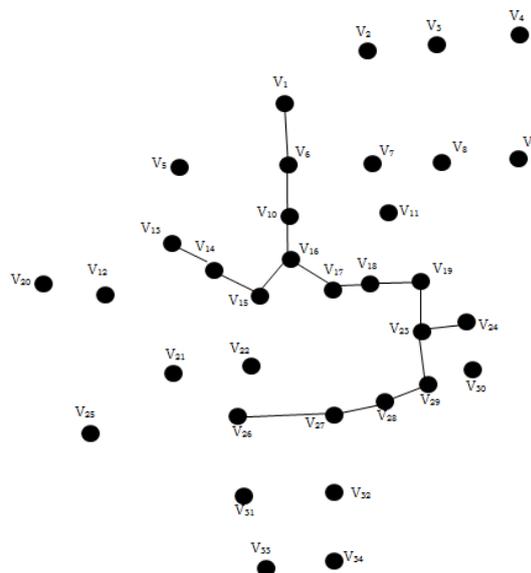


**Gambar 4.16** Iterasi 14 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

#### 15. Iterasi 15

Sisi  $V_{13}$ - $V_{14}$  dengan panjang 100 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

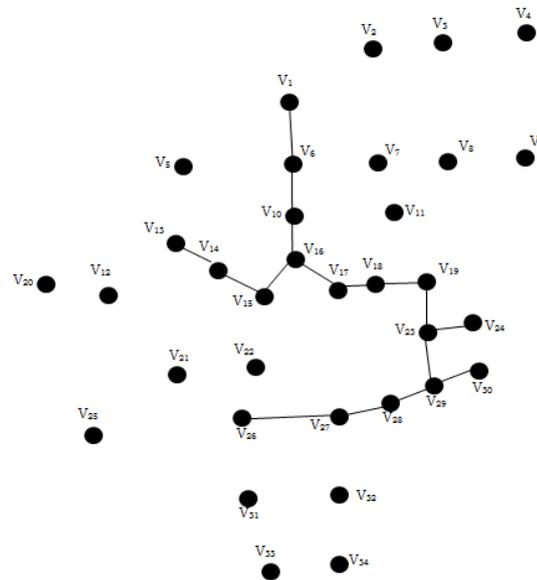


**Gambar 4.17** Iterasi 15 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 16. Iterasi 16

Sisi  $V_{29}$ - $V_{30}$  dengan panjang 100 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

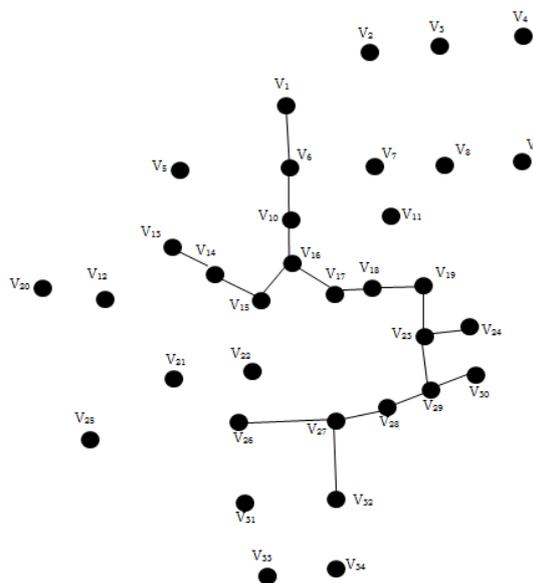


**Gambar 4.18** Iterasi 16 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 17. Iterasi 17

Sisi  $V_{27}$ - $V_{32}$  dengan panjang 105 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

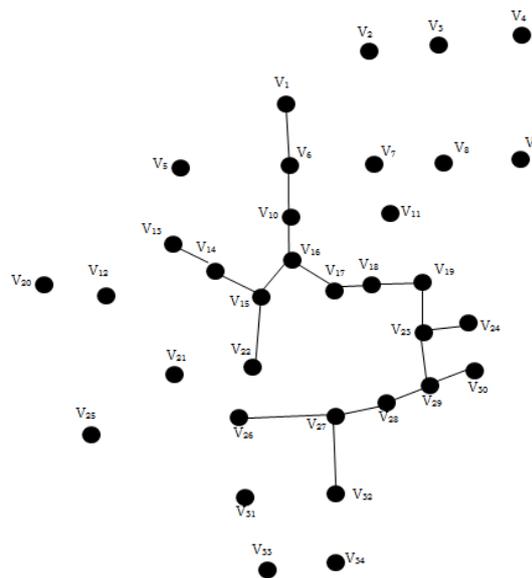


**Gambar 4.19** Iterasi 17 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

18. Iterasi 18

Sisi  $V_{15}$ - $V_{22}$  dengan panjang 105 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

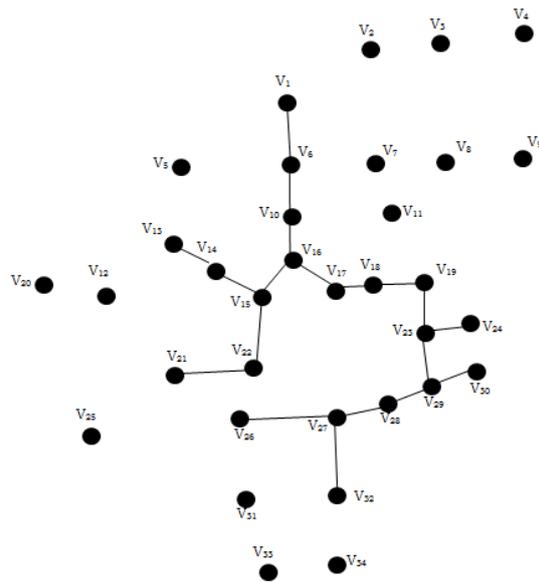


**Gambar 4.20** Iterasi 18 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

19. Iterasi 19

Sisi  $V_{21}$ - $V_{22}$  dengan panjang 118 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

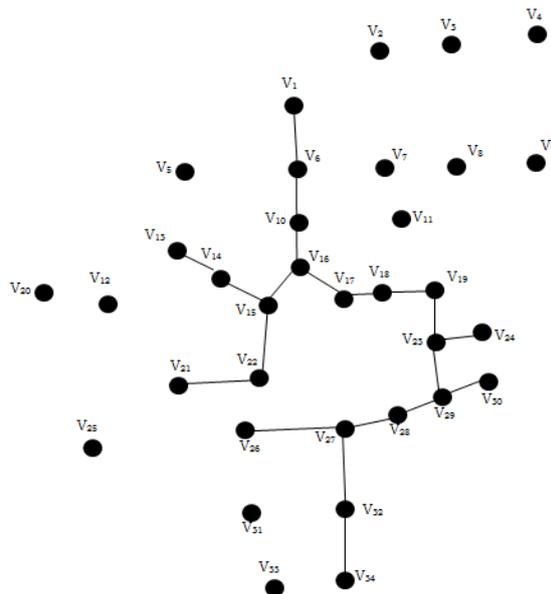


**Gambar 4.21** Iterasi 19 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

20. Iterasi 20

Sisi  $V_{32}-V_{34}$  dengan panjang 125 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

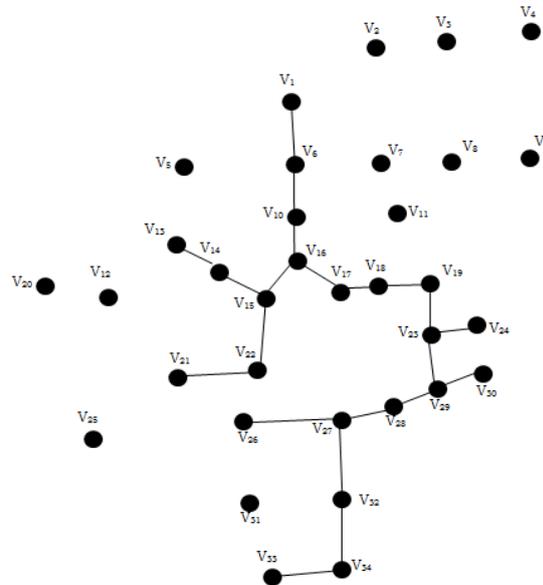


**Gambar 4.22** Iterasi 20 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 21. Iterasi 21

Sisi  $V_{33}$ - $V_{34}$  dengan panjang 128 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

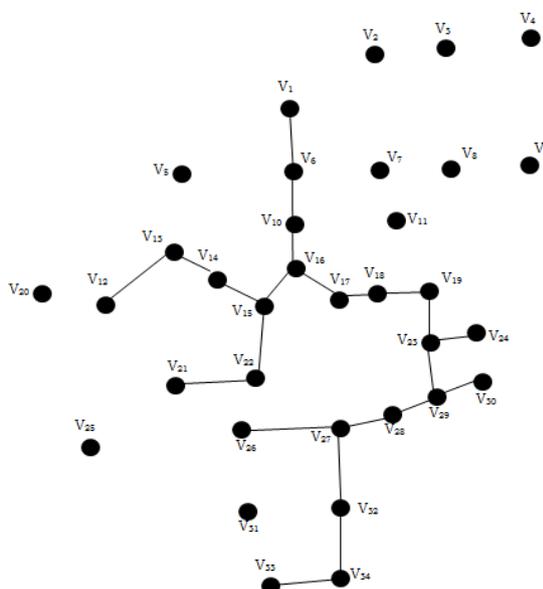


**Gambar 4.23** Iterasi 21 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 22. Iterasi 22

Sisi  $V_{12}$ - $V_{13}$  dengan panjang 129 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

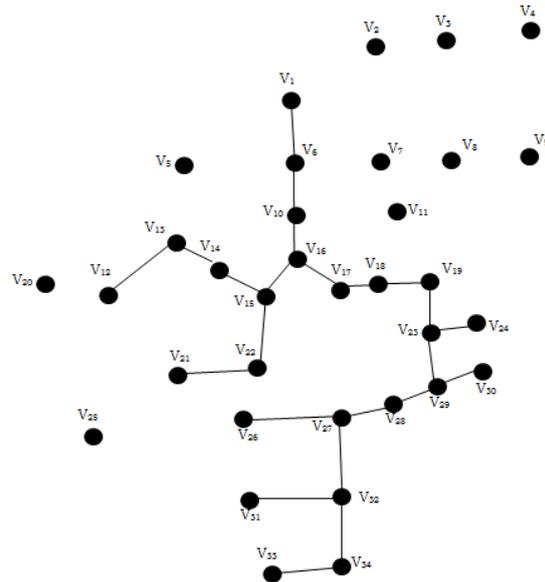


**Gambar 4.24** Iterasi 22 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

23. Iterasi 23

Sisi  $V_{31}$ - $V_{32}$  dengan panjang 132 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

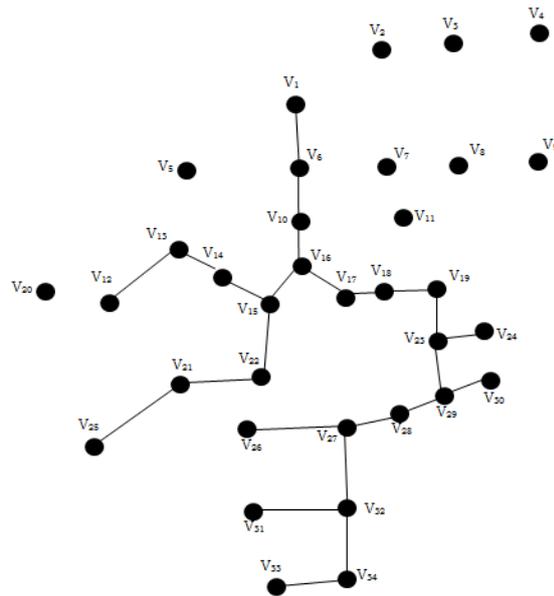


**Gambar 4.25** Iterasi 23 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

24. Iterasi 24

Sisi  $V_{21}$ - $V_{25}$  dengan panjang 135 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

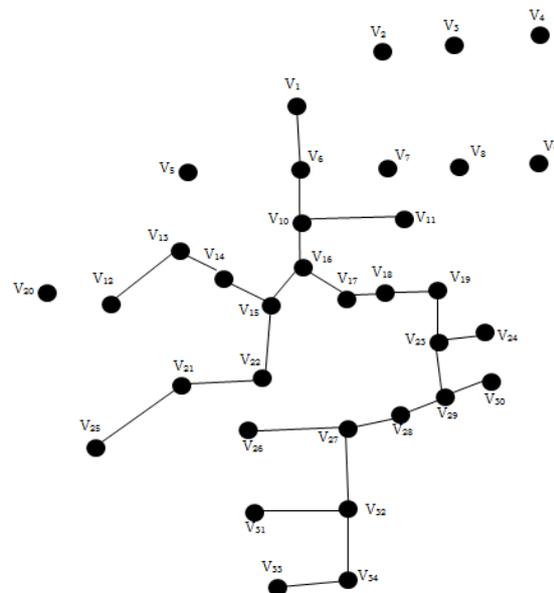


**Gambar 4.26** Iterasi 24 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

25. Iterasi 25

Sisi  $V_{10}$ - $V_{11}$  dengan panjang 136 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

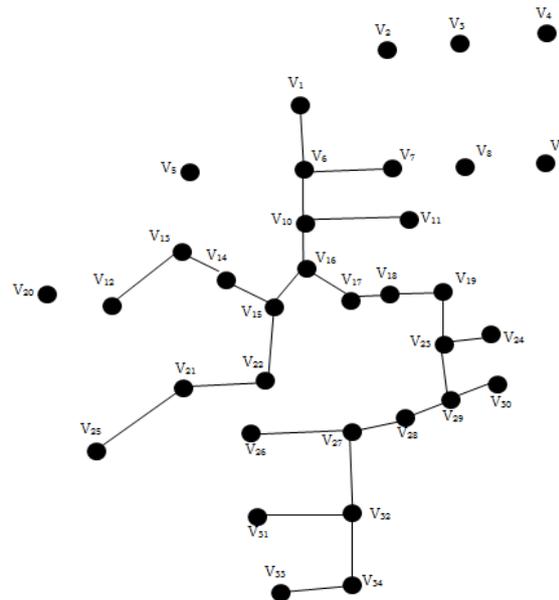


**Gambar 4.27** Iterasi 25 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 26. Iterasi 26

Sisi  $V_6-V_7$  dengan panjang 152 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

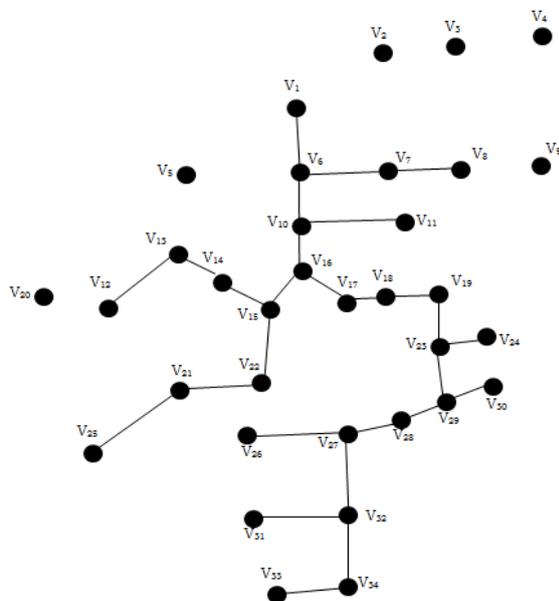


**Gambar 4.28** Iterasi 26 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 27. Iterasi 27

Sisi  $V_7-V_8$  dengan panjang 158 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

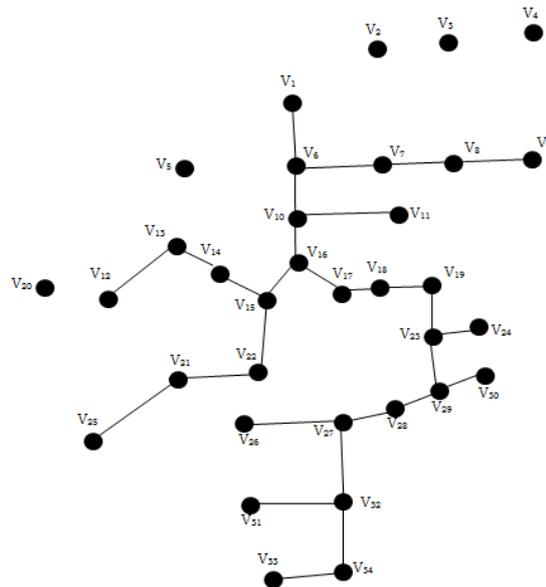


**Gambar 4.29** Iterasi 27 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

28. Iterasi 28

Sisi  $V_8$ - $V_9$  dengan panjang 160 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

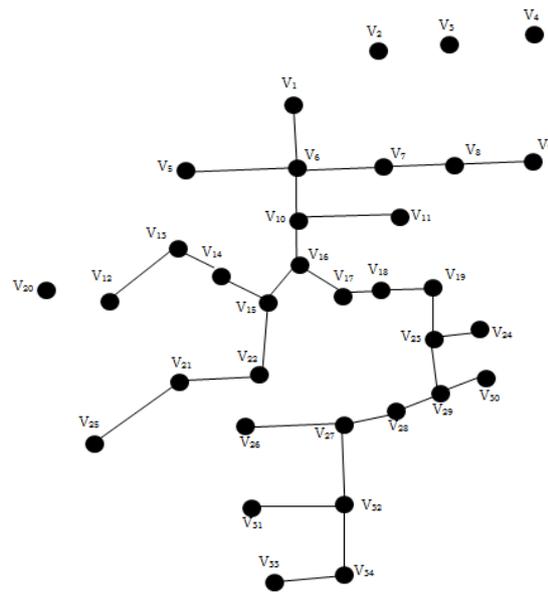


**Gambar 4.30** Iterasi 28 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

29. Iterasi 29

Sisi  $V_5$ - $V_6$  dengan panjang 176 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

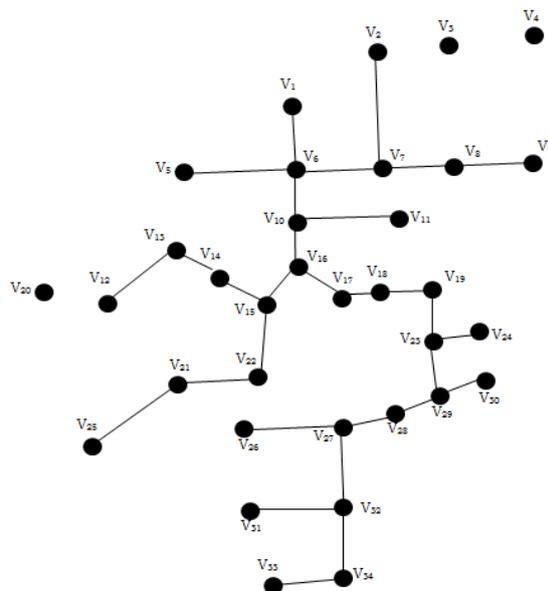


**Gambar 4.31** Iterasi 29 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

30. Iterasi 30

Sisi  $V_2$ - $V_7$  dengan panjang 176 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.

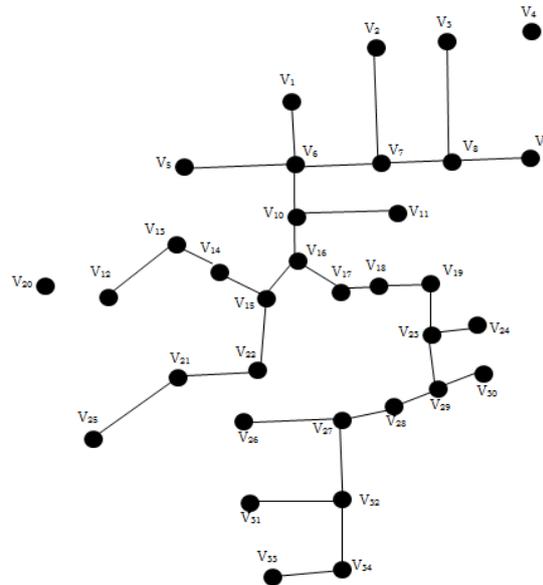


**Gambar 4.32** Iterasi 30 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 31. Iterasi 31

Sisi  $V_3$ - $V_8$  dengan panjang 188 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

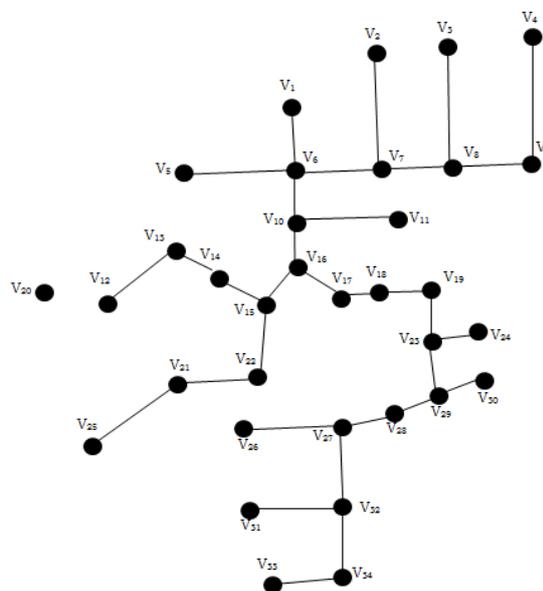


**Gambar 4.33** Iterasi 31 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

## 32. Iterasi 32

Sisi  $V_4$ - $V_9$  dengan panjang 193 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf  $T$ .

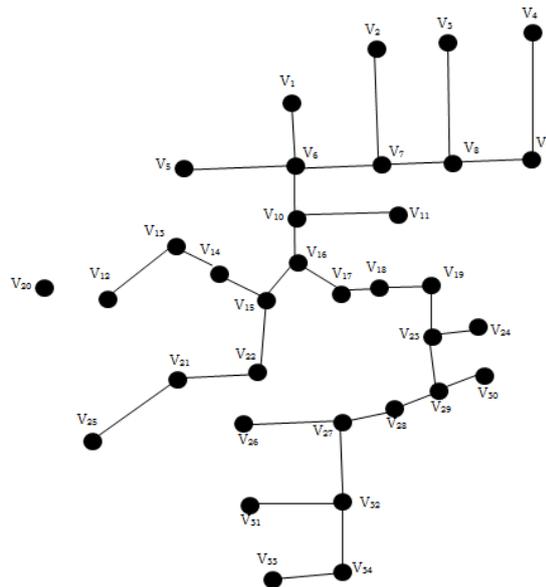


**Gambar 4.34** Iterasi 32 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

33. Iterasi 33

Sisi  $V_{17}$ - $V_{27}$  dengan panjang 193 tidak dipilih. Karena sisi tersebut membentuk sirkuit, maka tidak ditambahkan ke dalam graf T.

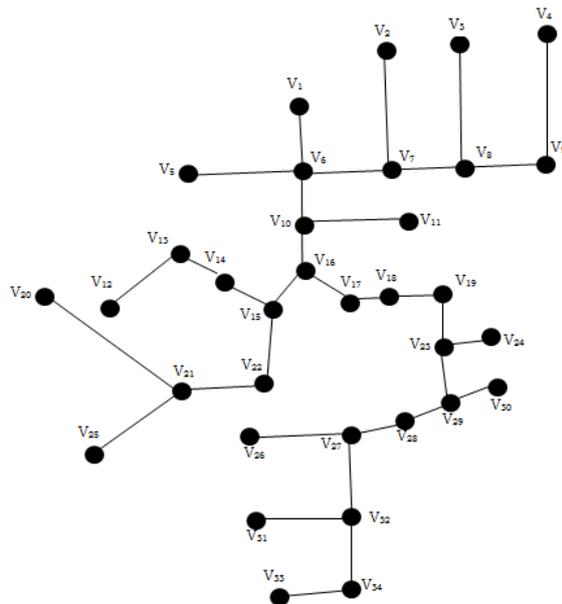


**Gambar 4.35** Iterasi 33 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Lanjutkan iterasi hingga pohon merentang minimum terbentuk.

34. Iterasi 34

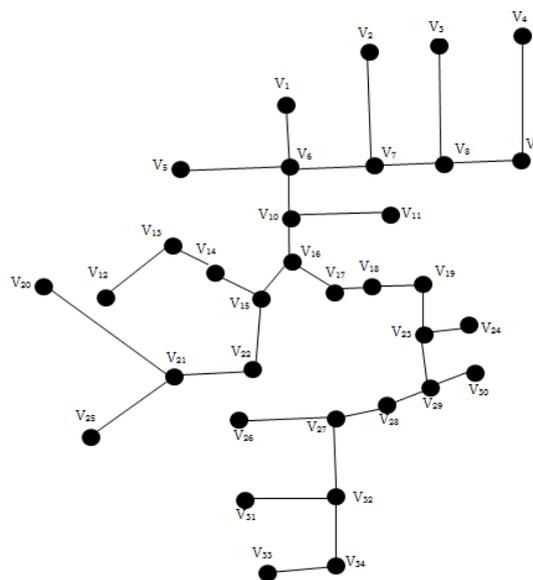
Sisi  $V_{20}$ - $V_{21}$  dengan panjang 240 dipilih. Karena sisi tersebut tidak membentuk sirkuit, maka ditambahkan ke dalam graf T.



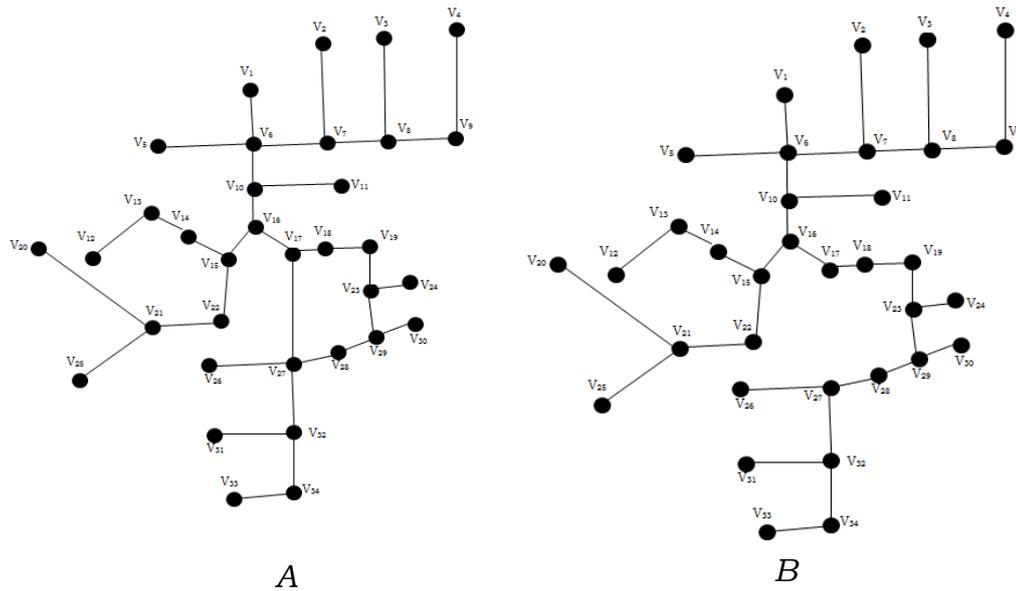
**Gambar 4.36** Iterasi 34 Pencarian Pohon Merentang Minimum dengan Algoritma Kruskal

Iterasi tidak dilanjutkan karena pohon merentang minimum telah terbentuk.

**Gambar 4.36** merupakan pohon merentang minimum yang terbentuk menggunakan Algoritma Kruskal. Secara keseluruhan graf  $T$  tersebut memiliki 34 titik dengan jumlah sisi sebanyak 34 dan total panjang 3921 meter.



**Gambar 4.37** Jaringan pipa primer Perumahan Aurduri dalam bentuk graf setelah dicari Pohon Merentang Minimum



**Gambar 4.38** Perbedaan jaringan pipa primer Perumahan Aurduri sebelum dan setelah dicari Pohon Merentang Minimumnya

A adalah Jaringan pipa primer Perumahan Aurduri dalam bentuk graf sebelum dicari Pohon Merentang Minimum sedangkan B setelah dicari.

Setelah menerapkan Algoritma Kruskal diperoleh banyaknya sisi pohon merentang minimum adalah 33 sisi dengan jumlah iterasi sebanyak 34 iterasi. Perbedaan jaringan pipa primer perumahan Aurduri sebelum dan setelah yaitu ada satu sisi yang dihapus. Sisi yang dihapus adalah sisi ( $V_{17}$ - $V_{27}$ ) sepanjang 193 meter. Maka diperoleh panjang pipa primer menggunakan Algoritma Kruskal adalah sepanjang 3728 meter. Sedangkan panjang jaringan pipa primer PDAM yang dihitung sebelum mencari pohon merentang minimum pada Perumahan Aurduri adalah sepanjang 3921 meter. Hasil perhitungan secara manual total panjang jaringan pipa primer PDAM menggunakan Algoritma Kruskal lebih minimum. Dibandingkan dengan total jaringan pipa primer PDAM yang terpasang di Perumahan Aurduri tersebut, yang berselisih 193 meter. Panjang pipa yang bisa dioptimalkan hingga 193 meter apabila diuangkan dengan merujuk pada pipa PVC-O diameter 16” per meter seharga Rp.1.400.000 maka efisiensi dana setelah dianalisis bisa dihemat hingga Rp.270.200.000.

## **V. KESIMPULAN DAN SARAN**

### **5.1 Kesimpulan**

Berdasarkan hasil dan pembahasan pada penelitian ini bahwa Algoritma Kruskal dapat digunakan dalam pencarian Pohon Merentang Minimum pada jaringan pipa primer PDAM di Perumahan Aurduri. Dengan menggunakan data yang ada, diperoleh banyaknya sisi pohon merentang minimum adalah 33 sisi dengan jumlah iterasi sebanyak 34 iterasi. Sisi yang awalnya berjumlah 34 dihapus satu sisi, yaitu sisi ( $V_{17}-V_{27}$ ) sepanjang 193 meter. Maka diperoleh panjang pipa primer menggunakan Algoritma Kruskal adalah sepanjang 3728 meter. Sedangkan panjang jaringan pipa primer PDAM yang dihitung pada Perumahan Aurduri adalah sepanjang 3921 meter. Hasil perhitungan secara manual total panjang jaringan pipa primer PDAM menggunakan Algoritma Kruskal lebih minimum dibandingkan dengan total jaringan pipa primer PDAM yang terpasang di Perumahan Aurduri tersebut, hingga berselisih sebesar 193 meter. Panjang pipa yang bisa dioptimalkan hingga 193 meter apabila diuangkan dengan merujuk pada pipa PVC-O diameter 16" per meter seharga Rp.1.400.000 maka efisiensi dana bisa dihemat hingga Rp.270.200.000

### **5.2 Saran**

Penulis menyarankan untuk melengkapi penelitian ini dengan menambahkan variabel seperti pemakaian pipa dan harga pipa dalam meter. Lalu dapat menambahkan sisi selain pipa primer, seperti menambah sisi pipa sekunder dan pipa tersier. Dikarenakan penelitian ini dihitung secara manual dapat dikembangkan dengan menggunakan aplikasi atau metode lainnya. Kemudian diharapkan bisa berkoordinasi dengan instansi yang bersangkutan agar terjadi kesinambungan dalam berbagai bidang ilmu dan dapat penerapannya langsung.

## DAFTAR PUSTAKA

- Abrori, M dan N. Ubaidillah. 2014. Pengujian Optimisasi Jaringan Kabel Fiber Optic Di Universitas Islam Indonesia Menggunakan *Minimum Spanning Tree* . *Jurnal Fourier*. Vol. 3 (1) : 49 – 58
- Anggraeni, W. 2015. Aplikasi Algoritma Sollin dalam Pencarian Pohon Perentang Minimum Provinsi Jawa Tengah. *Jurnal Faktor Exacta*. Vol 8 (4) : 381 – 391
- Faisol dan Mahluroh. 2015. Aplikasi Algoritma Prim untuk Mencari Optimisasi Jaringan Listrik Di Kabupaten Sampang. *Jurnal SEHATI*. Vol 1 : 537 – 540
- Dokumen RPI2-JM Kota Jambi Tahun 2016-2020
- Hayu, Wisra, Yuliani, dan Marwan Sam. 2017. Pembentukan Pohon Merentang Minimum Dengan Algoritma Kruskal. *Jurnal Scientific Pinisi*. Volume 3, Nomor 2: 108-115
- <http://sipkp.ciptakarya.pu.go.id> (Diakses pada Oktober 2020).
- <https://www.pu.go.id> (Diakses pada Oktober 2020).
- Ismail T, dan T. Setiadi. 2014. Media Pembelajaran Strategi Algoritma Pada Pokok Bahasan Pohon Merentang Minimum Dan Pencarian Lintasan Terpendek. *Jurnal Sarjana Teknik Informatika*. Vol 2 (2) : 1423 – 1430
- Joko, T. 2010. *Unit Air Baku dalam Sistem Penyediaan Air Minum*. Yogyakarta: Graha Ilmu
- Khanif, M dan Uszhani, U. 2010. *Perancangan Sistem Penyediaan Air Bersih Komplek Perkantoran Kabupaten Bandung Barat*. Bandung. Politeknik Negeri Bandung.
- Latifah, U. 2014. *Penerapan Algoritma Prim dan Kruskal pada Jaringan Distribusi Air PDAM Tirta Moedal Cabang Semarang Utara*. Universitas Negeri Semarang, Indonesia. Tersedia di <http://journal.unnes.ac.id/sju/index.php/ujm>.
- Monifani, E.M.Y, A. Fanggidae, dan T. Widiastuti. 2014. Penyelesaian *Minimum Spanning Tree* (MST) Pada Graf Lengkap Dengan Algoritma Genetika Menggunakan Teknik Prufer Sequenes. *Jurnal Komputer dan Informatika*. Vol 2 (2) : 84 – 91

- Mualimah, A dan Aris Fanani. 2020. Penggunaan Algoritma Kruskal Dalam Jaringan Pipa Pendistribusian Air Perusahaan Daerah Air Minum (PDAM) Tirta Dharma Lamongan. *Jurnal Mahasiswa Matematika ALGEBRA*. Vol 1 (1) : 150-156
- Munir, R. 2005. *Matematika Diskrit*. Bandung: Informatika.
- Munir, R. 2016. *Matematika Diskrit*. Bandung : Informatika.
- Nugraha, D.W. 2011. Aplikasi Algoritma Prim Untuk Menentukan *Minimum Spanning Tree* Suatu Graf Berbobot Dengan Menggunakan Pemrograman Berorientasi Objek. *Jurnal Ilmiah Foristek*. Vol.1 (2): 70 – 79
- Prasetyo, V.Z. 2013. Penerapan Algoritma Dijkstra Untuk Perutean Adaptif Pada Jaringan Pendistribusian Air PDAM di Kabupaten Demak. *Skripsi*. Tidak Diterbitkan. Fakultas Matematika dan Ilmu Pengetahuan Alam. Universitas Negeri Semarang: Semarang.
- Pratama A.P, DKK. 2013. Penggunaan Algoritma Kruskal Dalam Jaringan Pipa Air Minum Kecamatan Nganjuk Kabupaten Nganjuk. *Jurnal Sains dan Seni Pomits*. Vol 1 (1) : 1 – 6
- Ramadhan, A.F. 2017. Aplikasi Algoritma Prim Dalam Penentuan Pohon Merentang Minimum Untuk Jaringan Pipa PDAM Kota Tangerang. *Jurnal Ilmiah*. Vol 2 (1): 30 – 38
- Utami E, dan Sukrisno. 2005. *10 Langkah Belajar Logika dan Algoritma, Menggunakan Bahasa C dan C++ di GNU/Linux*. Yogyakarta: Andi
- Wamiliana, D. Kurniawan, dan C. Shavitri N.F. 2014. Perbandingan Kompleksitas Algoritma Prim, Algoritma Kruskal dan Algoritma Sollin untuk Menyelesaikan Masalah *Minimum Spanning Tree* . *Jurnal Komputasi*. Vol 2 (1) : 61 – 67
- Wattimena, Z. A dan Lawatama, S. 2013. *Aplikasi Algoritma Kruskal dalam Pengoptimalan Panjang Pipa*. FMIPA UNPATI. Poka-Ambon Maluku
- Wayangkau, I.H. 2015. Optimisasi Particle Swarm pada Pemasangan Jaringan Pipa Air PDAM. *Jurnal Ilmiah Mustek Anim Ha*. Vol 4 (1) : 1 - 6