

**PENERAPAN ALGORITMA *BELLMAN-FORD* DALAM MENENTUKAN
RUTE TERPENDEK OBJEK WISATA KOTA BANDAR LAMPUNG**

(Skripsi)

Oleh

**SAFHIRA AULIA HIDAYAT
2017031047**



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

ABSTRACT

APPLICATION OF *BELLMAN-FORD* ALGORITHM IN DETERMINING THE SHORTEST PATH OF TOURISM OBJECTS IN BANDAR LAMPUNG CITY

By

SAFHIRA AULIA HIDAYAT

In everyday life, tourists are certainly looking for the most efficient way to reach tourist destinations in the hope of shortening time and saving costs. In this research, the Bellman-Ford algorithm to determine the shortest route to various tourist destinations in Bandar Lampung City. This algorithm is effective in calculating the shortest distance from a point in a weighted and directed graph. This study shows that the Bellman-Ford algorithm can be used in determining the shortest route to tourist attractions in Bandar Lampung City. In this study, tourist attractions in Bandar Lampung City were modeled in the form of a graph, with 14 tourist objects modeled manually and 55 tourist objects modeled with the python programming language.

Keywords: *Graph, Shortest Path, Bellman-Ford Algorithm*

ABSTRAK

PENERAPAN ALGORITMA *BELLMAN-FORD* DALAM MENENTUKAN RUTE TERPENDEK OBJEK WISATA KOTA BANDAR LAMPUNG

Oleh

SAFHIRA AULIA HIDAYAT

Dalam kehidupan sehari – hari, para wisatawan tentu mencari cara paling efisien untuk mencapai destinasi wisata dengan harapan dapat mempersingkat waktu dan menghemat biaya. Pada penelitian ini algoritma *Bellman-Ford* untuk menentukan rute terpendek ke berbagai destinasi wisata Kota Bandar Lampung. Algoritma ini efektif dalam menghitung jarak terpendek dari satu titik dalam sebuah graf berbobot dan berarah. Pada penelitian ini menunjukkan bahwa algoritma *Bellman-Ford* dapat digunakan dalam menentukan rute terpendek ke tempat – tempat wisata di Kota Bandar Lampung. Dalam penelitian ini, tempat – tempat wisata di Kota Bandar Lampung dimodelkan dalam bentuk graf, dengan 14 objek wisata dimodelkan secara manual dan 55 objek wisata dimodelkan dengan bahasa pemrograman *python*.

Kata Kunci: Graf, Rute Terpendek, Algoritma *Bellman-Ford*

**PENERAPAN ALGORITMA *BELLMAN-FORD* DALAM MENENTUKAN
RUTE TERPENDEK OBJEK WISATA KOTA BANDAR LAMPUNG**

Oleh

SAFHIRA AULIA HIDAYAT

Skripsi

Sebagai Salah Satu Syarat Untuk Memperoleh Gelar
SARJANA MATEMATIKA

Pada

Jurusan Matematika
Fakultas Matematika Dan Ilmu Pengetahuan Alam
Universitas Lampung



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

Judul Skripsi : **PENERAPAN ALGORITMA *BELLMAN-FORD*
DALAM MENENTUKAN RUTE TERPENDEK
OBJEK WISATA KOTA BANDAR LAMPUNG**

Nama Mahasiswa : **Safhira Aulia Hidayat**

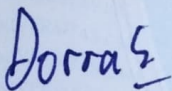
Nomor Pokok Mahasiswa : **2017031047**

Program Studi : **Matematika**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**

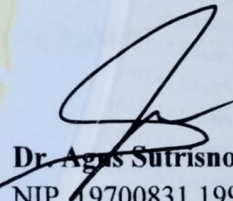
MENYETUJUI

1. Komisi Pembimbing



Dra. Dorrah Azis, M.Si.

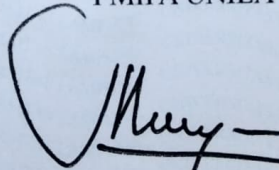
NIP. 19610128 198811 2 001



Dr. Agus Sutrisno, S.Si, M.Si.

NIP. 19700831 199903 1 002

2. Ketua Jurusan Matematika
FMIPA UNILA



Dr. Aang Nurjaman, S.Si., M.Si.

NIP. 19740316 200501 1 001

MENGESAHKAN

1. Tim Penguji

Ketua : **Dra. Dorrah Azis, M.Si.**

Dorrah
.....

Sekretaris : **Dr. Agus Sutrisno, S.Si, M.Si.**

Agus
.....

Penguji
Bukan Pembimbing : **Dr. Muslim Ansori, S.Si, M.Si.**

Muslim
.....

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Heri Satria
Dr. Eng Heri Satria, S.Si., M.Si.
NIP. 19711001 200501 1 002

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

PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini :

Judul Skripsi : **PENERAPAN ALGORITMA *BELLMAN-FORD* DALAM MENENTUKAN RUTE TERPENDEK OBJEK WISATA KOTA BANDAR LAMPUNG**

Nama Mahasiswa : **Safhira Aulia Hidayat**

Nomor Pokok Mahasiswa : **2017031047**

Jurusan : **Matematika**

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

Bandar Lampung, 19 April 2024
Penulis,



Safhira Aulia Hidayat
NPM. 2017031047

RIWAYAT HIDUP

Penulis lahir di Kotabumi, pada 17 Juli tahun 2002. Merupakan anak pertama dari empat bersaudara, dari Bapak Hidayat dan Ibu Rosmiah.

Riwayat pendidikan Taman Kanak-Kanak di TK Muslimin Kotabumi, Kabupaten Lampung Utara lulus tahun 2008. Sekolah Dasar di SD N 2 Kotabumi Tengah, Kabupaten Lampung Utara lulus tahun 2014. Sekolah Menengah Pertama di SMP N 1 Kotabumi, Kabupaten Lampung Utara lulus tahun 2017. Sekolah Menengah Atas di SMA N 4 Kotabumi, Kabupaten Lampung Utara lulus tahun 2020.

Penulis melanjutkan jenjang pendidikan di Universitas Lampung jalur SBMPTN dan terdaftar sebagai mahasiswa jurusan Matematika FMIPA. Riwayat organisasi yaitu Anggota Bidang Eksternal HIMATIKA (2021), Anggota Departemen Dana dan Usaha UKM SAINTEK (2021), Sekretaris Departemen Dana dan Usaha UKM SAINTEK (2022).

Sebagai aplikasi bidang ilmu kepada masyarakat, penulis telah menyelesaikan Kuliah Kerja Nyata di Desa Munca Kecamatan Teluk Pandan Kabupaten Pesawaran pada tahun 2023, dan Kerja Praktik (KP) di Asuransi BUMIDA Kecamatan Teluk Betung Utara Kota Bandar Lampung pada Tahun 2023.

KATA INSPIRASI

“Man Jadda Wajada”

“Innal A'maalu Binniyaat”

(HR. Bukhari dan Muslim)

“Maka, sesungguhnya beserta kesulitan ada kemudahan.

Sesungguhnya beserta kesulitan ada kemudahan.”

(QS. Al-Insyirah: 5-6)

“Ketika saya melibatkan Allah dalam semua rencana dan impian saya, dengan penuh keikhlasan dan keyakinan, aku percaya tidak ada yang tidak mungkin untuk diraih”

“Kesuksesan dan kebahagiaan terletak pada diri sendiri. Tetaplah berbahagia karena kebahagiaanmu dan kamu yang akan membentuk karakter kuat untuk melawan kesulitan”

(Hellen Keller)

“Terkadang keberhasilan sudah di depan mata, saat hati dan pikiran ingin menyerah. Jadi istirahatlah sejenak, ambil nafasmu dan pikirkan lagi dengan tenang.”

PERSEMBAHAN

Alhamdulillahirobbil'amin,

Akhirnya pekerjaan besar ini selesai. Pekerjaan besar yang harus dibanggakan khususnya untuk diri sendiri. Sudah berapa banyak kesedihan, mental, batin yang terluapkan, sudah berapa banyak emosi yang terbuang, sudah berapa banyak keprihatinan yang tersimpan, dan berapa besar harapan yang terenggam. Puji syukur kehadiran Allah SWT, sebuah karya yang penuh perjuangan telah terselesaikan.

Saya persembahkan karya ini untuk:

Kedua orang tua tersayang dan tercinta:

Terutama Mamah yang telah melahirkan Saya, orang yang selalu mendorong untuk menyelesaikan skripsi dan tak pernah berhenti untuk selalu mendoakan anaknya. Dan Ayah yang selalu membantu, mendukung dan mendoakan anaknya, agar sukses dan selalu membanggakan beliau.

Kepada adik perempuan yang amat saya sayangi:

Naura Salsabila Rania, Jihan Aqila Hidayat, dan Janeta kamila Hidayat

Bapak Ibu Dosen Pembimbing dan Pembahas.

Terimakasih telah memberikan cinta dan kasih sayangnya.

Terimakasih telah memberikan doa, dukungan, serta semangatnya.

Terimakasih telah memberikan kesabarannya.

SANWACANA

Puji syukur penulis ucapkan kepada Allah SWT atas segala ridha, rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi ini. Skripsi dengan judul “Penerapan Algoritma *BellMan-Ford* Dalam Menentukan Rute Terpendek Objek Wisata Kota Bandar Lampung” disusun untuk memenuhi salah satu syarat memperoleh gelar sarjana Matematika di Universitas Lampung.

Pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Ibu Dra. Dorrah Azis, M.Si. selaku Pembimbing I yang telah dengan sabar membimbing, memotivasi, dan memberikan arahan sehingga skripsi ini dapat terselesaikan tepat waktu.
2. Bapak Dr. Agus Sutrisno, S.Si., M.Si. selaku Pembimbing II yang telah memberikan dukungan, arahan, masukan, dan waktunya untuk membimbing dalam proses penyusunan skripsi ini.
3. Bapak Dr. Muslim Ansori, S.Si. M.Si. selaku Pembahas atas kesediaannya untuk menguji dan dengan sabar memberikan masukan, kritik, dan saran.
4. Ibu Prof. Dr. Ir. Netti Herawati, M.Sc. selaku dosen pembimbing akademik yang senantiasa memberikan motivasi dan membimbing selama menjalani perkuliahan.
5. Bapak Dr. Aang Nuryaman, S.Si., M.Si. selaku Ketua Jurusan Matematika FMIPA Universitas Lampung.
6. Bapak Dr. Eng. Heri Satria, S.Si., M.Si. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung.
7. Ibu Prof. Dr. Ir. Lusmeilia Afriani, D.E.A., IPM., ASEAN Eng. selaku Rektor Universitas Lampung.

8. Seluruh Dosen, staf, dan karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
9. Kedua orang tua saya, Mamah tersayang, Rosmiah dan Ayah tercinta, Hidayat. Terima kasih sebesar – besarnya penulis sampaikan kepada beliau yang hingga detik ini terus berjuang yang terbaik kepada putrinya baik secara materi maupun dukungan, semangat dan doa.
10. Untuk ketiga adik saya, Naura, Jihan dan Janet. Terimakasih sudah menjadi alasan penulis untuk pulang kerumah.
11. Sahabat tersayangku, Chintia Wati dan Nadhir Rotun Nikmah. Terima kasih banyak telah menjadi motivasi dan pendengar yang baik saat berkeluh kesah, dan telah membantu selama proses penyusunan skripsi ini.
12. Dan yang terakhir, kepada diri saya sendiri, Safhira Aulia Hidayat. Terima kasih karena telah mampu berusaha keras dan berjuang sejauh ini. Mampu mengendalikan diri dari berbagai tekanan diluar keadaan dan tak pernah memutuskan menyerah sesulit apapun proses penyusunan skripsi ini dengan menyelesaikan sebaik dan semaksimal mungkin, ini merupakan pencapaian yang patut dibanggakan untuk diri sendiri.

Bandar Lampung, 19 April 2024
Penulis,

Safhira Aulia Hidayat
NPM. 2017031047

DAFTAR ISI

	Halaman
DAFTAR TABEL	vi
DAFTAR GAMBAR	vii
I. PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Tujuan Penelitian.....	3
1.3 Manfaat Penelitian	4
II. TINJAUAN PUSTAKA	
2.1 Teori Graf.....	5
2.1.1 Definisi Graf.....	5
2.1.2 Jenis-Jenis Graf	7
2.2 Lintasan Terpendek(<i>Shortest Path</i>).....	9
2.3 Algoritma <i>Bellman-Ford</i>	10
2.3.1 Langkah-Langkah Algoritma <i>Bellman-Ford</i>	12
2.3.2 Contoh Penerapan Algoritma <i>Bellman-Ford</i>	13
2.4 Google Maps	20
2.5 Pemrograman Python	20
III. METODE PENELITIAN	
3.1 Waktu dan Tempat Penelitian.....	21
3.2 Data Penelitian	21
3.3 Metode Penelitian.....	21
3.4 Flowchart Algoritma <i>Bellman-Ford</i>	23
IV. HASIL DAN PEMBAHASAN	
4.1 Data Objek Wisata.....	24
4.2 Pemodelan Graf dari Data Objek Wisata	27
4.2.1. Pemodelan Graf 1 (Kantor Camat Kemiling)	27
4.2.2 Pemodelan Graf 2 (Kantor Camat Teluk Betung Barat)	29

4.3 Penerapan Algoritma <i>Bellman-Ford</i>	31
4.3.1 Perhitungan Rute Terpendek pada Graf 1 (Kantor Camat Kemiling)	31
4.3.2 Perhitungan Rute Terpendek pada Graf 2 (Kantor Camat Teluk Betung Barat)	41
4.4 Graf dan Lintasan Lokasi Objek Wisata Kota Bandar Lampung.....	51
4.5 Penerapan Masalah Rute Terpendek Dengan Algoritma <i>Bellman- Ford</i> dalam Bahasa Python	53
V. KESIMPULAN	
5.1 Kesimpulan	58
5.2 Saran.....	59
DAFTAR PUSTAKA.....	60

DAFTAR TABEL

	Halaman
Tabel 2.1 Hasil Contoh Langkah 1	13
Tabel 2.2 Hasil Contoh Langkah 2	16
Tabel 2.3 Hasil Contoh Langkah 3	19
Tabel 4.1 Data Objek Wisata di Kota Bandar Lampung	24
Tabel 4.2 Titik dari Data Objek Wisata di Kota Bandar Lampung Untuk Graf 1	28
Tabel 4.3 Sisi dari Data Objek Wisata di Kota Bandar Lampung Untuk Graf 1	28
Tabel 4.4 Titik dari Data Objek Wisata di Kota Bandar Lampung Untuk Graf 2	29
Tabel 4.5 Sisi dari Data Objek Wisata di Kota Bandar Lampung Untuk Graf 2	30
Tabel 4.6 Hasil Perhitungan Rute Terpendek pada Graf 1 (Kantor Camat Kemiling)	31
Tabel 4.7 Hasil Langkah 1 Graf 1	33
Tabel 4.8 Hasil Langkah 2 Graf 1	40
Tabel 4.9 Hasil Perhitungan Rute Terpendek pada Graf 2 (Kantor Camat Teluk Betung Barat)	41
Tabel 4.10 Hasil Langkah 1 Graf 2	42
Tabel 4.11 Hasil Langkah 2 Graf 2	50
Tabel 4.12 Titik dari Data Objek Wisata Kota Bandar Lampung	51
Tabel 4.13 Lintasan Terpendek dari Titik Awal (v_1) ke Semua Titik	52
Tabel 4.14 Data Objek Wisata Kota Bandar Lampung pada Pemrograman <i>Python</i>	53

DAFTAR GAMBAR

	Halaman
Gambar 2.1. Contoh Graf dengan 6 Titik dan 8 Sisi.....	6
Gambar 2.2 Graf Sederhana.....	7
Gambar 2.3 Graf Tak Sederhana.....	8
Gambar 2.4 Graf Tak Berarah.....	8
Gambar 2.5 Graf Berarah.....	9
Gambar 2.6 Contoh Algoritma <i>Bellman-Ford</i>	13
Gambar 2.7 Hasil Rute Terpendek dengan Algoritma <i>Bellman-Ford</i>	19
Gambar 3.1 Flowchart Algoritma <i>Bellman-Ford</i>	23
Gambar 4.1 Lokasi Objek Wisata Kota Bandar Lampung Berdasarkan Kecamatan.....	25
Gambar 4.2 Lokasi Objek Wisata Kota Bandar Lampung.....	26
Gambar 4.3 Pemodelan Graf 1 Objek Wisata pada Kecamatan Kemiling.....	29
Gambar 4.4 Pemodelan Graf 2 Objek Wisata pada Kecamatan Teluk Betung Barat.....	30
Gambar 4.5 Hasil Rute Terpendek dengan Langkah – Langkah Algoritma <i>Bellman-Ford</i> pada Graf 1.....	40
Gambar 4.6 Hasil Perhitungan Algoritma <i>Bellman-Ford</i> pada Graf 1.....	40
Gambar 4.7 Hasil Rute Terpendek dengan Langkah – Langkah Algoritma <i>Bellman-Ford</i> pada Graf 2.....	50
Gambar 4.8 Hasil Perhitungan Algoritma <i>Bellman-Ford</i> pada Graf 2.....	50
Gambar 4.9 Graf dan Lintasan 14 Titik Objek Wisata Kota Bandar Lampung.....	52
Gambar 4.10 Impor Skrip dari <i>Library</i>	54
Gambar 4.11 Representasi Graf.....	55
Gambar 4.12 Fungsi Menambahkan Sisi pada Graf.....	55

Gambar 4.13 Fungsi Utilitas.....	.55
Gambar 4.14 <i>Syntax</i> yang Dipakai jika Terdapat Bobot Negatif.....	55
Gambar 4.15 Menghitung Total Bobot Rute Terpendek	56
Gambar 4.16 <i>Syntax</i> Untuk Mendapatkan Graf.....	56
Gambar 4.17 Input Data Jarak Objek Wisata Kota bandar Lampung.....	57

I. PENDAHULUAN

1.1 Latar Belakang

Bandar Lampung sebuah kota di Provinsi Lampung, menawarkan beragam topografi yang berpotensi sebagai destinasi pariwisata yang menarik wisatawan baik wisatawan domestik maupun mancanegara. Dengan kekayaan alam yang beragam, mulai dari pantai – pantai yang memukau hingga perbukitan yang hijau, Bandar Lampung memiliki potensi besar untuk pengembangan sektor pariwisata. Keanekaragaman kondisi fisik ini menjadi daya tarik utama bagi wisatawan, dan diharapkan dapat menjadi pertumbuhan ekonomi kota dengan menarik lebih banyak kunjungan, baik dari wisatawan domestik maupun mancanegara.

Berdasarkan penyampaian dari Eva Dwiana selaku Wali Kota Bandar Lampung, perkembangan jumlah kunjungan wisatawan baik wisatawan domestik maupun mancanegara pada tahun 2022 mencapai 1,73 juta orang. Dibandingkan dengan tahun sebelumnya, yakni tahun 2021 yang mencatat sebanyak 810 ribu orang, terdapat peningkatan sebesar 210% dalam jumlah kunjungan wisatawan ke Bandar Lampung pada tahun 2022 (Rostanti, 2023).

Dalam kehidupan sehari-hari, seringkali kita melakukan perjalanan dari satu lokasi ke lokasi lainnya. Saat hendak menuju tempat wisata di Kota Bandar Lampung, terdapat beberapa rute yang dipilih. Para wisatawan tentunya menginginkan rute yang paling efisien agar dapat menghemat waktu dan biaya perjalanan. Untuk mencari rute yang paling efisien ini, wisatawan perlu membuat keputusan tentang rute mana yang akan mereka ambil dari sekian banyak pilihan yang tersedia. Dalam konteks matematika, untuk memecahkan masalah pengambilan keputusan

semacam ini, salah satu Teknik yang digunakan adalah dengan menggunakan algoritma *Bellman-Ford*, yang dikenal sebagai algoritma *shortest path* (rute terpendek).

Menurut Serdano, et al. (2019), bahwa algoritma *Bellman-Ford* adalah sistem algoritma yang digunakan untuk mencari jalur terpendek dalam suatu jaringan, mirip dengan fungsi algoritma *Dijkstra*. Namun, keunggulan algoritma *Bellman-Ford* terletak pada kemampuannya untuk mengakomodasi bobot negatif dalam perhitungan jarak terpendek pada graf yang memiliki bobot. Algoritma ini akan memberikan hasil yang akurat asalkan dalam graf tersebut tidak terdapat siklus yang memiliki bobot negatif yang berasal dari titik asal. Dibandingkan dengan algoritma *Dijkstra*, algoritma *Bellman-Ford* memiliki kompleksitas waktu yang lebih besar. Adapun kompleksitas waktu untuk algoritma *Dijkstra* dicatat sebagai $O(|E| + |V| \log|V|)$, menunjukkan efisiensi yang lebih tinggi dibandingkan dengan algoritma *Bellman-Ford* dalam konteks waktu pemrosesan.

Berdasarkan penelitian dari Moch Abdul Aziz (2023), yang berjudul implementasi algoritma *Dijkstra* dalam menentukan rute terpendek terhadap destinasi wisata Kabupaten Bojonegoro berhasil mengimplementasikan algoritma *Dijkstra* dalam permasalahan pencarian rute terpendek destinasi wisata Kabupaten Bojonegoro.

Berdasarkan penelitian dari Rozzi Kesuma Dinata, et al. (2022), yang berjudul algoritma *Dijkstra* dan algoritma *Bellman-Ford* dalam sistem pemetaan *Barbershop* di Kota Lhokseumawe berhasil mengimplementasikan algoritma *Dijkstra* dan algoritma *Bellman-Ford* untuk melakukan pencarian rute terdekat berdasarkan jarak yang ditempuh terhadap *Barbershop* di Kota Lhokseumawe.

Berdasarkan penelitian dari Basriati, et al. (2022), yang berjudul implementasi algoritma *Bellman-Ford* dalam menentukan lintasan terpendek truk pembuangan sampah berhasil mengimplementasikan algoritma *Bellman-Ford* untuk melakukan pencarian rute terdekat berdasarkan jarak yang ditempuh pada truk pembuangan sampah.

Berdasarkan penelitian dari Sulaiman, et al. (2018), yang berjudul *Bellman Ford algorithm In Routing Information Protocol (RIP)* berhasil mengimplementasikan algoritma *Bellman-Ford* pada *Routing Information Protocol (RIP)* ke dalam sebuah sistem telekomunikasi.

Berdasarkan penelitian dari Hutasoit (2019), yang berjudul pencarian rute terpendek menggunakan algoritma *Bellman-Ford* berhasil mengimplementasikan algoritma *Bellman-Ford* untuk melakukan pencarian rute terdekat berdasarkan jarak yang ditempuh ketika mengantar barang kepada konsumen dengan studi kasus yang digunakan yaitu PT. JNE.

Dari beberapa penelitian diatas dapat disimpulkan bahwa baik algoritma *Bellman-Ford* atau algoritma *Dijkstra* sama-sama dapat menyelesaikan masalah rute terpendek. Akan tetapi, algoritma *Bellman-Ford* mampu menangani bobot positif dan negatif pada pencarian jarak terpendek pada graf berbobot. Sedangkan, algoritma *Dijkstra* hanya mampu menangani bobot positif pada pencarian jarak terpendek pada graf berbobot.

Dengan demikian, penelitian ini bertujuan untuk menganalisis rute dan jarak terpendek guna menentukan rute terbaik bagi wisatawan yang ingin mengunjungi lokasi wisata di Kota Bandar Lampung, menggunakan algoritma *Bellman-Ford*. Penelitian ini akan mengimplementasikan skrip program berbasis algoritma *Bellman-Ford* untuk memudahkan proses iterasi dalam mengidentifikasi solusi rute terpendek, dibandingkan dengan metode perhitungan manual antar titik referensi tanpa dukungan skrip program.

1.2 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk mengetahui rute terpendek objek wisata Kota Bandar Lampung dengan menggunakan algoritma *Bellman-Ford*.

1.3 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah:

1. Sebagai sarana untuk memperdalam pengetahuan mengenai penerapan metode algoritma *Bellman-Ford* untuk menentukan rute terpendek objek wisata kota Bandar Lampung.
2. Sebagai referensi untuk menambah wawasan dan pemahaman bagi para pembaca mengenai penerapan algoritma *Bellman-Ford*.
3. Dapat dimanfaatkan oleh wisatawan sebagai rekomendasi jalur terbaik untuk mempersingkat waktu dan menghemat biaya saat akan berkunjung ke objek wisata Kota Bandar Lampung.

II. TINJAUAN PUSTAKA

2.1 Teori Graf

Menurut Abdussakir, (2009), graf adalah kumpulan simpul (*node*) yang dihubungkan satu sama lain melalui sisi/busur (*edges*). Suatu graf G terdiri dari dua himpunan yaitu himpunan V dan himpunan E . V (simpul) yaitu himpunan simpul yang terbatas dan tidak kosong, E (sisi/busur) yaitu himpunan busur yang menghubungkan sepasang simpul. Dengan demikian, graf dapat dijelaskan sebagai koleksi dari simpul yang dihubungkan oleh sisi. Sebagai contoh, peta merupakan salah satu bentuk representasi visual dari graf. Representasi ini membuka peluang untuk berbagai analisis, seperti mencari jalur terpendek antar tempat, mewarnai dua kota tetangga dengan warna yang berbeda pada peta, merancang tata letak jalur transportasi, mengelola jaringan telekomunikasi atau internet, dan masih banyak lagi. Di luar peta, terdapat berbagai aspek dalam kehidupan nyata yang dapat direpresentasikan sebagai graf.

2.1.1 Definisi Graf

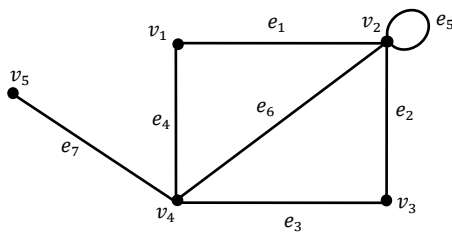
Menurut Munir, (2012), Graf G didefinisikan sebagai pasangan himpunan (V, E) ditulis dengan notasi $G = (V, E)$ yang dalam hal ini V adalah himpunan tidak kosong dari titik-titik (*vertices* atau *node*) dan E adalah himpunan sisi (*edges* atau *arcs*) yang menghubungkan sepasang titik.

Definisi graf G menyatakan bahwa V tidak boleh kosong, sedangkan E boleh kosong. Jadi, sebuah graf dimungkinkan tidak mempunyai sisi satu buah pun, tetapi titiknya harus ada minimal satu. Graf yang hanya mempunyai satu buah titik tanpa

sebuah sisi dinamakan graf trivial. Secara geometri graf digambarkan sebagai sekumpulan noktah (titik) di dalam bidang dwimatra yang dihubungkan dengan sekumpulan garis (sisi).

Titik pada graf dapat diberi nama dengan huruf, seperti a, b, c, \dots atau dengan bilangan asli $1, 2, 3, \dots$ atau gabungan keduanya. Sedangkan sisi yang menghubungkan titik v_i dengan v_j dinyatakan dengan pasangan (v_i, v_j) atau dinyatakan dengan lambing e_1, e_2, e_3, \dots . Dengan kata lain, jika e adalah sisi yang menghubungkan titik v_i dengan titik v_j , maka e dapat ditulis sebagai $e = (v_i, v_j)$.

Suatu graf $G = (V, E)$ terdiri dari himpunan tak kosong titik $V = \{v_1, v_2, v_3, \dots, v_n\}$ dan himpunan sisi $E = \{e_1 = v_1v_2, e_2 = v_3v_4, \dots, e_{n-1} = v_{n-1}v_n\}$. Banyaknya titik dari graf $G = (V, E)$ disebut *order*, yang dinotasikan dengan $|V|$. Sedangkan banyaknya sisi dari graf $G = (V, E)$ disebut *size*, yang dinotasikan dengan $|E|$.



Gambar 2.1 Contoh Graf dengan 5 Titik dan 7 Sisi

Pada Gambar 2.1, graf tersebut merupakan graf (V, E) dengan himpunan titik $V(G) = \{v_1, v_2, v_3, v_4, v_5\}$ dan himpunan sisi $E(G) = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$. Titik yang bertetangga dengan v_1 adalah titik v_2 dan v_4 sedangkan titik v_1 dan v_2 menempel dengan e_1 . Sebaliknya, sisi e_1 menempel pada titik v_1 dan v_2 .

Derajat (*degree*) suatu titik v pada graf G yaitu banyaknya sisi yang menempel pada titik v yang dinotasikan dengan $d(v)$, pada Gambar 2.1 $d(v_1) = 2, d(v_2) = 4, d(v_3) = 2, d(v_4) = 4$. Daun (*pendant*) adalah titik yang berderajat satu yaitu v_5 . Graf teratur yaitu graf yang semua titiknya berderajat sama.

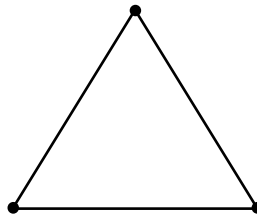
Menurut Deo (1989), *loop* adalah sebuah sisi yang mempunyai titik awal dan titik akhir yang sama, sedangkan sisi paralel adalah dua sisi atau lebih yang menghubungkan sepasang titik yang sama. Pada Gambar 2.1, sisi-sisi paralel nya tidak ada sedangkan e_5 merupakan *loop*. Jalan (*walk*) adalah barisan berhingga titik-titik dan sisi sedemikian sehingga setiap sisi menempel pada titik sebelum dan sesudahnya. Pada jalan (*walk*) boleh terjadi pengulangan titik atau sisi. Contoh jalan pada Gambar 2.1 adalah $v_1 - e_1 - v_2 - e_5 - v_2 - e_2 - v_3 - e_3 - v_4 - e_7 - v_5$. Lintasan (*Path*) adalah jalan yang semua titik yang dilewati berbeda, jika titik awal dan akhirnya sama maka disebut lintasan tertutup. Contoh lintasan pada Gambar 2.1 adalah $v_1 - e_1 - v_2 - e_2 - v_3 - e_3 - v_4 - e_7 - v_5$.

2.1.2 Jenis-Jenis Graf

Berdasarkan ada tidaknya sisi ganda (paralel) atau sisi gelang (*loop*) pada suatu graf, maka secara umum graf dapat dibedakan menjadi dua jenis, yaitu (Munir, 2012):

1. Graf Sederhana (*Simple Graph*)

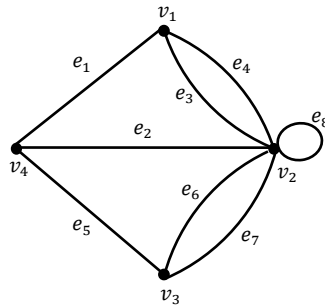
Graf sederhana adalah graf yang tidak mengandung sisi ganda (paralel) maupun sisi gelang (*loop*). Contoh dari graf sederhana dapat dilihat pada Gambar 2.2.



Gambar 2.2 Graf Sederhana

2. Graf Tak Sederhana (*Unsimple Graph*)

Graf tak sederhana adalah graf yang mengandung sisi ganda (paralel) atau sisi gelang (*loop*). Graf tak sederhana memiliki dua macam, yaitu graf ganda (*multigraph*) dan graf semu (*pseudograph*). Contoh dari graf tak sederhana dapat dilihat pada Gambar 2.3.



Gambar 2.3 Graf Tak Sederhana

Berdasarkan orientasi arah pada sisinya, maka secara umum graf dapat dibedakan menjadi dua jenis, yaitu (Munir, 2012):

1. Graf Tak Berarah (*Undirected Graph*)

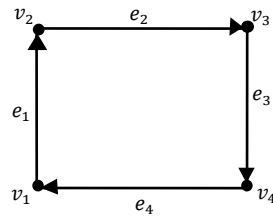
Graf tak berarah adalah graf yang sisinya tidak mempunyai orientasi arah. Pada graf tak berarah, urutan pasangan titik yang dihubungkan oleh sisi tidak diperhatikan. Jadi, $(u, v) = (v, u)$ adalah sisi yang sama. Berikut contoh graf tak berarah:



Gambar 2.4 Graf Tak Berarah

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

Graf berarah adalah graf yang setiap sisinya diberikan orientasi arah. Berikut contoh graf berarah:



Gambar 2.5 Graf Berarah

2.2 Lintasan Terpendek (*Shortest Path*)

Menurut Djafar dan Faizal (2015), lintasan terpendek adalah lintasan minimum yang harus ditempuh untuk mencapai sebuah destinasi dari suatu titik awal. Untuk menemukan jalur minimum ini, dapat digunakan konsep graf, khususnya graf berbobot, di mana setiap sisi graf diberikan nilai atau bobot tertentu. Dalam konteks penyelesaian masalah jalur terpendek, yang sering disebut dengan algoritma pencarian jalur, terdapat berbagai algoritma yang bisa digunakan, termasuk algoritma *Bellman-Ford*. Pada penelitian ini diasumsikan bahwa semua bobot bernilai positif, bobot yang dimaksud berupa jarak antar objek wisata.

Ada beberapa macam persoalan lintasan terpendek, antara lain:

- a. Lintasan terpendek antara dua buah titik tertentu.
- b. Lintasan terpendek antara semua pasangan titik.
- c. Lintasan terpendek dari satu titik ke semua titik lainnya.
- d. Lintasan terpendek antara dua titik yang melewati beberapa titik tertentu.

Fungsi kendala pada algoritma *Bellman-Ford* yang diterapkan adalah untuk memastikan bahwa jarak terpendek dari titik awal ke semua titik lainnya ditemukan dengan benar, serta menghindari terbentuknya siklus negatif. Fungsi kendala yang berlaku dalam algoritma *Bellman-Ford* sebagai berikut:

1. Kendala jarak terpendek

Memastikan bahwa jarak terpendek dari titik awal ke setiap titik lainnya ditemukan dengan benar. Dirumuskan dalam bentuk:

$$d[v] \leq d[u] + w(u, v)$$

Keterangan:

$d[v]$: Jarak terpendek dari titik awal ke titik v

$d[u]$: Jarak terpendek dari titik awal ke titik u

$w(u, v)$: Bobot dari tepi yang menghubungkan simpul u dan v

2. Kendala siklus negatif

Jika terbentuk siklus negatif, algoritma tidak akan konvergen dan hasilnya tidak akan valid.

Fungsi objektif pada algoritma *Bellman-Ford* yaitu untuk menentukan jalur terpendek dari titik awal ke semua titik lainnya dalam graf. Oleh karena itu, fungsi objektif pada algoritma *Bellman-Ford* yaitu untuk meminimalkan total bobot dari jalur yang dipilih dari titik awal ke semua titik lainnya.

Secara matematis, jika $d[v]$ sebagai jarak terpendek dari titik awal ke titik v , dan $w(u, v)$ adalah bobot dari tepi yang menghubungkan simpul u dan v , maka fungsi objektif dari algoritma *Bellman-Ford* dapat dirumuskan sebagai berikut:

$$\text{Minimize } \sum_{v \in V} d[v]$$

Keterangan:

V : Himpunan semua titik dalam graf

v : Titik v

$d[v]$: Jarak terpendek dari titik awal ke titik v

2.3 Algoritma *Bellman-Ford*

Menurut Cormen (2009), bahwa algoritma *Bellman-Ford*, yang diciptakan oleh Richard Bellman dan Lester Ford, dirancang untuk menghitung jarak terpendek dari sebuah titik sumber ke semua titik lain dalam graf yang memiliki bobot. Algoritma ini menentukan jarak terpendek dengan memulai dari satu titik dan menghitung jarak ke titik lainnya. Dalam algoritma ini, $d[u]$ sebagai batas atas dengan jarak $d[u, v]$ dari u ke v . Algoritma ini memulai dengan mengatur jarak dari titik asal ke nol dan jarak ke titik lain sebagai tak terhingga. Kemudian, secara bertahap

algoritma memperbaiki estimasi jarak ini untuk setiap titik dari titik sumber ke titik v di dalam V , sampai semua jalur terpendek ditemukan atau hingga terbukti bahwa graf mengandung siklus berbobot negatif, yang ditandai dengan kondisi Boolean TRUE (jika siklus berbobot negatif tidak ada dan semua titik bisa dijangkau dari titik sumber) atau FALSE (dalam kasus lain). Algoritma *Bellman-Ford* sebagai berikut:

```

BELLMAN-FORD ( $G, w, s$ )
  INITIALIZE-SINGLE-SOURCE ( $G, s$ )
  for each vertex  $i = 1$  to  $V[G] - 1$  do
    for each edge  $(u, v)$  in  $E[G]$  do
      RELAX ( $u, v, w$ )
  for each edge  $(u, v)$  in  $E[G]$  do
    if  $d[u] + w(u, v) < d[v]$  then
      return FALSE
  return TRUE

```

Menurut Serdano, et al. (2019), menjelaskan bahwa algoritma *Bellman-Ford* dalam menemukan rute terpendek dalam graf, bahkan ketika ada sisi dengan bobot negatif. Berikut ciri-ciri algoritma *Bellman-Ford*:

1. Kemampuan untuk mengatasi masalah pencarian rute terpendek dalam graf yang mengandung sisi yang bernilai negatif.
2. Kebutuhan akan graf yang memiliki arah.
3. Keahlian dalam mengidentifikasi keberadaan siklus yang memiliki bobot negatif di dalam graf jika ada.

Algoritma *Bellman-Ford* memiliki beberapa syarat yang membatasi kinerjanya yaitu sebagai berikut:

1. Graf terhubung yaitu algoritma *Bellman-Ford* hanya berlaku untuk graf yang terhubung, graf yang tidak terhubung akan menghasilkan hasil yang tidak valid karena tidak mungkin mencapai semua simpul.
2. Tidak ada siklus negatif yaitu algoritma *Bellman-Ford* memastikan bahwa tidak ada siklus negatif dalam graf, karena siklus negatif akan membuat algoritma

Bellman-Ford tidak konvergen, sehingga tidak dapat menghasilkan solusi yang benar.

3. Bobot negatif yaitu algoritma *Bellman-Ford* dapat menangani grafik dengan bobot negatif. Namun jika terdapat siklus negatif, algoritma *Bellman-Ford* tidak akan konvergen dan tidak akan memberikan hasil yang valid. Bobot negatif bisa berupa biaya pengiriman pada jalur yang menghasilkan keuntungan atau diskon, waktu tempuh jika ada jalur alternatif yang lebih cepat bobotnya akan menjadi negatif.
4. Kompleksitas waktu yaitu algoritma *Bellman-Ford* memiliki kompleksitas waktu $O(VE)$, Dimana V adalah jumlah titik dan E adalah jumlah sisi. Jika jumlah titik dan sisi dalam graf sangat besar, maka tidak efisien. Algoritma *Bellman-Ford* lebih cocok untuk graf dengan ukuran sedang atau kecil.

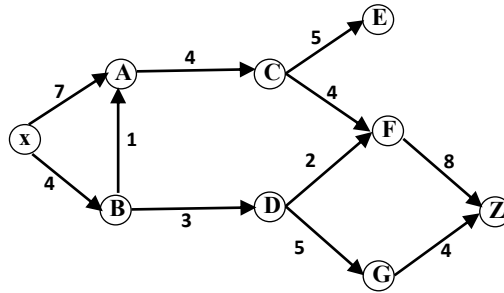
2.3.1 Langkah-Langkah Algoritma *Bellman-Ford*

Menurut Azdy dan Darnis (2019), berikut langkah-langkah algoritma *Bellman-Ford* yaitu sebagai berikut:

1. Menentukan titik asal dan mendaftar semua titik maupun sisi.
2. Memberi nilai untuk titik asal sama dengan nol dan titik lainnya dengan tak hingga.
3. Memulai iterasi dari titik asal. berlanjut dengan meninjau setiap titik yang terkait. Gunakan formula seperti berikut:
 U merupakan titik asal
 V merupakan titik tujuan
 UV merupakan sisi yang menghubungkan U dan V ,
 Jika jarak V lebih besar dari jarak $U + \text{bobot } UV$ maka jarak V diisi dengan jarak $U + \text{bobot } UV$, dilakukan hingga semua titik terjelajahi.
4. Melakukan iterasi untuk semua sisi yang ada untuk mengecek apakah ada siklus negatif dalam graf tersebut, kemudian melakukan pengecekan. Untuk semua sisi UV , jika jarak V lebih besar dari jarak $U + \text{bobot } UV$ maka sudah jelas bahwa graf tersebut memiliki siklus negatif.

2.3.2 Contoh Penerapan Algoritma *Bellman-Ford*

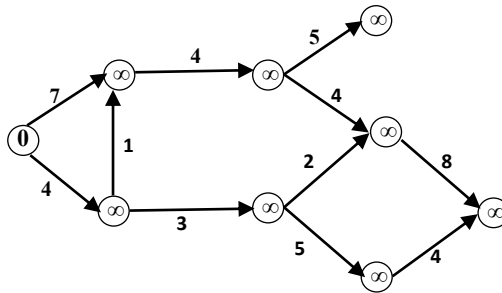
Seseorang berada di lokasi *X* ingin menuju ke lokasi *Z*. Tentukan rute yang paling dekat menurut algoritma *Bellman-Ford*.



Gambar 2.6 Contoh Algoritma *Bellman-Ford*

Langkah 1

Buat titik awal = 0 dan titik lainnya dengan nilai tak hingga.



Tabel 2.1 Hasil Contoh Langkah 1

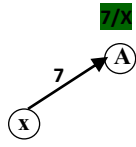
	X	A	B	C	D	E	F	G	Z
d[V]	0	∞	∞	∞	∞	∞	∞	∞	∞

Langkah 2

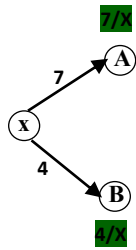
Hitung semua titik.

1. Titik *X* = 0.
2. Titik *A* = 7 melewati titik *X*.

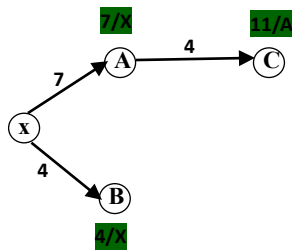
$$X + \text{bobot } XA = 0 + 7 = 7.$$



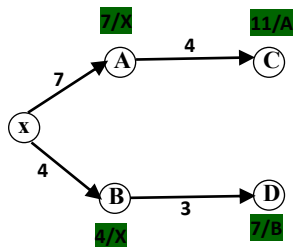
3. Titik $B = 4$ melewati titik X .
 $X + \text{bobot } XB = 0 + 4 = 4$.



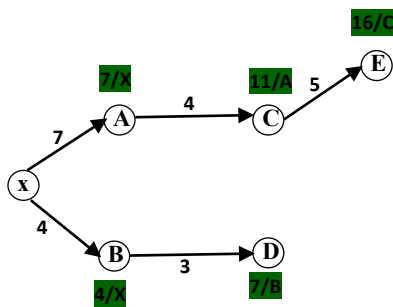
4. Titik $C = 11$ melewati titik A .
 $A + \text{bobot } AC = 7 + 4 = 11$.



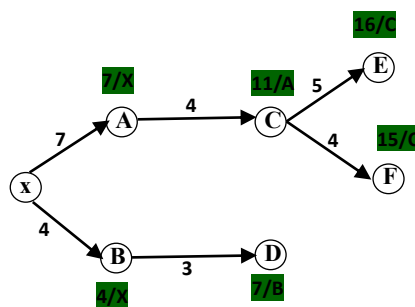
5. Titik $D = 7$ melewati titik B .
 $B + \text{bobot } BD = 4 + 3 = 7$.



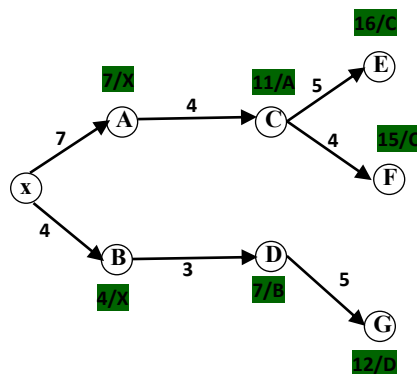
6. Titik $E = 16$ melewati titik C .
 $C + \text{bobot } CE = 11 + 5 = 16$.



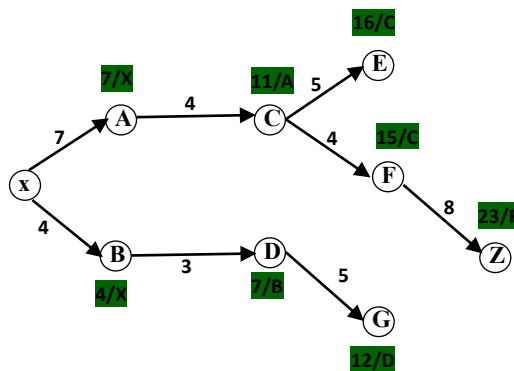
7. Titik $F = 15$ melewati titik C .
 $C + \text{bobot } CF = 11 + 4 = 15$.



8. Titik $G = 12$ melewati titik D .
 $D + \text{bobot } DG = 7 + 5 = 12$.



9. Titik $Z = 23$ melewati titik F .
 $F + \text{bobot } FZ = 15 + 8 = 23$.

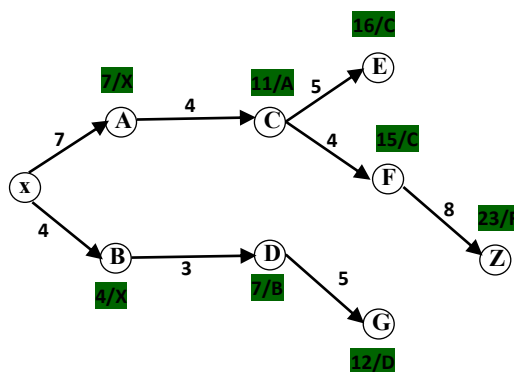


Tabel 2.2 Hasil Contoh Langkah 2

	X	A	B	C	D	E	F	G	Z
d[V]	0	7	4	11	7	16	15	12	23
Melalui Titik	0	X	X	A	B	C	C	D	F

Langkah 3

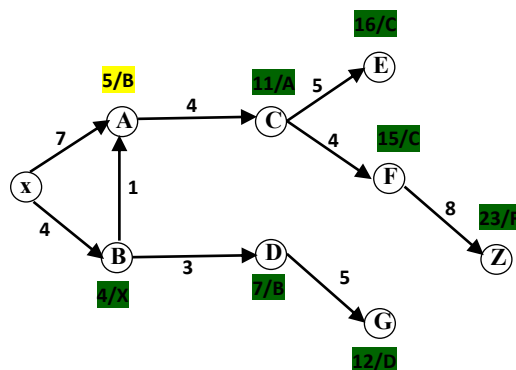
Hitung kembali semua titik yang belum terlewati.



1. Titik $X = 0$.
2. Titik $A = 5$ melewati titik B .

$$B + \text{bobot } BA = 4 + 1 = 5.$$

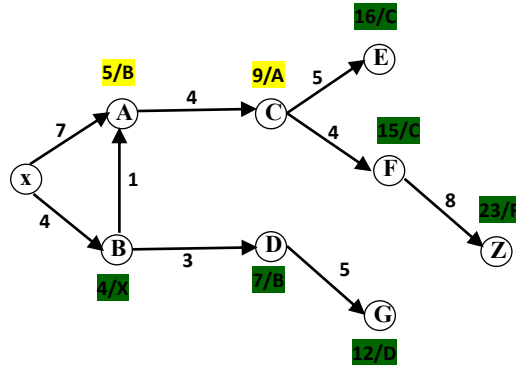
$7 > 5$ maka dipilih 5 sebagai jarak terpendek antara X dan A .



3. Titik $C = 9$ melewati titik A .

$$A + \text{bobot } AC = 5 + 4 = 9.$$

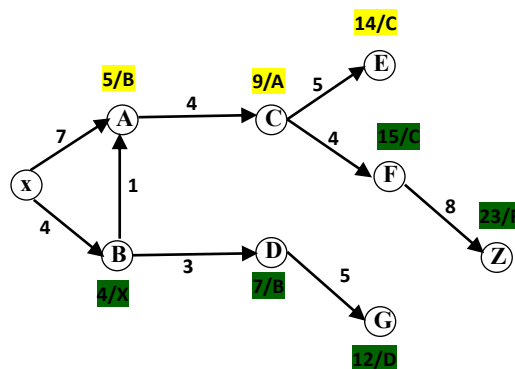
$11 > 9$ maka dipilih 9 sebagai jarak terpendek antara X dan C .



4. Titik $E = 14$ melewati titik C .

$$C + \text{bobot } CE = 9 + 5 = 14.$$

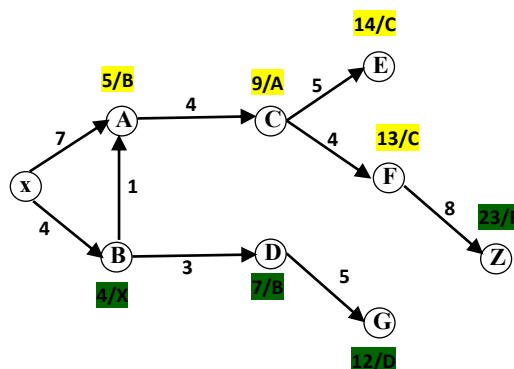
$16 > 14$ maka dipilih 14 sebagai jarak terpendek antara X dan E .



5. Titik $F = 13$ melewati titik C .

$$C + \text{bobot } CF = 9 + 4 = 13.$$

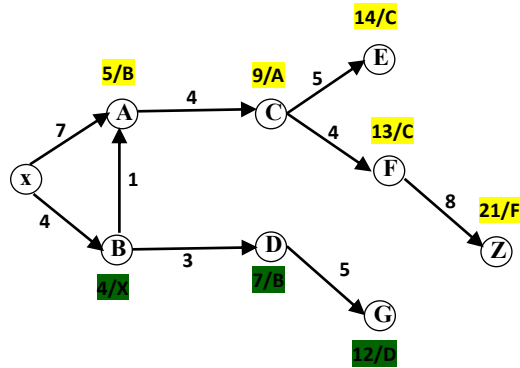
$15 > 13$ maka dipilih 13 sebagai jarak terpendek antara X dan F .



6. Titik $Z = 21$ melewati titik F .

$$F + \text{bobot } FZ = 13 + 8 = 21.$$

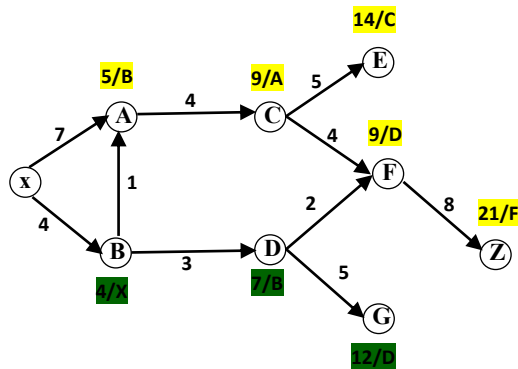
$23 > 21$ maka dipilih 21 sebagai jarak terpendek antara X dan Z .



7. Titik $F = 9$ melewati titik D .

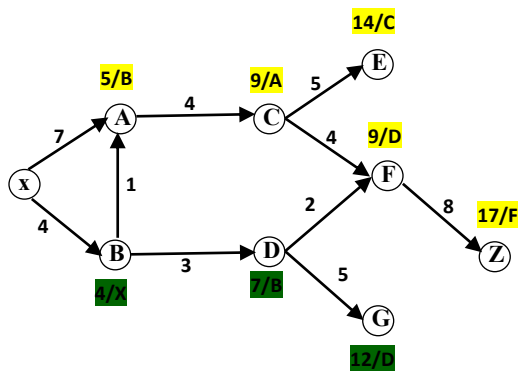
$$D + \text{bobot } DF = 7 + 2 = 9.$$

$13 > 9$ maka dipilih 9 sebagai jarak terpendek antara X dan F .



8. Titik $Z = 17$ melewati titik F .

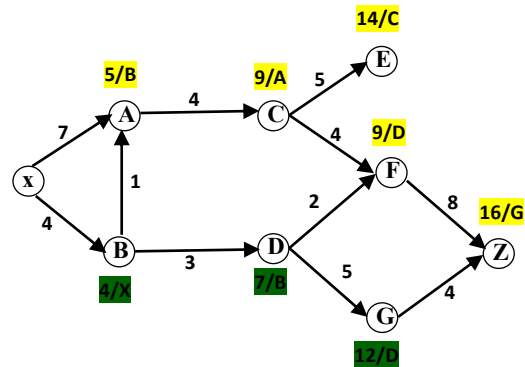
$$F + \text{bobot } FZ = 9 + 8 = 17.$$



Jika melewati titik G , maka $Z = 16$.

$G + \text{bobot } GZ = 12 + 4 = 16$.

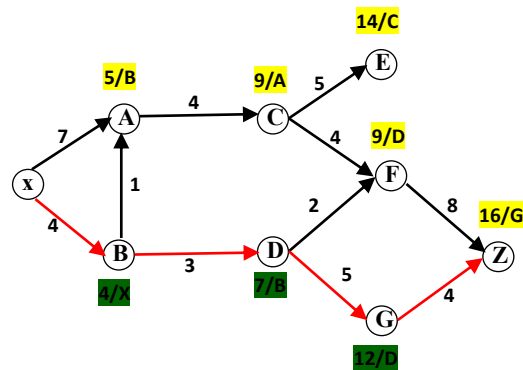
$17 > 16$ maka dipilih 16 sebagai jarak terpendek antara X dan Z .



Tabel 2.3 Hasil Contoh Langkah 3

	X	A	B	C	D	E	F	G	Z
$d[V]$	0	5	4	9	7	14	9	12	16
Melalui titik	0	B	X	C	B	C	D	D	G

Tampilan hasil yang telah di uji dengan menggunakan algoritma *Bellman-Ford* untuk mendapatkan rute yang terpendek dapat dilihat seperti pada Gambar 2.7.



Gambar 2.7 Hasil Rute Terpendek dengan Algoritma *Bellman-Ford*

Keterangan gambar:

- Merupakan nilai awal
- Merupakan nilai yang berubah setelah menghitung kembali titik belum terlewati.
- Merupakan rute terpendek yang akan dilalui.

Dari gambar nomor 8 di atas maka jarak orang yang berada di lokasi X menuju lokasi Z dengan rute $X - B - D - G - Z$.

2.4 *Google Maps*

Menurut Muslim dan Sunyoto (2012), *google maps* adalah layanan peta virtual gratis yang disediakan oleh *Google*. Layanan ini menyediakan peta yang dapat digeser serta gambar satelit untuk seluruh dunia. *Google Maps* juga berfungsi sebagai aplikasi pemandu perjalanan yang memberikan informasi penting tentang suatu lokasi. Dalam konteks penelitian ini, *Google Maps* digunakan sebagai gambaran peta lokasi. Dengan kata lain, peran *Google Maps* dalam penelitian ini adalah untuk membantu menentukan jarak antara objek wisata yang terletak di Kota Bandar Lampung.

2.5 *Pemrograman Python*

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. *Python* diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. *Python* dirancang untuk memberikan kemudahan bagi *programmer* melalui segi efisiensi waktu, kemudahan dalam pengembangan dan kompatibilitas dengan sistem. *Python* bisa digunakan untuk membuat aplikasi *standalone* (berdiri sendiri) dan pemrograman *script* (*scripting programming*).

III. METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian ini dilakukan pada semester ganjil tahun akademik 2023/2024 di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

3.2 Data Penelitian

Data yang dipergunakan mencakup jarak antara objek wisata yang ada di Kota Bandar Lampung, diperoleh dari Dinas Kebudayaan dan Pariwisata Kota Bandar Lampung. Informasi mengenai jarak titik keberangkatan awal menuju berbagai destinasi wisata dengan menggunakan bantuan *google maps*.

3.3 Metode Penelitian

Adapun metode penelitian yang digunakan dalam penyusunan proposal penelitian ini adalah:

1. Mempelajari prinsip – prinsip dasar teori graf.
2. Mempelajari penerapan algoritma *Bellman-Ford* untuk menentukan rute terpendek.
3. Melakukan analisis kasus terhadap data lokasi objek wisata di Kota Bandar Lampung.
4. Menyusun permasalahan dalam bentuk graf berarah dan berbobot.
5. Menggunakan algoritma *Bellman-Ford* untuk menentukan rute terpendek objek wisata di Kota Bandar Lampung, dengan langkah - langkah yaitu:

- a. Menentukan titik keberangkatan dan mendaftarkan semua titik maupun sisi.
- b. Memberikan nilai titik keberangkatan sama dengan nol dan titik-titik lainnya dengan tak hingga.
- c. Memulai iterasi ke semua titik yang ada, dimulai melalui titik keberangkatan. Untuk menentukan jarak semua titik terhadap titik keberangkatan yaitu dengan formula:

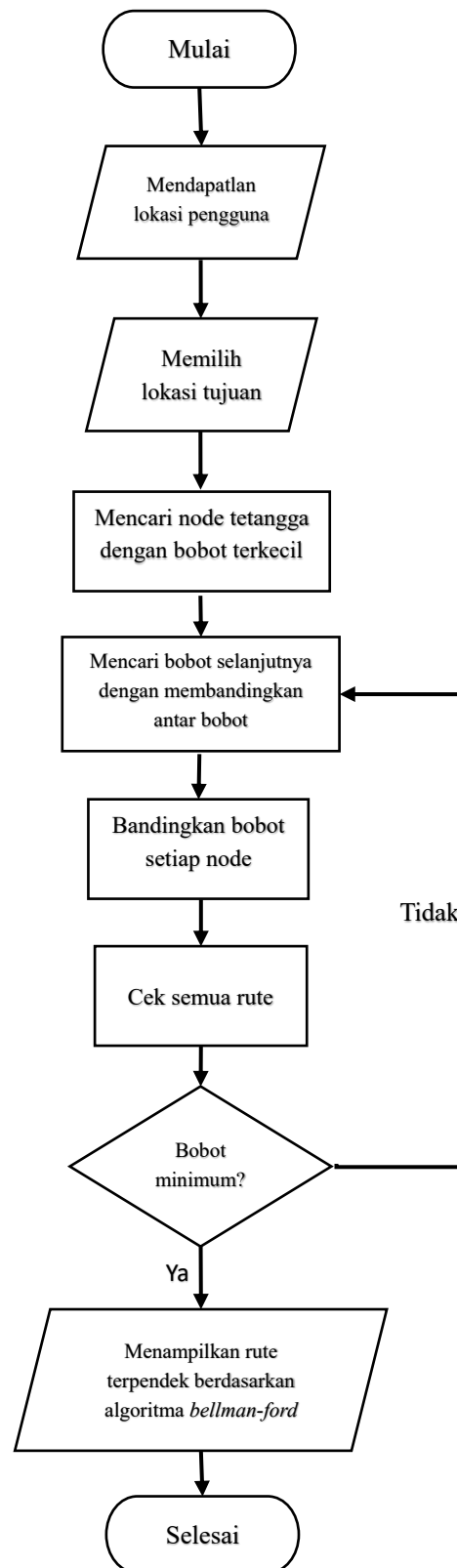
U = Titik asal/ Titik keberangkatan.

V = Titik tujuan/Objek wisata.

UV = Sisi yang menghubungkan U dengan V .

Jika jarak V lebih besar dari jarak $U + \text{bobot } UV$ maka jarak V diisi dengan jarak $U + \text{bobot } UV$, dan dilakukan sampai semua titik dijelajahi.

3.4 Flowchart Algoritma *Bellman-Ford*



Gambar 3.1 Flowchart Algoritma *Bellman-Ford*

V. KESIMPULAN

5.1 Kesimpulan

Pada penelitian ini menunjukkan bahwa algoritma *Bellman-Ford* efektif dalam menentukan rute terpendek menuju objek wisata Kota Bandar Lampung. Dalam penelitian ini, perhitungan rute terpendek dengan menggunakan algoritma *Bellman-Ford* dilakukan melalui dua cara yaitu secara manual dan secara komputerisasi dengan bahasa pemrograman *python* pada objek wisata di Kota Bandar Lampung. Direpresentasikan ke dalam model graf secara manual dengan 14 objek wisata Kota Bandar Lampung dan secara komputerisasi dengan bahasa pemrograman *python* dengan 55 objek wisata Kota Bandar Lampung.

Secara manual didapat rute terpendek objek wisata Kota Bandar Lampung dari Kantor Camat Kemiling menuju objek wisata Pintu Langit 2, melewati objek wisata Taman Agrowisata Kemiling, objek wisata Tebing Vietnam, objek wisata Kampung Vietnam, objek wisata Puncak Mas dan objek wisata Kampung Pelangi Umbul kapuk yaitu 20,72 km. Secara komputerisasi dengan bahasa pemrograman *python* didapat rute terpendek objek wisata Kota Bandar Lampung dari Kecamatan Teluk Betung Utara menuju objek wisata Rumah Adat Jajar Intan, melewati objek wisata Wira Garden, objek wisata Taman Wisata Bumi Kedaton, objek wisata Gunung Kunyit, objek wisata Kampung De Brow, objek wisata Makam Muhammad Al-Atas, objek wisata Jembatan Beton Way Balau, objek wisata Hutan Raya Wan Abdurahman, objek wisata Taman Kupu – Kupu, objek wisata Tebing Vietnam, objek wisata Kampung Vietnam, objek wisata Taman Hutan Kera Gunung Banten,

objek wisata Kampung Edukasi D'lima dan objek wisata Kedatun Keagungan yaitu 58,12 km.

5.2 Saran

Berdasarkan hasil dan pembahasan pada penelitian ini, disarankan untuk penelitian selanjutnya mengintegrasikan faktor – faktor tambahan seperti waktu perjalanan atau biaya yang dikeluarkan untuk mengunjungi objek wisata. Selain itu, hasil penelitian ini diharapkan dapat memberikan masukan berharga bagi pemerintah Kota Bandar Lampung untuk mempertimbangkan pengembangan fasilitas transportasi umum yang dapat meningkatkan aksesibilitas bagi wisatawan saat mengunjungi objek wisata.

DAFTAR PUSTAKA

- Abdussakir. (2009). Teori Graf Topik Dasar Untuk Tugas Akhir dan Skripsi. *UIN Malang Press*. Malang.
- Azdy, R.A., Darnis, F. (2019). Implementasi *Bellman-Ford* untuk Optimasi Rute Pengambilan Sampah di Kota Palembang. *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*. 8(4): 327-333.
- Aziz, M. A. (2023). Implementasi Algoritma *Dijkstra* dalam Menentukan Rute Terpendek Terhadap Destinasi Wisata Kabupaten Bojonegoro. *Prosiding Lomba dan Seminar Pendidikan Matematika, Matematika, dan Statistik*. 7(1): 1-8.
- Basriati, S., Safitri, E., Yesti, S. A., Andiraja, N. (2022). Implementasi Algoritma *Bellman-Ford* dalam Menentukan Lintasan Terpendek Truk Pembuangan Sampah. *Seminar Nasional Teknologi Informasi, Komunikasi dan Industri (SNTIKI)*. Riau.
- Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C. (2009). *Introduction to Algorithms. 3rd Edition*. The MIT (Massachusetts Institute of Technology) Press, United State of America.
- Dinata, R. K., Bustami, Razi, A., Arasyi, M. (2022). Algoritma *Dijkstra* dan Algoritma *Bellman-Ford* dalam Sistem Pemetaan Barbershop di Kota Lhokseumawe. *Informatics Journal*. 7(2): 128-137.
- Djafar, I., Faizal. (2015). Single-Source Shortest Path pada Graf Berbobot Menggunakan Algoritma *Dijkstra* dan *Bellman-Ford*. *Prosiding Seminar Ilmiah Sistem Informasi dan Teknologi Informasi*. Makassar. hlm. 172-181
- Hutasoit, E.T. (2019). Pencarian Rute Terpendek Menggunakan Algoritma *Bellman-Ford* (Studi Kasus: PT. JNE Medan). *Jurnal Sistem Komputer dan Informatika (JSON)*. 1(1): 20-25.

- Munir, R. (2012). *Matematika Diskrit*. Revisi Kelima. Informatika Bandung, Bandung.
- Muslim, N., Sunyoto, A. (2012). Sistem Informasi Geografis Berbasis Web Pemetaan Potensi Panas Bumi di Indonesia Menggunakan Google Maps. *Jurnal Data Manajemen dan Teknologi Informasi (DASI)*. 13(2): 60-64.
- Rostanti, Q. (2023). Bandar Lampung Optimistis Capai Target Dua Juta Kunjungan Wisatawan. <https://news.republika.co.id/berita/rw549o425/bandar-lampung-optimistis-capai-target-dua-juta-kunjungan-wisatawan>. Diakses Pada 18 November 2023, 17.34.
- Serdano, A., Zarlis, M., Dedy, H. (2019). Perbandingan Algoritma *Dijkstra* dan *Bellman-Ford* dalam Pencarian Jarak Terpendek pada SPBU. *Prosiding Seminar Nasional Sains & Teknologi Informasi (SENSASI)*, Medan. hlm. 259-264.
- Sulaiman, O. K., Siregar, A. M., Nasution, K., Haramaini, T. (2018). Bellman Ford algorithm-In Routing Information Protocol (RIP). *Journal of Physics: Conference Series* 1007(1):1-9.