

***MINIMUM SPANNING TREE* PADA RANCANGAN PEMBANGUNAN  
JALUR KORIDOR BERATAP DI UNIVERSITAS LAMPUNG**

**(Skripsi)**

**Oleh**

**Atma Rahmawati**



**JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG**

**2024**

## **ABSTRACT**

### **Minimum Spanning Tree in the Design of Roofed Corridor Path Development at the University of Lampung**

**By**

**Atma Rahmawati**

The movement of Lampung University students from one building to another in any weather to be able to participate in all existing activities, both academic and non-academic, will be very easy if there is a road or covered corridor that connects the buildings. This problem can be viewed as a minimum spanning tree problem. The aim of this research is to determine the minimum spanning tree (MST) or shortest path using the Prim Algorithm in order to minimize the construction of covered corridors between buildings at the University of Lampung. In this research, POM-QM software was used to avoid human error in the search process. From the results of searching for the minimum spanning tree (MST) by applying the Prim Algorithm, both manually and using POM-QM Software, the optimal result for the distance between buildings at the University of Lampung is 6,889 meters, more than a minimum of 56% of the initial distance with 51 points. and 50 sides, while the initial data amount was 51 points and 76 sides with a total distance of 15,669 meters.

**Keywords:** spanning tree, minimum spanning tree, shortest path, Prim Algorithm, POM-QM software

## ABSTRAK

### ***Minimum Spanning Tree* Pada Rancangan Pembangunan Jalur Koridor Beratap di Universitas Lampung**

Oleh

**Atma Rahmawati**

Pergerakan mahasiswa Universitas Lampung dari satu gedung ke gedung lain dalam cuaca apapun untuk dapat mengikuti seluruh kegiatan yang ada baik itu akademik maupun nonakademik akan sangat mudah apabila terdapat jalan atau koridor beratap yang menghubungkan antar gedung. Masalah ini dapat dipandang sebagai masalah pohon merentang minimum. Tujuan dari dilakukannya penelitian ini adalah menentukan *minimum spanning tree* (MST) atau jalur terpendek dengan menggunakan Algoritma Prim agar dapat meminimumkan pembangunan jalur koridor beratap antar gedung di Universitas Lampung. Pada penelitian ini, digunakan *software* POM-QM untuk menghindari terjadinya *human error* atau kesalahan oleh manusia dalam proses pencarian. Dari hasil pencarian *minimum spanning tree* (MST) dengan menerapkan Algoritma Prim, baik secara manual dan menggunakan *software* POM-QM, maka diperoleh hasil optimal jarak antar gedung yang ada di Universitas Lampung adalah 6.889 meter lebih minimum sebesar 56 % dari jarak awal dengan 51 titik dan 50 sisi sedangkan jumlah data awal sebanyak 51 titik dan 76 sisi dengan total jarak 15.669 meter.

Kata kunci: pohon rentang, *minimum spanning tree*, jalur terpendek, Algoritma Prim, *software* POM-QM

***MINIMUM SPANNING TREE* PADA RANCANGAN PEMBANGUNAN  
JALUR KORIDOR BERATAP DI UNIVERSITAS LAMPUNG**

**Atma Rahmawati**

**Skripsi**

Sebagai Salah Satu Syarat Untuk Memperoleh Gelar  
SARJANA MATEMATIKA

Pada

Jurusan Matematika  
Fakultas Matematika dan Ilmu Pengetahuan Alam



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2024**

Judul : **MINIMUM SPANNING TREE PADA RANCANGAN PEMBANGUNAN JALUR KORIDOR BERATAP DI UNIVERSITAS LAMPUNG**

Nama Mahasiswa : **Atma Rahmawati**

NPM : 1817031085

Jurusan : Matematika

Fakultas : Matematika dan Ilmu Pengetahuan Alam



**MENYETUJUI**

1. **Komisi Pembimbing**

**Dr. Aang Nuryaman, S. Si., M.Si.**  
NIP. 197403162005011001

**Dina Eka Nurvazly, M.Si**  
NIP. 199311062019032018

2. **Ketua Jurusan Matematika**

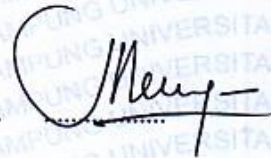
**Dr. Aang Nuryaman, S. Si., M.Si.**  
NIP. 197403162005011001



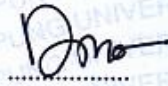
**MENGESAHKAN**

**1. Tim Penguji**

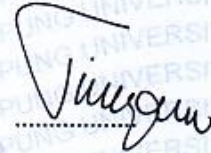
**Ketua : Dr. Aang Nuryaman, S. Si., M.Si.**



**Sekretaris : Dina Eka Nurvazly, M.Si.**



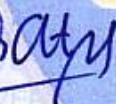
**Penguji  
Bukan Pembimbing : Drs. Tiryono Ruby, M.Sc., Ph.D.**



**2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam**



**Dr. Eng. Heri Satria, S.Si., M.Si.**  
**NIP. 197110012005011002**



**Tanggal Lulus Ujian Skripsi : 19 Desember 2024**

## PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini :

Nama : Atma Rahmawati

Nomor Pokok Mahasiswa : 1817031085

Jurusan : Matematika

Judul Skripsi : *Minimum Spanning Tree* Pada Rancangan  
Pembangunan Jalur Koridor Beratap di Universitas  
Lampung

Dengan ini menyatakan bahwa skripsi ini adalah hasil pekerjaan saya sendiri. Apabila kemudian hari terbukti bahwa skripsi ini merupakan hasil salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, Desember 2024

Yang Menyatakan,



Atma Rahmawati  
NPM. 1817031085

## **RIWAYAT HIDUP**

Penulis bernama lengkap Atma Rahmawati lahir di Rajabasa Lama 10 November 2000. Penulis merupakan anak pertama dari 4 bersaudara pasangan Bapak Nuryadianto dan Ibu Mutini.

Penulis menempuh pendidikan di TK PERTIWI 2 Lampung Timur pada tahun 2005 sampai dengan 2006. Kemudian melanjutkan sekolah dasar di SD Negeri 2 Rajabasa Lama pada tahun 2006 sampai dengan 2012. Kemudian melanjutkan pendidikan di SMP N 1 Labuhan Ratu pada tahun 2012 sampai dengan 2015. Kemudian menempuh pendidikan SMA di SMA N 1 Way Jepara pada tahun 2015 sampai dengan 2018.

Pada tahun 2018, melalui jalur SBMPTN penulis melanjutkan pendidikan di perguruan tinggi dan terdaftar sebagai mahasiswa S1 di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung. Selama menjadi mahasiswa penulis aktif dalam organisasi kemahasiswaan tingkat jurusan seperti Anggota Gematika 2018/2019, dan tingkat fakultas yaitu anggota biro kesekretariatan pada tahun 2019 yang kemudian menjadi sekretaris biro kesekretariatan pada tahun 2020. Pada tahun 2021 penulis aktif pada organisasi Dewan Perwakilan Mahasiswa (DPM) yang ada di Fakultas FMIPA sebagai sekretaris komisi 1 (Legislasi).



Sebagai penerapan ilmu yang didapat, pada awal semester VI penulis melaksanakan Kerja Praktik (KP) di Badan Perencanaan Pembangunan Daerah (BAPPEDA) Kota Metro Provinsi Lampung pada tanggal 01 Februari 2021 – 02 Maret 2021. Kemudian, pada semester VII penulis melaksanakan Kuliah Kerja Nyata di Desa Rajabasa Lama, Kecamatan Labuhan Ratu, Kabupaten Lampung Timur pada tanggal 1 Agustus 2021 sampai dengan 10 September 2021.

## **KATA INSPIRASI**

*“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya”*

**(Q.S Al-Baqarah : 286)**

*“ Karena sesungguhnya sesudah kesulitan itu ada kemudahan, sesungguhnya sesudah kesulitan itu ada kemudahan.”*

**(Q.S Al-Insyirah : 5-6)**

*“Raihlah ilmu! Dan untuk meraih ilmu, belajarlah untuk tenang dan sabar”*

**(Umar bin Khattab)**

*“Hidup itu seperti bersepeda. Kalau kamu ingin menjaga keseimbanganmu, maka kamu harus terus bergerak maju.”*

**(Albert Einstein)**

## **PERSEMBAHAN**

Alhamdulillahrabbi'l'amin

Dengan mengucapkan puji dan syukur yang tiada hentinya kepada Allah Subhanahu Wata'ala karna limpahan nikmat dan karunia-Nya sehingga skripsi ini dapat diselesaikan. Taklupa shalawat beserta salam selalu tercurah kepada junjungan kita Nabi Muhammad Shallallahu'Alaihi Wasallam.

Sengan penuh syukur dan ketulusan, kupersembahkan karya kecil dan sederhana ini sebagai suatu tanda cinta kepada semua orang yang senantiasa mendukung, mendampingi, dan mendoakan kelancaran terciptanya karya ini.

Mamah, Papah, dan adik-adik yang telah memberikan banyak masukan dan pengarahan serta menjadi motivasi terbesar selama ini.

Dosen pembimbing dan penguji yang senantiasa membimbing penulis.

Dosen Jurusan Matematika FMIPAUnila yang telah memberikan banyak ilmu.

Untuk siapapun yang selalu bertanya "kapan skripsimu selesai?"

Terlambat lulus atau lulus tidak tepat waktu bukan sebuah kejahatan, bukan sebuah aib. Alangkah kerdilnya jika mengukur kepintaran seseorang hanya dari siapa yang paling cepat lulus. Bukankah sebaik-baiknya skripsi adalah yang selesai? Baik itu selesai tepat waktu maupun tidak tepat waktu

## SANWACANA

Puji dan syukur penulis panjatkan atas kehadiran Allah SWT yang senantiasa melimpahkan rahmat dan hidayah-Nya kepada penulis, sehingga penulis dapat menyelesaikan skripsi ini yang berjudul “ *Minimum Spanning Tree* Pada Rancangan Pembangunan Jalur Koridor Beratap di Universitas Lampung” dengan baik. Selama proses penulisan dan penyelesaian skripsi ini, penulis banyak memperoleh bimbingan dan bantuan dari berbagai pihak. oleh karena itu penulis ingin mengucapkan terima kasih kepada:

1. Bapak Dr. Aang Nuryaman, S.Si., M.Si. selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung dan selaku Pembimbing I yang selalu bersedia menyempatkan waktu untuk memberikan arahan, bimbingan, saran, serta dukungan dalam berbagai hal kepada penulis sehingga dapat menyelesaikan skripsi ini.
2. Ibu Dina Eka Nurvazly M.Si., selaku Pembimbing II yang telah menyempatkan waktu untuk membimbing dan memberikan arahan serta dukungan kepada penulis.
3. Bapak Drs. Tiryono Rubi, M.Sc., Ph.D selaku Penguji yang telah bersedia memberikan kritik dan saran serta evaluasi kepada penulis sehingga dapat lebih baik lagi.
4. Ibu Dra. Dorrah Aziz, M. Si., selaku Pembimbing Akademik yang selalu bersedia memberikan bimbingan, saran serta dukungan kepada penulis terhadap hal yang berkaitan dengan Akademik. .
5. Bapak Ahmad Faisol, S.Si., M.Sc. selaku Sekretaris Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam UniversitasLampung



yang sudah membantu penulis sampai penulis dapat menyelesaikan studinya.

6. Seluruh dosen, staff dan karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
7. Orang tuaku Bapak Nuryadianto dan Ibu Mutini yang menemani dan mendukung seluruh perjalanan penulis dalam menempuh pendidikan sarjananya, yang selalu mendoakan dan berjuang bagi penulis.
8. Adikku Galih, Aura, dan Syafiqa yang sangat aku sayangi, serta semua saudaraku yang senantiasa mendukungku.
9. Teman terbaikku Rahma, Riris yang selalu memberi semangat.
10. Bernie teman seperjuangan sebagai mahasiswa akhir.
11. Badan Perencanaan Pembangunan Kota Metro yang telah memberikan ilmu dan pengalaman kerja kepada penulis.
12. Teman-teman mahasiswa Jurusan Matematika angkatan 2018.
13. Teman-teman dan pihak yang membantu dalam pengerjaan skripsi ini yang tidak dapat disebutkan satu-persatu dan juga kepada pihak-pihak yang mendoakan penulis.
14. Almamater tercinta Universitas Lampung.

Semoga seluruh bantuan dan dukungan yang telah diberikan kepada penulis mendapat balasan dari Allah SWT. Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, karena terbatasnya kemampuan, pengetahuan, dan pengalaman yang dimiliki. Oleh karena itu penulis mohon maaf dan sangat mengharapkan kritik, saran, dan masukan dari semua pihak demi kesempurnaan skripsi ini. Semoga skripsi ini dapat memberi manfaat bagi semua pihak, penyusun pada khususnya dan pembaca pada umumnya.

Bandar Lampung, Desember 2024

Penulis

Atma Rahmawati

## DAFTAR ISI

	Halaman
<b>DAFTAR ISI .....</b>	<b>xiv</b>
<b>DAFTAR TABEL .....</b>	<b>xvi</b>
<b>DAFTAR GAMBAR.....</b>	<b>xvii</b>
<b>I. PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah .....	4
1.3 Tujuan .....	4
1.4 Manfaat .....	4
<b>II. TINJAUAN PUSTAKA .....</b>	<b>5</b>
2.1 Definisi Graf .....	5
2.2 Terminologi Graf .....	6
2.3 Jenis-Jenis Graf.....	11
2.4 Pohon .....	14
2.5 Pohon Merentang ( <i>Spanning Tree</i> ).....	15
2.6 <i>Minimum Spanning Tree</i> (MST) .....	16
2.7 Algoritma Prim .....	17
2.8 <i>POM-QM</i> .....	18

<b>III. METODOLOGI PENELITIAN .....</b>	<b>22</b>
3.1 Waktu dan Tempat Penelitian.....	22
3.2 Sumber Data Penelitian .....	22
3.3 Metode Penelitian .....	23
<b>IV. HASIL DAN PEMBAHASAN.....</b>	<b>24</b>
4.1 Data Penelitian.....	24
4.2 Analisis dan Pembahasan.....	30
<b>V. KESIMPULAN DAN SARAN.....</b>	<b>40</b>
5.1 Kesimpulan .....	40
5.2 Saran .....	41
<b>DAFTAR PUSTAKA .....</b>	<b>42</b>
<b>LAMPIRAN.....</b>	<b>44</b>

## DAFTAR TABEL

Tabel	Halaman
1. Data jarak antar gedung Universitas Lampung .....	25
2. Letak titik atau nama gedung Universitas Lampung .....	27
3. Sisi yang dipilih dari hasil iterasi menggunakan Algoritma Prim .....	34
4. Tabel Sisi yang dihapus dari hasil iterasi Algoritma Prim dan <i>software</i> POM-QM .....	38



## DAFTAR GAMBAR

Gambar	Halaman
1. Graf sederhana .....	6
2. Contoh titik terpencil .....	7
3. Graf kosong .....	7
4. Contoh graf terhubung dan tak terhubung .....	9
5. Contoh <i>upagraf</i> .....	9
6. Contoh <i>cut-set</i> .....	10
7. Graf berbobot .....	10
8. Contoh sisi ganda dan <i>loop</i> .....	11
9. Graf lengkap .....	12
10. Graf tak berarah .....	13
11. Graf berarah .....	13
12. Contoh pohon dan bukan pohon .....	14
13. Graf lengkap $G$ dengan 4 jenis hasil pohon merentangannya.....	15
14. Langkah ke 1 <i>POM-QM</i> .....	19
15. Langkah ke 2 <i>POM-QM</i> .....	19
16. Langkah ke 3 <i>POM-QM</i> .....	20
17. Langkah ke 4 <i>POM-QM</i> .....	20
18. Langkah ke 5 <i>POM-QM</i> .....	21

19. <i>Solution steps POM-QM</i> .....	21
20. Peta kampus Universitas Lampung .....	24
21. Graf awal jalur antar gedung Universitas Lampung .....	28
22. Iterasi ke 1 pencarian MST dengan Algoritma Prim .....	29
23. Iterasi ke 2 pencarian MST dengan Algoritma Prim .....	30
24. Iterasi ke 3 pencarian MST dengan Algoritma Prim .....	30
25. Iterasi ke 4 pencarian MST dengan Algoritma Prim .....	30
26. Iterasi ke 5 pencarian MST dengan Algoritma Prim .....	31
27. Iterasi ke 6 pencarian MST dengan Algoritma Prim .....	31
28. Iterasi ke 50 pencarian MST dengan Algoritma Prim .....	32
29. <i>Input data</i> pencarian MST dengan <i>software</i> POM-QM .....	35
30. <i>Output data</i> pencarian MST dengan <i>software</i> POM-QM .....	36

## I. PENDAHULUAN

### 1.1 Latar Belakang

Dalam kehidupan sehari-hari sering kali seseorang melakukan perjalanan dari satu tempat ke tempat yang lain, dan sangat umum apabila memilih jalan terdekat atau jarak terpendek untuk dapat meminimalisir waktu, tenaga dan biaya. Kesulitan menentukan jarak terpendek timbul karena terdapat banyak jalur yang ada pada tiap tempat karena dalam kenyataannya dari tempat A ke tempat Z tidak hanya memiliki satu jalur saja, banyak sekali jalur yang dapat dilalui sehingga terbentuk suatu jaringan/lintasan. Suatu jaringan memiliki banyak jalur alternatif dari kedudukan semula ke kedudukan yang dikehendaki. Untuk mendapatkan rute perjalanan yang paling optimum, rangkaian satu tempat ke tempat lainnya yang dikunjungi harus membentuk suatu jalur sedemikian sehingga tempat-tempat yang dituju tersebut hanya dilewati satu kali dan kemudian kembali lagi ke tempat awal dengan mempertimbangkan jarak tempuh yang efektif dan efisien (Munir, 2010).

Dalam menemukan rute perjalanan optimum ini juga terdapat beberapa parameter yang perlu ditentukan sebelumnya untuk memperhitungkan nilai ongkos (*cost*) yaitu jarak antara titik awal ke tempat tujuan dan jarak antara tempat satu ke tempat lainnya. Titik tujuan dari tiap tempat akan diasosiasikan sebagai titik dan titik awal sebagai akar. Peta yang menggambarkan lokasi dari titik awal dan titik-titik tujuan akan direpresentasikan dalam bentuk graf lengkap (Munir, 2010).

Pencarian pohon merentang merupakan lintasan dengan bobot yang lebih kecil dari lintasan awal yang dilalui. Bobot di sini dapat berupa jarak, waktu tempuh atau bisa juga ongkos transportasi dari satu tujuan ke tujuan lainnya yang berbentuk seperti rute tertentu sedangkan persoalan pohon rentang minimum atau *Minimum Spanning Tree* (MST) adalah variasi dari persoalan rute terpendek juga, namun perbedaannya terletak pada lintasan atau rute yang akan dicari, di mana harus ditentukan sisi paling minimal yang menghubungkan titik-titik yang ada pada jaringan, hingga pada akhirnya diperoleh panjang sisi total dengan hasil yang minimum. Dengan diperolehnya *minimum spanning tree* dari graf, maka akan diketahui keoptimalan suatu jaringan graf tersebut. Algoritma yang biasanya digunakan dalam mencari *minimum spanning tree* yaitu Algoritma Prim dan Algoritma Kruskal. Walaupun ada lebih dari satu algoritma yang berbeda, namun hasil yang didapat akan sama panjangnya atau jumlah bobot minimal yang didapatkan akan sama besar (Siang, 2004).

Permasalahan dalam menentukan rute terpendek ini telah diterapkan dalam beberapa penelitian, seperti penelitian Annisa dan Mulyani (2020) yang bertujuan untuk melihat pengimplementasian dari metode *minimum spanning tree* Algoritma Kruskal dalam sistem jaringan listrik di Kota Langsa agar dapat meminimumkan kabel listrik yang digunakan. Hasil yang diperoleh yaitu adanya perbedaan panjang kabel dan total sisi yang digunakan untuk sistem jaringan listrik sebelum menggunakan metode *minimum spanning tree* Algoritma Kruskal dan setelah menggunakannya. Panjang kabel listrik sebelum menggunakan metode *minimum spanning tree* Algoritma Kruskal adalah sepanjang 8.834 meter dengan total sisi 40, sedangkan setelah menggunakan metode *spanning tree* Algoritma Kruskal adalah sepanjang 7.581 meter dengan total sisi 34.

Penelitian lain yang berkaitan juga dilakukan oleh Priyono (2021) yang bertujuan untuk menentukan lintasan terpendek pada *effective call salesman* di PT. BSP Purwokerto dengan menggunakan metode *minimum spanning tree* Algoritma Kruskal. Hasil menunjukkan bahwa jarak yang ditempuh setelah melakukan perhitungan dengan metode *minimum spanning tree* Algoritma Kruskal adalah sejauh 14.110 meter, sedangkan sebelum melakukan perhitungan dengan metode



*minimum spanning tree* Algoritma Kruskal jarak yang ditempuhnya adalah sejauh 16.660 meter.

Penelitian terdahulu selanjutnya adalah penelitian yang diteliti oleh Aji, dkk (2021) dengan tujuan untuk mencari jalur terpendek yang harus dilalui kurir untuk mengirim barang dengan metode *assignment* dan *networking* dengan hasil jarak optimum yang didapatkan adalah 34 km. Ada juga penelitian lain yang disusun oleh Fatimah dan Sam (2020) bertujuan untuk menentukan lintasan yang mengarah ke pusat perbelanjaan di Kota Palopo menggunakan metode *minimum spanning tree* algoritma semut dengan data jarak antar 8 pusat perbelanjaan di Kota Palopo dan mendapatkan jarak *minimum* yang diperoleh yaitu sejauh 4,21 Km. Selanjutnya penelitian yang dilakukan oleh Subidyo, dkk (2020) dengan tujuan untuk mencari penyelesaian *minimum spanning tree* terpendek untuk mendistribusikan naskah USBN SD/MI di Kabupaten Sragen dengan metode *minimum spanning tree* Algoritma Kruskal. Hasil jarak terpendek yang diperoleh untuk mendistribusikan naskah USBN SD/MI dengan perhitungan *manual* dan bantuan *software* TORA yaitu sejauh 112,2 km.

Pergerakan mahasiswa Universitas Lampung dari satu gedung ke gedung lain untuk dapat mengikuti seluruh kegiatan yang ada baik itu tentang hal akademik maupun nonakademik akan sangat mudah apabila terdapat jalan atau koridor beratap yang menghubungkan antar gedung. Adanya jalan atau koridor beratap membuat mahasiswa tetap dapat berpindah dari satu gedung ke gedung lain tanpa merasa khawatir terlambat ketika cuaca sedang panas dan hujan sekalipun. Berdasarkan latar belakang tersebut, tujuan dari penelitian ini adalah mengetahui jarak terpendek antar gedung sehingga dapat meminimalkan biaya pembangunan jalur koridor beratap di sana. Untuk menghindari terjadinya *human error* atau kesalahan oleh manusia dalam proses pencarian *minimum spanning tree* tersebut, maka diperlukanlah suatu aplikasi ataupun *software* yang efisien dan efektif untuk dijadikan sebagai pembanding dalam memecahkan permasalahan tersebut. Salah satu aplikasi yang dapat digunakan untuk menyelesaikan masalah ini yaitu dengan menggunakan *software* POM-QM.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka rumusan masalah yang dibahas pada penelitian ini adalah bagaimana penyelesaian dan hasil *Minimum Spanning Tree* (MST) pada rancangan pembangunan jalan berkanopi di Universitas Lampung agar dapat meminimalisir biaya pembangunannya dengan menerapkan Algoritma Prim.

## 1.3 Tujuan Penelitian

Berdasarkan rumusan masalah yang ada, maka tujuan dari dilakukannya penelitian ini adalah menentukan jalur minimum atau jalur terpendek agar dapat meminimumkan pembangunan jalur koridor beratap antar gedung di Universitas Lampung.

## 1.4 Manfaat

Manfaat dari penelitian ini adalah:

1. Dapat digunakan sebagai langkah awal dalam pemanfaatan teknologi informasi dan sebagai media informasi untuk pembangunan jalan agar lebih efisien dengan penentuan jarak terpendek.
2. Memberi kemudahan pengguna untuk mengetahui jarak terpendek yang harus dilalui untuk mencapai suatu tempat.

## II. TINJAUAN PUSTAKA

### 2.1 Definisi Graf

Menurut Siang (2004) graf adalah suatu diagram yang memuat informasi tertentu jika diinterpretasikan secara tepat. Dalam Kehidupan sehari-hari graf digunakan untuk menggambarkan berbagai macam struktur yang ada. Tujuannya adalah sebagai visualisasi objek agar mudah dimengerti. Beberapa contoh graf yang sering dijumpai dalam kehidupan sehari-hari seperti struktur organisasi, peta, rangkaian listrik, dan lain-lain.

Sebuah graf  $G$  terdiri dari dua bagian:

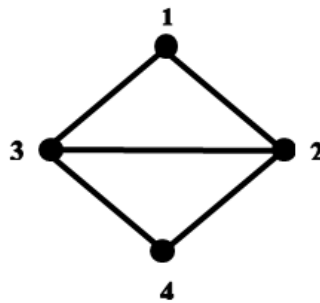
1. Sebuah himpunan  $V = V(G)$  memiliki elemen-elemen yang dinamakan titik atau *node*.
2. Sebuah kumpulan  $E = E(G)$  merupakan pasangan tak berurut dari titik-titik yang berbeda dinamakan sisi (*edge*). Dituliskan  $G = (V, E)$  bila ingin menyatakan dua bagian dari  $G$  (Lipschutz dan Lipson, 2002).

## 2.2 Terminologi Graf

Dalam terminologi graf, graf memiliki banyak istilah-istilah yaitu:

### 1. Ketetanggaan (*Adjacent*)

Dua buah titik pada graf tak berarah  $G$  dikatakan bertetangga bila keduanya terhubung langsung dengan sebuah sisi (Munir, 2010).



Gambar 2.1 Graf sederhana (Munir, 2010)

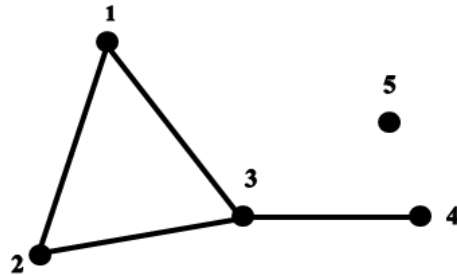
Contoh *Adjacent* dapat dilihat pada Gambar 2.1 yaitu titik 1 bertetangga dengan titik 2 dan 3, titik 1 tidak bertetangga dengan titik 4.

### 2. Menempel (*Incidency*)

Untuk sembarang sisi  $e = (v_j, v_k)$  dikatakan  $e$  menempel dengan titik  $v_j$ , atau  $e$  menempel dengan titik  $v_k$  (Munir, 2010). Contoh *Incidency* dapat kita lihat pada Gambar 2.1 yaitu sisi (2,3) menempel pada titik 2 dan titik 3, sisi (2,4) menempel pada dengan titik 2 dan 4.

### 3. Titik Terpencil

Titik terpencil ialah titik yang tidak mempunyai sisi yang menempel dengannya (Munir, 2010).

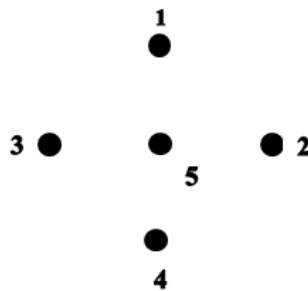


Gambar 2.2 Contoh titik terpencil (Munir, 2010)

Contoh titik terpencil dapat dilihat pada Gambar 2.2 yaitu titik 5 yang merupakan titik terpencil karena tidak menempel pada sisi manapun.

### 4. Graf Kosong

Sebuah graf yang hanya terdiri dari titik terpencil atau tidak memiliki sisi disebut graf kosong (Munir, 2010).



Gambar 2.3 Graf kosong (Munir, 2010)

## 5. Derajat

Derajat adalah jumlah sisi yang bersisian dengan titik tersebut. Derajat suatu titik disimbolkan dengan  $d(v)$  (Munir, 2010). Contoh Derajat dapat dilihat pada Gambar 2.1 yaitu derajat pada titik 1 =  $d(1) = 2$  dan derajat pada titik 2 =  $d(2) = 3$ .

## 6. Lintasan

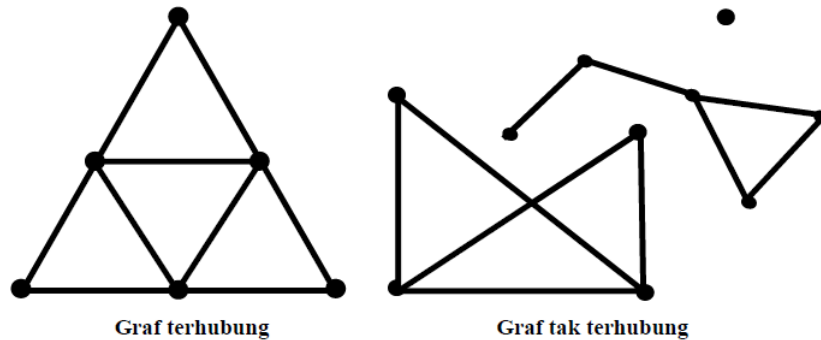
Lintasan (*path*) yang memiliki panjang  $n$  dari suatu titik awal  $v_0$  ke titik tujuan  $v_n$  dalam graf  $G$  adalah barisan yang berselang-seling dari titik-titik dan sisi-sisi yang berbentuk  $v_0, e_1, v_1, e_2, v_2, \dots, v_{n-1}, e_n, v_n$  sedemikian sehingga  $e_1 = (v_0, v_1), e_2 = (v_1, v_2), \dots, e_n = (v_{n-1}, v_n)$  merupakan sisi pada graf  $G$ . Panjang lintasan merupakan jumlah sisi yang berada pada lintasan tersebut (Wahyuningrum dan Usada, 2016). Contoh lintasan dapat dilihat pada Gambar 2.1 yaitu lintasan 1, 2, 4, 3 adalah lintasan dengan barisan sisi (1,2), (2,4), (4,3) dan memiliki panjang 3.

## 7. Siklus

Lintasan yang berawal dan berakhir pada titik yang sama disebut sirkuit atau siklus (Purwanto, dkk, 2006). Panjang sebuah sirkuit adalah jumlah sisi pada sirkuit tersebut. Contoh siklus yang dapat kita lihat pada Gambar 2.1 yaitu lintasan 1, 2, 3, 1 adalah sebuah sirkuit. Sirkuit 1, 2, 3, 1 memiliki panjang 3.

## 8. Keterhubungan

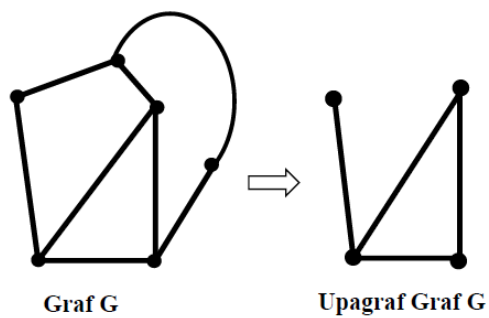
Titik  $a$  dan titik  $b$  disebut terhubung jika terdapat paling sedikit satu buah lintasan dari  $a$  ke  $b$ . Disebut graf terhubung (*connected graph*) jika untuk setiap pasang titik dalam himpunan  $V$  terdapat lintasan dari  $v_i$  ke  $v_j$ . Jika tidak, maka  $G$  disebut graf tak-terhubung (*disconnected graph*) (Munir, 2010).



Gambar 2.4 Contoh graf terhubung dan tak terhubung

## 9. Upagraf (*Subgraph*)

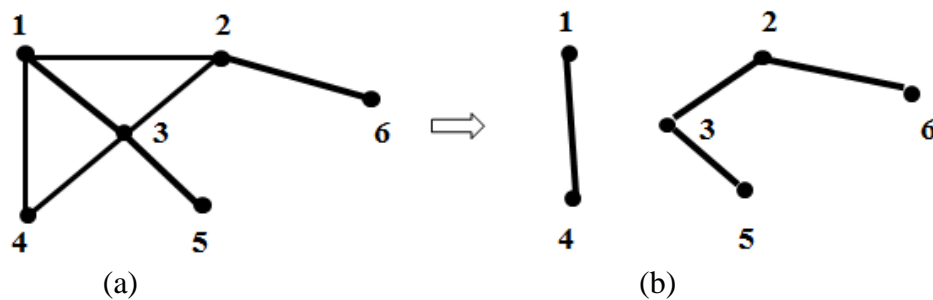
Misalkan  $G = (V, E)$  adalah sebuah graf.  $G_1 = (V_1, E_1)$  adalah *subgraph* dari  $G$  jika  $V_1 \subseteq V$  dan  $E_1 \subseteq E$  (Munir, 2010).



Gambar 2.5 Contoh upagraf (*Subgraph*) (Munir, 2010)

## 10. Cut-Set

*Cut-set* dari graf terhubung  $G$  adalah himpunan sisi minimal yang bila dibuang dari  $G$  menyebabkan  $G$  tidak terhubung. Jadi, *cut-set* selalu menghasilkan dua buah komponen terhubung (Munir, 2010).

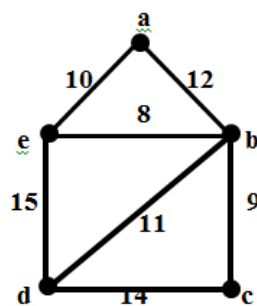


Gambar 2.6 Contoh *cut-set* (Munir, 2010)

Pada gambar 2.6 bagian (a) jika dibuang sisi (1,2) maka graf masih terhubung. Jika membuang sisi (1,2) dan (1,5) graf masih tetap terhubung. Tetapi, jika membuang sisi-sisi dalam himpunan  $\{(1,2), (1,5), (3,5), (3,4)\}$  seperti pada gambar (b) barulah graf menjadi tidak terhubung. Jadi,  $\{(1,2), (1,5), (3,5), (3,4)\}$  adalah *cut-set*.

## 11. Graf Berbobot

Graf berbobot adalah graf yang setiap sisinya diberi sebuah harga atau bobot (Munir, 2010).

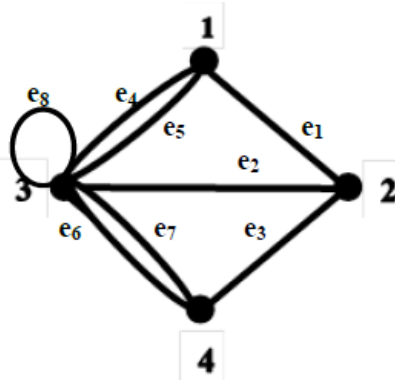


2.7 Graf berbobot (Munir, 2010)



## 12. Sisi Ganda dan Sisi Gelang (*Loop*).

Sisi ganda adalah dua sisi yang terhubung pada dua titik yang sama, sedangkan *loop* adalah sisi yang berawal dan berakhir pada titik yang sama (Munir, 2010). Contoh sisi ganda dan *loop* dapat dilihat pada gambar berikut di mana sisi  $e_4$  dan  $e_5$  adalah sisi ganda karena sama-sama terhubung pada titik 1 dan 3 sedangkan sisi  $e_8$  merupakan contoh *loop* karena berawal dan berakhir pada titik 3.



Gambar 2.8 Contoh sisi ganda dan *loop* (Munir, 2010)

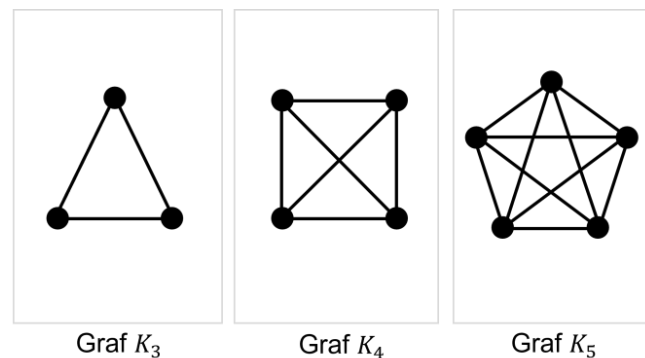
### 2.3 Jenis-Jenis Graf

Menurut Munir (2010), graf dapat dikelompokkan menjadi beberapa jenis tergantung dari sudut pandang pengelompokannya. Pengelompokan graf dapat dipandang berdasarkan dari ada tidaknya sisi ganda, berdasarkan keterhubungan titik, atau berdasarkan orientasi arah pada sisi.

Berdasarkan ada tidaknya *loop* atau sisi ganda pada suatu graf, maka secara umum graf dapat digolongkan menjadi dua jenis, yaitu:

### 1. Graf sederhana (*Simple Graph*)

Graf yang tidak mengandung *loop* maupun sisi ganda dinamakan graf sederhana. Pada graf sederhana, sisi adalah pasangan tak-terurut. Jadi penulisan sisinya  $(u, v)$  sama saja dengan  $(v, u)$ . Graf sederhana juga dapat didefinisikan  $G = (V, E)$  di mana  $V$  terdiri dari himpunan tidak kosong titik-titik dan  $E$  adalah himpunan pasangan tak terurut yang berbeda disebut sisi. Gambar 2.1 merupakan Graf sederhana yang tidak mengandung *loop* dan sisi ganda. Salah satu contoh dari graf sederhana adalah graf lengkap yang di mana setiap titiknya bertetangga dengan titik lainnya. Graf lengkap  $n$  buah titik dilambangkan dengan  $K_n$ . Setiap titik pada  $K_n$  berderajat  $n - 1$ .



Gambar 2.9 Graf lengkap (Munir, 2010)

### 2. Graf tak sederhana (*Unsimple Graph*)

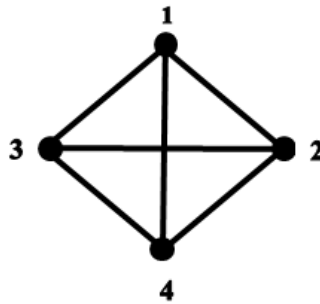
Graf yang mengandung sisi ganda atau gelang (*loop*) dinamakan graf tak sederhana. Ada dua jenis graf tak sederhana, yaitu:

- Graf ganda ialah graf yang mengandung sisi ganda. Sisi ganda yang menghubungkan sepasang titik bisa lebih dari dua buah.
- Graf semu ialah graf yang mengandung gelang (*loop*).

Berdasarkan keterhubungan titik maka graf dapat dibedakan menjadi dua jenis yaitu graf terhubung dan graf tak terhubung. Kemudian berdasarkan orientasi arah pada sisi, maka secara umum graf dibedakan atas dua jenis, yaitu:

1. Graf tak berarah (*Undirected Graph*)

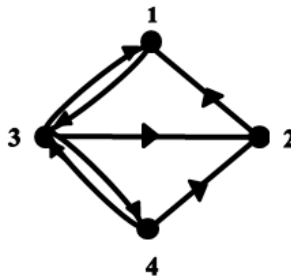
Graf yang sisinya tidak mempunyai orientasi arah disebut graf tak-berarah, urutan pasangan titik yang dihubungkan oleh sisi tidak diperhatikan. Jadi,  $(u, v) = (v, u)$  adalah sisi yang sama.



Gambar 2.10 Graf tak berarah (Munir, 2010)

2. Graf berarah (*Directed Graph* atau *Digraph*)

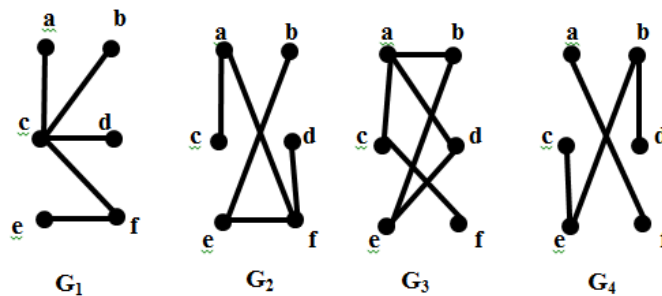
Graf berarah merupakan graf yang setiap sisinya mempunyai arah dan tidak mempunyai dua sisi yang berlawanan antara dua buah titik. Pada graf berarah,  $(u, v)$  dan  $(v, u)$  menyatakan dua buah sisi yang berbeda, dengan kata lain  $(u, v) \neq (v, u)$ . Untuk sisi  $(u, v)$ , titik  $u$  dinamakan titik asal (*initial node*) dan titik  $v$  dinamakan titik terminal (*terminal node*).



Gambar 2.11 Graf berarah (Munir, 2010)

## 2.4 Pohon

Pohon adalah graf terhubung yang tidak memiliki siklus (*cycle*). Di dalam suatu pohon tidak memuat sisi-sisi yang *paralel* dan *loop*. Karena itu pohon juga merupakan graf sederhana. Pohon merupakan suatu graf yang tidak memiliki arah. Pohon (*tree*) membentuk salah satu subkelas dari graf yang paling banyak digunakan. Subkelas dari pohon misalnya pohon berakar dan biner serta beberapa penerapan dari pohon (misalnya pohon rentang, pohon keputusan, dan pohon permainan) (Siang, 2004). Contoh pohon dapat dilihat pada gambar berikut dimana  $G_1$  dan  $G_2$  adalah pohon,  $G_3$  bukan pohon karena terdapat *cycle* dan  $G_4$  juga bukan pohon karena grafnya tak terhubung.



Gambar 2.12 Contoh pohon dan bukan pohon (Munir, 2010)

Pohon dapat dilambangkan sebagai  $G = (V, E)$  dengan titik sebanyak  $n$  dan memiliki properti di antaranya:

1.  $G$  merupakan sebuah pohon.
2. Masing-masing titik dalam  $G$  saling terhubung dengan lintasan yang tunggal.
3.  $G$  terhubung dan memiliki sisi sebanyak  $n - 1$ .
4.  $G$  tidak memiliki sirkuit.
5. Penambahan satu buah sisi pada pohon  $G$  hanya akan menghasilkan satu buah sirkuit.
6. Semua sisi dalam pohon  $G$  adalah jembatan.

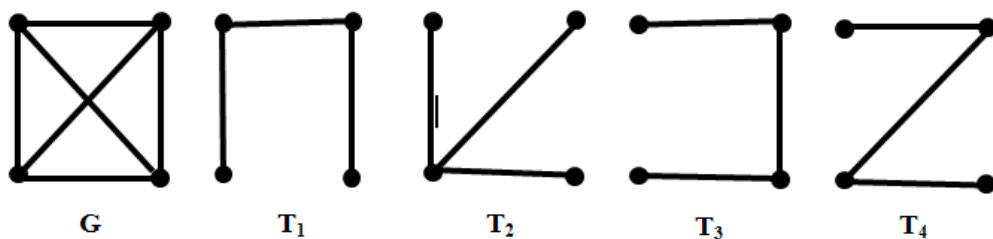
(Chartrand & Lesniak, 1986).

## 2.5 Pohon Merentang (*Spanning Tree*)

Menurut Siang (2004) *Spanning tree* dari sebuah graf yang terhubung merupakan sebuah *subgraf* merentang berupa pohon. Sembarang pohon yang dibentuk dari suatu graf disebut *spanning tree*. *Spanning tree* dihasilkan dari pemutusan sirkuit dalam sebuah graf. Misalkan  $G = (V, E)$  adalah graf tak berarah terhubung yang memiliki beberapa sirkuit.  $G$  dapat diubah menjadi pohon  $T = (V_i, E_i)$  dengan cara menghilangkan sirkuit-sirkuit yang sebelumnya terhubung. Caranya yaitu pada tahap awal pilih sebuah sirkuit, kemudian hapus sebuah sisi dari sirkuit tersebut. Pemutusan sirkuit tersebut tidak akan mengubah  $G$  menjadi tidak terhubung, hanya saja jumlah sirkuitnya berkurang satu. Lakukan proses tersebut sampai semua sirkuit pada  $G$  hilang dan menjadi sebuah pohon  $T$  yang dinamakan *spanning tree* (Munir, 2010).

Setiap graf terhubung mempunyai sekurang-kurangnya satu *spanning tree*. Kenyataan ini dapat ditunjukkan melalui cara membuat *spanning tree* dari grafnya. Jika suatu graf  $G$  terhubung dan tidak mempunyai sirkuit maka *spanning tree*nya adalah  $G$ . Graf yang tidak terhubung dengan  $n$  buah komponen memiliki  $n$  buah hutan yang merentang. Graf yang terhubung dan berbobot bisa memiliki lebih dari satu *spanning tree* (Siang, 2004).

Perhatikan gambar di bawah ini:



Gambar 2.13 Graf lengkap  $G$  dengan 4 contoh *spanning tree* yang terbentuk (Munir, 2010)

Sebuah sisi dalam suatu *spanning tree*  $T$  disebut dengan sebuah cabang (*branch*) dan sebuah sisi dari yang tidak dimuat oleh *spanning tree* disebut dengan tali (*chord*). Suatu sisi bisa saja merupakan *branch* untuk suatu *spanning tree* tetapi merupakan *chord* untuk *spanning tree* yang lainnya (Munir, 2010).

## 2.6 Minimum Spanning Tree

Misalkan  $G$  adalah graf berbobot, maka bobot *spanning tree*  $T$  dari  $G$  didefinisikan sebagai jumlah bobot semua sisi di  $T$ . *Spanning tree* yang berbeda mempunyai bobot yang berbeda pula. Dari semua *spanning tree* di  $G$ , yang berbobot minimum dinamakan *Minimum Spanning Tree* (MST). Persoalan ini merupakan variasi dari persoalan rute terpendek yang perbedaannya terletak pada lintasan yang dicari. Pada rute terpendek, yang dicari adalah lintasan/rute dari sumber ke tujuan yang memberikan total jarak minimum, sedangkan pada persoalan MST, yang dipersoalkan ialah menentukan sisi-sisi yang menghubungkan titik-titik yang ada pada jaringan, sehingga diperoleh panjang sisi total yang minimum (Munir, 2010).

Menurut Siang (2004) cara paling sederhana menyelesaikan *minimum spanning tree* adalah dengan mendaftarkan semua pohon rentang yang mungkin dibuat dan menghitung total bobot tiap-tiap pohon rentang. Selanjutnya dipilih pohon rentang dengan total bobot yang paling kecil. Namun metode ini kurang efisien, terutama pada graf yang cukup besar karena terdapat banyak sekali pohon merentang yang akan didapatkan, maka dari itu dibutuhkanlah suatu algoritma dalam penyelesaiannya.

## 2.7 Algoritma Prim

Algoritma ini ditemukan pada tahun 1930 oleh seorang matematikawan Voljtêch Jarnik, lalu secara terpisah oleh ahli komputer Robert C. Prim di tahun 1957, kemudian dikembangkan lagi oleh Dijkstra di tahun 1959. Algoritma Prim adalah sebuah algoritma dalam teori graf untuk mencari *minimum spanning tree* sebuah graf berbobot yang saling terhubung. Ini berarti bahwa sebuah himpunan bagian dari sisi yang membentuk suatu pohon yang mengandung titik, di mana bobot keseluruhan dari semua sisi dalam pohon diminimalkan. Algoritma Prim menitikberatkan pada pemilihan bobot minimal berdasarkan titik yang diambil, dan karena tidak perlu mengurutkan terlebih dahulu, Algoritma Prim cocok untuk graf dengan jumlah titik yang banyak (Siang, 2004).

Algoritma Prim menyusun pohon dengan memilih sembarang titik dan kemudian suatu sisi dengan bobot terkecil pada titik itu. Berikutnya pohon itu dikembangkan dengan memilih suatu sisi berbobot terkecil yang dengan sisi terpilih sebelumnya membentuk pohon. Pohon itu masih dikembangkan lagi dengan memilih suatu sisi berbobot terkecil yang membentuk pohon dengan dua sisi terpilih sebelumnya. Proses ini dilanjutkan sampai terpilih sebuah pohon jumlah, yang juga merupakan *Minimum Spanning Tree* (Dossey, 1978).

Langkah- langkah dalam penyelesaian Algoritma Prim yaitu:

1. Ambil titik dari graf  $G = (V, E)$ , masukkan ke dalam  $T = (V_T, E_T)$ .
2. Pilih sisi  $(v_i, v_j)$  yang memiliki bobot minimum dan bersisian dengan titik di  $T$ , tetapi  $(v_i, v_j)$  tidak membentuk sirkuit di  $T$ . Tambahkan  $(v_i, v_j)$  ke dalam  $T$ .
3. Ulangi langkah 2 sampai sisi  $n - 1$  dengan  $n$  merupakan jumlah titik.

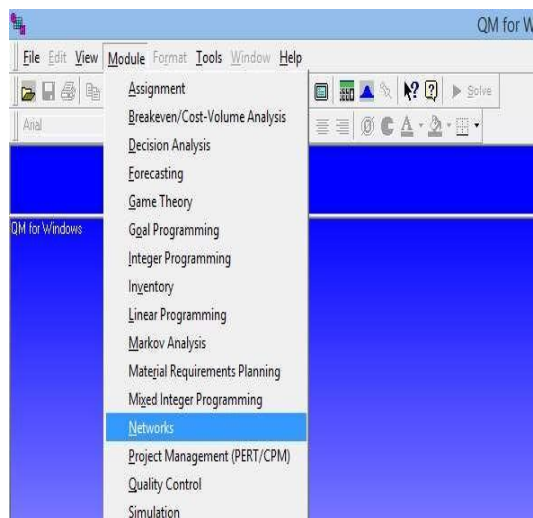
(Vasuder, 2009)

## 2.8 POM-QM

*Software* yang digunakan dalam penelitian ini adalah *Software POM-QM for Windows 3* dirancang untuk melakukan perhitungan yang diperlukan pihak manajemen untuk mengambil keputusan di bidang produksi dan pemasaran. *Software* ini dirancang oleh Howard J. Weiss pada tahun 1996 untuk membantu manajer produksi khususnya dalam menyusun prakiraan dan anggaran untuk produksi bahan baku menjadi produk jadi atau setengah jadi dalam proses pabrikan. Tetapi sekarang modul-modul yang terdapat di *software* ini sudah lebih berkembang dan dapat digunakan untuk menghitung berbagai permasalahan seperti *quantitative methods* dan *management science*. *Software* ini dibekali berbagai macam modul, namun penggunaan yang berkaitan dengan penulisan ini adalah modul *network* bagian *minimum spanning tree* (Weiss, 2006).

Adapun langkah yang dilakukan untuk melakukan perhitungan pada *software POM-QM* adalah sebagai berikut:

1. Pada *menu bar* klik “*module*” kemudian sorot lalu klik “*networks*”, seperti pada gambar di bawah:

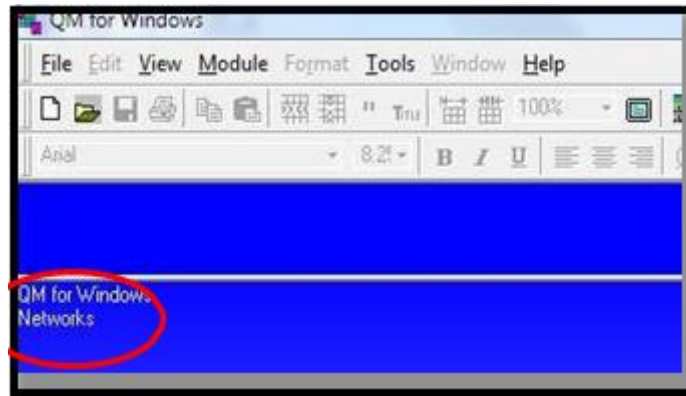


Gambar 2.14 Langkah ke 1 *POM-QM*

Sumber: *Software POM-QM For Windows V4*



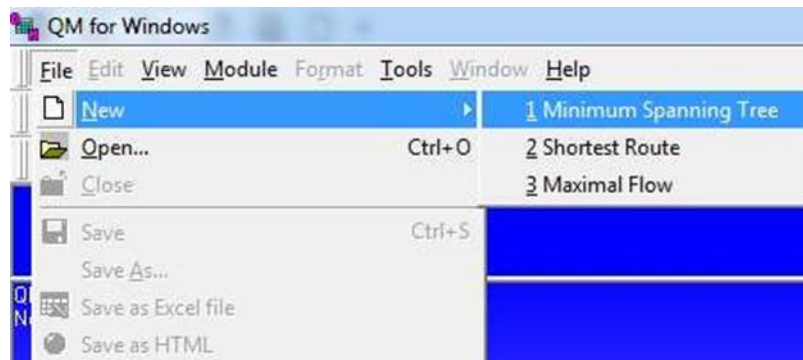
2. POM-QM *for Windows* akan menunjukkan dalam keadaan aktif untuk penyelesaian masalah *network*:



Gambar 2.15 Langkah ke 2 *POM-QM*

Sumber: *Software POM-QM For Windows V4*

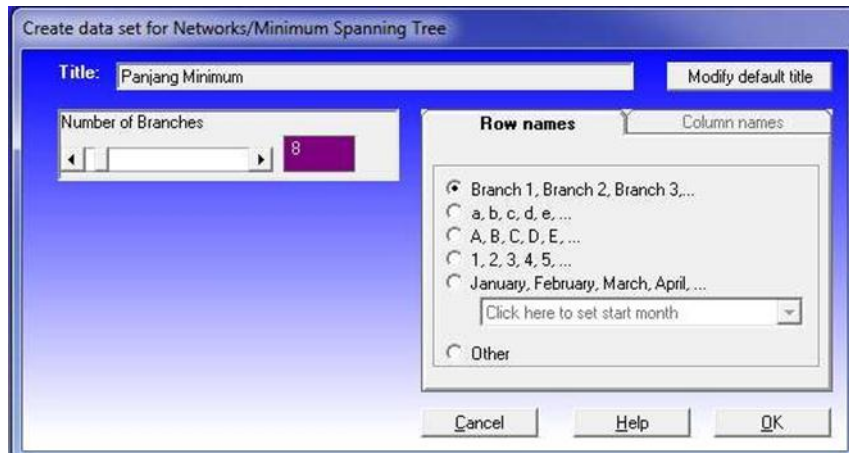
3. Kemudian klik *menu bar "file"* kemudian sorot "*new*" pilih dan klik pilihan ke-1 yaitu "*Minimum Spanning Tree*", seperti pada gambar di bawah:



Gambar 2.16 Langkah ke 3 *POM-QM*

Sumber: *Software POM-QM For Windows V4*

4. Kemudian masukkan *title* atau judul setelah itu *input* jumlah sisi yang akan di uji, lalu pilih salah satu jenis *node* yang diinginkan, kemudian klik “OK”:



Gambar 2.17 Langkah ke 4 *POM-QM*

Sumber: *Software POM-QM For Windows V4*

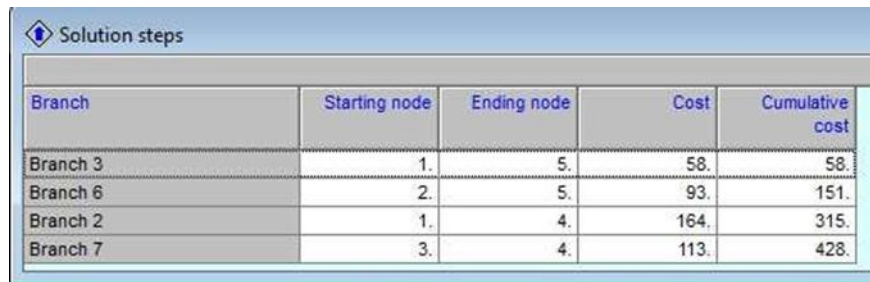
5. Muncul tampilan seperti berikut dan masukkan data/jarak masing- masing *arc*:

Starting node for iterations		Instruction	
1		Enter the cost	
Branch name	Start node	End node	Cost
Branch 1	1	3	217
Branch 2	1	4	164
Branch 3	1	5	58
Branch 4	2	3	304
Branch 5	2	4	215
Branch 6	2	5	93
Branch 7	3	4	113

Gambar 2.18 Langkah ke 5 *POM-QM*

Sumber: *Software POM-QM For Windows V4*

6. Klik *solve* dan lihat hasilnya pada *solution steps*:



Branch	Starting node	Ending node	Cost	Cumulative cost
Branch 3	1.	5.	58.	58.
Branch 6	2.	5.	93.	151.
Branch 2	1.	4.	164.	315.
Branch 7	3.	4.	113.	428.

Gambar 2.19 *Solution steps POM-QM*

Sumber: *Software POM-QM For Windows V4*

### **III. METODOLOGI PENELITIAN**

#### **3.1 Lokasi dan Waktu Penelitian**

Penelitian ini dilakukan pada semester ganjil tahun ajaran 2024/2025 di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

#### **3.2 Sumber Data**

Sumber data dalam penelitian ini yaitu data berupa denah Universitas Lampung dan data jarak antar gedung Universitas Lampung yang diperoleh dari Google Maps yang dapat ditempuh dengan berjalan kaki guna pembuatan jalur koridor beratap Universitas Lampung yang memungkinkan.

### 3.3 Metode Penelitian

Langkah-langkah yang dilakukan pada penelitian ini adalah sebagai berikut:

#### 1. Pengambilan Data

Data berupa gambar denah Universitas Lampung dan data jarak pada *maps* untuk menentukan bobot graf yang akan dibentuk. Pada tahapan ini dilakukan pembentukan graf awal dari jalur pejalan kaki. Setelah itu dicari pohon merentang minimumnya.

#### 2. Pencarian *Minimum Spanning Tree*

Setelah dibentuk graf berbobot, maka dapat dilakukan perhitungan dan penganalisan data tersebut dengan menggunakan Algoritma Prim untuk menentukan *Minimum Spanning Tree* (MST) dan penggunaan *software POM-QM* untuk perbandingan hasil akhir.

#### 3. Interpretasi Hasil

Pada tahap ini dilakukan perbandingan hasil antara pencarian MST menggunakan Algoritma Prim dengan jalur awal jalan yang ada di Universitas Lampung.

#### 4. Penarikan Kesimpulan

Penarikan kesimpulan dapat dilakukan setelah didapatkan perbedaan antara jumlah bobot awal jaringan sebelum dan sesudah dilakukan analisis. Jika bobot jaringan awal setelah dianalisis menjadi lebih pendek, maka pencarian MST menggunakan Algoritma Prim berhasil.

## V. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Berdasarkan hasil dan pembahasan pada BAB IV, maka diperoleh kesimpulan sebagai berikut:

1. Dari hasil pencarian *Minimum Spanning Tree* (MST) dengan menerapkan Algoritma Prim, baik secara manual dan menggunakan Software POM-QM, maka diperoleh panjang total jarak antar gedung yang ada di Universitas Lampung adalah 6.889 meter dengan 51 titik dan 50 sisi dari graf awal sebanyak 51 titik dan 76 sisi dengan panjang jaringan 15.669 meter.
2. Berdasarkan data awal panjang jarak antar gedung di Universitas Lampung yaitu sepanjang 15.669 meter, sedangkan hasil dari perhitungan *Minimum Spanning Tree* dengan menerapkan Algoritma Prim, total jarak antar gedung di Universitas Lampung lebih minimum sebesar 56 % dari jarak awal, yaitu 8.780 meter, hal ini mengakibatkan optimalnya pembangunan yang akan dilakukan sehingga Algoritma Prim dapat diterapkan untuk menentukan *Minimum Spanning Tree* dalam mengoptimalkan panjang jalur koridor beratap di Universitas Lampung.

## 5.2 Saran

1. Kekurangan dari pencarian *Minimum Spanning Tree* pada penelitian ini adalah dalam mengoptimalkan pembangunan jalan koridor beratap hanya menghitung panjang jarak atau jalur yang akan dibuat dan bukan biaya yang dikeluarkan, serta wilayah yang diteliti merupakan dataran datar. Oleh karena itu, disarankan agar penelitian selanjutnya dapat memperhatikan variabel tersebut.
2. Berdasarkan hasil penelitian dan pembahasan pada BAB IV yaitu pencarian *Minimum Spanning Tree* menggunakan Algoritma Prim dan *Software POM-QM*, penulis menyarankan untuk peneliti selanjutnya dapat menggunakan metode dan *Software* lainnya untuk mendapatkan model *Minimum Spanning Tree* dalam bentuk yang berbeda.

## DAFTAR PUSTAKA

- Aji, A. T. S., Sari, R. P., dan Dede, D. 2021. *Analisis Optimalisasi Penempatan dan Efisiensi Jarak Perjalanan Kurir Ke Berbagai Tujuan dengan Metode Assignment dan Networking*. Barometer. vol. 6. no. 2. pp. 343-351.
- Annisa, F., dan Muliani, F. 2020. Penerapan Algoritma Kruskal dalam Sisrem Jaringan Listrik di Kecamatan Langsa Baro. *Jurnal Gamma-Pi. Jurnal Matematika Terapan dan Pendidikan Matematika*, vol. 2, no. 2, pp. 5-9.
- Chartrand, G., & Lesniak, L. 1986. *Graphs And Diagraphs*. Prentice-Hall International. California
- Dossey, J. A. 1978. *Matematika Diskrit I*. MD: Computer Science Press.
- Fatimah, dan Sam, M. 2020. Aplikasi Algoritma dalam Menentukan Pohon Merentang Minimum (*Minimum Spanning Tree*) terhadap Lintasan yang Mengarah ke Pusat Perbelanjaan di Kota Palopo. *Infinity Jurnal Matematika & Aplikasinya*. vol. 1. no. 1. pp. 1-7.
- Lipschutz, S., dan Lipson, M. L. 2002. *Matematika Diskrit Jilid 2*. Salemba Teknika. Jakarta
- Munir, R. 2010. *Matematika Diskrit edisi ke 3 revisi keempat*. Informatika. Bandung.
- Priyono. 2021. Algoritma Kruskal Menentukan Lintasan Terpendek *Efektif Call Salesman*. *Fusioma (Fundamental Scientific Journal of Mathematics)*. *Jurnal Ilmiah Matematika Dasar*. vol. 1, no. 1, pp. 25-32.



Purwanto, H., Indriani, G., dan Dayanti, E. 2006. *Matematika Diskrit*. Ercontara Rajawali. Cirebon.

Siang, J. J. 2004. *Matematika Diskrit dan Aplikasinya pada Ilmu Komputer*. ANDI. Yogyakarta.

Sudiby, N.A., Purwanto, T., dan Rahmadi, D. 2020. *Minimum Spanning Tree pada Distribusi Bahan Naskah USBN SD/MI di Kabupaten Sragen. Research of Mathematics and Mathematics Education*, 2(2), 64-69.

Vasuder, C. 2009. *Graph Theory with Applications*. New Age International (P) Ltd. Publishers. New Delhi.

Wahyuningrum, T., dan Usada, E. 2016. *Matematika Diskrit Dan Penerapannya Dalam Dunia Informatika*. Deepublish – CV Budi Utama. Sleman.

Weiss H. J. 2006. *POM-QM for Windows (Version 3)*. Pearson/Prentice Hall.