

**PERBANDINGAN ALGORITMA CHEAPEST INSERTION HEURISTIC
(CIH) DAN ALGORITMA STEEPEST ASCENT HILL CLIMBING
(SAHC) DALAM PENDISTRIBUSIAN SURAT
SUARA PEMILIHAN UMUM**

(Skripsi)

Oleh

**ELA NOVITASARI
2017031045**



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

ABSTRACT

COMPARISON OF THE CHEAPEST INSERTION HEURISTIC (CIH) ALGORITHM AND THE STEEPEST ASCENT HILL CLIMBING (SAHC) ALGORITHM IN LETTER DISTRIBUTION GENERAL ELECTION VOTES

By

ELA NOVITASARI

William Rowan Hamilton and Thomas Penyngton are the first people to discover the Travelling Salesman Problem (TSP). TSP can be described as a problem in determining the shortest route that a salesman must take, starting from the home city and visiting every city exactly once before returning to the departure city. The aim of this research is to compare the results obtained to determine the Travelling Salesman Problem from one TPS to 27 other TPS in South Pringsewu Village. The results obtained show that the solution obtained using the Steepest Ascent Hill Climbing (SAHC) algorithm method is better than the solution obtained using the Cheapest Insertion Heuristic (CIH) method.

Keywords: *Travelling Salesman Problem, Cheapest Insertion Heuristic Algorithm, Steepest Ascent Hill Climbing Algorithm.*

ABSTRAK

PERBANDINGAN ALGORITMA CHEAPEST INSERTION HEURISTIC (CIH) DAN ALGORITMA STEEPEST ASCENT HILL CLIMBING (SAHC) DALAM PENDISTRIBUSIAN SURAT SUARA PEMILIHAN UMUM

Oleh

ELA NOVITASARI

William Rowan Hamilton dan Thomas Penyngton merupakan orang yang pertama kali menemukan Travelling Salesman problem (TSP). TSP dapat digambarkan sebagai masalah dalam menentukan jalur terpendek yang harus dilalui oleh seorang salesman, yang dimulai dari kota asal dan menyinggahi setiap kota tepat satu kali kemudian kembali lagi ke kota asal keberangkatannya. Adapun tujuan yang ingin didapatkan dari penelitian ini membandingkan hasil yang didapat untuk menentukan Travelling Salesman Problem dari satu TPS ke 27 TPS lainnya di Kelurahan Pringsewu Selatan. Hasil yang didapat menunjukkan bahwa solusi yang didapat dengan menggunakan metode Algoritma Steepest Ascent Hill Climbing (SAHC) lebih baik dari solusi yang didapat dengan menggunakan metode Cheapest Insertion Heuristic (CIH).

Kata kunci: *Travelling Salesman Problem, Algoritma Cheapest Insertion Heuristic, Algoritma Steepest Ascent Hill Climbing.*

**PERBANDINGAN ALGORITMA CHEAPEST INSERTION HEURISTIC
(CIH) DAN ALGORITMA STEEPEST ASCENT HILL CLIMBING
(SAHC) DALAM PENDISTRIBUSIAN SURAT
SUARA PEMILIHAN UMUM**

Oleh

**ELA NOVITASARI
2017031045**

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA MATEMATIKA

Pada

Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Lampung



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

Judul Skripsi : **PERBANDINGAN ALGORITMA CHEAPEST INSERTION HEURISTIC (CIH) DAN ALGORITMA STEEPEST ASCENT HILL CLIMBING (SAHC) DALAM PENDISTRIBUSIAN SURAT SUARA PEMILIHAN UMUM**

Nama Mahasiswa : **Ela Novitasari**

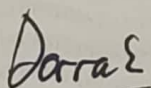
No. Pokok Mahasiswa : **2017031045**

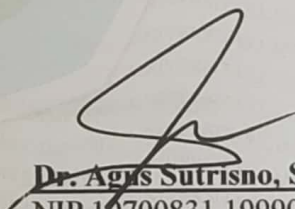
Jurusan : **Matematika**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**

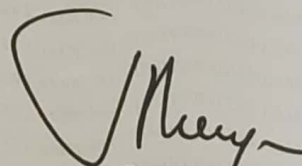
MENYETUJUI

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

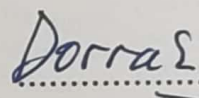
2. Ketua Jurusan Matematika


Dr. Aang Nuryaman S.Si. M.Si.
NIP. 19740316 200501 1 001

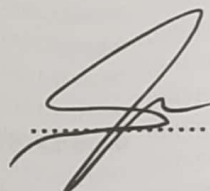
MENGESAHKAN

1. Tim Penguji

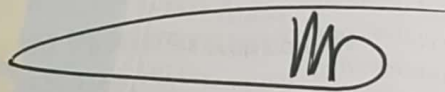
Ketua : **Dra. Dorrah Azis, M.Si.**



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



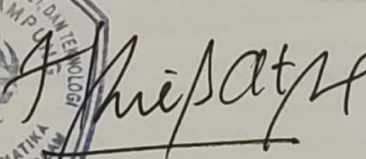
Penguji
Bukan Pembimbing : **Dr. Muslim Ansori, S.Si., M.Si.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



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 dibawah ini:

Nama : **Ela Novitasari**

No. Pokok Mahasiswa : **2017030145**

Jurusan : **Matematika**

Judul : **PERBANDINGAN ALGORITMA CHEAPEST
INSERTION HEURISTIC (CIH) DAN
ALGORITMA STEEPEST ASCENT HILL
CLIMBING (SAHC) DALAM PENDISTRIBUSIAN
SURAT SUARA PEMILIHAN UMUM.**

Dengan ini menyatakan bahwa penelitian ini adalah hasil pekerjaan saya sendiri, dan 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, 19 April 2024
Penulis,



Ela Novitasari
NPM. 2017031045

RIWAYAT HIDUP

Penulis bernama lengkap Ela Novitasari, Anak kedua dari tiga bersaudara yang dilahirkan di Adiluwih pada tanggal 10 Januari 2002 oleh pasangan Bapak Slamet dan Ibu Setyaningsih. Penulis memiliki kakak laki-laki bernama Kiki Febrianto dan adik perempuan bernama Alike Almira Shanum.

Penulis menempuh Pendidikan Taman Kanak-kanak Ma'arif Adiluwih pada tahun 2007-2008, kemudian menempuh Pendidikan Sekolah Dasar diselesaikan di SD Negeri 2 Adiluwih pada tahun 2008-2014, lalu bersekolah di SMP Negeri 1 Adiluwih pada tahun 2014-2017, dan bersekolah SMA Negeri 1 Adiluwih pada tahun 2017-2020.

Pada tahun 2020 penulis melanjutkan Pendidikan di perguruan tinggi dan terdaftar sebagai mahasiswa S1 Jurusan Matematika melalui jalur SBMPTN. Selama menjadi mahasiswi penulis ikut serta dalam organisasi Himpunan Mahasiswa Jurusan Matematika (HIMATIKA) FMIPA UNILA sebagai anggota Biro Dana dan Usaha (2021), dan organisasi Keluarga Mahasiswa Nadlatul Ulama (KMNU) Universitas Lampung sebagai Kepala Departemen Kewirausahaan (2023).

Sebagai aplikasi bidang ilmu kepada masyarakat, penulis telah menyelesaikan Kuliah Kerja Nyata di Desa Wayakrui Kecamatan Kalirejo Kabupaten Lampung Tengah pada tahun 2023, dan Kerja Praktik (KP) di PLN UP2D Bandar Lampung pada Tahun 2023.

KATA INSPIRASI

“Barangsiapa yang mengerjakan kebaikan sekecil apapun, niscaya dia akan melihat (balasan)nya”
(Q.S Al-Zalzalah: 7)

“Karena sesungguhnya sesudah kesulitan itu ada kemudahan, sesungguhnya sesudah kesulitan itu ada kemudahan.”
(Q.S Al-Insyirah: 5-6)

“Barangsiapa yang menempuh jalan untuk mencari ilmu, maka Allah akan mudahkan baginya jalan menuju surga”
(HR. Muslim)

“Ketika kamu sudah berusaha dan berdoa maksimal, tetapi hasil berkata tidak atau belum. Yakinlah, bahwa rencana Allah itu lebih presisi dari rencana manusia”
(Birrul Qodriyyah)

“Jangan bandingkan prosesmu dengan proses orang lain, karena tak semua bunga tumbuh dan mekar bersamaan”

PERSEMBAHAN

Alhamdulillahirobbil'alamin,

Puji dan syukur kepada Allah SWT yang telah memberikan petunjuk dan kemudahan untuk menyelesaikan skripsi ini. Penghargaan bagi diri sendiri yang sudah mampu bertahan sejauh ini melewati berbagai kesedihan.

Kupersembahkan karya kecil dan sederhana ini kepada:

Ibu dan Ayah Tercinta

Sebagai tanda bukti, sayang, dan rasa terima kasih kepada mamak dan bapak yang tidak pernah lelah untuk selalu berkorban, mendoakan, selalu memberi dukungan, serta kasih sayang yang tidak mungkin terbalas dengan selembar kata cinta dan persembahan ini. Semoga ini menjadi langkah awal untuk membuat mamak dan bapak bangga

Kakak dan Adik Tersayang

Terima kasih kepada Mas Kiki yang telah memberikan semangat dan doanya. Kepada Adekku Alike Almira Shanum semoga apa yang telah kakakmu lakukan bisa menjadi contoh dan motivasi

Bapak Ibu Dosen Pembimbing dan Pembahas

Yang senantiasa mengarahkan, memberikan doa dan dukungan, serta semangatnya. Terima kasih telah memberikan cinta, kasih sayang dan kesabarannya.

Almamater Tercinta, Universitas Lampung

SANWACANA

Alhamdulillah segala puji bagi Allah SWT yang telah melimpahkan segala rahmat, ridho dan hidayah-Nya sehingga skripsi ini dapat diselesaikan. Shalawat serta salam semoga selalu tercurah kepada junjungan alam Nabi Muhammad SAW., penuntun jalan bagi seluruh umat manusia. Dalam penulisan skripsi ini, penulis banyak mendapat bantuan, bimbingan dan dorongan baik langsung maupun tidak langsung dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Ibu Dra. Dorrah Aziz, 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 kesediannya 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. Kepada kedua orang tua tercinta, terima kasih telah memberikan dukungan, motivasi dan selalu menyertai penulis dalam doanya untuk melaksanakan dan menyelesaikan skripsi.
10. Untuk Mas Kiki dan adik saya Alike. Terima kasih sudah memberi dukungan dan doa dalam menyelesaikan skripsi ini.
11. Sahabat-sahabat tersayangku, Novitasari Anggreani, Fatimatuzzahra, dan Annisa Salsabila. Terima kasih telah menjadi pendengar yang baik dan selalu memberi dukungan serta motivasi selama proses penyusunan skripsi.
12. Sahabat-sahabat SMA tercintaku, Mustika Diningsih, Lisa Anggraini dan Erna Fitriyanti. Terima kasih atas dukungan, semangat, canda tawa, serta kebersamaannya selama ini.
13. Sahabat-sahabat terkasihku, Diah Pratiwi, Laras Marvenda, dan Nopi Pebriani. Terima kasih telah memberi semangat, dukungan dan kebersamaan selama ini.
14. Keluarga besar KMNU Universitas Lampung atas pengalaman yang dibrikan serta kebersamaan selama ini.
15. Seluruh pihak yang telah memotivasi, membantu, dan memdoakan penulis yang tidak dapat disebutkan satu persatu.

Penulis sangat menyadari skripsi ini jauh dari kata sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun dan dapat memberikan manfaat bagi penulis pribadi dan bagi para pembaca.

Bandar Lampung, 19 April 2024
Penulis,

Ela Novitasari
NPM. 2017031045

DAFTAR ISI

	Halaman
DAFTAR TABEL.....	vi
DAFTAR GAMBAR.....	vii
I. PENDAHULUAN.....	1
1.1 Latar Belakang dan Masalah	1
1.2 Tujuan Penelitian.....	4
1.3 Manfaat Penelitian.....	4
II. TINJAUAN PUSTAKA	5
2.1 Graf	5
2.2 Lintasan Terpendek.....	6
2.3 <i>Travelling Salesman Problem (TSP)</i>	7
2.4 Algoritma <i>Cheapest Insertion Heuristic (CIH)</i>	7
2.5 Algoritma <i>Steepest Ascent Hill Climbing (SAHC)</i>	9
2.6 <i>Google Maps</i>	11
III. METODOLOGI PENELITIAN	13
3.1 Waktu dan Tempat Penelitian	13
3.2 Data Penelitian	13
3.3 Metode Penelitian.....	13
IV. HASIL DAN PEMBAHASAN.....	16
4.1 <i>Travelling Salesman Problem</i> dan Proses Pendistribusian Surat Suara Pemilu di Pringsewu Selatan	16
4.2 Proses Penyelesaian Menggunakan Algoritma <i>Cheapest Insertion Heuristic</i>	21
4.3 Proses Penyelesaian Masalah Menggunakan Algoritma <i>Steepest Ascent Hill Climbing (SAHC)</i>	28

V. KESIMPULAN DAN SARAN	34
5.1 Kesimpulan.....	34
5.2 Saran.....	34
DAFTAR PUSTAKA	36
LAMPIRAN	39

DAFTAR TABEL

Tabel	Halaman
Tabel 1 Data jarak contoh	8
Tabel 2 Tabel iterasi contoh CIH	8
Tabel 3 Tabel iterasi ke-2 CIH.....	9
Tabel 4 Tabel contoh SAHC	10
Tabel 5 Data letak TPS di Kelurahan Pringsewu Selatan	17
Tabel 6 jarak antar lokasi TPS di Kelurahan Pringsewu Selatan.....	19
Tabel 7 penambahan <i>subtour</i> ke 1 (iterasi 1).....	21
Tabel 8 Iterasi CIH yang telah dimodifikasi	22

DAFTAR GAMBAR

Gambar	Halaman
Gambar 1 Graf Sederhana.....	5
Gambar 2 Graf Ganda.....	6
Gambar 3 Graf Semu.....	6
Gambar 4 Graf tak-berarah.....	6
Gambar 5 Graf berarah.....	6
Gambar 6 Gambar Diagram Metode Penelitian.....	15
Gambar 7 Letak titik TPS pada peta.....	16
Gambar 8 <i>Tour</i> 28 TPS di Pringsewu Selatan.....	25
Gambar 9 <i>Tour</i> pada <i>google maps</i>	26
Gambar 10 Penghapusan dan Penambahan sisi pada <i>tour</i>	26
Gambar 11 <i>Tour</i> CIH setelah direvisi.....	27
Gambar 12 <i>Tour</i> CIH pada <i>google maps</i> setelah direvisi.....	27
Gambar 13 iterasi 1 <i>Steepest Ascent Hill Climbing</i>	28
Gambar 14 subtour <i>Steepest Ascent Hill Climbing</i>	30
Gambar 15 <i>Subtour Steepest Ascent Hill Climbing</i>	31
Gambar 16 Penambahan dan pengurangan sisi pada <i>tour</i> SAHC.....	31
Gambar 17 <i>Tour</i> SAHC setelah revisi.....	32
Gambar 18 <i>Tour</i> SAHC di <i>google maps</i> setelah direvisi.....	33

I. PENDAHULUAN

1.1 Latar Belakang dan Masalah

Pemilihan Umum (Pemilu) sering disebut sebagai pesta demokrasi negara (Wardhani, 2018). Pemilu merupakan salah satu pilar penting dalam pembentukan opini publik dan juga merupakan proses demokratis dalam memilih pemimpin. Surat suara merupakan salah satu hal terpenting dalam pemilu. Pendistribusian surat suara yang cepat memerlukan jalur pendistribusian yang efisien dari Panitia Pemungutan Suara (PPS) hingga Tempat Pemungutan Suara (TPS). Oleh karena itu, akan dipilih jalur terpendek untuk pendistribusian surat suara tersebut.

Penentuan rute merupakan aktivitas penting dalam proses pendistribusian, proses penentuan rute adalah langkah-langkah atau tahapan yang dilakukan untuk menentukan jalur atau rute yang akan diambil dari satu titik ke titik lain dalam suatu sistem transportasi yang bertujuan untuk memodelkan pelaku pergerakan dalam memilih rute yang diyakini sebagai rute terbaik. Dengan penentuan rute ini diharapkan permasalahan yang ada dalam pemilu akan dapat diminimalisir. Misalnya keterlambatan pendistribusian surat suara sehingga pelaksanaan pemungutan suara menjadi terhambat.

Pencarian rute terpendek ini dalam optimasi sering kita jumpai dalam permasalahan *Travelling Salesman Problem* (TSP). TSP dapat digambarkan sebagai masalah dalam menentukan jalur terpendek yang harus dilalui oleh seorang salesman, yang dimulai dari kota asal dan menyinggahi setiap kota tepat satu kali kemudian

kembali lagi ke kota asal keberangkatannya. Kota dapat dinyatakan sebagai simpul graf, sedangkan sisi menyatakan jalan yang menghubungkan antar dua buah kota. Bobot pada sisi menyatakan jarak antara dua kota (Syarif et al., 2008). Masalah TSP dapat diselesaikan dengan menggunakan beberapa algoritma, antara lain yaitu *Algoritma Branch and Bound*, *Algoritma Brute Force*, *Algoritma Dijkstra*, *Heuristic* dan lain-lain. Salah satu algoritma heuristic yang cukup efektif dalam menyelesaikan masalah TSP adalah algoritma *Cheapest Insertion Heuristic* (CIH) dan algoritma lain diantaranya *Steepest Ascent Hill Climbing* (SAHC).

Algoritma Cheapest Insertion Heuristic (CIH) adalah algoritma yang membentuk suatu tour dengan membuat rute jalur terpendek dengan bobot minimal dan secara berturut-turut ditambah dengan tempat baru. Pemilihan titik baru tersebut dilakukan bersamaan dengan pemilihan sisi sehingga didapatkan nilai penyisipan minimum. Selanjutnya tempat baru tersebut disisipkan di antara dua tempat yang membentuk sisi yang telah terpilih, algoritma ini seringkali memberikan solusi yang cukup baik karena untuk proses seleksi tempat yang akan disisipkan dilakukan pada setiap tempat di luar tour dan setiap sisi di dalam tour (Saleh et al., 2015).

Algoritma Steepest Ascent Hill Climbing (SAHC) merupakan algoritma yang banyak juga digunakan untuk permasalahan optimasi. Salah satu penerapannya adalah untuk mencari rute yang terpendek dengan cara memaksimalkan atau meminimumkan nilai dari fungsi optimasi yang ada. Secara harafiah *steepest* berarti paling tinggi, sedangkan *ascent* berarti kenaikan. Dengan demikian *steepest ascent* berarti kenaikan paling tinggi. Jadi prinsip dasar dari metode ini adalah mencari kenaikan paling tinggi dari keadaan sekitar untuk mencapai nilai yang paling optimal (Zemma et al., 2017). Algoritma pencarian SAHC pada dasarnya hampir sama dengan algoritma pencarian *Simple Hill Climbing* (SHC), yang membedakannya adalah gerakan pencarian yang tidak dimulai dari posisi paling kiri namun gerakan selanjutnya dicari berdasarkan nilai heuristik terbaik (Elvina & Hakim, 2018).

Ada beberapa penelitian sebelumnya dalam penyelesaian TSP seperti penelitian yang dilakukan oleh (Hignasari, 2020) tentang penyelesaian Travelling Salesman Problem. Dalam penelitian tersebut dijelaskan bahwa Algoritma CIH dan Greedy memiliki hasil optimasi yang cukup baik dan sama-sama memberikan hasil yang hampir sama. Yang membedakan hanyalah pemilihan rute perjalanan yang dilalui. Penelitian yang dilakukan oleh (Beutari,C.A.S., 2020) tentang penentuan rute terpendek pendistribusian surat suara di kota Lhokseumawe menggunakan Algoritma *Best First Search* berbasis Android dapat mengefisiensikan aplikasi tersebut sekitar 90%. Pada penelitian (Yuniawanti, 2015) dengan algoritma SAHC telah mengoptimalkan rute distribusi sales pulsa sehingga dapat menghemat 18,11% biaya transportasi. Pada penelitian lain (Fitiriati & Nessrayasa, 2019) dapat mengoptimalkan jarak terpendek kota wisata di Indonesia dengan nilai akurasi sebesar 93,3%.

Berdasarkan penelitian-penelitian yang telah dilakukan terkait pengaplikasian Algoritma CIH dan SAHC, dan masalah kendala lapangan yang mungkin terjadi pada saat pendistribusian surat suara seperti, sarana dan prasarana pendistribusian yang belum memadai, kondisi jalan yang kurang baik, dan letak beberapa TPS yang sulit dijangkau yang dapat menghambat proses pendistribusian surat suara. Permasalahan ini sangat menarik dan penting untuk dilakukan penelitian lebih lanjut dalam rangka meningkatkan efisiensi jarak pendistribusian, sehingga penulis tertarik untuk mengambil topik tentang penyelesaian masalah *Travelling Salesman Problem* dengan menggunakan metode *Cheapest Insertion Heuristic* (CIH) dan *Steepest Ascent Hill Climbing* (SAHC) dalam pendistribusian surat suara pemilihan umum. Dalam penelitian ini kondisi jalan diasumsikan sama.

1.2 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk membandingkan penyelesaian masalah *Travelling salesman Problem* (TSP) dengan menggunakan Algoritma *Cheapest Insertion Heuristic* (CIH) dan Algoritma *Steepest Ascent Hill Climbing* (SAHC)

1.3 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut :

1. Mengetahui perbandingan penyelesaian masalah *Travelling salesman Problem* dengan menggunakan Algoritma *Cheapest Insertion Heuristic* (CIH) dan Algoritma *Steepest Ascent Hill Climbing* (SAHC).
2. Menambah wawasan pembaca terkait *Travelling Salesman Problem* (TSP), Algoritma *Cheapest Insertion Heuristic* (CIH), dan Algoritma *Steepest Ascent Hill Climbing* (SAHC).

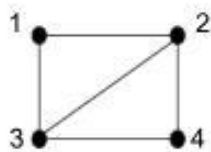
II. TINJAUAN PUSTAKA

2.1 Graf

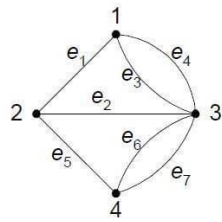
Graf adalah pasangan himpunan *vertex*/simpul dan *edges*/sisi, dimana setiap sisi berhubungan dengan satu atau dua buah simpul. (Dalem, 2018) Secara matematis, graf didefinisikan sebagai berikut: Graf G didefinisikan sebagai pasangan himpunan (V, E) yang dalam hal ini: V = himpunan tidak-kosong dari simpul-simpul (*vertices* atau *node*) = $\{v_1, v_2, \dots, v_n\}$ dan E = himpunan sisi (*edges* atau sisi) yang menghubungkan sepasang simpul = $\{e_1, e_2, \dots, e_n\}$ Atau dapat ditulis singkat notasi $G = (V, E)$, dengan V tidak boleh kosong, sedangkan E boleh kosong Jadi, sebuah graf dimungkinkan tidak mempunyai sisi satu buah pun, tetapi simpulnya harus ada, minimal satu (Salaki, 2011)

Berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, maka secara umum graf dapat digolongkan menjadi dua jenis (Pradhana, 2013):

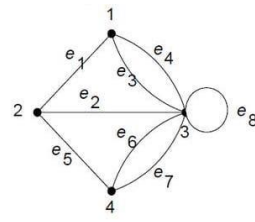
1. Graf sederhana (*simple graph*). Graf yang tidak memiliki gelang maupun sisi-ganda.
2. Graf tak-sederhana (*unsimple-graph*). Graf yang memiliki gelang maupun sisi ganda. Ada dua macam graf tak-sederhana, yaitu graf ganda (*multigraph*) dan graf semu (*pseudograph*). Graf ganda adalah graf yang memiliki sisi ganda, sedangkan graf semu adalah graf yang memiliki gelang.



Gambar 1 Graf Sederhana



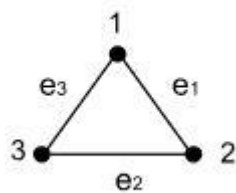
Gambar 2 Graf Ganda



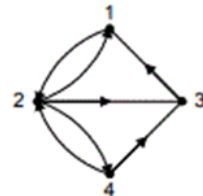
Gambar 3 Graf Semu

Berdasarkan orientasi arah pada sisi graf dapat digolongkan menjadi dua jenis (Mediputra, 2011):

1. Graf tak-berarah graf yang sisinya tidak mempunyai orientasi arah disebut graf tak-berarah..
2. Graf berarah graf yang setiap sisinya diberikan orientasi arah disebut graf berarah.



Gambar 4 Graf tak-berarah



Gambar 5 Graf berarah

2.2 Lintasan Terpendek

Persoalan lintasan terpendek di dalam graf merupakan salah satu persoalan optimasi. Graf yang digunakan dalam pencarian lintasan terpendek adalah graf berbobot (*weighted graph*), yaitu graf yang setiap sisinya diberikan suatu nilai atau bobot. Bobot pada sisi graf dapat dinyatakan sebagai jarak antar kota, waktu pengiriman pesan dan lain-lain. Lintasan terpendek dapat juga di artikan sebagai jalur yang dilalui dari suatu *node* ke *node* lain dengan besar atau nilai pada sisi yang jumlah akhirnya dari node awal ke *node* akhir paling kecil (Dangkua et al., 2015).

2.3 *Travelling Salesman Problem (TSP)*

William Rowan Hamilton dan *Thomas Penyngton* merupakan orang yang pertama kali menemukan *Travelling Salesman problem*. Permainan *Icosian Hamilton* merupakan asal mula ditemukannya TSP. Dalam permainan tersebut diminta untuk menyelesaikan perjalanan dari 20 titik berbeda yang melewati jalur-jalur tertentu (Usman & Oktiarso, 2019). Seorang salesman yang terlibat dalam TSP, yang mana dia wajib mengunjungi beberapa kota untuk mendistribusikan suatu barang. Dalam melakukan kunjungan, salesman harus merencanakan suatu rute agar kota-kota tersebut hanya dilalui satu kali saja, kemudian salesman kembali ke kota asal (Ramadhania & Rani, 2021) .

2.4 *Algoritma Cheapest Insertion Heuristic (CIH)*

Algoritma CIH adalah Algoritma *Insertion* yang pada setiap penambahan kota baru yang akan disisipkan ke dalam *subtour* yang mempunyai bobot penyisipan paling minimal. Algoritma ini memberikan rute perjalanan yang berbeda tergantung dari urutan penyisipan kota- kota pada *subtour* yang bersangkutan dan Algoritma CIH ini baik digunakan untuk kasus TSP dengan jumlah kota yang besar (Utomo et al., 2018) .

Berikut ini adalah tata urutan algoritma CIH sebagai berikut :

1. Pilih 2 titik awal secara acak yang memiliki bobot paling kecil
2. Jumlahkan kedua titik tersebut
3. Cari kandidat titik yang akan dimasukkan kedalam *subtour* baru
4. Hitung setiap kandidat garis yang terpilih dengan tambahan jarak terkecil.

Jarak diperoleh dari Persamaan 1 :

$$C_{jk} = C_{ik} + C_{kj} - C_{ij} \quad (1)$$

Keterangan :

C_{ik} adalah jarak kota i ke kota k

C_{kj} adalah jarak kota k ke kota j

C_{ij} adalah jarak kota i ke kota j

5. Pilih bobot nilai yang paling terkecil dari kandidat yang telah dihitung
6. Ulangi tahapan nomor 3 hingga seluruh titik sudah dikunjungi dan masuk dalam rute
7. Hentikan proses jika semua titik berhasil dikunjungi dan membuat suatu rute.

Contoh 1

Tabel 1 Data jarak contoh

Kota/kota	1	2	3	4
1	0	1	3	2
2	1	0	4	3
3	3	4	0	2
4	2	3	2	0

Langkah-langkah untuk menyelesaikan permasalahan lintasan terpendek:

1. Pilih 2 sisi dengan bobot terkecil, maka dari $(1,2) = 1$ dan $(2,1) = 1$ dengan total jarak 1 km.
2. Pilih semua kandidat sisi yang belum terlewati:
 $(1,3) = 3$, $(1,4) = 2$, $(2,3) = 4$, $(2,4) = 3$
3. Hitung pertambahan bobot dari kandidat:

Tabel 2 Tabel iterasi contoh CIH

Arc yang akan dibuang	Arc yang akan ditambahkan	Perhitungan sisipan jarak
(1,2)	$(1,3) - (3,2)$	$C_{13} + C_{32} - C_{12} = 6$
(1,2)	$(1,4) - (4,2)$	$C_{14} + C_{42} - C_{12} = 4$
(2,1)	$(2,3) - (3,1)$	$C_{23} + C_{31} - C_{21} = 6$
(2,1)	$(2,4) - (4,1)$	$C_{24} + C_{41} - C_{21} = 4$

4. Pilih kandidat dengan bobot terkecil, karena terdapat dua kandidat dengan nilai bobot sama, pilih salah satu saja maka terpilih (1,4), untuk sisi yang dibuang (1,2) dan ditambahkan sisi baru (4,2), dengan bobot sisipan terkecil yaitu 4 km.
5. Didapat *subtour* baru 1-3-2-1.

Karena masih terdapat titik yang belum terhubung dengan *subtour*, maka dilakukan kembali perhitungan yang sama dimulai dari langkah 2 untuk iterasi ke 2 ini:

1. Pilih kandidat sisi yang belum terlewati:
 $(1,4) = 2, (3,4) = 2, (2,4) = 3.$
2. Hitung perhitungan dari penyisipan kandidat:

Tabel 3 Tabel iterasi ke-2 CIH

Arc yang akan dibuang	Arc yang akan ditambahkan	Perhitungan sisipan jarak
(1,3)	(1,4) – (4,3)	$C_{14} + C_{43} - C_{13} = 1$
(3,2)	(3,4) – (4,2)	$C_{34} + C_{42} - C_{32} = 1$
(2,1)	(2,4) – (4,1)	$C_{24} + C_{41} - C_{21} = 4$

3. Karena terdapat dua kandidat yang memiliki penyisipan bobot sama, maka pilih salah satu yaitu (3,4), untuk sisi yang dibuang (3,2) dan ditambahkan sisi baru (4,2) dengan total bobot penyisipan 1 km.
4. Didapat *subtour* baru 1-3-4-2-1.

Maka dari Langkah-langkah yang sudah diselesaikan diperoleh lintasan terpendek untuk seorang salesman mengunjungi 4 lokasi tepat satu kali dengan lintasan 1-3-4-2-1 yang memiliki jarak 9 km.

2.5 Algoritma *Steepest Ascent Hill Climbing* (SAHC)

Algoritma *Steepest Ascent Hill Climbing* (SAHC) adalah algoritma yang menggunakan estimasi jarak terdekat (*cost* / jarak sebenarnya) untuk mencapai tujuan (*goal*). Algoritma SAHC hampir sama dengan algoritma *simple hill climbing*, yang membedakan keduanya adalah pada pencarian atau pertukaran kombinasi tiap *node* yang bila menemukan satu *node* tidak langsung berhenti tetapi

dilanjutkan dengan mencari apakah ada *node* lain yang memiliki nilai heuristik yang lebih baik (Fitiriati & Nessrayasa, 2019)

Tahapan-tahapan Algoritma SAHC sebagai berikut :

- a. Dibentuk lintasan awal sebagai keadaan awal (*initial state*), kemudian dilakukan pengujian dari keadaan awal tersebut. Pengujian dilakukan dengan menghitung total jarak yang ditempuh pada *initial state*. Pengujian tersebut dilakukan untuk mendapatkan nilai heuristik sebagai pembandingan terhadap nilai – nilai heuristik yang ada setelah dilakukan kombinasi pertukaran dua kota. Hasil dari pengujian tersebut adalah jarak minimal yang ditempuh pada permasalahan TSP. Hasil tersebut akan digunakan sebagai pembandingan hasil dari langkah berikutnya.
- b. Setelah dibentuk lintasan awal sebagai keadaan awal, langkah selanjutnya yaitu melakukan kombinasi penukaran dua kota kemudian dilakukan pengujian seperti pada poin a, hingga mencapai kondisi *goal*, jika tidak maka pilihlah elemen yang lain hingga mencapai *goal*. Kondisi *goal* diartikan sebagai keadaan yang merupakan solusi optimum dari permasalahan TSP dengan kondisi: panjang lintasan baru < panjang lintasan lama.
- c. Jika kondisi *goal* telah tercapai maka kondisi tersebut adalah solusinya, jika tidak, maka ulangi langkah tersebut hingga panjang lintasan baru tidak kurang panjang lintasan sebelumnya.

Contoh 2

Tabel 4 Tabel contoh SAHC

Kota/kota	1	2	3	4
1	0	1	3	2
2	1	0	4	3
3	3	4	0	2
4	2	3	2	0

Berikut Langkah-langkah pencarian rute terpendek pada metode SAHC

1. Bentuk lintasan awal (*initial state*) yaitu 1-2-3-4-1 dengan total bobot 9 km.

2. Melakukan kombinasi penukaran dua kota menggunakan rumus:

$$\frac{n!}{2!(n-2)!}$$

Keterangan:

n = banyaknya titik (kota)

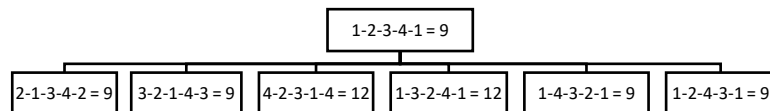
Dari masalah diatas terdapat 4 kota sehingga banyaknya kombinasi penukaran dua kota sebagai berikut:

$$\frac{n!}{2!(n-2)!}$$

$$\frac{4!}{2!(4-2)!} = \frac{4 \times 3 \times 2!}{2!(2)!} = \frac{12}{2} = 6 \text{ lintasan}$$

- Tukar 1 ke 2 = 2-1-3-4-2
- Tukar 1 ke 3 = 3-2-1-4-3
- Tukar 1 ke 4 = 4-2-3-1-4
- Tukar 2 ke 3 = 1-3-2-4-1
- Tukar 2 ke 4 = 1-4-3-2-1
- Tukar 3 ke 4 = 1-2-4-3-1

Dapat dituliskan dalam bentuk diagram berikut:



3. Karena panjang lintasan baru tidak lebih kecil dari *inisial state* maka permasalahan TSP tersebut sudah optimum dengan kondisi *goal state* 1-2-3-4-1 dengan bobot 9 km.

2.6 Google Maps

Google Maps merupakan sebuah jasa peta globe *virtual* dan *online*. Menawarkan peta yang dapat diseret yang diambil dari satelit. Dengan teknologi LBS yang dimiliki android pencarian informasi geografis suatu wisata kuliner dan rute terpendek dapat dilakukan kapanpun dan dimanapun (Marcelina & Yulianti, 2020).

Layanan peta *Google Maps* secara resmi dapat diakses melalui situs <http://maps.google.com>. Pada situs tersebut dapat dilihat informasi geografis pada hampir semua permukaan di bumi kecuali daerah kutub utara dan selatan. Layanan ini dibuat sangat interaktif, karena di dalamnya peta dapat digeser sesuai keinginan pengguna, mengubah level zoom, serta mengubah tampilan jenis peta. *Google Maps* mempunyai banyak fasilitas yang dapat dipergunakan misalnya pencarian lokasi dengan memasukkan kata kunci, kata kunci yang dimaksud seperti nama tempat, kota, atau jalan, fasilitas lainnya yaitu perhitungan rute perjalanan dari satu tempat ke tempat lainnya (M Syaiful Amri, 2016)

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

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

3.2 Data Penelitian

Dalam penelitian ini data yang digunakan adalah data lokasi TPS yang ada di Kelurahan Pringsewu Selatan kabupaten Pringsewu yang diperoleh dari Komisi Pemilihan Umum (KPU) Pringsewu.

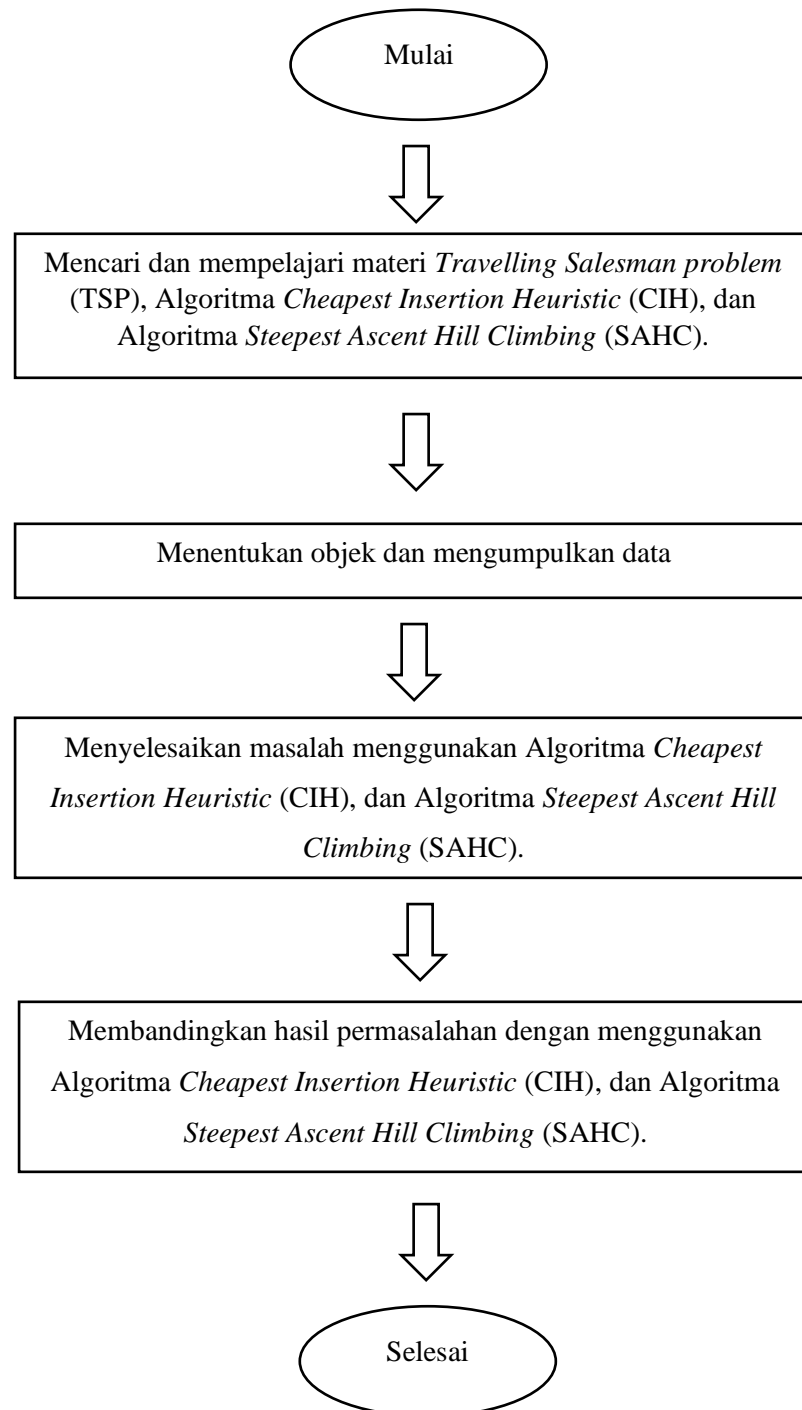
3.3 Metode Penelitian

Penulis melakukan penelitian ini dengan menggunakan studi literatur secara sistematis yang diperoleh dari buku, jurnal maupun dari media yang bisa memberikan informasi kepada penulis untuk mendukung penelitian ini. Metode yang digunakan dalam penelitian ini terdiri dari beberapa langkah sebagai berikut :

1. Mencari bahan dan materi *Travelling Salesman problem* (TSP), Algoritma *Cheapest Insertion Heuristic* (CIH), dan Algoritma *Steepest Ascent Hill Climbing* (SAHC).
2. Mengumpulkan data ke Instansi terkait data yang dibutuhkan.

3. Menghitung jarak dari setiap titik ke titik yang lainnya dengan bantuan *Google Maps*.
4. Mencari penyelesaian permasalahan *Travelling Salesman problem* (TSP) dengan menggunakan Algoritma *Cheapest Insertion Heuristic* (CIH), dan Algoritma *Steepest Ascent Hill Climbing* (SAHC).
5. Menarik kesimpulan berdasarkan penyelesaian masalah yang diperoleh.

Berikut diberikan diagram metode penelitian :



Gambar 6 Gambar Diagram Metode Penelitian

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Pada penelitian ini hasil yang didapatkan dari dua algoritma yang diujikan mendapatkan solusi yang terbaik yaitu menggunakan *Steepest Ascent Hill Climbing* dengan mendapat lintasan terpendek yang lebih baik dibandingkan dengan Algoritma *Cheapest Insertion Heuristic*. Didapat juga solusi menggunakan bantuan *software Python* pada Algoritma *Cheapest Insertion Heuristic* dengan total lintasan 8710 m, dan setelah dilakukan perbaikan lintasan pada garis yang masih berpotongan didapat hasil dengan total lintasan 8770 m atau 8,77 km. Untuk Algoritma *Steepest Ascent Hill Climbing* didapat total lintasan 8170 m atau 8,17 km, dan setelah dilakukan perbaikan lintasan agar tidak ada sisi yang berpotongan, didapat hasil dengan total lintasan 8080 m atau 8,08 km. Dari hasil yang didapat, dapat disimpulkan bahwa Algoritma *Steepest Ascent Hill Climbing* lebih baik dari Algoritma *Cheapest Insertion Heuristic* untuk kasus penyelesaian masalah *Travelling Salesman Problem (TSP)* pada pendistribusian surat suara pemilu dengan perbandingan Algoritma *Cheapest Insertion Heuristic* dengan Algoritma *Steepest Ascent Hill Climbing*.

5.2 Saran

Adapun saran yang dapat diberikan perlu adanya metode penelitian lebih lanjut untuk mendapatkan hasil yang optimal dengan menggunakan algoritma lain serta peneliti lebih memperhatikan kondisi *real* secara nyata di lapangan.

DAFTAR PUSTAKA

- Cut, A. S. B. 2020. Rute terpendek pendistribusian surat suara di kota lhokseumawe menggunakan algoritma *best first search* berbasis android (*Doctoral dissertation*, Universitas Malikussaleh).
- Dalem, I. B. G. W. A. (2018). Penerapan Algoritma A* (Star) Menggunakan Graph Untuk Menghitung Jarak Terpendek. *Jurnal RESISTOR (Rekayasa Sistem Komputer)*, 1(1), 41–47.
- Dangkua, E. V., Gunawan, V., & Adi, K. (2015). Penerapan Metode Hill Climbing Pada Sistem Informasi Geografis Untuk Mencari Lintasan Terpendek. *Jurnal Sistem Informasi Bisnis*, 5(1), 19–25.
- Elvina, & Hakim, L. (2018). Modifikasi Algoritma Steepest-Ascent Hill Climbing Dan Backtracking Untuk Pencarian Lintasan Kritis Proyek Modified Steepest-Ascent Hill Climbing and Backtracking Algorithm for Project Critical Path Finding. *Cogito Smart Journal*, 4(2), 268–282.
- Fitiriati, D., & Nessrayasa, N. M. (2019). Implementasi Algoritma Hill Climbing Pada Penentuan Jarak Terpendek Kots wisata di Indonesia. *Jurnal Riset Informatika*, 1(3), 127–132.
- Hignasari, L. V. (2020). Komparasi Algoritma Cheapest Insertion Heuristic (CIH) Dan Greedy Dalam Optimasi Rute Pendistribusian Barang. *Jurnal Ilmiah Vastuwidya*, 2(2), 31–39.

- M Syaiful Amri. (2016). *Membangun Sistem Navigasi di Surabaya menggunakan Google Maps API*. 15(1), 165–175.
- Marcelina, D., & Yulianti, E. (2020). Aplikasi Pencarian Rute Terpendek Lokasi Kuliner Khas Palembang Menggunakan Algoritma Euclidean Distance Dan a*(Star). *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, 9(2), 195–202.
- Mediputra, A. (2011). *Aplikasi Shortest Path dengan Menggunakan Graf dalam*. 1–5.
- Pradhana, B. A. (2013). *Studi dan Implementasi Persoalan Lintasan Terpendek Suatu Graf Dengan Algoritma Dijkstra dan Algoritma Bellman-Ford*. 1–6.
- Ramadhania, S. E., & Rani, S. (2021). Implementasi Kombinasi Algoritma Genetika dan Tabu Search untuk Penyelesaian Travelling Salesman Problem. *Automata*, 2(1), 99–106.
- Salaki, D. T. (2011). Penentuan Lintasan Terpendek Dari FMIPA ke Rektorat dan fakultas Lain di UNSRAT Manado Menggunakan Algoritma Dijkstra. *Jurnal Ilmiah Sains*, 11(1), 1–4.
- Saleh, K., Helmi, & Prihandono, B. (2015). Penentuan Rute Terpendek Dengan Menggunakan Algoritma Cheapest Insertion Heuristic (Studi Kasus: PT. Wicaksana Overseas International Tbk. Cabang Pontianak). *Buletin Ilmiah Math. Stat. Dan Terapannya (Bimaster)*, 04(3), 295–304.
- Syarif, A., Wamiliana, & Wijaya, Y. (2008). Evaluasi Kinerja Metode-Metode Heuristik Untuk Penyelesaian Travelling Salesman Problem. *Jurnal Sains MIPA*, 14(1), 7–11.
- Usman, M. Z., & Oktiarso, T. (2019). Implementasi Algoritma Greedy Untuk

Menyelesaikan Travelling Salesman Problem di Distributor PT. Z. *Journal of Integrated System*, 1(2), 216–229.

Utomo, R. G., Maylawati, D. S., & Alam, C. N. (2018). Implementasi Algoritma Cheapest Insertion Heuristic (CIH) dalam Penyelesaian Travelling Salesman Problem (TSP). *Jurnal Online Informatika*, 3(1), 61.

Wardhani, P. S. N. (2018). Partisipasi Politik Pemilih Pemula dalam Pemilihan Umum. *Jupiis: Jurnal Pendidikan Ilmu-Ilmu Sosial*, 10(1), 57–62.

Yuniawanti, Endah P. 2015. Optimasi Rute Distribusi Menggunakan Steepest - Ascent Hill Climbing. Tugas Akhir, Universitas Negeri Yogyakarta

Zemma, L. A., Herfina, & Qur'ania, A. (2017). *Penerapan Metode Steepest Ascent Hill Climbing Pada Model Pencarian Rute Terdekat Fasilitas Pelayanan Darurat Di Kota Bogor Berbasis Android*.

LAMPIRAN

Lampiran 1 Output Algoritma *Cheapest Insertion Heuristic* (CIH) Menggunakan Program Python

```
start_point = 12 # Ganti dengan titik awal yang diinginkan
tour = cheapest_insertion_heuristic(distances_matrix, start_point)
total_cost = calculate_total_cost(tour, distances_matrix)

print(f'Tur terpendek menggunakan Cheapest Insertion Heuristic dengan titik awal {start_point}: {tour}')
print(f'Panjang Tur (Total Biaya): {total_cost}')

Tur terpendek menggunakan Cheapest Insertion Heuristic dengan titik awal 12: [12, 18, 25, 24, 10, 4, 5, 19, 14, 15, 21, 20, 27, 22, 16, 17, 13, 7, 8, 23, 6, 3, 1, 0, 2, 9, 26, 11, 12]
Panjang Tur (Total Biaya): 8710
```

Lampiran 2 Output Algoritma *Steepest Ascent Hill Climbing* (SAHC) Menggunakan Program Python

```
# Menyelesaikan TSP menggunakan algoritma Steepest Ascent Hill Climbing
best_path, best_cost = steepest_ascent_hill_climbing(initial_state, distances)
print("Best Path:", best_path)
print("Best Cost:", best_cost)

Initial Path: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27]
Initial Cost: 15220
Iteration: 1
Swap: (23, 27)
New Path: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 27, 24, 25, 26, 23]
New Cost: 13070
Iteration: 2
Swap: (2, 27)
New Path: [0, 1, 23, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 27, 24, 25, 26, 2]
New Cost: 12880
Iteration: 3
Swap: (2, 3)
New Path: [0, 1, 3, 23, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 27, 24, 25, 26, 2]
New Cost: 11480
Iteration: 4
Swap: (13, 18)
New Path: [0, 1, 3, 23, 4, 5, 6, 7, 8, 9, 10, 11, 12, 18, 14, 15, 16, 17, 13, 19, 20, 21, 22, 27, 24, 25, 26, 2]
New Cost: 10930
Iteration: 5
Swap: (4, 8)
New Path: [0, 1, 3, 23, 8, 5, 6, 7, 4, 9, 10, 11, 12, 18, 14, 15, 16, 17, 13, 19, 20, 21, 22, 27, 24, 25, 26, 2]
New Cost: 10560
Iteration: 6
Swap: (10, 26)
New Path: [0, 1, 3, 23, 8, 5, 6, 7, 4, 9, 26, 11, 12, 18, 14, 15, 16, 17, 13, 19, 20, 21, 22, 27, 24, 25, 10, 2]
New Cost: 10230
Iteration: 7
Swap: (5, 7)
New Path: [0, 1, 3, 23, 8, 7, 6, 5, 4, 9, 26, 11, 12, 18, 14, 15, 16, 17, 13, 19, 20, 21, 22, 27, 24, 25, 10, 2]
New Cost: 10000
Iteration: 8
Swap: (4, 6)
New Path: [0, 1, 3, 23, 6, 7, 8, 5, 4, 9, 26, 11, 12, 18, 14, 15, 16, 17, 13, 19, 20, 21, 22, 27, 24, 25, 10, 2]
New Cost: 9790
Iteration: 9
Swap: (3, 4)
New Path: [0, 1, 3, 6, 23, 7, 8, 5, 4, 9, 26, 11, 12, 18, 14, 15, 16, 17, 13, 19, 20, 21, 22, 27, 24, 25, 10, 2]
New Cost: 9590
Iteration: 10
Swap: (5, 6)
New Path: [0, 1, 3, 6, 23, 8, 7, 5, 4, 9, 26, 11, 12, 18, 14, 15, 16, 17, 13, 19, 20, 21, 22, 27, 24, 25, 10, 2]
New Cost: 9360
Iteration: 11
Swap: (24, 25)
New Path: [0, 1, 3, 6, 23, 8, 7, 5, 4, 9, 26, 11, 12, 18, 14, 15, 16, 17, 13, 19, 20, 21, 22, 27, 25, 24, 10, 2]
New Cost: 9170
Iteration: 12
Swap: (20, 22)
New Path: [0, 1, 3, 6, 23, 8, 7, 5, 4, 9, 26, 11, 12, 18, 14, 15, 16, 17, 13, 19, 22, 21, 20, 27, 25, 24, 10, 2]
New Cost: 9020
Iteration: 13
Swap: (15, 19)
New Path: [0, 1, 3, 6, 23, 8, 7, 5, 4, 9, 26, 11, 12, 18, 14, 19, 16, 17, 13, 15, 22, 21, 20, 27, 25, 24, 10, 2]
New Cost: 8790
Iteration: 14
Swap: (14, 15)
New Path: [0, 1, 3, 6, 23, 8, 7, 5, 4, 9, 26, 11, 12, 18, 19, 14, 16, 17, 13, 15, 22, 21, 20, 27, 25, 24, 10, 2]
New Cost: 8740
Iteration: 15
Swap: (16, 18)
Iteration: 16
Swap: (17, 19)
New Path: [0, 1, 3, 6, 23, 8, 7, 5, 4, 9, 26, 11, 12, 18, 19, 14, 13, 15, 16, 17, 22, 21, 20, 27, 25, 24, 10, 2]
New Cost: 8230
Iteration: 17
Swap: (18, 20)
New Path: [0, 1, 3, 6, 23, 8, 7, 5, 4, 9, 26, 11, 12, 18, 19, 14, 13, 15, 22, 17, 16, 21, 20, 27, 25, 24, 10, 2]
New Cost: 8220
Iteration: 18
Swap: (19, 20)
New Path: [0, 1, 3, 6, 23, 8, 7, 5, 4, 9, 26, 11, 12, 18, 19, 14, 13, 15, 22, 16, 17, 21, 20, 27, 25, 24, 10, 2]
New Cost: 8170
Best Path: [0, 1, 3, 6, 23, 8, 7, 5, 4, 9, 26, 11, 12, 18, 19, 14, 13, 15, 22, 16, 17, 21, 20, 27, 25, 24, 10, 2, 0]
Best Cost: 8170
```