

**PENERAPAN ALGORITMA A-STAR UNTUK Mencari
RUTE TERPENDEK DESTINASI WISATA BUDAYA
KOTA YOGYAKARTA**

SKRIPSI

**OLEH
ADITYA DWI CAHYONO
NIM. 18610059**



**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022**

**PENERAPAN ALGORITMA A-STAR UNTUK Mencari
RUTE TERPENDEK DESTINASI WISATA BUDAYA
KOTA YOGYAKARTA**

SKRIPSI

**Diajukan Kepada
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Matematika (S.Mat)**

**Oleh
Aditya Dwi Cahyono
NIM. 18610059**

**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022**

**PENERAPAN ALGORITMA *A-STAR* UNTUK Mencari
RUTE TERPENDEK DESTINASI WISATA BUDAYA
KOTA YOGYAKARTA**

SKRIPSI

Oleh
Aditya Dwi Cahyono
NIM. 18610059


Telah Diperiksa dan Disetujui Untuk Diuji
Malang, 22 Juni 2022

Dosen Pembimbing I



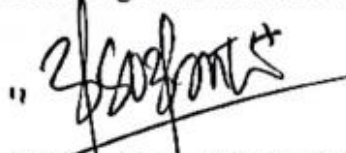
Juhari, M.Si.
NIDT. 19840209 20160801 1 055

Dosen Pembimbing II



Ach. Nasichuddin, M.A.
NIP. 19730705 200003 1 002

Mengetahui,
Ketua Program Studi Matematika




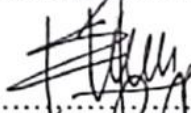
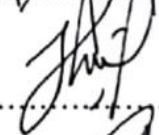
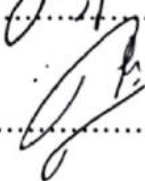
Dr. Elly Susanti, S.Pd., M.Sc.
NIP. 19741129 200012 2 005

**PENERAPAN ALGORITMA *A-STAR* UNTUK Mencari
RUTE TERPENDEK DESTINASI WISATA BUDAYA
KOTA YOGYAKARTA**

SKRIPSI

**Oleh
Aditya Dwi Cahyono
NIM. 18610059**

Telah Dipertahankan di Depan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)
Tanggal 23 Juni 2022

Ketua Penguji	: Ari Kusumastuti, M.Si., M.Pd.	
Anggota Penguji 1	: Dr. Heni Widayani, M.Si.	
Anggota Penguji 2	: Juhari, M.Si.	
Anggota Penguji 3	: Ach. Nasichuddin, M.A.	

Mengetahui,
Ketua Program Studi Matematika



Dr. Elly Susanti, S.Pd., M.Sc.
NIP. 19741129 200012 2 005

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Aditya Dwi Cahyono

NIM : 18610059

Program Studi : Matematika

Fakultas : Sains dan Teknologi

Judul Skripsi : Penerapan Algoritma *A-Star* untuk Mencari Rute Terpendek
Destinasi Wisata Budaya Kota Yogyakarta.

Menyatakan dengan sebenarnya bahwa skripsi ini saya tulis benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilan data, tulisan, atau pikiran orang lain yang saya akui sebagai tulisan saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar rujukan. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima saksi atas perbuatan tersebut.

Malang, 12 Juni 2022
Yang membuat pernyataan



Aditya Dwi Cahyono
NIM. 18610059

MOTO

*“Dan aku belum pernah kecewa dalam berdoa kepada engkau, ya Rabbku”
(QS. Maryam:4)*

PERSEMBAHAN

Dan skripsi ini penulis persembahkan untuk:

Ayahanda Ina Mariana yang selalu mendukung dan mendoakakan setiap langkah dan keputusan yang penulis ambil. Ibunda Ina Mariana yang tidak henti-hentinya mendoakan semua anaknya di sepertiga malam terakhir. Teman-teman yang selalu membantu dalam menyelesaikan kesulitan penulis. Dosen-dosen yang telah membantu menyelesaikan skripsi penulis. Serta diri saya sendiri yang mau diajak untuk memaksakan diri dan bersusah payah dalam menyelesaikan segala tantangan dan tekanan selama menempuh pendidikan strata satu.

KATA PENGANTAR

Segala puji bagi Allah subhanahu wa ta'ala yang telah melimpahkan nikmat dan karunianya, sehingga penulis dapat menyelesaikan penulisan skripsi yang berjudul “Penerapan Algoritma *A-Star* untuk Mencari Rute Terpendek Destinasi Wisata Budaya Kota Yogyakarta” dengan lancar, sebagai syarat untuk memperoleh gelar sarjana dalam bidang matematika di Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Dalam penyusunan skripsi ini tidak lepas bimbingan dan arahan baik secara langsung maupun tidak langsung dari berbagai pihak. Untuk itu ucapan terima kasih yang sebesar-besarnya dan penghargaan yang setinggi-tingginya penulis sampaikan terutama kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim.
2. Dr. Sri Harini, M.Si, selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim.
3. Dr. Elly Susanti, S.Pd., M.Sc, selaku ketua Program Studi Matematika, Universitas Islam Negeri Maulana Malik Ibrahim.
4. Juhari, M.Si, selaku dosen pembimbing I yang telah banyak memberikan arahan, nasihat, motivasi, dan pengalaman berharga kepada penulis.
5. Ach. Nasichuddin, M.A. selaku dosen pembimbing II yang telah banyak memberikan arahan dan berbagi ilmunya kepada penulis.
6. Dr. Heni Widayani, M.Si, selaku dosen penguji seminar proposal, seminar hasil, dan sidang skripsi yang telah memberikan saran dan kritikan yang bermanfaat bagi penulis.
7. Ari Kusumastuti, M.Si., M.Pd, selaku dosen penguji seminar hasil dan sidang skripsi yang juga telah memberikan saran yang bermanfaat bagi penulis.
8. Seluruh dosen Program Studi Matematika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim

9. Ayahanda Hasim Asari dan Ibunda Ina Mariana, serta kakak dan adik tercinta yang selalu memberikan semangat, dukungan, dan motivasi kepada penulis.
10. Segenap sivitas akademika Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
11. Semua pihak yang tidak dapat disebutkan satu-persatu yang telah memberikan bantuan kepada penulis dalam menyelesaikan skripsi ini baik moril maupun materil.

Malang, 11 Juni 2022

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTO	vi
PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii
ABSTRAK	xiii
ABSTRACT	xiv
ملخص	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian	4
1.4 Manfaat Penelitian	5
1.5 Batasan Masalah	5
BAB II KAJIAN TEORI	6
2.1 Teori Pendukung	6
2.1.1 Definisi Graf	6
2.1.2 Jenis-jenis Graf	6
2.1.3 Lintasan Terpendek	8
2.1.4 Algoritma <i>A-Star</i>	9
2.1.5 Wisata Budaya	11
2.2 Kajian Integrasi dengan Al-Qur'an dan Hadits	11
2.3 Kajian Penerapan Algoritma <i>A-Star</i> Dengan Teori Pendukung	13
BAB III METODE PENELITIAN	16
3.1 Jenis Penelitian	16
3.2 Data dan Sumber Penelitian	16
3.3 Tahap Penelitian	16
BAB IV HASIL DAN PEMBAHASAN	20
4.1 Deskripsi Data	20
4.2 Pengujian Pencarian Rute Terpendek	27
4.3 Hasil Pengujian	66
4.4 Evaluasi Pengujian	68
BAB V PENUTUP	72
5.1 Kesimpulan	72
5.2 Saran untuk Penelitian Lanjutan	73
DAFTAR PUSTAKA	74
RIWAYAT HIDUP	77

DAFTAR GAMBAR

Gambar 2.1 Contoh dari Graf Sederhana.....	6
Gambar 2.2 Contoh dari Graf Tidak Sederhana.....	7
Gambar 2.3 Contoh dari Graf Berbobot.....	8
Gambar 2.4 Contoh dari Graf berarah.....	8
Gambar 4.1 Screenshoot data Tempat dan Jalur Kota Yogyakarta	21
Gambar 4.2 Pembentukan Grid berdasarkan Screenshot pada Maps	23
Gambar 4.3 Visualisasi Rute Optimal Setelah Perhitungan.....	65
Gambar 4.4 Perhitungan Rute Hasil Penelitian di Google Maps.....	66
Gambar 4.5 Perhitungan Rute Pemandangan di Google Maps	67

DAFTAR TABEL

Tabel 4.1 Tabel Destinasi Wisata Budaya Kota Yogyakarta.....	20
Tabel 4.2 Data Titik-Titik Indeks dari Setiap Persimpangan.....	25
Tabel 4.3 Pembagian Cluster dari Destinasi yang Berdekatan	26
Tabel 4.4 Pembentukan Titik-Titik Koordinat Grid Setelah Cluster	26
Tabel 4.5 Sampel Titik Awal dan Titik Tujuan	27
Tabel 4.6 Tabel Hasil Proses Pengujian.....	68
Tabel 4.7 Tabel Hasil Nilai Efektifitas	69

ABSTRAK

Cahyono, Aditya Dwi. 2022. **Penerapan Algoritma *A-Star* Untuk Mencari Rute Terpendek Destinasi Wisata Budaya Kota Yogyakarta**. Skripsi. Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing (1) : Juhari, M.Si, Pembimbing (2) : Ach. Nasichuddin, M.A.

Kata Kunci : Algoritma *A-Star*, nilai heuristik, rute terpendek, wisata budaya

Berkunjung ke wisata budaya merupakan pilihan yang bagus untuk menghabiskan liburan sekaligus mengingat sejarah perjuangan. Salah satu kota di provinsi Yogyakarta yang menjadi tujuan favorit dengan banyak wisata budayanya adalah Kota Yogyakarta. Kota ini mempunyai lebih dari 20 destinasi wisata budaya yang terbuka. Seringkali para wisatawan yang datang tidak hanya ingin pergi ke salah satu tempat saja, namun ke beberapa tempat sekaligus. Oleh karena itu diperlukan suatu metode bantuan untuk menentukan destinasi mana saja yang disarankan untuk dikunjungi terlebih dahulu agar perjalanan berwisata menjadi lebih efektif dan efisien. Metode yang digunakan dalam pencarian rute terpendek pada penelitian ini menggunakan Algoritma *A-Star*, yang dalam proses perhitungannya dipengaruhi oleh nilai heuristik. Nilai heuristik didapat dari perhitungan jarak setiap titik yang dipilih menuju titik tujuan. Proses perhitungan rute terpendek dimulai dengan memilih titik awal yang diinginkan, kemudian dapat dipilih beberapa pilihan destinasi wisata budaya. Setelah itu akan dilakukan penyeleksian untuk mencari rute terpendek menggunakan algoritma *A-Star* pada setiap destinasi melalui metode grid, kemudian didapatkan perhitungan paling minimum untuk dipilih menjadi destinasi yang akan dikunjungi terlebih dulu, proses perhitungan terus berlanjut hingga destinasi terakhir. Pengujian keefektifan rute yang didapat dari proses perhitungan dilakukan melalui perbandingan dengan rute yang disajikan oleh google maps. Berdasarkan 10 kali percobaan yang dilakukan pada lebih dari satu destinasi, didapatkan hasil perhitungan bahwa rata-rata perhitungan rute terpendek menggunakan algoritma *A-Star* adalah 21,85% lebih pendek dibandingkan jarak yang tersaji pada google maps. Perhitungan dengan algoritma *A-Star* diharapkan dapat membantu para wisatawan untuk bisa menentukan destinasi wisata budaya sesuai prioritas agar bisa lebih menghemat waktu dan tenaga yang dibutuhkan.

ABSTRACT

Cahyono, Aditya Dwi. 2022. **The Application of the A-Star Algorithm to Find the Shortest Route to Yogyakarta City Cultural Tourism Destinations.** Thesis. Mathematics Department of Mathematics, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Supervisor (1) : Juhari, M.Si, Supervisor (2) : Ach. Nasichuddin, M.A.

Keywords : *A-Star* Algorithm, heuristic value, shortest path, cultural tourism

Visiting cultural tours is a good choice to spend a vacation while remembering the history of the struggle. One of the cities in the province of Yogyakarta which is a favorite destination with a lot of cultural tourism is the city of Yogyakarta. The city has more than 20 open cultural tourism destinations. Often the tourists who come not only want to go to one place, but to several places at once. Therefore, a method of assistance is needed to determine which destinations are recommended to be visited first so that the trip becomes more effective and efficient. The method used in finding the shortest route in this study uses the A-Star Algorithm, which in the calculation process is influenced by the heuristic value. The heuristic value is obtained from the calculation of the distance of each selected point to the destination point. The process of calculating the shortest route begins by selecting the desired starting point, then several choices of cultural tourism destinations can be selected. After that, a selection will be made to find the shortest route using the A-Star algorithm at each destination through the grid method, then the minimum calculation will be obtained to be selected as the destination to be visited first, the calculation process continues until the last destination. Testing the effectiveness of the route obtained from the calculation process is carried out by comparison with the route presented by google maps. Based on 10 experiments conducted at more than one destination, it was found that the average calculation of the shortest route using the A-Star algorithm was 21.85% shorter than the distance presented on google maps. Calculations with the A-Star algorithm are expected to help tourists to be able to determine cultural tourism destinations according to priority in order to save more time and energy needed.

ملخص

كاهيونو ، أديتيا دوي. 2022. تطبيق خوارزمية A-Star لإيجاد أقصر طريق إلى وجهات السياحة الثقافية في مدينة يوجياكارتا. فرضية. برنامج دراسة الرياضيات ، كلية العلوم والتكنولوجيا ، مولانا مالك إبراهيم الدولة الإسلامية جامعة مالانج. مشرف (1): جهري ، محمد علي ، مستشار (2): أحمد نصي الدين ، ماجستير

الكلمات المفتاحية: خوارزمية A-Star ، القيمة الإرشادية ، أقصر طريق ، السياحة الثقافية

تعد زيارة الجولات الثقافية اختيارًا جيدًا لقضاء إجازة مع تذكر تاريخ النضال. مدينة يوجياكارتا هي إحدى المدن في منطقة يوجياكارتا الخاصة والتي تعد مكانًا مميزًا مع العديد من الجولات الثقافية. هذه المدينة لديها ما يقرب من 20 وجهة سياحية ثقافية. عادة السياح الذين يرغبون في زيارة ليس فقط مكان واحد ، ولكن عدة أماكن. نتيجة لذلك ، هناك حاجة إلى المساعدة لتحديد الوجهات التي يجب زيارتها أولاً حتى تكون رحلات السفر الخاصة بهم فعالة. تستخدم عملية العثور على أقصر طريق في هذه الدراسة خوارزمية A-Star ، والتي تعتبر في هذه العملية القيمة التجريبية حقًا. يتم الحصول على القيمة الإرشادية من حساب المسافة من كل نقطة إلى نقطة الوجهة. تتم عملية اختبار أقصر طريق عن طريق اختيار نقطة البداية ، ثم اختيار عدة خيارات لوجهات السياحة الثقافية. بعد ذلك ، سيتم إجراء تحديد للعثور على أقصر طريق باستخدام خوارزمية A-Star في كل وجهة من خلال طريقة الشبكة ، ثم سيتم الحصول على الحد الأدنى من الحساب ليتم اختياره كوجهة يجب زيارتها أولاً. وهكذا حتى الوجهة النهائية. يتم اختبار فاعلية المسار من خلال المقارنة مع المسار الذي قدمته خرائط جوجل. بناءً على نتائج 10 تجارب على عدة وجهات شاملة ، وجد أن متوسط البحث عن المسار باستخدام خوارزمية A-Star كان 25.40٪ أقصر من ذلك المعروض على خرائط جوجل. من المتوقع أن تساعد الحسابات باستخدام خوارزمية A-Star السائحين على تحديد الوجهات السياحية الثقافية وفقًا للأولوية من أجل توفير المزيد من الوقت والطاقة اللازمين.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sektor perjalanan pariwisata menjadi salah satu sektor penting sebagai penggerak ekonomi. Seperti berbagai kawasan perkotaan yang ada di Pulau Jawa dan terkenal dengan destinasi liburannya, kota Yogyakarta merupakan salah satu wisata andalan masyarakat perkotaan yang ada di kawasan pulau Jawa karena memiliki banyak sekali tempat wisata budaya yang dapat didatangi oleh wisatawan/pendatang. Dengan banyaknya pilihan destinasi wisata, maka wisatawan pasti akan memikirkan efisiensi penggunaan waktu untuk pergi ke berbagai lokasi wisata yang ada. Oleh karena itu, metode *Shortest Path* bisa dimanfaatkan menjadi suatu cara untuk memperkirakan jarak tercepat yang bisa ditempuh untuk pergi ke berbagai lokasi wisata budaya yang ada di kota Yogyakarta sehingga bisa menghemat waktu dan biaya.

Pencarian jalur terpendek ketika sedang dalam berkunjung ke suatu daerah wisata adalah hal yang sangat penting untuk dipertimbangkan ketika merencanakan destinasi wisata. Pencarian rute terpendek bertujuan untuk meminimalisir akomodasi dan tidak boros saat perjalanan. Nasihat berperilaku hemat dan tidak boros menurut Islam tercantum dalam Al-Qur'an surat Al-Isra' ayat 26 hingga 27 yang artinya:

“Dan janganlah kamu menghambur-hamburkan (hartamu) secara boros. Sesungguhnya pemboros-pemboros itu adalah saudara-saudara syaitan, dan syaitan itu adalah sangat ingkar kepada Rabbnya.” (QS. Al-Isra'/17:26-27).

Dari ayat di atas menerangkan betapa tidak terpujinya sifat boros. Dalam Tafsir Al Munir, Syaikh Wahbah Az Zuhaili menerangkan sekaligus menjelaskan

bahwa Allah mengharamkan untuk bersifat boros, sebagaimana perkataan Imam Syafi'i bahwa sifat boros adalah mengeluarkan harta tidak pada sesuai kebutuhannya, dan bukanlah pemborosan jika membelanjakannya untuk kebaikan. Sebab, sifat boros tersebut bisa memicu perasaan kufur nikmat dan kurang bersyukur kepada nikmat dan rezeki yang sudah diberikan oleh Allah subhanahu wa ta'ala.

Rute terpendek merupakan suatu permasalahan yang sangat erat hubungannya dengan teori Graf. Teori Graf adalah himpunan dari titik-titik $V(G)$ dan himpunan sisi $E(G)$ yang kemudian menghubungkan suatu titik dengan titik yang lain, atau yang disebut dengan *edge* (Munir, 2016). Penelitian dalam teori ini sering diimplementasikan pada banyak permasalahan, beberapa contohnya adalah dalam menentukan runtutan jadwal kegiatan, perhitungan jalur terpendek, penyusunan dalam penyaluran listrik, dan lainnya. Kaitan penelitian dalam teori ini dengan destinasi wisata budaya Kota Yogyakarta adalah tentang rute paling efisien untuk mengunjungi wisata budaya Kota Yogyakarta. Lokasi-lokasi wisata budaya Kota Yogyakarta dapat disusun menjadi titik koordinat sesuai lokasi yang di data, dan dari titik koordinat tersebut barulah bisa dibentuk menjadi suatu graf. Titik-titik tersebut dihubungkan untuk mencari solusi optimal bagaimana misalnya jika diasumsikan contoh sembarang titik, dan akan dicari jalur terpendek dan efisien untuk berkunjung ke destinasi wisata budaya yang diinginkan. Ada beberapa algoritma yang dapat digunakan dalam menentukan jalur terpendek, salah satunya adalah algoritma *A-Star*.

Dalam menentukan rute terpendek, ada banyak jenis algoritma yang dapat diimplementasikan, seperti *Bellman-Portage*, *Dijkstra*, dan *Floyd-Warshall*. Dalam

menentukan rute terpendek, perhitungan algoritma *Dijkstra* dapat lebih cepat, tetapi perhitungan menggunakan algoritma *Dijkstra* yang mengimplementasikan prinsip *greedy* umumnya tidak pasti berhasil dalam menghasilkan rute yang optimal dalam kasus penentuan rute terpendek, karena perhitungan *Dijkstra* hanya mengutamakan solusi yang terbaik untuk dibuat pada setiap langkah tanpa mempertimbangkan imbas yang dihasilkan ke tahap selanjutnya, dan juga perhitungan *dijkstra* tidak dapat menyelesaikan sisi graff dengan bobot yang negatif (Rofiq dan Uzzu, 2014).

Selanjutnya, pada jaringan dinamis yang kompleks, algoritma *Dijkstra* menjadi kurang optimal dikarenakan banyaknya simpul yang ada pada suatu rute graf dan simpul tersebut akan dihampiri kembali, akhirnya membuat banyak perhitungan dan operasi komputasi yang diulangi (Kumar dan Kumar, 2010). Selain itu, metode *Bellman-Portage* dapat menjadi penyelesaian masalah rute tercepat pada sisi berbobot negatif dari suatu graf, namun menghabiskan waktu yang lebih lama dan tidak efisien (Kamayudi, 2006). Pada perhitungan dengan algoritma *A-Star*, penyelesaiannya menggunakan program dinamis dan akan lebih terjamin tingkat keberhasilannya untuk menentukan solusi yang paling efisien (minimum), karena algoritma ini menggunakan nilai fungsi heuristic yang bisa mengabaikan hal-hal kurang penting yang dapat menghambat untuk mencapai solusi optimal, serta dapat membandingkan seluruh kemungkinan yang ada di suatu lintasan graf pada setiap sisinya dari tiap simpul yang ditempuh (Aprian dan Novandi, 2007).

Implementasi Algoritma *A-Star* sudah sering dikerjakan pada penelitian sebelumnya dengan berbagai jenis permasalahan yang bervariasi. Penelitian Kurniawan (2016) dalam Susilawati (2020) memakai algoritma *A-Star* sebagai

metode Perhitungan Rute terpendek Pada aplikasi *Maze*, dan mendapatkan hasil pengujian bahwa Algoritma *A-Star* dapat memperhitungkan rute terpendek yang telah dilewati dari node awal menuju *goal node* yang diinginkan.

Oleh karena itu, untuk menentukan cara yang paling efisien dalam mengunjungi lokasi wisata budaya yang ada di kota Yogyakarta, penelitian ini memakai teknik *A-Star* yang dipakai untuk membuat perhitungan jalur terpendek dengan waktu yang efisien. Metode ini dipakai karena dapat memperhitungkan jalur yang paling singkat, dan dapat menganalisis semua jalur lintasan yang efisien pada suatu graf untuk tiap tepi dari seluruh simpul yang dibuat. Oleh karenanya, metode ini layak dipakai dalam menyelesaikan masalah penentuan jalur yang paling singkat, terutama dalam penerapannya pada pencarian lokasi wisata budaya yang ada di kota Yogyakarta.

1.2 Rumusan Masalah

Berdasarkan permasalahan di atas, rumusan masalah yang akan dikaji pada penelitian ini yaitu bagaimana penerapan algoritma *A-Star* dalam mencari rute terpendek dalam mengunjungi lokasi-lokasi wisata budaya yang ada di kota Yogyakarta?

1.3 Tujuan Penelitian

Tujuan yang ingin diraih penulis melalui penelitian ini yaitu untuk memahami bagaimana penerapan dari algoritma *A-Star* dalam mencari rute terpendek dalam mengunjungi lokasi-lokasi wisata budaya yang ada di kota Yogyakarta.

1.4 Manfaat Penelitian

Manfaat penelitian ini secara umum yaitu untuk dapat mengaplikasikan pengetahuan dan ilmu matematika, serta penerapannya dalam kehidupan yang dijelaskan pada interpretasi tentang penerapan algoritma *A-Star* dalam penentuan rute yang paling pendek dan efisien. Selain itu, penelitian ini diharapkan juga dapat dimanfaatkan sebagai pedoman dan sumber bacaan tambahan untuk penelitian lain dengan studi kasus dan metode yang berbeda.

1.5 Batasan Masalah

Agar penyelesaian masalah yang diangkat bisa diselesaikan dengan maksimal, maka diperlukan suatu batasan-batasan pada permasalahan sedang dibahas. Adapun batasan masalah yang ditentukan dalam penelitian ini, yaitu:

1. Lokasi destinasi wisata budaya yang menjadi data penelitian adalah destinasi wisata budaya yang ada di Kota Yogyakarta, bukan provinsi Daerah Istimewa Yogyakarta.
2. Pengambilan rute ditentukan pada rute jalan utama menuju lokasi yang diambil dalam data, yakni destinasi wisata budaya di kota Kota Yogyakarta.
3. Perhitungan dari penelitian ini hanya mengambil bobot jarak, namun untuk waktu diabaikan karena keadaan kemacetan jalan di Kota Yogyakarta yang tidak bisa diprediksi.
4. Setiap destinasi wisata yang berdekatan dengan jarak 2x3 grid akan dijadikan satu *cluster* titik wisata.
5. Data penelitian yang diolah didapatkan lewat *Google Maps*.

BAB II KAJIAN TEORI

2.1 Teori Pendukung

2.1.1 Definisi Graf

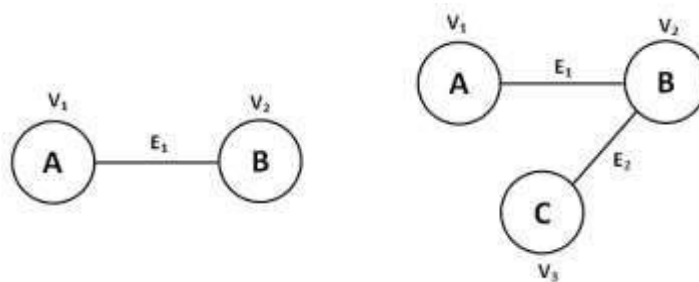
Suatu graf G dapat didefinisikan sebagai suatu himpunan (V, E) yang berpasangan dengan V , dan pasangan tersebut adalah himpunan yang tak kosong dan berhingga dari suatu obyek sering dikatakan sebagai suatu titik dan E merupakan himpunan pasangan yang tidak berurut di titik yang tidak sama di V , dan dapat dikatakan sebagai suatu sisi (Mandailina 2009).

2.1.2 Jenis-jenis Graf

Menurut ada atau tidaknya sisi ganda atau gelang yang ada pada sebuah graf, graf dapat dibagi dalam beberapa jenis (Zuhri 2016), yaitu :

a. Graf Sederhana

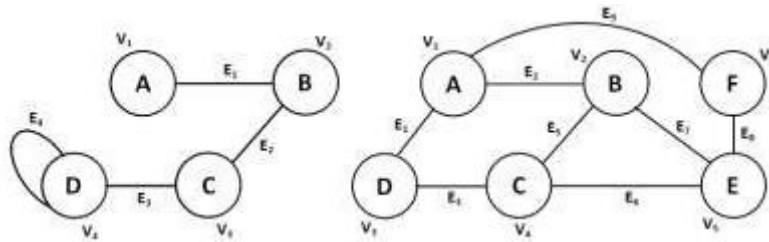
Graf Sederhana dapat didefinisikan sebagai suatu graf yang tidak mempunyai gelang (*loop*), serta hanya memiliki satu *vertex* (simpul) yang mempunyai garis paralel, atau suatu graf yang tidak mempunyai garis paralel. Garis paralel adalah suatu kondisi dimana satu simpul dapat terhubung kepada satu garis atau lebih. Garis gelang merupakan suatu kondisi ketika suatu garis yang dimulai dari suatu simpul dan akan kembali lagi ke simpul awal.



Gambar 2.1 Contoh dari Graf Sederhana

b. Graf Tak Sederhana

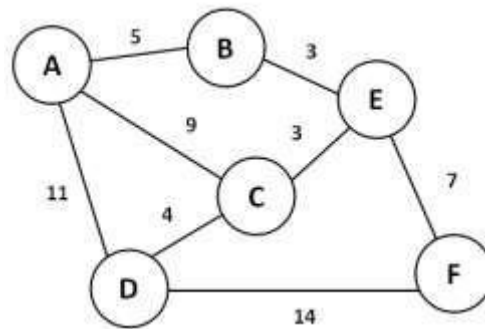
Graf tidak sederhana didefinisikan sebagai suatu graf yang mempunyai simpul lebih dari satu yang terhubung ke suatu garis paralel dan di salah satu simpulnya memiliki gelang.



Gambar 2.2 Contoh dari Graf Tidak Sederhana

c. Graf Berbobot

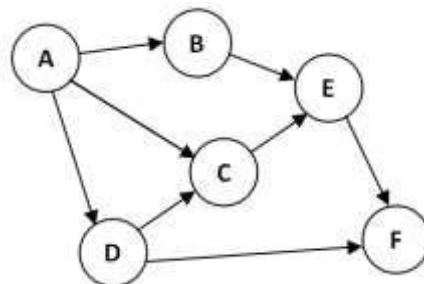
Graf Berbobot merupakan graf yang di tiap garisnya (*edge*) memiliki sebuah nilai tertentu. Pada setiap garis, nilainya dapat tak sama dan tergantung ke model suatu permasalahan dan di interpretasikan melalui suatu graf. Implementasinya dapat dipakai dalam penataan jalur lalu lintas dimana nilai dari jarak antara dua lokasi pada suatu daerah yang menjadi suatu nilai bobot pada rute lalu lintas. Misalkan suatu graf mewakili suatu denah dari jalur umum antar daerah, maka jarak antar daerah yang dihubungkan oleh *edge* tersebut ditunjukkan oleh suatu bilangan pada *edge*.



Gambar 2.3 Contoh dari Graf Berbobot

d. Graf Berarah

Graf berarah didefinisikan sebagai suatu graf yang mempunyai arah tujuan dari suatu simpul menuju ke simpul lain di tiap sisinya. Setiap pasang *verteks* yang terurut dapat dihubungkan oleh *edge*, misalnya ada suatu pasangan *verteks* terurut (C,D) yang dihubungkan oleh *edge* b , maka bisa juga ditulis dengan notasi $b=(C,D)$, yang menjelaskan suatu *edge* dari C ke D .



Gambar 2.4 Contoh dari Graf berarah

2.1.3 Lintasan Terpendek

Salah satu permasalahan yang bisa dipecahkan menggunakan graf adalah permasalahan tentang Lintasan terpendek. Dalam suatu graf berbobot, permasalahan lintasan terpendek adalah suatu cara kita untuk mencari suatu jalur pada graf yang dapat membuat jumlah bobot dari sisi pembentuk jalur tersebut menjadi seminimal mungkin.

Ada beberapa macam tipe permasalahan tentang lintasan terpendek, yaitu:

- a. Lintasan terpendek yang menghubungkan dua buah simpul tertentu.
- b. Lintasan terpendek yang menghubungkan semua pasangan simpul.
- c. Lintasan terpendek yang menghubungkan simpul tertentu ke semua simpul yang lainnya.
- d. Lintasan terpendek antara dua simpul yang melewati beberapa simpul tertentu.

Ada beberapa jenis algoritma yang dapat digunakan untuk memecahkan persoalan ini, diantaranya yaitu algoritma *Dijkstra*, algoritma *Bellman-Ford*, dan algoritma *Floyd-Warshall*. Tiap-tiap algoritma yang bisa menyelesaikan permasalahan lintasan terpendek memiliki fungsi dan kriterianya tersendiri.

2.1.4 Algoritma A-Star

Algoritma *A-Star* merupakan algoritma yang menerapkan perhitungan nilai fungsi heuristik, algoritma ini mengabaikan langkah yang dapat menghambat jalannya operasi algoritma dengan asumsi bahwa langkah-langkah yang diabaikan adalah langkah yang tidak efisien untuk mencapai solusi yang diharapkan dengan menggunakan suatu nilai heuristik. Nilai heuristik adalah nilai yang diberikan pada setiap simpul serta memudahkan algoritma *A-Star* dalam mencapai solusi yang efisien. Dengan nilai heuristik, maka algoritma *A-Star* pasti akan mendapatkan solusi (jika memang ada solusinya) (Kusumadewi 2005).

A-Star adalah salah contoh algoritma perhitungan rute terpendek yang paling efisien pada suatu graf yang dapat membuat solusi jalur dengan *cost* paling sedikit dari titik koordinat awal sampai ke titik koordinat yang diinginkan. Perhitungan algoritma ini mengaplikasikan fungsi *distance-plus-cost* (di

notasikan dengan $f(n)$) untuk memilih urutan kunjungan pememtuatan node pada suatu Graf. *Distance-plus-cost* didapatkan dengan menjumlahkan dua fungsi, yakni $h(n)$ dan $g(n)$. Fungsi *path-cost* yang dinotasikan dengan $h(n)$. Fungsi ini merupakan penjumlahan biaya (beban jarak tempuh) yang dibutuhkan dari node awal menuju node yang diinginkan. Fungsi kedua fungsi heuristik, fungsi ini merupakan nilai jarak yang dikalkulasikan dari titik ke- $n+1$ menuju titik yang diinginkan, dinotasikan dengan $g(n)$.

Nilai heuristik dapat dimanfaatkan dengan tujuan untuk meminimalisir pencarian ruang. Algoritma *A-Star* dapat mengkontruksi rute tercepat mulai dari tempat permulaan menuju tempat yang diinginkan. Algoritma ini mempunyai formula sebagai berikut :

$$f(n) = g(n) + h(n) \quad (1) \quad (2.1)$$

Keterangan:

$h(n)$ = biaya estimasi dari node n ke tujuan.

$g(n)$ = biaya path / perjalanan

$f(n)$ = solusi biaya estimasi termurah node n untuk mencapai tujuan.

$h(n)$ dapat dicari melalui perhitungan dengan rumus fungsi *Heuristic Euclidean Distance* (HED). Berikut merupakan rumus untuk mencari HED di mana a dan b merupakan koordinat dua titik yang akan dicari:

$$d(a,b) = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2} \quad (2.2)$$

Keterangan :

$d(a,b)$ = jarak antara titik a dan b

x = nilai absis suatu titik

y = nilai ordinat suatu titik

Semakin tepat nilai heuristik yang dipakai, maka semakin sedikit pula perulangan yang dihitung oleh algoritma *A-Star*, sehingga semakin cepat pula perhitungan bisa dilakukan. Diasumsikan jika nilai heuristik yang ditemukan tepat (selalu mengembalikan jarak rute minimum antara dua node sangat tepat), maka algoritma *A-Star* akan menghasilkan pada hasil perhitungan yang akurat sehingga algoritma bisa menjadi $O(p)$, di mana p adalah banyak jumlah langkah dalam *path* (rute). Pemilihan nilai heuristik yang akurat atau tidaknya tergantung pada apakah nilai heuristik yang telah di hitung tersebut menjadi terlalu rendah atau terlalu tinggi (Fernando, 2020).

2.1.5 Wisata Budaya

Pengertian Wisata Budaya adalah kegiatan wisata yang dipicu dengan tersedianya objek-objek wisata dan peninggalan berwujud hasil-hasil peninggalan seni budaya daerah sekitar, seperti adat istiadat, tradisi, upacara peringatan, perayaan agama, tata cara hidup setempat, peninggalan dan bangunan sejarah, seni kerajinan tangan, dan lain sebagainya.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa wisata budaya adalah salah satu jenis wisata yang menjadikan objek-objek wisata seperti adat istiadat, upacara-upacara, agama, tata hidup masyarakat setempat, peninggalan-peninggalan sejarah, hasil-hasil seni, kerajinan rakyat dan lain sebagainya, menjadi daya tarik tersendiri di setiap daerah.

2.2 Kajian Integrasi dengan Al-Qur'an dan Hadits

Islam sebagai agama *Rahmatan lil 'Alamin* telah mengatur dan menjelaskan segala aspek tentang tata cara yang baik dalam kehidupan manusia, salah satunya dalam aspek berhemat. Islam sangat berperan penting sebagai agama yang mulia

serta Al-Qur'an dan hadits yang berperan sebagai pedoman hidup dan pendukung dalam menjalani kehidupan yang sesuai dengan syari'at islam. Berpegang teguh kepada Al-Qur'an dan hadits adalah suatu kewajiban dan hal tersebut akan sangat berpengaruh dalam kehidupan kita, baik dalam bersikap maupun menyelesaikan masalah yang ada. Di zaman yang serba modern yang serba ada dan mudah ini, cenderung membuat pola hidup manusia menjadi boros dan hedonis dalam mencapai kepuasan diri. Agar terhindar dari pola hidup seperti itu, kita harus melatih diri untuk bisa berhemat. Seperti yang tercantum dalam Al-Qur'an surat Al-Isra' ayat 29 yang artinya:

“Dan janganlah engkau jadikan tanganmu terbelenggu pada lehermu dan jangan (pula) engkau terlalu mengulurkannya (sangat pemurah) nanti kamu menjadi tercela dan menyesal.” (QS. Al-Isra'/17:29).

Menurut tafsir Ibnu Katsir, dalam ayat tersebut Allah subhanahu wa ta'ala menginstruksikan pada hamba-hamba-Nya untuk bersikap hemat dan efisien dalam kehidupan, mencela sifat kikir, serta di waktu yang sama juga melarang sifat boros/berlebihan.

“Dan janganlah engkau jadikan tanganmu terbelenggu pada lehermu”

Dengan maksud, janganlah kita menjadi orang yang selalu menolak orang lain yang meminta bantuan kepada kita, serta tidak pernah sekalipun memberikan sesuatu kepada seseorang sebagai bentuk bantuan, serta jangan sampai kita menjadi orang yang kikir. Orang-orang yahudi mengatakan bahwa tangan Allah terbelenggu. Maksud mereka ialah Allah bersifat kikir, padahal kenyataannya Allah Maha Tinggi lagi Maha Suci, Maha Mulia dan Maha Pemberi.

“dan jangan (pula) engkau terlalu mengulurkannya”

Artinya janganlah kamu berlebihan dalam membelanjakan hartamu dengan

cara memberi di luar kemampuanmu dan mengeluarkan biaya lebih dari pemasukanmu.

“nanti kamu menjadi tercela dan menyesal”

Menurut riwayat Said Ibnu Manshur dalam Tafsir Al-Munir dari hadits yang diriwayatkan oleh Sayar Abdul Hakam, bahwa Rasulullah Shallallahu 'Alaihi wa Sallam pernah mendapatkan beberapa pakaian, namun Rasulullah adalah seorang yang sangat baik dan dermawan, sehingga beliau membagikan pakaian yang didapatnya tersebut kepada orang yang lebih membutuhkan. Kemudian ada suatu kaum menghampiri Rasulullah untuk meminta beberapa pakaian, namun pakaian itu telah habis dibagikan, maka Allah mewahyukan ayat tersebut.

Dalam ayat tersebut juga tersirat bahwa dalam menggunakan harta haruslah dalam hal-hal kebaikan dan sebisa mungkin menjauhi sikap kikir. Kemudian dalam memakai harta juga tidak diperbolehkan untuk membuatnya menjadi suatu kemubadziran. Hal ini membuat manusia untuk tidak dianjurkan bersikap berlebihan atau boros (Winaryo, 2011).

2.3 Kajian Penerapan Algoritma *A-Star* Dengan Teori Pendukung

Penerapan Algoritma *A-Star* bertujuan untuk menganalisis rute terpendek pada jalur destinasi wisata budaya yang ada di kota Yogyakarta. Dengan mengetahui penerapan algoritma *A-Star* untuk mencari rute terpendek di kota Yogyakarta, diharapkan dapat membantu wisatawan asing, maupun lokal, serta para pengembang aplikasi navigasi, untuk lebih memahami dan mengoptimalkan pertimbangan jarak dan tempat yang harus dikunjungi, agar para pengunjung dan wisatawan dapat lebih menghemat tenaga dan pikiran untuk bisa mengunjungi

banyak destinasi wisata budaya yang ada di kota Yogyakarta dengan lebih efisien. Dengan hal itu, diharapkan kedepannya pengunjung lokal maupun internasional dapat lebih meningkat di setiap tahunnya.

Penerapan algoritma *A-Star* dilakukan dengan beberapa langkah antara lain menentukan titik koordinat dari lokasi yang dicari, mengkonversi nilai koordinat derajat menjadi radian, menghitung dengan metode *Euclidean*, dengan menggunakan metode yang sama, hitung jarak node yang lain ke node tujuan untuk mencari nilai $h(n)$, mengkonversi nilai koordinat derajat menjadi radian, menghitung kembali dengan metode *Euclidean*, memasukkan nilai $g(n)$ dan $h(n)$ yang sudah didapat untuk mendapatkan nilai $f(n)$, melakukan perhitungan yang sama untuk node-node yang lain menggunakan metode yang seperti sebelumnya, dan melakukan eksperimen untuk menentukan rute terpendek.

Pada penelitian ini, algoritma *A-Star* akan dikaji lebih lanjut sehingga dapat menggambarkan rute terpendek melalui suatu graf dalam mengunjungi destinasi wisata budaya yang ada di kota Yogyakarta. Ada banyak sekali metode dan algoritma yang dapat digunakan untuk mencari rute terpendek dari suatu graf, algoritma *A-Star* adalah salah satunya. Padahal, ada banyak sekali wisata-wisata di Indonesia yang masih belum di modelkan untuk bisa dicari rute terpendeknya. Sehingga pada penelitian ini, algoritma *A-Star* akan dikaji ulang dimana akan ditentukan rute terpendek dari 20 destinasi wisata budaya kota Yogyakarta. Algoritma ini mempunyai nilai heuristik dalam perhitungannya, yakni suatu nilai yang didapatkan dari tiap simpul dan akan menyusun algoritma *A-Star* dalam mendapatkan solusi yang diinginkan. Algoritma ini juga membuang langkah-langkah yang tidak perlu untuk bisa mencapai solusi yang diinginkan dengan lebih

efisien.

Eksperimen untuk menentukan rute terpendek bertujuan untuk membandingkan hasil yang telah didapat pada penelitian sebelumnya atau aplikasi *google maps*. Graf yang diperoleh dari hasil eksperimen akan menggambarkan rute destinasi-destinasi budaya mana saja yang paling efisien untuk di kunjungi di kota Yogyakarta.

BAB III METODE PENELITIAN

3.1 Jenis Penelitian

Dalam penelitian ini akan dilakukan studi penelitian evaluasi secara kuantitatif dan pendekatan studi lapangan melalui *Google Maps* dan *Google Earth*. Studi penelitian evaluasi secara kuantitatif merupakan studi penelitian pada suatu sistem yang dijalankan dan bertujuan untuk melihat suatu hasil keefektifan sistem tersebut saat digunakan. Studi lapangan dapat dilakukan dengan mencari sumber-sumber informasi dengan mengumpulkan data dari obyek yang akan diteliti. Setelah melalui proses pengolahan data, kemudian didapatkan suatu simpulan secara kuantitatif dari pengujian sudah dilakukan.

3.2 Data dan Sumber Penelitian

Data yang dipakai sebagai obyek dalam penelitian ini adalah data sekunder yang didapatkan dari website resmi Dinas Pariwisata Kota Yogyakarta. <https://pariwisata.jogjakota.go.id/>.

3.3 Tahap Penelitian

Tahapan penelitian menggunakan algoritma *A-Star* tersusun dari beberapa tahap. Tahapan-tahapan yang harus dilakukan adalah sebagai berikut:

a. Mendapatkan Data Dari Google Maps Berupa Capture Jalur Yang Akan Dilewati

Tahap pertama yang harus dilakukan adalah mengamati dan mengumpulkan data obyek yang akan dijadikan sebagai bahan penelitian. Data yang dihimpun

berupa data jarak berdasarkan grid dan titik yang diperoleh dari data spasial *Google Maps*. Pada penelitian ini, ada beberapa hal penting yang akan dibahas, mulai dari penangkapan gambar (*capturing*) yang berupa screenshot titik-titik destinasi budaya, titik-titik lokasi yang sudah didapatkan akan dibuat menjadi node sebuah graf, edge pada graf diambil sekaligus dari rute perjalanan, dan menganalisa apakah edge tersebut berbentuk dua arah atau satu arah. Hal-hal tersebut akan menjadi pertimbangan ketika mengumpulkan dan menganalisa data yang dibutuhkan.

b. Menentukan dan Mengumpulkan Koordinat Dari Lokasi-Lokasi Destinasi Wisata Budaya Yang Diperlukan

Tahap penentuan titik dari tiap-tiap node adalah penentuan titik koordinat lokasi destinasi wisata budaya dan titik persimpangan jalan yang ditentukan dengan cara mengambil titik *longitude* dan *latitude* pada *Google Maps*. Setelah hasil dari penentuan tersebut didapatkan, hasilnya akan dijadikan menjadi acuan dalam menghitung nilai heuristik yang akan digunakan dalam proses pencarian rute terpendek menggunakan algoritma *A-Star*.

c. Menyusun Grid

Tahapan proses penyusunan grid dilakukan dengan mengilustrasikan seluruh data yang telah didapatkan, dimulai dari titik-titik dan sisi-sisi yang menghubungkan antar titik yang satu dengan titik yang lainnya. Sisi yang sudah terilustrasi akan ditandai dengan lebel bobot jarak. Perolehan lebel tersebut juga dibantu menggunakan aplikasi bantuan google maps. Jarak yang digunakan memakai satuan km (kilometer). Penentuan satuan km dinilai lebih familiar dengan para pembaca ketika akan membahas dan mempejari tentang jarak rute

terpendek untuk mengunjungi beberapa tempat sekaligus.

d. Menghitung Nilai Heuristik

Dalam tahapan pencarian dan perhitungan nilai heuristik, dibutuhkan suatu metode untuk mengevaluasi data dari suatu keadaan-keadaan tertentu untuk menentukan seberapa berpengaruh hal tersebut dapat digunakan untuk mendapatkan solusi yang efisien sesuatu data yang diinginkan. Fungsi heuristik dapat berbeda-beda tergantung titik tujuan yang diinginkan, sehingga fungsi ini seringkali menjadi parameter penting dalam mencari solusi dari suatu permasalahan. Fungsi heuristik di aplikasikan untuk mencari jarak dari dua buah *vertex*.

Tahapan perhitungan untuk mencari fungsi heuristik dari suatu graf dengan menggunakan algoritma *A-Star* yaitu, menentukan titik koordinat dari lokasi yang dicari, menghitung dengan metode *Euclidean*, dengan menggunakan metode yang sama, akan dihitung jarak node awal ke node tujuan untuk mendapatkan nilai $h(n)$, menghitung nilai dengan metode *Euclidean*, kemudian menjumlahkan nilai $g(n)$ dan $h(n)$ yang sudah didapat untuk mendapatkan nilai $f(n)$, dan akan dilakukan perhitungan yang sama untuk node-node yang lain, dan melakukan eksperimen untuk mencari rute terpendek kepada suatu studi kasus.

e. Melakukan Eksperimen untuk Menentukan Rute Terpendek

Pengumpulan data yang sudah dibentuk menjadi suatu grid melalui titik koordinat yang telah didapat, maka tahapan selanjutnya adalah melakukan eksperimen pengujian rute terpendek. Dalam melakukan suatu pengujian, akan ditentukan beberapa lokasi titik destinasi wisata budaya yang akan dilewati,

kemudian menentukan titik destinasi mana yang akan menjadi rute awal sampai akhir. Proses pengujian dilakukan beberapa kali sehingga bisa menjangkau pengujian untuk seluruh destinasi wisata budaya di Kota Yogyakarta. Penentuan rute dalam pengujian ini adalah dengan menggunakan algoritma *A-Star* serta penambahan seleksi destinasi terkhusus wisata budaya sebagai pengembangan dari penelitian sebelumnya.

BAB IV HASIL DAN PEMBAHASAN

4.1 Deskripsi Data

Dalam pengerjaannya, telah dikumpulkan beberapa data yang akan digunakan dalam penelitian ini, yang kemudian dibagi menjadi dua kategori, yaitu data dari titik jalan serta persimpangan dan data dari titik destinasi wisata budaya. Berikut merupakan penjelasan dari masing-masing kategori data :

1. Data Destinasi Wisata Budaya

Data dari titik-titik lokasi wisata budaya yang ada di kota Yogyakarta, berikut tabel penjabaran data tersebut.

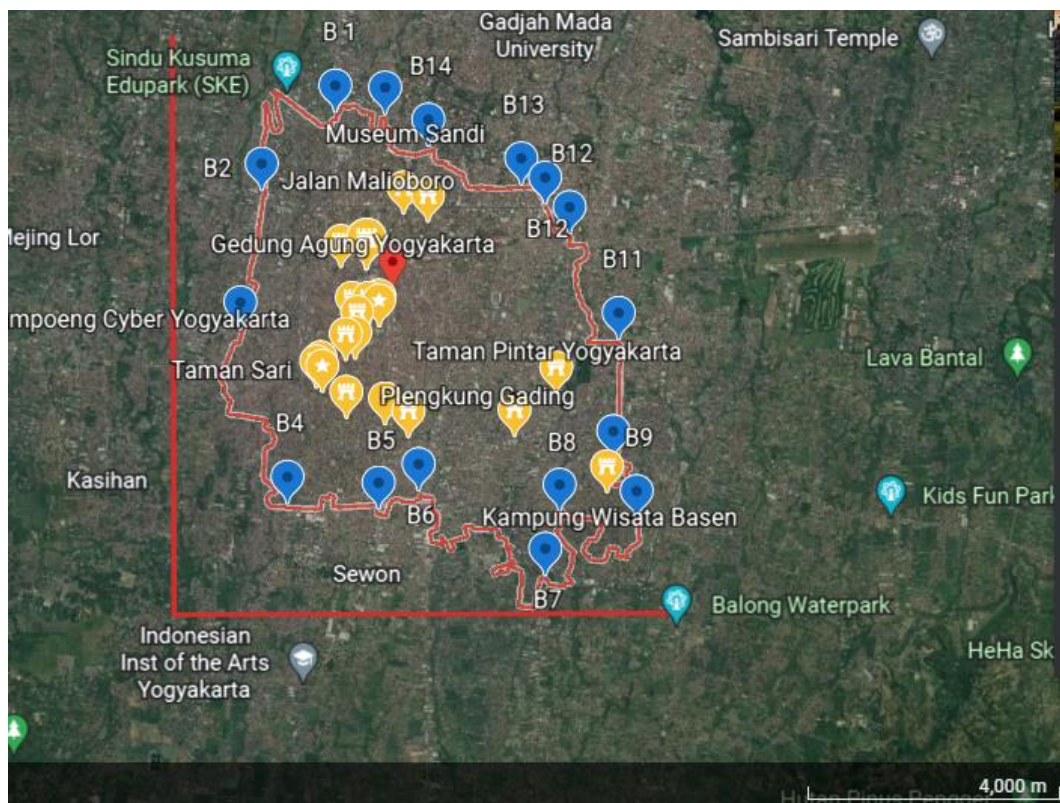
Tabel 4.1 Tabel Destinasi Wisata Budaya Kota Yogyakarta

No	Nama Destinasi	Simbol
1	Kraton Yogyakarta	W1
2	Taman Pintar Yogyakarta	W2
3	Kampung Wisata Taman Sari	W3
4	Situs Warungboto	W4
5	Museum Sonobudoyo Unit I	W5
6	Bentara Budaya Yogyakarta	W6
7	Taman Budaya Yogyakarta	W7
8	Kampoeng Cyber Yogyakarta	W8
9	Wisata Malioboro	W9
10	De Mataa Museum Jogja	W10
11	Plengkung Nirbaya Gading	W11
12	Sumur Gumuling	W12
13	Kampung Wisata Sosromenduran	W13
14	Museum Perjuangan	W14
15	Museum Kereta Keraton Yogyakarta	W15
16	Museum Sandi	W16
17	Gedung Agung Yogyakarta	W17
18	Pojok Benteng Wetan	W18
19	Kampung Wisata Basen	W19
20	Museum Benteng Vredeburg	W20

Titik-titik wisata budaya di atas disimbolkan dengan huruf W. Hal ini dilakukan untuk menjadi penanda antara titik-titik persimpangan yang akan dipakai didalam data penelitian. Pendanda simbol tersebut digunakan agar pemilihan data lokasi yang diinginkan menjadi lebih mudah.

2. Data Titik Persimpangan Jalan

Data titik persimpangan jalan pada penelitian ini diambil dari jenis jalan yang berfokus pada jalan utama. Gambaran jalan yang dijadikan acuan pengambilan data tersaji dalam gambar 4.1 yang didapatkan dari website <http://geoportal.jogjakota.go.id/pencarian?kategori=Transportasi>.



Gambar 4.1 Screenshoot data Tempat dan Jalur Kota Yogyakarta

Titik-titik kuning dari gambar di atas menyimbolkan titik-titik desinasi wisata budaya yang dipilih dalam penelitian ini. Titik-titik biru menyimbolkan

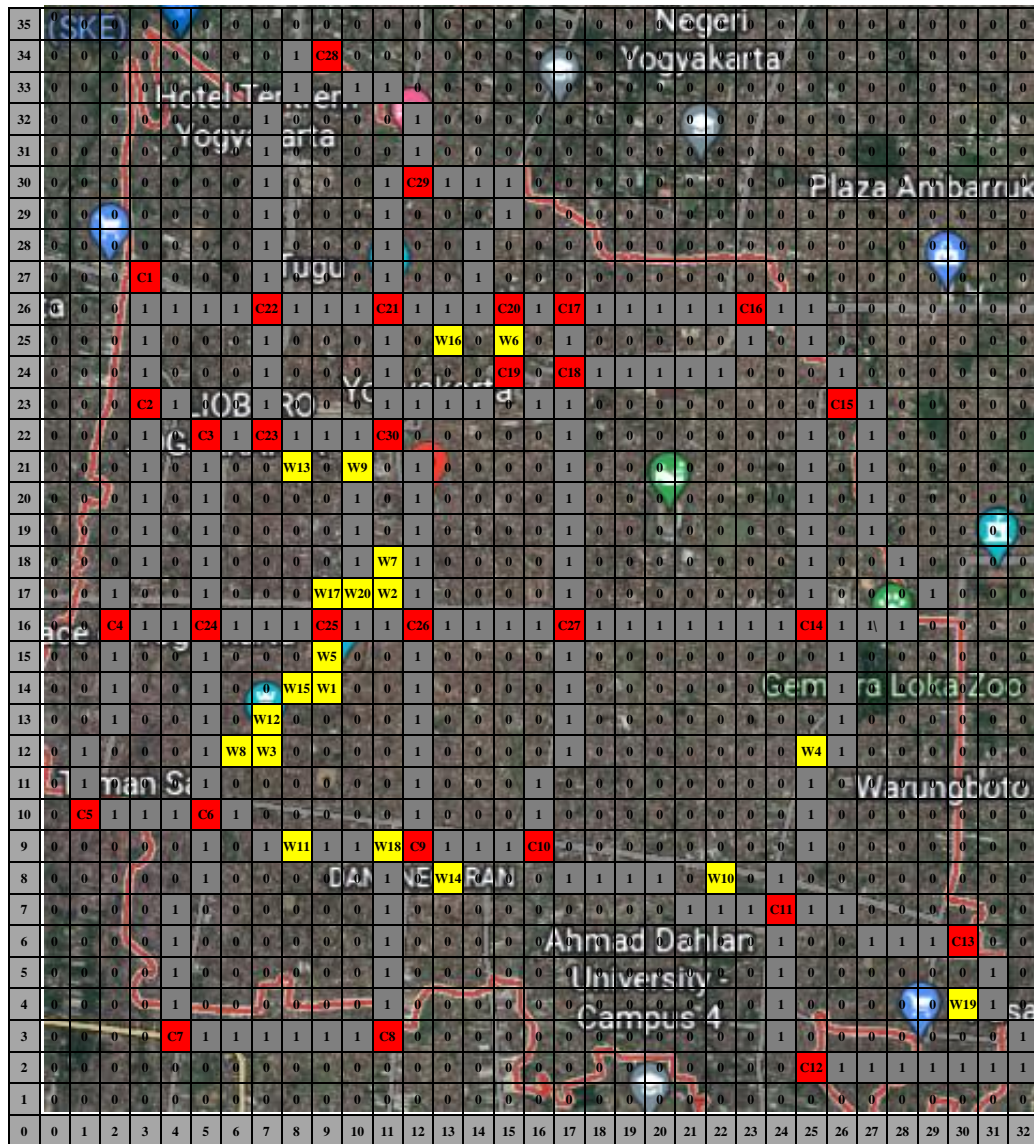
perbatasan jalan besar antara provinsi dan kota yogyakarta. Garis merah menyimbolkan batasana secara menyeluruh antara kota dan provinsi yogyakarta.

Setelah dilakukan pemilihan data dari google maps, didapatkan 30 titik jalan utama yang menjadi jalan bercabang. Setiap titik disimbolkan dengan kode huruf C sesuai banyaknya persimpangan.

3. Pembentukan Grid

Setelah diketahui gambaran umum mengenai rute dan persimpangan jalan menuju Kecamatan Menes berdasarkan data spasial Google Maps, maka langkah selanjutnya adalah mengubahnya ke dalam bentuk titik-titik koordinat dalam grid. Di dalam grid-grid tersebut ditampilkan node berdasarkan data google maps. Titik grid warna merah menyimbolkan persimpangan, titik grid warna kuning menyimbolkan tempat wisata, dan grid warna abu-abu melambangkan jalan utama yang bisa dilalui. Dari data node tersebut kemudian akan dijadikan data perhitungan menggunakan algoritma *A-Star*. Angka 1 yang ada d idalam masing-masing kotak grid menyimbolkan jalan yang dilalui bernilai 1 grid. Angka 0 yang ada di dalam kotak grid menyimbolkan jalan utama yang tidak dilalui.

Gambar 4.2 Pembentukan Grid berdasarkan Screenshot pada Maps



yang telah dibatasi pada batasan masalah. Adapun nama-nama setiap persimpangan beserta indeksnya tersaji dalam tabel berikut:

Tabel 4.2 Data Titik-Titik Indeks dari Setiap Persimpangan

Simbol	Keterangan	Indeks	Simbol	Keterangan	Indeks
C1	Persimpangan 1	(3,27)	C16	Persimpangan 16	(23,26)
C2	Persimpangan 2	(3,23)	C17	Persimpangan 17	(17,26)
C3	Persimpangan 3	(5,22)	C18	Persimpangan 18	(17,24)
C4	Persimpangan 4	(2,16)	C19	Persimpangan 19	(15,24)
C5	Persimpangan 5	(1,10)	C20	Persimpangan 20	(15,26)
C6	Persimpangan 6	(5,10)	C21	Persimpangan 21	(11,26)
C7	Persimpangan 7	(4,3)	C22	Persimpangan 22	(7,26)
C8	Persimpangan 8	(11,3)	C23	Persimpangan 23	(7,22)
C9	Persimpangan 9	(12,9)	C24	Persimpangan 24	(5,16)
C10	Persimpangan 10	(16,9)	C25	Persimpangan 25	(9,16)
C11	Persimpangan 11	(24,7)	C26	Persimpangan 26	(12,16)
C12	Persimpangan 12	(25,2)	C27	Persimpangan 27	(17,16)
C13	Persimpangan 13	(30,6)	C28	Persimpangan 28	(9,34)
C14	Persimpangan 14	(25,16)	C29	Persimpangan 29	(12,30)
C15	Persimpangan 15	(26,23)	C30	Persimpangan 30	(11,22)

Destinasi wisata budaya yang saling berdekatan dengan radius 2x3 grid akan dijadikan sebagai satu *cluster* karena destinasi-destinasi tersebut bisa dikunjungi sekaligus ketika berkunjung ke salah satu destinasi wisatanya. Pengelompokan *cluster* dibuat khusus untuk mewakili destinasi-destinasi yang berdekatan, sebagaimana yang ada pada tabel berikut.

4.2 Pengujian Pencarian Rute Terpendek

Pengujian rute terpendek dari beberapa sampel titik awal akan dilakukan sesuai dengan tahapan-tahapan penelitian menggunakan Algoritma *A-Star* yang telah dijelaskan pada bab 3. Berikut merupakan alur yang dilakukan dalam penelitian untuk menghitung pengujian rute terpendek:

a. Memilih Titik Awal dan Titik Tujuan

Langkah awal pada penelitian ini adalah memilih titik awal perjalanan. Titik awal ini dipilih berdasarkan titik yang menjadi jalan utama dan ramai dikunjungi, yakni C1, C2, C4, C7, C8, C12, C14, C20, C28, C29. Namun, penentuan titik-titik awal harus memerhatikan keadaan tiap persimpangan dan juga batas daerah pada setiap batasan kota dan provinsi Yogyakarta. Batas pemilihan destinasi wisata budaya yang dapat dikunjungi dilihat dari jumlah titik pada deskripsi data yang telah dipaparkan sebelumnya.

Penentuan lokasi destinasi wisata bisa dipilih untuk beberapa lokasi wisata/*cluster* sekaligus. Hal ini dikarenakan jumlah data lokasi destinasi wisata budaya yang ada pada data penelitian hanya berjumlah 20 destinasi. Format untuk penulisan beberapa titik wisata adalah dipisahkan dengan tanda koma (.). Sampel penentuan beberapa titik awal dan beberapa sampel lokasi tujuan tersaji dalam tabel 4.8.

Tabel 4.5 Sampel Titik Awal dan Titik Tujuan

Titik Awal	Titik-titik Tujuan Destinasi
C1	W13, W9, W16
C2	CA, CB
C4	CC, CD
C7	CE, CF
C8	W4, W10

b. Penentuan Rute

Penentuan rute terpendek akan dilakukan dengan menyeleksi titik awal terhadap setiap lokasi titik tujuan menggunakan algoritma *A-Star*. Berikut merupakan penjelasan dan perhitungan dari beberapa sampel titik awal dan titik tujuan dalam perhitungan dan penentuan rute paling efisien berdasarkan contoh pada poin a.

a) Titik Awal C1

- **Penentuan Rute C1 → W13**

C1 merupakan titik awal permulaan untuk menuju destinasi, sehingga C1 bisa dimasukkan ke dalam *List* Tutup.

List Tutup = [['C1'], 0]

Setelah itu akan dicari titik mana saja yang rute jalan utamanya terhubung dengan C1 dan perhitungan nilai heuristik masing-masing titik tersebut. Nilai heuristik bisa didapatkan melalui perhitungan rumus jarak *Euclid* pada persamaan (2.2). Perhitungan ini akan menggunakan titik-titik koordinat yang ada pada visualisasi grid yang telah dibentuk. C1 terhubung dengan C2 dan C22, kemudian diperoleh hasil perhitungan dari grid berupa panjang rute dan nilai heuristiknya sebagai berikut:

Terpilih = {'C2':(4,5.4), 'C22':(4,5.1)}

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C2) &= g(C2) + h(C2) \\ &= (0 + 4) + 5.4 \\ &= 9.4 \end{aligned}$$

$$\begin{aligned}
 f(C22) &= g(C22) + h(C22) \\
 &= (0 + 4) + 5.1 \\
 &= 9.1
 \end{aligned}$$

Hasil yang didapatkan dari perhitungan diatas menunjukkan bahwa C22 memiliki hasil perhitungan yang minimum, sehingga hasil minimum tersebut bisa dimasukkan ke dalam *list* tutup menjadi:

$$List\ Tutup = [[\text{'C1'}, \text{'C22'}], 4]$$

Sama dengan proses sebelumnya, akan dicari titik mana saja yang rute jalan utamanya terhubung dengan C22, kemudian didapatkan hasil:

$$Terpilih = \{\text{'C21'}:(4,5.8), \text{'C28'}:(9,13), \text{'C23'}:(4,1.4)\}$$

$$\begin{aligned}
 f(C21) &= g(C21) + h(C21) \\
 &= (4+4) + 5.8 \\
 &= 13.8
 \end{aligned}$$

$$\begin{aligned}
 f(C28) &= g(C28) + h(C28) \\
 &= (4+9) + 13 \\
 &= 26
 \end{aligned}$$

$$\begin{aligned}
 f(C23) &= g(C23) + h(C23) \\
 &= (4+4) + 1.4 \\
 &= 9.4
 \end{aligned}$$

Melalui proses perhitungan Algoritma A-Star, didapatkan hasil bahwa pilihan rute yang memiliki perhitungan minimum adalah titik C23. Persimpangan C23 bisa dimasukkan dalam *list* tutup.

$$List\ Tutup = [[\text{'C1'}, \text{'C22'}, \text{'C23'}], 8]$$

C23 terhubung dengan C3, C30 dan W13. Panjang rute dan nilai heuristiknya adalah:

Terpilih = {'C3': (2,2), 'C30':(6,6), 'W13':(1,0)}

$$\begin{aligned} f(C30) &= g(C30) + h(L8) \\ &= (8+4) + 4 \\ &= 16 \end{aligned}$$

$$\begin{aligned} f(W13) &= g(W13) + h(W13) \\ &= (8+1) + 0 \\ &= 9 \end{aligned}$$

$$\begin{aligned} f(C3) &= g(C3) + h(C3) \\ &= (8+2) + 2 \\ &= 12 \end{aligned}$$

W13 terpilih karena memiliki hasil nilai perhitungan yang paling minimum sekaligus menjadi destinasi akhir yang berarti telah sampai pada destinasi wisata yang di inginkan. Sehingga didapatkan hasil:

List Tutup = [['C1', 'C22', 'C23', 'W13'], 9]

- **Penentuan Rute C1 → W16**

Proses dalam pencarian rute kali ini mirip dengan perhitungan rute sebelumnya. C1 merupakan titik awal permulaan untuk menuju destinasi, sehingga C1 bisa dimasukkan ke dalam *List* Tutup.

List Tutup = [['C1'], 0]

Selanjutnya akan dicari titik mana saja yang rute jalan utamanya terhubung dengan C1 dan perhitungan nilai heuristik masing-masing titik tersebut, setelah

perhitungan didapatkan hasil berupa panjang rute dan nilai heuristiknya berdasarkan grid sebagai berikut:

Terpilih = {'C2':(4 ,10.2), 'C22':(4 ,4.1)}

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C2) &= g(C2) + h(C2) \\ &= (0 + 4) + 10.2 \\ &= 14.2 \end{aligned}$$

$$\begin{aligned} f(C22) &= g(C22) + h(C22) \\ &= (0 + 4) + 4.1 \\ &= 8.1 \end{aligned}$$

Hasil perhitungan yang diperoleh menunjukkan bahwa C22 merupakan hasil perhitungan yang paling minimum, sehingga dapat dimasukkan ke dalam *list* tutup menjadi:

List Tutup = [['C1', 'C22'], 4]

Sama halnya dengan proses perhitungan sebelumnya, akan dicari titik mana saja yang rute jalan utamanya terhubung dengan C22, kemudian didapatkan hasil:

Terpilih = {'C21':(4,2.2), 'C23':(4,7.1), 'C28':(9,9.8)}

$$\begin{aligned} f(C21) &= g(C3) + h(C3) \\ &= (4+4) + 2.2 \\ &= 10.2 \end{aligned}$$

$$\begin{aligned} f(C23) &= g(C4) + h(C4) \\ &= (4+4) + 7.1 \end{aligned}$$

$$= 15.1$$

$$f(C28) = g(C28) + h(C28)$$

$$= (4+9) + 9.8$$

$$= 22.8$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C21. C21 kita masukkan dalam *list* tutup.

List Tutup = [['C1', 'C22', 'C21'], 8]

C21 terhubung dengan W16, C30 dan C20. Panjang rute dan nilai heuristiknya adalah:

Terpilih = {'W16': (3, 0), 'C30': (4, 3.6), 'C20': (4, 2.1)}

$$f(W16) = g(W16) + h(W16)$$

$$= (8+3) + 0$$

$$= 11$$

$$f(C30) = g(C30) + h(C30)$$

$$= (8+4) + 3.6$$

$$= 15.6$$

$$f(C20) = g(C20) + h(C20)$$

$$= (8+4) + 2.1$$

$$= 14.1$$

W16 terpilih karena memiliki nilai perhitungan yang paling minum sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi tujuan. Sehingga didapatkan hasil rute terpendeknya:

List Tutup = [['C1', 'C22', 'C21', 'W16], 11]

- **Penentuan Rute C1 → W9**

C1 merupakan titik awal permulaan untuk menuju destinasi, sehingga C1 bisa dimasukkan ke dalam *List* Tutup.

$$List\ Tutup = [['C1'], 0]$$

Selanjutnya akan dicari titik mana saja yang terhubung dengan C1 dan berapa nilai masing-masing heuristik titik tersebut. C1 terhubung dengan C2 dan C22, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

$$Terpilih = \{ 'C2':(4, 7.2), 'C22':(4, 5.8) \}$$

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C2) &= g(C2) + h(C2) \\ &= (0 + 4) + 7.2 \\ &= 11.2 \end{aligned}$$

$$\begin{aligned} f(C22) &= g(C22) + h(C22) \\ &= (0 + 4) + 5.8 \\ &= 9.8 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C22 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$$List\ Tutup = [['C1', 'C22'], 4]$$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C22, kemudian didapatkan hasil:

$$Terpilih = \{ 'C21':(4,5.1), 'C28':(9,13), 'C23':(4,3.1) \}$$

$$f(C21) = g(C21) + h(C21)$$

$$= (4+4) + 5.1$$

$$= 13.1$$

$$f(C28) = g(C28) + h(C28)$$

$$= (4+9) + 13$$

$$= 26$$

$$f(C23) = g(C23) + h(C23)$$

$$= (4+4) + 3.1$$

$$= 11.1$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C23. C23 kita masukkan dalam *list* tutup.

List Tutup = [['C1', 'C22', 'C23'], 8]

C23 terhubung dengan C3, C30 dan W9. Kemudian kita ketahui informasi di dalamnya tentang bobot dan nilai heuristiknya adalah:

Terpilih = { 'C3':(2.5.1), 'C30': (4 ,1.4), 'W9':(4,0) }

$$f(C3) = g(C3) + h(C3)$$

$$= (8+2) + 5.1$$

$$= 15.1$$

$$f(C30) = g(C30) + h(L8)$$

$$= (8+4) + 1.4$$

$$= 13.4$$

$$f(W9) = g(W9) + h(W9)$$

$$= (8+4) + 0$$

$$= 12$$

W9 terpilih karena memiliki nilai heuristik 0 dan menghasilkan perhitungan yang paling minum, sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi tujuan. Sehingga didapatkan hasil:

List Tutup = [['C1','C22','C23','W9'], 12]

Berdasarkan perhitungan di atas, antara titik awal yaitu C1 terhadap masing-masing titik didapatkan bahwa destinasi terdekat untuk dikunjungi terlebih dulu adalah W13. Sehingga W13 dapat kita masukkan dalam *list* rute.

List Tutup = [['C1','C22','C23','W13'], 9]

Langkah selanjutnya adalah menyeleksi dari titik W13 ke titik W9 dan W16. Akan ditentukan destinasi mana yang memiliki rute terpendek untuk dikunjungi terlebih dahulu.

- **Penentuan Rute W13 → W9**

W13 terhubung dengan titik C23, C30 dan W9. Kedua titik tersebut mengandung informasi sebagai berikut:

Terpilih = {'C23':(2,2.2), 'C30':(4,1.4), 'W9':(4,0)}

$$\begin{aligned} f(C23) &= g(C23) + h(C23) \\ &= (0+4) + 2.2 \\ &= 6.2 \end{aligned}$$

$$\begin{aligned} f(C30) &= g(C30) + h(C30) \\ &= (0 + 4) + 1.4 \\ &= 5.4 \end{aligned}$$

$$\begin{aligned} f(W9) &= g(W9) + h(W9) \\ &= (0 + 4) + 0 \\ &= 4 \end{aligned}$$

W9 terpilih karena memiliki nilai perhitungan yang paling minum dan juga memiliki nilai heuristik 0 yang menandakan bahwa perhitungan telah sampai pada lokasi yang diinginkan. Sehingga didapatkan hasil:

List Tutup = [['W13', 'W9'], 4]

- **Penentuan Rute W13 → W16**

W13 terhubung dengan titik C23 dan C30. Kedua titik tersebut mengandung informasi sebagai berikut:

Terpilih = {'C23':(2,6.7), 'C30':(4,4.2)}

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C23) &= g(C23) + h(C23) \\ &= (0 + 2) + 6.7 \\ &= 9.7 \end{aligned}$$

$$\begin{aligned} f(C30) &= g(C30) + h(C30) \\ &= (0 + 4) + 4.2 \\ &= 8.2 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C30 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

List Tutup = [['W13', 'C30'], 4]

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C30, kemudian didapatkan hasil:

Terpilih = {'C21':(4,2.2), 'C26':(6,13.3), 'C19':(5,2.2)}

$$\begin{aligned} f(C21) &= g(C21) + h(C21) \\ &= (4+4) + 2.2 \end{aligned}$$

$$\begin{aligned}
 &= 10.2 \\
 f(C26) &= g(C26) + h(C26) \\
 &= (4+6) + 13.3 \\
 &= 23.3 \\
 f(C19) &= g(C19) + h(C19) \\
 &= (4+5) + 2.2 \\
 &= 11.2
 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C21. C21 kita masukkan dalam *list* tutup.

$$List\ Tutup = [['W13', 'C30', 'C21'], 8]$$

Selanjutnya, akan dicari titik yang terhubung dengan C21, kemudian didapatkan hasil:

$$Terpilih = \{ 'C22':(4,6.1), 'C29':(5,5.1), 'C20':(4,2.2), 'W16':(3,0) \}$$

$$\begin{aligned}
 f(C22) &= g(C22) + h(C22) \\
 &= (8+4) + 6.1 \\
 &= 18.1 \\
 f(C29) &= g(C29) + h(C29) \\
 &= (8+5) + 5.1 \\
 &= 18.1 \\
 f(C20) &= g(C20) + h(C20) \\
 &= (8+4) + 2.2 \\
 &= 14.2 \\
 f(W16) &= g(W16) + h(W16) \\
 &= (8+3) + 0
 \end{aligned}$$

$$= 11$$

W16 terpilih karena memiliki nilai perhitungan yang paling minum dan juga memiliki nilai heuristik 0 yang menandakan bahwa perhitungan telah sampai pada lokasi yang diinginkan. Sehingga didapatkan hasil:

$$List\ Tutup = [[\text{'W13'}, \text{'C30'}, \text{'C21'}, \text{'W16'}], 11]$$

Berdasarkan perhitungan di atas, dari titik W13, rute terdekat yang dapat dikunjungi terlebih dahulu adalah destinasi wisata budaya W9. Sehingga kita dapat masukkan W9 ke dalam list rute.

$$List\ Rute = [[\text{'C1'}, \text{'C22'}, \text{'C23'}, \text{'W13'}], [\text{'W13'}, \text{'W9'}], 13]$$

Destinasi yang tersisa adalah destinasi W16. Pencarian rute menuju W16 akan dimulai dari titik W9. Di sini tidak akan dilakukan perbandingan kembali karena W16 menjadi destinasi terakhir.

- **Perhitungan Rute W9 → W16**

W9 terhubung dengan titik C23 dan C30. Kedua titik tersebut memuat informasi sebagai berikut:

$$Terpilih = \{\text{'C23'}:(4,6.7), \text{'C30'}:(2,4.2)\}$$

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C23) &= g(C23) + h(C23) \\ &= (0 + 4) + 6.7 \\ &= 10.7 \end{aligned}$$

$$\begin{aligned} f(C30) &= g(C30) + h(C30) \\ &= (0 + 2) + 4.2 \\ &= 6.2 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C30 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

List Tutup = [['W9', 'C30'], 2]

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C30, kemudian didapatkan hasil:

Terpilih = {'C21':(4,2.2), 'C26':(6,13.3), 'C19':(5,2.2)}

$$\begin{aligned} f(C21) &= g(C21) + h(C21) \\ &= (2+4) + 2.2 \\ &= 8.2 \end{aligned}$$

$$\begin{aligned} f(C26) &= g(C26) + h(C26) \\ &= (2+6) + 13.3 \\ &= 21.3 \end{aligned}$$

$$\begin{aligned} f(C19) &= g(C19) + h(C19) \\ &= (2+5) + 2.2 \\ &= 9.2 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C21. C21 kita masukkan dalam *list* tutup.

List Tutup = [['W9', 'C30', 'C21'], 6]

Selanjutnya, akan dicari titik yang terhubung dengan C21, kemudian didapatkan hasil:

Terpilih = {'C22':(4,6.1), 'C29':(5,5.1), 'C20':(4,2.2), 'W16':(3,0)}

$$\begin{aligned} f(C22) &= g(C22) + h(C22) \\ &= (6+4) + 6.1 \\ &= 16.1 \end{aligned}$$

$$\begin{aligned}
 f(C29) &= g(C29) + h(C29) \\
 &= (6+5) + 5.1 \\
 &= 16.1
 \end{aligned}$$

$$\begin{aligned}
 f(C20) &= g(C20) + h(C20) \\
 &= (6+4) + 2.2 \\
 &= 12.2
 \end{aligned}$$

$$\begin{aligned}
 f(W16) &= g(W16) + h(W16) \\
 &= (6+3) + 0 \\
 &= 9
 \end{aligned}$$

W16 terpilih karena memiliki nilai perhitungan yang paling minum dan juga memiliki nilai heuristik 0 yang menandakan bahwa perhitungan telah sampai pada lokasi yang diinginkan. Sehingga didapatkan hasil:

List Tutup = [['W9', 'C30', 'C21', 'W16'], 11]

Rute terakhir yang ada pada *list* tutup di atas dapat dimasukkan ke dalam *list* rute. Kemudian dijumlahkan total jarak minimum yang dapat ditempuh.

List Rute

List Rute = [['C1', 'C22', 'C23', 'W13'], ['W13', 'W9'], ['W9', 'C30', 'C21', 'W16'], 24]

Berdasarkan list rute di atas, melalui titik C1, Destinasi pertama yang menjadi tujuan adalah W13. Setelah itu dilanjutkan ke destinasi kedua yaitu W9. Setelah melewati W9, perjalanan dilanjutkan ke destinasi terakhir yaitu W16.

b) Titik Awal C2

- **Penentuan Rute C2 → CA**

C2 merupakan titik awal permulaan untuk menuju destinasi, sehingga C2 bisa dimasukkan ke dalam *List* Tutup.

$$List\ Tutup = [[\text{'C2'}], 0]$$

Selanjutnya akan dicari titik mana saja yang terhubung dengan C2 dan berapa nilai masing-masing heuristik titik tersebut. Nilai heuristik diambil melalui perhitungan jarak *Euclid* pada persamaan (2.2). Perhitungan ini diambil dari titik koordinat yang ada pada gambar visualisasi graf. C2 terhubung dengan C3 dan C4, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

$$Terpilih = \{\text{'C3'}:(2, 10.2), \text{'C4'}:(7, 4.5)\}$$

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C3) &= g(C3) + h(C3) \\ &= (0 + 2) + 10.2 \\ &= 12.2 \end{aligned}$$

$$\begin{aligned} f(C4) &= g(C4) + h(C4) \\ &= (0 + 7) + 4.5 \\ &= 11.5 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C4 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$$List\ Tutup = [[\text{'C2'}, \text{'C4'}], 7]$$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C4, kemudian didapatkan hasil:

$$\text{Terpilih} = \{ 'C24':(3,2.2), 'C5':(5,6.3) \}$$

$$\begin{aligned} f(C24) &= g(C24) + h(C24) \\ &= (7+3) + 2.2 \\ &= 12.2 \end{aligned}$$

$$\begin{aligned} f(C5) &= g(C5) + h(C5) \\ &= (7+5) + 6.3 \\ &= 18.3 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C24. C24 kita masukkan dalam *list* tutup.

$$\text{List Tutup} = [['C2', 'C4', 'C24'], 10]$$

C24 terhubung dengan CA, C6 dan C25. Panjang rute dan nilai heuristiknya adalah:

$$\text{Terpilih} = \{ 'CA':(6,0), 'C6':(6,2.8), 'C25':(4,4.5) \}$$

$$\begin{aligned} f(CA) &= g(CA) + h(CA) \\ &= (10+6) + 0 \\ &= 16 \end{aligned}$$

$$\begin{aligned} f(C6) &= g(C6) + h(C6) \\ &= (10+6) + 2.8 \\ &= 18.8 \end{aligned}$$

$$\begin{aligned} f(C25) &= g(C25) + h(C25) \\ &= (10+4) + 4.5 \\ &= 18.5 \end{aligned}$$

CA terpilih karena memiliki nilai perhitungan yang paling minum sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi *cluster* tujuan. Sehingga didapatkan hasil:

$List\ Tutup = [[\text{'C2'}, \text{'C4'}, \text{'C24'}, \text{'CA'}], 16]$

- **Penentuan Rute C2 → CB**

Proses dalam pencarian rute kali ini mirip dengan perhitungan rute sebelumnya. C2 merupakan titik awal permulaan untuk menuju destinasi, sehingga C2 bisa dimasukkan ke dalam *List Tutup*.

$List\ Tutup = [[\text{'C2'}], 0]$

Selanjutnya akan dicari titik mana saja yang terhubung dengan C2 dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

$Terpilih = \{\text{'C3'}:(2, 8.9), \text{'C4'}:(7, 7.3)\}$

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C3) &= g(C3) + h(C3) \\ &= (0 + 2) + 8.9 \\ &= 10.9 \end{aligned}$$

$$\begin{aligned} f(C4) &= g(C4) + h(C4) \\ &= (0 + 7) + 7.3 \\ &= 14.3 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C3 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$List\ Tutup = [[\text{'C2'}, \text{'C3'}], 2]$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C3, kemudian didapatkan hasil:

$$\text{Terpilih} = \{ 'C23':(2,8.2), 'C24':(6,4.5) \}$$

$$\begin{aligned} f(C23) &= g(C23) + h(C23) \\ &= (2+2) + 8.2 \\ &= 12.2 \end{aligned}$$

$$\begin{aligned} f(C24) &= g(C24) + h(C24) \\ &= (2+6) + 4.5 \\ &= 12.5 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C23. C23 kita masukkan dalam *list* tutup.

$$\text{List Tutup} = [['C2', 'C3', 'C23'], 4]$$

C23 terhubung dengan C25, C30 dan C22. Panjang rute dan nilai heuristiknya adalah:

$$\text{Terpilih} = \{ 'C25': (10, 2), 'C30':(4,8.2), 'C22':(4,12.2) \}$$

$$\begin{aligned} f(C25) &= g(C25) + h(C25) \\ &= (4+10) + 2 \\ &= 16 \end{aligned}$$

$$\begin{aligned} f(C30) &= g(C30) + h(C30) \\ &= (4+4) + 8.2 \\ &= 16.2 \end{aligned}$$

$$\begin{aligned} f(C22) &= g(C22) + h(C22) \\ &= (4+4) + 12.2 \\ &= 20.2 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C25. C25 kita masukkan dalam *list* tutup.

List Tutup = [['C2','C3','C23'], 14]

C25 terhubung dengan CB, C24 dan C26. Panjang rute dan nilai heuristiknya adalah:

Terpilih = {'CB': (2, 0), 'C24':(4,4.5), 'C26':(4,3.7)}

$$\begin{aligned} f(\text{CB}) &= g(\text{CB}) + h(\text{CB}) \\ &= (14+2) + 0 \\ &= 16 \end{aligned}$$

$$\begin{aligned} f(\text{C24}) &= g(\text{C24}) + h(\text{C24}) \\ &= (14+4) + 4.5 \\ &= 22.5 \end{aligned}$$

$$\begin{aligned} f(\text{C26}) &= g(\text{C26}) + h(\text{C26}) \\ &= (14+4) + 3.7 \\ &= 21.7 \end{aligned}$$

CB terpilih karena memiliki nilai perhitungan yang paling minum sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi tujuan. Sehingga didapatkan hasil rute terpendeknya:

List Tutup = [['C2','C3','C23','CB'], 16]

Berdasarkan perhitungan di atas, antara titik awal yaitu C2 terhadap masing-masing titik didapatkan bahwa destinasi terdekat untuk dikunjungi terlebih dulu adalah CB. Sehingga W13 dapat kita masukkan dalam *list* rute.

List rute = [['C2','C3','C23','CB'], 16]

Langkah selanjutnya adalah menyeleksi dari titik W13 ke titik W9 dan W16. Akan ditentukan destinasi mana yang memiliki rute terpendek untuk dikunjungi terlebih dahulu.

Destinasi yang tersisa adalah destinasi CA. Pencarian rute menuju CA akan dimulai dari titik CB. Di sini tidak akan dilakukan perbandingan kembali karena CA menjadi destinasi terakhir.

- **Perhitungan Rute CB → CA**

Proses dalam pencarian rute kali ini mirip dengan perhitungan rute sebelumnya. CB merupakan titik awal permulaan untuk menuju destinasi, sehingga CB bisa dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [['CB'], 0]$$

Selanjutnya akan dicari titik mana saja yang terhubung dengan CB dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

CB terhubung dengan titik C24 dan C26. Kedua titik tersebut memuat informasi sebagai berikut:

$$Terpilih = \{ 'C24':(6,4.5), 'C26':(5,6.4) \}$$

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C24) &= g(C24) + h(C24) \\ &= (0 + 6) + 4.5 \\ &= 10.5 \end{aligned}$$

$$\begin{aligned} f(C26) &= g(C26) + h(C26) \\ &= (0 + 5) + 6.4 \end{aligned}$$

$$= 11.4$$

Hasil yang diperoleh menunjukkan bahwa C24 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$$\text{List Tutup} = [[\text{'CB'}, \text{'C24 '}], 6]$$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C24, kemudian didapatkan hasil:

$$\text{Terpilih} = \{\text{'C6'}:(6,2.8), \text{'CA'}:(6,0)\}$$

$$\begin{aligned} f(\text{C6}) &= g(\text{C6}) + h(\text{C6}) \\ &= (6+6) + 2.8 \\ &= 14.8 \end{aligned}$$

$$\begin{aligned} f(\text{CA}) &= g(\text{CA}) + h(\text{CA}) \\ &= (6+6) + 0 \\ &= 12 \end{aligned}$$

CA terpilih karena memiliki nilai perhitungan yang paling minum dan juga memiliki nilai heuristik 0 yang menandakan bahwa perhitungan telah sampai pada lokasi yang diinginkan. Sehingga didapatkan hasil:

$$\text{List Tutup} = [[\text{'CB'}, \text{'C24'.'CA'}], 12]$$

Rute terakhir yang ada pada *list* tutup di atas dapat dimasukkan ke dalam *list* rute. Kemudian dijumlahkan total jarak minimum yang dapat ditempuh.

List Rute

$$\text{List rute} = [[\text{'C2'}, \text{'C3'}, \text{'C23'}, \text{'CB'}], [\text{'CB'}, \text{'C24'.'CA'}], 28]$$

Berdasarkan perhitungan dari list rute di atas, melalui titik C2, destinasi pertama yang menjadi tujuan adalah *cluster* A (CA). Setelah itu dilanjutkan ke destinasi kedua yaitu *cluster* B (CB), dengan jarak 28 grid.

c) **Titik Awal C4**

- **Penentuan Rute C4 → CC**

C4 merupakan titik awal permulaan untuk menuju destinasi, sehingga C4 bisa dimasukkan ke dalam *List Tutup*.

List Tutup = [['C4'], 0]

Selanjutnya akan dicari titik mana saja yang terhubung dengan C4 dan berapa nilai masing-masing heuristik titik tersebut. Nilai heuristik diambil melalui perhitungan jarak *Euclid* pada persamaan (2.2). Perhitungan ini diambil dari titik koordinat yang ada pada gambar visualisasi graf. C4 terhubung dengan C2, C5, dan C24, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

Terpilih = {'C2':(7, 9.2), 'C5':(7, 11.4), 'C24':(3, 5.1)}

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C2) &= g(C2) + h(C2) \\ &= (0 + 7) + 9.2 \\ &= 16.2 \end{aligned}$$

$$\begin{aligned} f(C5) &= g(C5) + h(C5) \\ &= (0 + 7) + 11.4 \\ &= 18.4 \end{aligned}$$

$$\begin{aligned} f(C24) &= g(C5) + h(C5) \\ &= (0 + 3) + 5.1 \\ &= 8.1 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C24 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$$\text{List Tutup} = [[\text{'C4'}, \text{'C24'}], 3]$$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C24, kemudian didapatkan hasil:

$$\text{Terpilih} = \{\text{'C3'}:(6, 7.1), \text{'C6'}:(6,8.6), \text{'C25'}:(4,1.4)\}$$

$$\begin{aligned} f(\text{C3}) &= g(\text{C3}) + h(\text{C3}) \\ &= (3+6) + 7.1 \\ &= 16.1 \end{aligned}$$

$$\begin{aligned} f(\text{C6}) &= g(\text{C6}) + h(\text{C6}) \\ &= (3+6) + 8.6 \\ &= 17.6 \end{aligned}$$

$$\begin{aligned} f(\text{C25}) &= g(\text{C25}) + h(\text{C25}) \\ &= (3+4) + 1.4 \\ &= 8.4 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C25. C25 kita masukkan dalam *list* tutup.

$$\text{List Tutup} = [[\text{'C4'}, \text{'C24'}, \text{'C25'}], 7]$$

C25 terhubung dengan CC dan C26. Panjang rute dan nilai heuristiknya adalah:

$$\text{Terpilih} = \{\text{'CC'}:(6, 0), \text{'C26'}:(3,2.2)\}$$

$$\begin{aligned} f(\text{CC}) &= g(\text{CC}) + h(\text{CC}) \\ &= (7+6) + 0 \\ &= 13 \end{aligned}$$

$$\begin{aligned}
 f(C26) &= g(C26) + h(C26) \\
 &= (7+3) + 2.2 \\
 &= 12.2
 \end{aligned}$$

CC terpilih karena memiliki nilai perhitungan yang paling minimum sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi *cluster* tujuan. Sehingga didapatkan hasil:

$$List\ Tutup = [['C4', 'C24', 'C25', 'CC'], 13]$$

- **Penentuan Rute C4 → CD**

Proses dalam pencarian rute kali ini mirip dengan perhitungan rute sebelumnya. C4 merupakan titik awal permulaan untuk menuju destinasi, sehingga C4 bisa dimasukkan ke dalam *List Tutup*.

$$List\ Tutup = [['C4'], 0]$$

Selanjutnya akan dicari titik mana saja yang terhubung dengan C4 dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

$$Terpilih = \{ 'C5':(6, 11.1), 'C24':(3, 10.3) \}$$

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned}
 f(C5) &= g(C5) + h(C5) \\
 &= (0 + 6) + 11.1 \\
 &= 17.1
 \end{aligned}$$

$$\begin{aligned}
 f(C24) &= g(C24) + h(C24) \\
 &= (0 + 3) + 10.3 \\
 &= 13.3
 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C24 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$$\text{List Tutup} = [[\text{'C4'}, \text{'C24'}], 3]$$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C24, kemudian didapatkan hasil:

$$\text{Terpilih} = \{\text{'C6'}:(6, 7.2), \text{'C3'}:(6, 15.2), \text{'C25'}:(4, 8.1)\}$$

$$\begin{aligned} f(\text{C6}) &= g(\text{C6}) + h(\text{C6}) \\ &= (3+6) + 7.2 \\ &= 16.2 \end{aligned}$$

$$\begin{aligned} f(\text{C3}) &= g(\text{C3}) + h(\text{C3}) \\ &= (3+6) + 15.2 \\ &= 24.2 \end{aligned}$$

$$\begin{aligned} f(\text{C25}) &= g(\text{C25}) + h(\text{C25}) \\ &= (3+4) + 8.1 \\ &= 15.1 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C25. C25 kita masukkan dalam *list* tutup.

$$\text{List Tutup} = [[\text{'C4'}, \text{'C24'}, \text{'C25'}], 7]$$

C25 terhubung dengan C26 dan C30. Panjang rute dan nilai heuristiknya adalah:

$$\text{Terpilih} = \{\text{'C26'}:(3, 7.5), \text{'C30'}:(7, 13.5)\}$$

$$\begin{aligned} f(\text{C26}) &= g(\text{C26}) + h(\text{C26}) \\ &= (7+3) + 7.5 \\ &= 17.5 \end{aligned}$$

$$\begin{aligned}
 f(C30) &= g(C30) + h(C30) \\
 &= (7+7) + 13.5 \\
 &= 27.5
 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C26. C26 kita masukkan dalam *list* tutup.

List Tutup = [['C4', 'C24', 'C25', 'C26'], 10]

C26 terhubung dengan C9, C30 dan C27. Panjang rute dan nilai heuristiknya adalah:

Terpilih = {'C9': (7, 0.25), 'C30': (6, 13.5), 'C27': (5, 9.0)}

$$\begin{aligned}
 f(C9) &= g(C9) + h(C9) \\
 &= (10+7) + 0.25 \\
 &= 17.25
 \end{aligned}$$

$$\begin{aligned}
 f(C30) &= g(C30) + h(C30) \\
 &= (10+6) + 13.5 \\
 &= 29.5
 \end{aligned}$$

$$\begin{aligned}
 f(C27) &= g(C27) + h(C27) \\
 &= (10+5) + 9.0 \\
 &= 24
 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C9. C9 kita masukkan dalam *list* tutup.

List Tutup = [['C4', 'C24', 'C25', 'C26', 'C9'], 17]

Selanjutnya akan dicari titik mana saja yang terhubung dengan C9 dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

Terpilih = {'CD':(1, 0), 'C10':(4, 4)}

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(\text{CD}) &= g(\text{CD}) + h(\text{CD}) \\ &= (17 + 1) + 0 \\ &= 18 \end{aligned}$$

$$\begin{aligned} f(\text{C24}) &= g(\text{C24}) + h(\text{C24}) \\ &= (17 + 4) + 4 \\ &= 25 \end{aligned}$$

CD terpilih karena memiliki nilai perhitungan yang paling minum sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi tujuan. Sehingga didapatkan hasil rute terpendeknya:

List Tutup = [['C4', 'C24', 'C25', 'C26', 'C9', 'CD'], 18]

Berdasarkan perhitungan di atas, antara titik awal yaitu C4 terhadap masing-masing titik didapatkan bahwa destinasi terdekat untuk dikunjungi terlebih dulu adalah CC. Sehingga CC dapat kita masukkan dalam *list* rute.

List rute = [['C4', 'C24', 'C25', 'CC'], 13]

Destinasi yang tersisa adalah destinasi CD. Pencarian rute menuju CD akan dimulai dari titik CC. Di sini tidak akan dilakukan perbandingan kembali karena CD menjadi destinasi terakhir.

- **Perhitungan Rute CC → CD**

Proses dalam pencarian rute kali ini mirip dengan perhitungan rute sebelumnya. CC merupakan titik awal permulaan untuk menuju destinasi, sehingga CC bisa dimasukkan ke dalam *List* Tutup.

$List\ Tutup = [[\text{'CC'}, 0]$

Selanjutnya akan dicari titik mana saja yang terhubung dengan CC dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

CC terhubung dengan titik C25, C26 dan C30. Kedua titik tersebut memuat informasi sebagai berikut:

$Terpilih = \{\text{'C25'}:(2,8.1), \text{'C26'}:(2,7.5), \text{'C30'}:(6,13.5)\}$

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C25) &= g(C25) + h(C25) \\ &= (0 + 2) + 8.1 \\ &= 10.1 \end{aligned}$$

$$\begin{aligned} f(C26) &= g(C26) + h(C26) \\ &= (0 + 2) + 7.5 \\ &= 9.5 \end{aligned}$$

$$\begin{aligned} f(C30) &= g(C30) + h(C30) \\ &= (0 + 6) + 13.5 \\ &= 19.5 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C26 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$List\ Tutup = [[\text{'CC'}, \text{'C26'}, 2]$

C26 terhubung dengan C9, C30 dan C27. Panjang rute dan nilai heuristiknya adalah:

$Terpilih = \{\text{'C9'}:(7, 0.25), \text{'C30'}:(6,13.5), \text{'C27'}:(5,9.0)\}$

$$\begin{aligned}
 f(C9) &= g(C9) + h(C9) \\
 &= (2+7) + 0.25 \\
 &= 9.25
 \end{aligned}$$

$$\begin{aligned}
 f(C30) &= g(C30) + h(C30) \\
 &= (2+6) + 13.5 \\
 &= 21.5
 \end{aligned}$$

$$\begin{aligned}
 f(C27) &= g(C27) + h(C27) \\
 &= (2+5) + 9.0 \\
 &= 16
 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C9. C9 kita masukkan dalam *list* tutup.

List Tutup = [['CC', 'C26', 'C9'], 9]

Selanjutnya akan dicari titik mana saja yang terhubung dengan C9 dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

Terpilih = {'CD':(1, 0), 'C10':(4, 4)}

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned}
 f(CD) &= g(CD) + h(CD) \\
 &= (9 + 1) + 0 \\
 &= 10
 \end{aligned}$$

$$\begin{aligned}
 f(C24) &= g(C24) + h(C24) \\
 &= (9 + 4) + 4 \\
 &= 17
 \end{aligned}$$

CD terpilih karena memiliki nilai perhitungan yang paling minum sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi tujuan. Sehingga didapatkan hasil rute terpendeknya:

List Tutup = [['CC', 'C26', 'C9', 'CD'], 10]

Rute terakhir yang ada pada *list* tutup di atas dapat dimasukkan ke dalam *list* rute. Kemudian dijumlahkan total jarak minimum yang dapat ditempuh.

List Rute

List rute = [['C4', 'C24', 'C25', 'CC'], ['CC', 'C26', 'C9', 'CD'], 23]

Berdasarkan perhitungan dari list rute di atas, melalui titik C4, destinasi pertama yang menjadi tujuan adalah *cluster* C (CC). Setelah itu dilanjutkan ke destinasi kedua yaitu *cluster* D (CD), dengan jarak 23 grid.

d) Titik Awal C7

- **Penentuan Rute C7 → CE**

C7 merupakan titik awal permulaan untuk menuju destinasi, sehingga C7 bisa dimasukkan ke dalam *List* Tutup.

List Tutup = [['C7'], 0]

Selanjutnya akan dicari titik mana saja yang terhubung dengan C4 dan berapa nilai masing-masing heuristik titik tersebut. Nilai heuristik diambil melalui perhitungan jarak *Euclid* pada persamaan (2.2). Perhitungan ini diambil dari titik koordinat yang ada pada gambar visualisasi graf. C7 terhubung dengan C6 dan C8, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

Terpilih = {'C6':(7, 11.7), 'C8':(7, 18.1)}

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C6) &= g(C6) + h(C6) \\ &= (0 + 7) + 11.7 \\ &= 18.7 \end{aligned}$$

$$\begin{aligned} f(C8) &= g(C8) + h(C8) \\ &= (0 + 7) + 18.1 \\ &= 25.1 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C6 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$$List\ Tutup = [['C7', 'C6'], 7]$$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C6, kemudian didapatkan hasil:

$$Terpilih = \{ 'C5':(4, 13.6), 'C24':(6,6.4) \}$$

$$\begin{aligned} f(C5) &= g(C5) + h(C5) \\ &= (7+4) + 13.6 \\ &= 24.6 \end{aligned}$$

$$\begin{aligned} f(C24) &= g(C24) + h(C24) \\ &= (7+6) + 6.4 \\ &= 19.4 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C24. C24 kita masukkan dalam *list* tutup.

$$List\ Tutup = [['C7', 'C6', 'C24'], 13]$$

C24 terhubung dengan C3, C4 dan C25. Panjang rute dan nilai heuristiknya adalah:

Terpilih = {'C3': (6, 4.1), 'C4': (3, 8.6), 'C25': (4, 5)}

$$\begin{aligned} f(C3) &= g(CC) + h(CC) \\ &= (13+6) + 4.1 \\ &= 23.1 \end{aligned}$$

$$\begin{aligned} f(C4) &= g(C4) + h(C4) \\ &= (13+3) + 8.6 \\ &= 24.6 \end{aligned}$$

$$\begin{aligned} f(C25) &= g(C25) + h(C25) \\ &= (13+4) + 5 \\ &= 22 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C24. C24 kita masukkan dalam *list* tutup.

List Tutup = [['C7', 'C6', 'C24', 'C25'], 17]

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C25, kemudian didapatkan hasil:

Terpilih = {'CE': (5, 0), 'C26': (3, 5.8)}

$$\begin{aligned} f(CE) &= g(CE) + h(CE) \\ &= (17+5) + 0 \\ &= 22 \end{aligned}$$

$$\begin{aligned} f(C26) &= g(C26) + h(C26) \\ &= (17+3) + 5.8 \\ &= 25.8 \end{aligned}$$

CE terpilih karena memiliki nilai perhitungan yang paling minum sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi *cluster* tujuan. Sehingga didapatkan hasil:

$List\ Tutup = [[\text{'C7'}, \text{'C6'}, \text{'C24'}, \text{'C25'}, \text{'CE'}], 22]$

- **Penentuan Rute C7 → CF**

Proses dalam pencarian rute kali ini mirip dengan perhitungan rute sebelumnya. C7 merupakan titik awal permulaan untuk menuju destinasi, sehingga C7 bisa dimasukkan ke dalam *List Tutup*.

$List\ Tutup = [[\text{'C7'}], 0]$

Selanjutnya akan dicari titik mana saja yang terhubung dengan C7 dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

$Terpilih = \{\text{'C6'}:(7, 17.5), \text{'C8'}:(7, 22.2)\}$

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C6) &= g(C6) + h(C6) \\ &= (0 + 6) + 17.5 \\ &= 23.5 \end{aligned}$$

$$\begin{aligned} f(C8) &= g(C8) + h(C8) \\ &= (0 + 7) + 22.2 \\ &= 29.2 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C6 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

$List\ Tutup = [[\text{'C7'}, \text{'C6'}], 6]$

Sama halnya dengan proses sebelumnya, akan dicari titik yang terhubung dengan C6, kemudian didapatkan hasil:

Terpilih = {'C9':(7, 16.1), 'C24':(6,12.7)}

$$\begin{aligned} f(C9) &= g(C9) + h(C9) \\ &= (6+7) + 16.1 \\ &= 29.1 \end{aligned}$$

$$\begin{aligned} f(C24) &= g(C24) + h(C24) \\ &= (6+6) + 12.7 \\ &= 24.7 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C24. C24 kita masukkan dalam *list* tutup.

List Tutup = [['C7', 'C6', 'C24'], 12]

C24 terhubung dengan C3 dan C25. Panjang rute dan nilai heuristiknya adalah:

Terpilih = {'C3': (6, 9.5), 'C25':(4,10.3)}

$$\begin{aligned} f(C3) &= g(C3) + h(C3) \\ &= (12+6) + 9.5 \\ &= 27.5 \end{aligned}$$

$$\begin{aligned} f(C25) &= g(C25) + h(C25) \\ &= (12+4) + 10.3 \\ &= 26.3 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C25. C25 kita masukkan dalam *list* tutup.

List Tutup = [['C7', 'C6', 'C24', 'C25'], 16]

C25 terhubung dengan C26 dan C30. Panjang rute dan nilai heuristiknya adalah:

$$\text{Terpilih} = \{ \text{'C26': (3, 9.2)}, \text{'C30': (7, 4.2)} \}$$

$$\begin{aligned} f(\text{C26}) &= g(\text{C26}) + h(\text{C26}) \\ &= (16+3) + 9.2 \\ &= 28.2 \end{aligned}$$

$$\begin{aligned} f(\text{C30}) &= g(\text{C30}) + h(\text{C30}) \\ &= (16+7) + 4.2 \\ &= 27.5 \end{aligned}$$

Dari proses perhitungan yang sama didapatkan hasil bahwa titik rute yang memiliki perhitungan paling minimum adalah titik C30. C30 kita masukkan dalam *list* tutup.

$$\text{List Tutup} = [[\text{'C7'}, \text{'C6'}, \text{'C24'}, \text{'C25'}, \text{'C30'}], 23]$$

Selanjutnya akan dicari titik mana saja yang terhubung dengan C30 dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

$$\text{Terpilih} = \{ \text{'C19': (5, 1.4)}, \text{'C21': (4, 3.2)} \}$$

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(\text{C19}) &= g(\text{C19}) + h(\text{C19}) \\ &= (23 + 5) + 1.4 \\ &= 29.4 \end{aligned}$$

$$\begin{aligned} f(\text{C21}) &= g(\text{C21}) + h(\text{C21}) \\ &= (23 + 4) + 3.2 \end{aligned}$$

$$= 30.2$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C19. C19 kita masukkan dalam *list* tutup.

$$List\ Tutup = [['C7', 'C6', 'C24', 'C25', 'C30', 'C19'], 28]$$

Selanjutnya akan dicari titik mana saja yang terhubung dengan C19 dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

$$Terpilih = \{ 'CF':(1, 0), 'C20':(2, 1.4) \}$$

$$\begin{aligned} f(CF) &= g(CF) + h(CF) \\ &= (28 + 1) + 0 \\ &= 29 \end{aligned}$$

$$\begin{aligned} f(C20) &= g(C20) + h(C20) \\ &= (28 + 2) + 1.4 \\ &= 31.4 \end{aligned}$$

CF terpilih karena memiliki nilai perhitungan yang paling minum sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi tujuan. Sehingga didapatkan hasil rute terpendeknya:

$$List\ Tutup = [['C7', 'C6', 'C24', 'C25', 'C30', 'C19', 'CF'], 29]$$

Berdasarkan perhitungan di atas, antara titik awal yaitu C7 terhadap masing-masing titik didapatkan bahwa destinasi terdekat untuk dikunjungi terlebih dulu adalah CE. Sehingga CE dapat kita masukkan dalam *list* rute.

$$List\ Tutup = [['C7', 'C6', 'C24', 'C25', 'CE'], 22]$$

Destinasi yang tersisa adalah destinasi CF. Pencarian rute menuju CF akan dimulai dari titik CE. Di sini tidak akan dilakukan perbandingan kembali karena CF menjadi destinasi terakhir.

- **Perhitungan Rute CE → CF**

Proses dalam pencarian rute kali ini mirip dengan perhitungan rute sebelumnya. CE merupakan titik awal permulaan untuk menuju destinasi, sehingga CE bisa dimasukkan ke dalam *List Tutup*.

List Tutup = [['CE'], 0]

Selanjutnya akan dicari titik mana saja yang terhubung dengan CE dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

CE terhubung dengan titik C23 dan C30. Kedua titik tersebut memuat informasi sebagai berikut:

Terpilih = {'C23':(3,7.6), 'C30':(2,4.2)}

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C23) &= g(C23) + h(C23) \\ &= (0 + 3) + 7.6 \\ &= 9.6 \end{aligned}$$

$$\begin{aligned} f(C30) &= g(C30) + h(C30) \\ &= (0 + 2) + 4.2 \\ &= 6.2 \end{aligned}$$

Hasil yang diperoleh menunjukkan bahwa C30 memiliki perhitungan yang minimum, sehingga kita masukkan ke dalam *list* tutup menjadi:

List Tutup = [['CE', 'C30'], 2]

Selanjutnya akan dicari titik mana saja yang terhubung dengan C30 dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

Terpilih = {'C19':(5, 1.4), 'C21':(4, 3.2)}

Kemudian kita seleksi melalui sebuah perhitungan dan dipilih nilai yang paling minimum.

$$\begin{aligned} f(C19) &= g(C19) + h(C19) \\ &= (2 + 5) + 1.4 \\ &= 8.4 \end{aligned}$$

$$\begin{aligned} f(C21) &= g(C21) + h(C21) \\ &= (2 + 4) + 3.2 \\ &= 9.2 \end{aligned}$$

Melalui proses yang sama didapatkan hasil bahwa titik yang memiliki perhitungan minimum adalah titik C19. C19 kita masukkan dalam *list* tutup.

List Tutup = [['CE', 'C30', 'C19'], 7]

Selanjutnya akan dicari titik mana saja yang terhubung dengan C19 dan berapa nilai masing-masing heuristik titik tersebut, kemudian didapatkan hasil berupa panjang rute dan nilai heuristiknya sebagai berikut:

Terpilih = {'CF':(1, 0), 'C20':(2, 1.4)}

$$\begin{aligned} f(CF) &= g(CF) + h(CF) \\ &= (7 + 1) + 0 \\ &= 8 \end{aligned}$$

$$f(C20) = g(C20) + h(C20)$$

$$= (7 + 2) + 1.4$$

$$= 10.4$$

CF terpilih karena memiliki nilai perhitungan yang paling minum sekaligus menjadi destinasi akhir yang menandakan telah sampai pada lokasi tujuan. Sehingga didapatkan hasil rute terpendeknya:

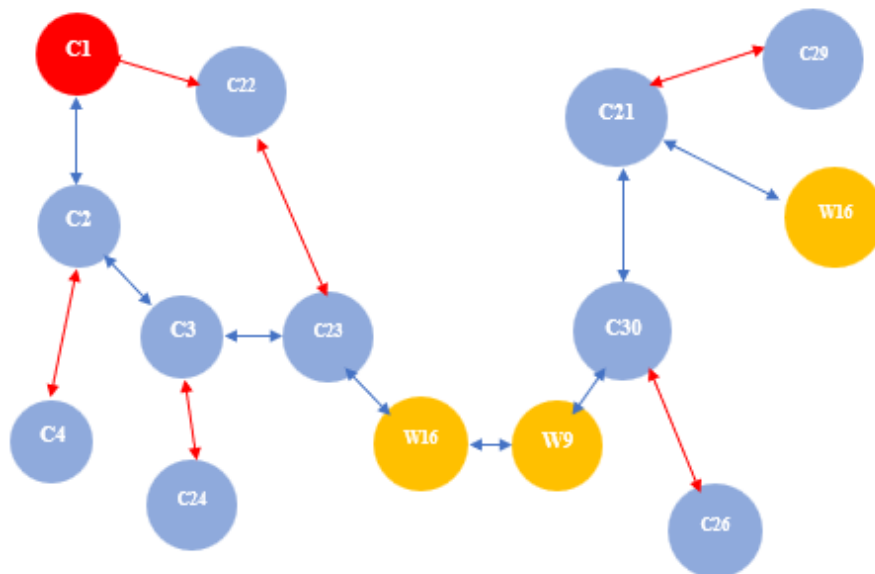
List Tutup = [['CE', 'C30', 'C19', 'CF'], 8]

Rute terakhir yang ada pada *list* tutup di atas dapat dimasukkan ke dalam *list* rute. Kemudian dijumlahkan total jarak minimum yang dapat ditempuh.

List rute = [['C7', 'C6', 'C24', 'C25', 'CE'], ['CE', 'C30', 'C19', 'CF'], 30]

Berdasarkan perhitungan dari list rute di atas, melalui titik C7, destinasi pertama yang menjadi tujuan adalah *cluster* E (CE). Setelah itu dilanjutkan ke destinasi kedua yaitu *cluster* F (CF), dengan jarak 30 grid.

4. Visualisasi Rute



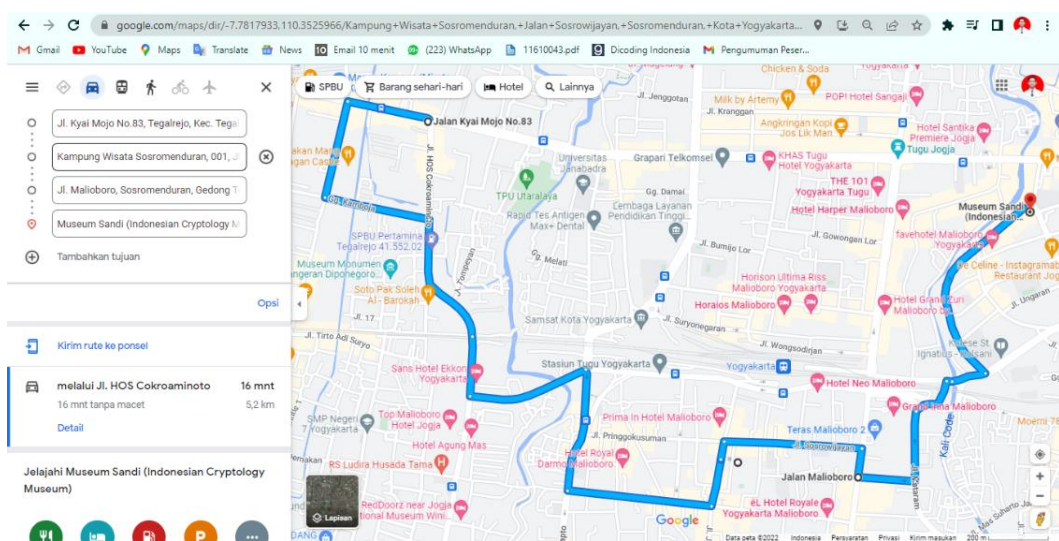
Gambar 4. 3 Visualisasi Rute Optimal Setelah Perhitungan

Visualisasi graf di atas adalah representasi dari sampel titik awal C1, simbol-simbol pada visualisasi tersebut pada umumnya bersimbol C untuk semua

titik persimpangan dan warna kuning untuk setiap titik destinasi wisata budaya yang dipilih. Garis yang berpanah menunjukkan graf berarah itu sendiri. Selain itu, untuk output hasil akan diberikan warna tersendiri untuk dibedakan. Titik berwarna merah menyimbolkan titik awal dalam melakukan perjalanan. Titik ini merupakan titik awal yang telah dimasukkan pada tahap awal. Titik yang berwarna kuning menandakan titik yang akan menjadi tujuan destinasi wisata. Warna garis penghubung diberi warna merah untuk menandakan jalur yang tidak efisien, dan garis jalur warna biru sebagai jalur yang efisien dan dipilih untuk dilewati.

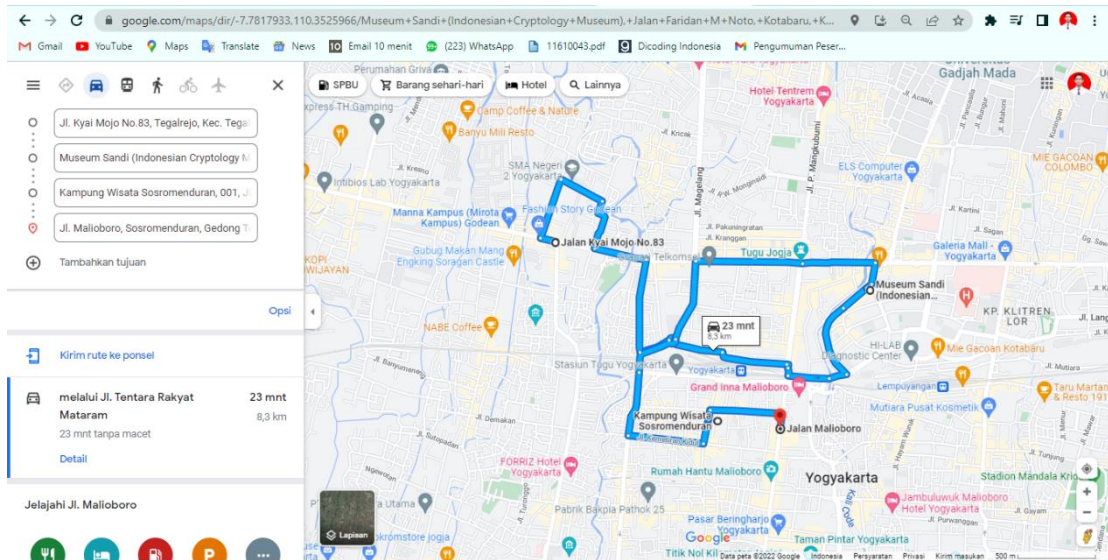
4.3 Hasil Pengujian

Penelitian rute terpendek ini akan dibuktikan dan divalidasi melalui google maps untuk beberapa sampel destinasi wisata budaya. Pembuktian ini akan dibandingkan dengan perhitungan menggunakan algoritma *A-Star*. Misalnya perhitungan yang telah dijelaskan di atas adalah dari C1 ke W19, W9, dan W16. Tersaji pada gambar berikut melalui google maps dengan jarak 5,2 km.



Gambar 4.4 Perhitungan Rute Hasil Penelitian di Google Maps

Maka pada google maps akan dibuktikan dengan penelusuran jalur melalui rute C1 menuju W16, W9, dan W13. Satuan jarak pada penelitian ini menggunakan satuan kilo meter (KM). Salah satu hasil yang ditampilkan google maps berdasarkan sampel di atas ada pada gambar berikut.



Gambar 4.5 Perhitungan Rute Pemandangan di Google Maps

Gambar di atas menunjukkan bahwa rute yang terbentuk adalah rute yang berurut sesuai dengan yang telah di masukkan pada kolom tujuan. Tidak terdapat penyeleksian destinasi mana yang terdekat dengan titik awal sehingga dapat dikunjungi terlebih dahulu. Jarak yang ditempuh adalah 8,3 km. Jarak ini lebih panjang daripada yang dihasilkan oleh Algoritma *A-Star*.

Proses pengujian ini akan dilakukan sebanyak 10 kali secara acak. Walaupun secara acak, pengujian ini akan dilakukan menyeluruh. Arti menyeluruh di sini adalah seluruh destinasi wisata akan mendapatkan bagian untuk dilakukan proses pengujian sesuai dengan program yang telah dibuat sebelumnya. Hasil dari proses pengujian ini akan ditampilkan melalui tabel berikut.

Tabel 4.6 Tabel Hasil Proses Pengujian

No	Titik Awal	Tujuan	Rute Hasil Penelitian	Jarak pada Penelitian (KM)	Rute pada Maps	Jarak pada Maps (KM)
1	C1	W9, W13, W16	W13, W9, W16	5.2	W16, W13, W9	8.3
2	C2	CA, CB	CA, CB	4.0	CB, CA	5.1
3	C4	CC, CD	CC, CD	4.6	CD, CC	6.5
4	C7	CE, CF	CE, CF	8.0	CF, CE	8.8
5	C8	W4, W10	W10, W4	4.2	W10, W4	5.2
6	C12	CA, CC, CD	CD, CC, CA	7.4	CA, CC, CD	10.0
7	C14	CB, CD, CE	CD, CB, CE	9.6	CE, CD, CB	11.2
8	C20	CA, CD, CF	W17, W18, W11	5.3	W18, W11, W17	6.6
9	C28	CB, CC, CF	W20, W5, W8	5.2	W8, W5, W20	6.2
10	C29	CB, CE, CF	W15, W13, W16	8.3	W16, W13, W15	10.2

4.4 Evaluasi Pengujian

Setelah dilakukan pengujian sebanyak beberapa kali didapatkan hasil bahwa pengujian yang dilakukan dengan algoritma *A-Star* memiliki jarak yang relatif lebih pendek daripada jarak yang ditampilkan melalui google maps. Persentase kehematan/ efektivitas jarak tempuh hasil pengujian didapatkan melalui:

$$\text{Nilai efektivitas} = \frac{JM - JP}{JM} \times 100\% \quad (3)$$

Keterangan:

JM = Jarak yang ditampilkan pada google maps

JP = Jarak yang ditampilkan pada penelitian

Hasil perhitungan nilai efektifitas tersebut tersaji pada tabel berikut.

Tabel 4.7 Tabel Hasil Nilai Efektifitas

Pengujian Ke-	Presentase (%)
1	37,35%
2	21,57 %
3	29,23 %
4	9,62 %
5	26,23 %
6	26,00 %
7	14,29 %
8	19,70 %
9	16,12 %
10	18,63 %
Rata-rata	21,85 %

Tabel di atas menunjukkan pengujian yang dilakukan menggunakan algoritma *A-Star* memiliki kehematan jarak tempuh dengan persentase rata-rata 21,85 %. Rendahnya jarak tempuh yang dihasilkan oleh Algoritma *A-Star* membuat algoritma ini lebih efektif. Ada beberapa alasan yang membuat hasil pengujian rute terpendek menuju beberapa destinasi wisata jauh lebih efektif. Berikut merupakan penjelasannya:

a. Perhitungan Nilai Heuristik

Keunikan khas yang dimiliki oleh algoritma *A-Star* adalah perhitungan nilai heuristiknya. Nilai heuristik merupakan jarak antara setiap titik yang dihubungkan dengan titik tujuan. Proses perhitungannya melalui perhitungan jarak euclidan yang diambil melalui titik koordinat masing-masing titik yang tersebar pada gambar.

Keefektifan nilai heuristik ini dapat diketahui pada pengujian ke-1. Pengujian yang dilakukan dengan titik awal adalah C1 dan penginputan destinasi tujuan W13, W9, dan W16. Hasil yang ditunjukkan oleh Algoritma *A-Star* dan google maps sama-sama menunjukkan bahwa destinasi yang dituju pertama adalah W13, kemudian W9, dan terakhir adalah W16.

Perbedaannya adalah pada jarak yang ditempuh. Jarak rute yang ditempuh berdasarkan algoritma *A-Star* sebesar 5.2 km. Hal ini lebih rendah daripada jarak yang dihasilkan oleh google maps dengan prioritas jalur yang berbeda sebesar 8.3 km. Walaupun prioritas destinasi wisata yang dituju sama, namun rute yang dilalui berbeda. Hal ini merupakan efek dari perhitungan nilai heuristik dalam penentuan rute.

b. Penyeleksian Prioritas Rute Destinasi

Proses penelitian ini tidak berfokus pada satu tujuan destinasi saja, melainkan dilakukan pada beberapa titik tujuan destinasi wisata. Ketika dipilih terdapat tiga tujuan, maka dari ke tiga tujuan tersebut akan diseleksi masing-masing dengan algoritma *A-Star*. Kemudian akan dipilih destinasi mana yang akan menjadi tujuan pertama. Untuk menentukan tujuan destinasi kedua juga akan dilakukan seperti halnya yang pertama. Hal ini dilakukan sampai pada tujuan destinasi terakhir.

Akibat dari proses desain tersebut, pengujian rute tersebut lebih efektif dari pada pengujian rute yang dilakukan melalui google maps. Hal ini dikarenakan pada google maps, tidak adanya penyeleksian destinasi wisata. Jika seseorang memilih W9, W13, dan W16 maka pada google maps akan menampilkan destinasi pertama W9, kemudian W13, dan terakhir W16. Berbeda dengan

pengujian dengan algoritma *A-Star*. Walaupun yang dipilih adalah W9, W13, dan W16, belum tentu W9 menjadi tujuan destinasi pertama. Destinasi yang pertama dikunjungi adalah destinasi yang memiliki rute terpendek dari titik asal. Sehingga desain algoritma ini juga turut andil dalam menentukan pemilihan rute yang paling optimal dan lebih efektif.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan dapat disimpulkan bahwa penentuan rute terpendek menggunakan algoritma *A-Star* dinilai lebih efektif dalam sisi jarak. Hal ini dibuktikan dengan pengujian rute terpendek yang telah dilakukan secara acak dan menyeluruh terhadap seluruh destinasi wisata budaya sebanyak beberapa kali. Hasilnya didapatkan bahwa rata-rata jarak yang dihasilkan lebih pendek/ efektif sekitar 21,85 % dibandingkan jarak rute yang dihasilkan pada google maps.

Penyebab keefektifan ini ada dua. Peratama adalah adanya nilai heuristik setiap titik terhadap titik tujuan. Kondisi ini menjadikan pemilihan rute lebih tepat, sehingga jarak yang ditempuh lebih pendek.

Penyebab keefektifan kedua adalah desain penelitian tentang pembuatan prioritas destinasi. Tujuan pembuatan prioritas destinasi adalah menyeleksi beberapa destinasi tujuan, kemudian menentukan destinasi mana yang akan dikunjungi terlebih dahulu yang memiliki jarak terdekat dengan titik awal. Dalam penyeleksian prioritas ini akan digunakan algoritma *A-Star*. Inovasi desain penelitian ini menjadikan pemilihan rute jauh lebih efektif dibandingkan pemilihan rute yang dilakukan melalui google maps. Google maps sendiri tidak melakukan penyeleksian prioritas, hanya menampilkan rute sesuai urutan penginputan destinasi tujuan.

5.2 Saran untuk Penelitian Lanjutan

Penelitian ini telah menghasilkan modifikasi baru terhadap penentuan beberapa rute destinasi tujuan menggunakan algoritma *A-Star*. Diharapkan adanya penelitian ini dapat menjadi pondasi untuk penelitian selanjutnya terutama dalam menentukan rute destinasi di luar wilayah Kota Yogyakarta. Selain itu diharapkan penelitian ini bisa dikembangkan lebih terutama dalam penampilan visualisasinya agar lebih rapi sesuai dengan gambar peta pada *google maps*. Pembuatan program juga bisa dilakukan untuk penelitian selanjutnya, karena perhitungan manualnya yang sangat panjang, maka pembuatan program untuk dapat menghitung algoritma *A-Star* pasti akan sangat membantu untuk implementasi algoritma ini dalam kehidupan sehari-hari.

DAFTAR PUSTAKA

- Al-Qur'an dan Terjemah. Cetakan ke-7: Al-Mizan Publishing House.
- Al-Imam Abul Fida Isma'il Ibnu Katsir ad-Dimasyqi, *Terjemah Tafsir Ibnu Katsir Juz 15*, Bandung: Sinar Baru al-Gensindo, 2002.
- Ananda Rahmasari. (2021). UNIVERSITAS SUMATERA UTARA Poliklinik UNIVERSITAS SUMATERA UTARA. *Jurnal Pembangunan Wilayah & Kota*, 1(3), 82–91.
- ASTRI, R., & SULARNO. (2021). Implementasi Algoritma A-Star Untuk Pencarian RuteTerdekat Titik. *Jurnal Rekayasa Sistem Dan Teknologi Informasi*, 1(10), 254–259.
- Candra, A., Budiman, M. A., & Pohan, R. I. (2021). Application of A-Star Algorithm on Pathfinding Game. *Journal of Physics: Conference Series*, 1898(1). <https://doi.org/10.1088/1742-6596/1898/1/012047>
- Cantone, D., & Faro, S. (2014). Fast shortest-paths algorithms in the presence of few destinations of negative-weight arcs. *Journal of Discrete Algorithms*, 24, 12–25. <https://doi.org/10.1016/j.jda.2013.03.005>
- Duchon, F., Babinec, A., Kajan, M., Beno, P., Florek, M., Fico, T., & Jurišica, L. (2014). Path planning with modified A star algorithm for a mobile robot. *Procedia Engineering*, 96, 59–69. <https://doi.org/10.1016/j.proeng.2014.12.098>
- Fernando, Yusra. (2020). *Penerapan Algoritma A-Star Pada Aplikasi Pencarian Lokasi Fotografi Di Bandar Lampung Berbasis Android* dalam *Jurnal TEKNOINFO Volume 14 Nomor 1 (hlm. 27–34)* , Lampung : Universitas Lampung.
- Geoportal Kota Yogyakarta. <https://geoportal.jogjakota.go.id/pencarian?kategori=Transportasi>
- Kamayudi, A. (2006). Studi dan Implementasi Algoritma Djikstra, Bellman-Ford dan Floyd-Warshall dalam menangani masalah lintasan terpendek dalam Graf. *Program Studi Teknik Informatika, Institut Teknologi Bandung*.

- Kane, S. N., Mishra, A., & Dutta, A. K. (2016). Preface: International Conference on Recent Trends in Physics (ICRTP 2016). *Journal of Physics: Conference Series*, 755(1). <https://doi.org/10.1088/1742-6596/755/1/011001>
- Kumar, R., & Kumar, M. (2010). Exploring genetic algorithm for shortest path optimization in data networks. *Global Journal of Computer Science and Technology*.
- Kusumadewi, Ida & Purnomo, H. (2005). *Penyelesaian Masalah Optimisasi dengan Teknik-teknik Heuristik*. Yogyakarta: Graha Ilmu.
- Mandailina, V. (2009). *Faktorisasi pada graf komplit* (Doctoral dissertation, Universitas Islam Negeri Maulana Malik Ibrahim).
- Munir, A. Q. (2016). Perancangan Sistem Informasi Geografis Penentuan Jalur Jalan Optimum Kodya Yogyakarta Menggunakan Algoritma Dijkstra. *Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer*, 6(2), 80-85.
- Novandi, R. A. D. (2007). Perbandingan Algoritma Dijkstra dan Algoritma Floyd-Warshall dalam Penentuan Lintasan Terpendek (Single Pair Shortest Path). *Makalah IF2251 Strategi Algoritmik*.
- Rofiq, M., & Uzzy, R. F. (2014). Penentuan Jalur Terpendek Menuju Cafe Di Kota Malang Menggunakan Metode Bellman-Ford Dengan Location Based Service Berbasis Android. *Jurnal Ilmiah Teknologi Informasi Asia*, 8(2), 49-64.
- Susilawati, S. (2020). Penerapan Metode A*Star Pada Pencarian Rute Tercepat Menuju Destinasi Wisata Cagar Budaya Menes Pandeglang. *Geodika: Jurnal Kajian Ilmu Dan Pendidikan Geografi*, 4(2), 192–199. <https://doi.org/10.29408/geodika.v4i2.2754>
- syihabuddin, R. F. (2021). *Implementasi Algoritma A-Star Dalam Menentukan Rute Terpendek Destinasi Wisata Kota Yogyakarta* (Vol. 4, Issue 1). Universitas Islam Negeri Maulana Malik Ibrahim Yogyakarta.
- Suprpto. (2017). Perbandingan Performansi Pencarian Jarak Terpendek Menggunakan Algoritma A Star (A*) Dan Algoritma Dijkstra (Studi Kasus: Jalan Pariwisata Kabupaten Pati). Universitas Negeri Semarang.

- Tang, G., Tang, C., Claramunt, C., Hu, X., & Zhou, P. (2021). Geometric A-Star Algorithm: An Improved A-Star Algorithm for AGV Path Planning in a Port Environment. *IEEE Access*, 9, 59196–59210. <https://doi.org/10.1109/ACCESS.2021.3070054>
- Wahbah az-Zuhaili, Tafsir al-Munir fi al-`aqidah wa asySyar`iah wa al-Manhaj, Suriah, (Damaskus : Darul Fikri, 1991)
- Wayahdi, M. R., Ginting, S. H. N., & Syahputra, D. (2021). Greedy, A-Star, and Dijkstra's Algorithms in Finding Shortest Path. *International Journal of Advances in Data and Information Systems*, 2(1), 45–52. <https://doi.org/10.25008/ijadis.v2i1.1206>
- Winaryo, Herimanto. 2011. *Ilmu Sosial dan Budaya Dasar*. Jakarta, Bumi Aksara
- Yuliani, Agus, F., & Kunci, K. (2013). WEBGIS PENCARIAN RUTE TERPENDEK MENGGUNAKAN ALGORITM A STAR (A*) (Studi Kasus: Kota Bontang). *Jurnal Informatika Mulawarman Edisi Juli*, 8(2), 50–55.
- Zuhri, M., & Soleh, A. (2016). Analisis Kinerja Keuangan Pemerintah Daerah Kabupaten Kaur. *EKOMBIS REVIEW: Jurnal Ilmiah Ekonomi Dan Bisnis*, 4(2).

RIWAYAT HIDUP



Aditya Dwi Cahyono atau akrab dipanggil Adit, lahir di Batam pada tanggal 27 Mei 2000. Anak kedua dari 5 bersaudara dari pasangan Bapak Hasim Asari dan Ibu Ina Mariana. Seorang aktivis relawan di *Indonesia Scholarship Center* (ISC) dan seorang yang gemar akan dunia data dan teknologi.

Pemuda yang berasal dari kota bumi melayu ini selalu bersemangat dalam mengembangkan potensi yang telah dianugerahkan oleh Tuhan. Hal ini dapat terwujud baik dalam Pendidikan formal maupun nonformal yang telah ditempuh. Dimulai dari SDN 001 Batam Kota (2006-2012), SMPN 6 Batam (2012-2015), SMAN 8 Batam (2015-2018), sampai UIN Maulana Malik Ibrahim Malang. Penulis juga mengenyam beberapa Pendidikan nonformal diantaranya Pelatihan JOO dari BPPTIK kominfo RI (2020), Digitalent Scholarship kominfo RI (2021–2022), Ma’had Sunan Ampel Al Aly (2017-2018), Machine Learning Student di Bangkit Academy by Goole, GoTo, Traveloka, dan lain sebagainya.

Selama menempuh pendidikan dikuliah penulis juga mempunyai beberapa prestasi. Prestasi tersebut diantaranya adalah Juara 1 Debat Ilmiah tingkat fakultas Saintek dan Kedokteran 2018, Juara 2 Duta Putra Fakultas Sains dan Teknologi 2020, Juara 3 Essay PHBI 2020, Juara 1 Essay PHBI 2021, dan terpilih sebagai Duta Festival Luar Negeri (FELARI) PPI Dunia 2 priode berturut-turut (2021-2022). Selain itu penulis juga pernah menjadi asisten praktikum pemodelan matematika, aktif dalam Forum Kegiatan Mahasiswa (FKM) PKPBA 2018, aktif dalam kegiatan HMJ “Integral” Matematika pada divisi Internal Public Relation dan Kematematikaan, dan juga aktif di beberapa kegiatan kerelawanan dalam volunteer Masuk PTN, Grand SBMPTN, dan (World Indonesia Scholarship) WISH Festival.



BUKTI KONSULTASI SKRIPSI

Nama : Aditya Dwi Cahyono
NIM : 18610059
Fakultas / Program Studi : Sains dan Teknologi / Matematika
Judul Skripsi : Penerapan Algoritma *A-Star* Untuk Mencari Rute Terpendek
Destinasi Wisata Budaya Kota Yogyakarta
Pembimbing I : Juhari, M.Si,
Pembimbing II : Ach. Nasichuddin, M.A.

No	Tanggal	Hal	Tanda Tangan
1.	18 MARET 2022	Konsultasi bab I, II, III	1.
2.	25 MARET 2022	Konsultasi bab I, II, III	2.
3.	4 APRIL 2022	Konsultasi bab I, II, III	3.
4.	6 APRIL 2022	Konsultasi Keagamaan	4.
5.	8 APRIL 2022	Acc bab I, II, III	5.
6.	20 APRIL 2022	Konsultasi bab IV	6.
7.	25 APRIL 2022	Konsultasi bab IV	7.
8.	23 MEI 2022	Konsultasi bab IV	8.
9.	26 MEI 2022	Konsultasi Keagamaan	9.
10.	30 MEI 2022	Revisi bab IV	10.
11.	6 JUNI 2022	Revisi bab IV	11.
12.	8 JUNI 2022	Konsultasi bab IV, V	12.
13.	13 JUNI 2022	Konsultasi bab IV, V	13.

Malang, 21 Juni 2022
Mengetahui,
Ketua Program Studi Matematika

Dr. Elly Susanti, S.Pd., M.Sc.
NIP.197411292000122005