

## Implementasi Algoritma *Depth First Search* Dalam Penyelesaian Permasalahan Lintasan dan Sirkuit Euler

Matheus Supriyanto Rumetna<sup>1\*</sup>, Tirsa Ninia Lina<sup>2</sup>, Agustinus Budi Santoso<sup>3</sup>, Reinhard  
Komansilan<sup>4</sup>, Jeni Karay<sup>5</sup>

<sup>1,2</sup>Program Studi Sistem Informasi, Universitas Victory Sorong

<sup>3</sup>Program Studi Sistem Informasi, Universitas Sains dan Teknologi Komputer

<sup>4</sup>Program Studi Teknik Informatika, Universitas Sam Ratulangi

<sup>5</sup>Program Studi Sistem Informasi, Universitas Ottow Geisler Papua

\*email: [matheus.rumetna@gmail.com](mailto:matheus.rumetna@gmail.com)

DOI: <https://doi.org/10.31603/komtika.v7i1.8672>

Received: 01-02-2023, Revised: 15-03-2023, Accepted: 18-03-2023

### ABSTRACT

*Complicated and takes a long time to solve the problem of redrawing Euler's paths and circuits without passing through the edges that have been drawn, are the problems discussed in this study. This can be solved by implementing the Depth First Search (DFS) algorithm into an application. The DFS search technique functions as a solution seeker so that all paths contained in the graph can be traversed once and specifically for the Euler circuit will end at the point of departure. This research resulted in an application to find a solution for graphing Euler circuits and paths and to be able to quickly and effectively simulate the completion steps that have been processed using the DFS algorithm.*

**Keywords:** *Depth First Search, Graf, Path, Euler Circuit.*

### ABSTRAK

Rumit serta memerlukan waktu yang lama untuk menyelesaikan permasalahan menggambarkan ulang lintasan dan sirkuit *Euler* tanpa melewati sisi yang telah digambar, merupakan permasalahan yang dibahas dalam penelitian ini. Hal ini dapat terselesaikan dengan mengimplementasikan algoritma *Depth First Search* (DFS) menjadi sebuah aplikasi. Teknik pencarian DFS berfungsi sebagai pencari solusi agar semua jalur yang terdapat dalam graf dapat dilalui satu kali dan khusus sirkuit *Euler* akan berakhir ke titik asal keberangkatan. Penelitian ini menghasilkan aplikasi mencari solusi penggambaran graf terhadap sirkuit dan lintasan *Euler* serta dapat mensimulasikan langkah-langkah penyelesaian yang telah diproses menggunakan algoritma DFS secara cepat dan efektif.

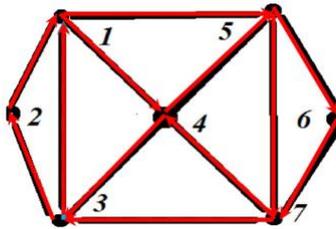
**Kata kunci:** *Depth First Search, Graf, Lintasan, Sirkuit Euler.*

### PENDAHULUAN

Salah satu ilmu pada bidang informatika atau komputer yang digunakan untuk membuat serta mempelajari, bagaimana komputer dapat melaksanakan hal-hal yang manusia laksanakan atau dapat dikatakan meniru pekerjaan manusia dengan baik adalah kecerdasan buatan atau *Artificial Intelligence* (AI)[1]–[4]. Awalnya komputer digunakan untuk menghitung atau alat hitung sederhana[5]–[7]. Teknologi yang digunakan dalam komputer berkembang pesat dan hingga saat ini AI sering digunakan untuk mencari penyelesaian atau solusi[8]–[10]. Dalam suatu kasus AI terdapat suatu ruang yang disebut ruang keadaan. Kondisi ruang keadaan meliputi: keadaan sekarang (*start state*), keadaan tujuan (*goal state*) adalah kondisi dimana suatu keadaan diubah sesuai dengan aturan dan batasannya, sehingga solusi akhir atau sasaran dapat dicapai[11]–[14]. Node (simpul) merupakan representasi dari

keadaan, kemudian *arc* (busur) adalah representasi dari aksi. Hal ini sering ditemui pada lintasan dan sirkuit *Euler*.

Sirkuit *Euler* merupakan sirkuit yang melewati setiap sisi dalam graf tepat satu kali dan kembali lagi ke titik awal keberangkatan. Sedangkan lintasan *Euler* adalah lintasan yang melalui setiap sisi dalam graf tepat satu kali dan tidak kembali ke titik awal keberangkatan. Teori dikemukakan *Euler* adalah “suatu graf yang terhubung namun tidak memiliki arah  $G$  merupakan graf *Euler* jika setiap simpulnya memiliki derajat genap”. Lintasan *Euler* merupakan lintasan terbuka yaitu lintasan tidak berupa sirkuit saat dilalui. Kondisinya adalah simpul awal dan akhir harus memiliki derajat ganjil, maka dikatakan graf memiliki lintasan *Euler*[11], [15], [16] seperti disajikan pada gambar 1.



Gambar 1. Graf *Euler*

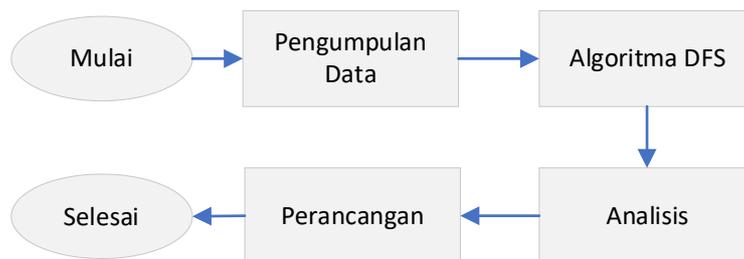
Sirkuit *Euler* yang dimiliki oleh graf tertentu menunjukkan bahwa graf tersebut juga memiliki lintasan *Euler*, namun tidak sebaliknya. Ada sebuah kondisi yang harus dicapai untuk membuat graf yang memiliki lintasan *Euler*, yaitu semua simpulnya wajib saling terkoneksi serta mempunyai dua buah simpul yang memiliki derajat ganjil. Masalah yang dihadapi saat ini adalah rumitnya menggambarkan ulang lintasan dan sirkuit *Euler* tersebut tanpa melewati sisi yang telah digambar, hal ini memerlukan waktu yang lama untuk menyelesaikannya.

Permasalahan lintasan dan sirkuit *Euler* dapat diselesaikan menggunakan algoritma *Depth First Search* (DFS). Teknik pencarian DFS berfungsi sebagai pencari solusi bagaimana agar semua jalur (sisi) yang terdapat dalam graf dapat dilalui satu kali dan khusus sirkuit *Euler* akan berakhir ke titik asal keberangkatan. Pencarian DFS dimulai dari simpul ter kiri di setiap level, apabila hingga level yang paling dalam belum juga menemukan hasil maka proses pencarian dilakukan pada simpul yang berada di kanan, sedangkan simpul yang berada di kiri dihilangkan dari memori [17], [18]–[22]. Apabila belum juga menemukan hasil hingga level terdalam, maka dilanjutkan lagi proses pencariannya ke level sebelumnya. Demikian seterusnya sampai ditemukannya solusi. Keunggulan dari algoritma ini yaitu membutuhkan memori relatif kecil, karena DFS tidak memerlukan pengujian yang banyak dalam ruang keadaan untuk mendapatkan hasil serta sistem penyimpanan hanya dilakukan untuk node pada lintasan yang aktif [23]–[29].

## METODE

Penelitian ini menggunakan beberapa pendekatan atau teknik sebagai berikut seperti disajikan pada Gambar 2:

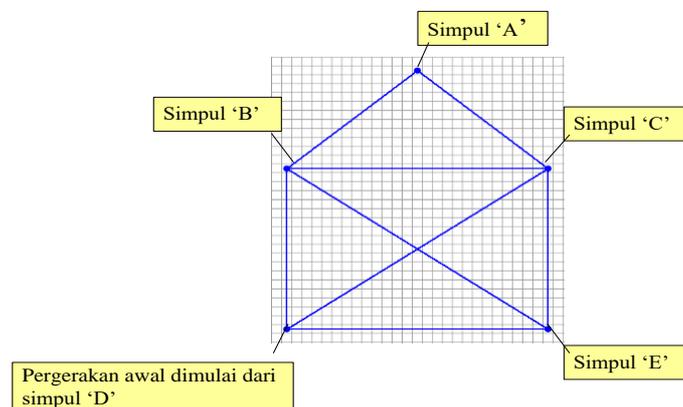
- 1) Pengumpulan data. Hal ini dilakukan melalui riset, mencari referensi buku maupun jurnal yang berhubungan dengan kecerdasan buatan, lintasan dan sirkuit *Euler* serta mempelajari algoritma DFS.
- 2) Algoritma DFS akan digunakan dalam pencarian solusi menyelesaikan permasalahan pada lintasan dan sirkuit *Euler*. Dalam tahap pencarian menggunakan DFS terdapat prosedur kerja yang harus dilalui dan waktu yang dibutuhkan dalam menemukan solusi. Setelah ditemukan solusi, maka akan dilakukan penggambaran ulang lintasan dan sirkuit *Euler* sesuai dengan solusi yang didapatkan oleh DFS.
- 3) Analisis, dilakukan untuk mengolah dan menganalisis kebutuhan informasi.
- 4) Perancangan, melakukan desain terhadap aplikasi yang dibuat, sehingga aplikasi dapat berjalan dengan baik.



Gambar 2. Alur Penelitian

## HASIL DAN PEMBAHASAN

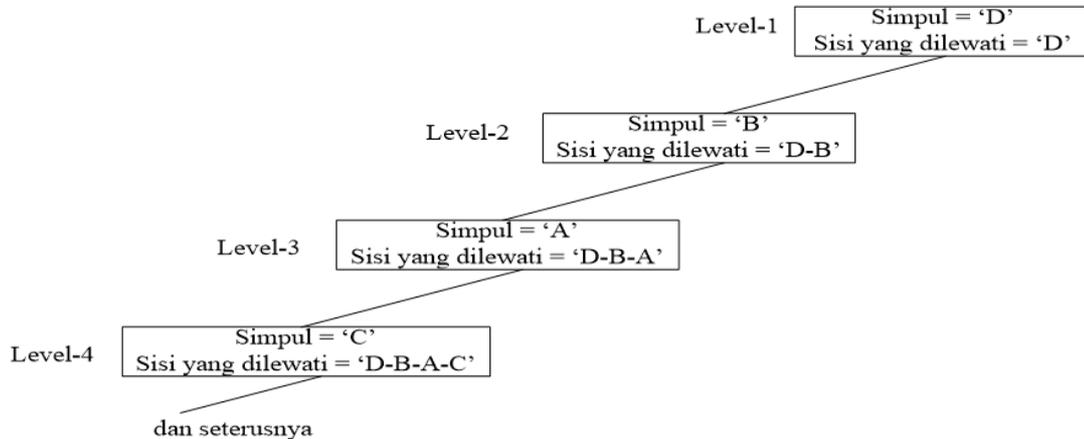
Pada bab ini akan dimulai dengan contoh kasus sederhana untuk menggambarkan permasalahan yang terjadi. Misalnya lintasan atau sirkuit *Euler* yang akan diselesaikan seperti disajikan pada Gambar 3.



Gambar 3. Pencarian Solusi di Lintasan *Euler*

Berdasarkan Gambar 3, terlihat hubungan antar simpul pada lintasan Euler yaitu (A – B), (A – C), (B – C), (B – D), (B – E), (C – E), (C – D) dan (D – E). Agar lebih jelas,

prosedur pencarian DFS untuk menyelesaikan graf pada Gambar 3, disajikan seperti pada Gambar 4.

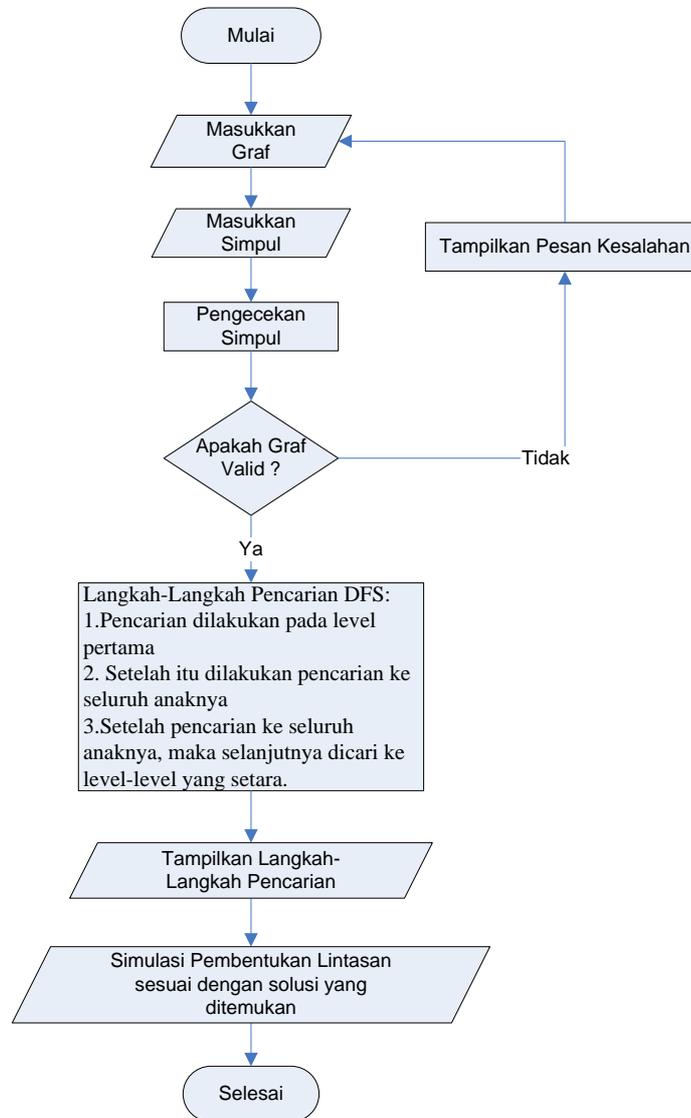


Gambar 4. Prosedur Pencarian DFS

Pencarian DFS hanya mengembangkan satu node anak dan dilanjutkan ke node anak yang baru terbentuk. Berdasarkan Gambar 4, pencarian akan dimulai dari node akar (simpul 'D'). Lalu dikembangkan satu node anak untuk menghubungkan simpul 'D' dan 'B', dilanjutkan dengan menghubungkan simpul 'B' dan 'A'. Begitu seterusnya hingga graf dilewati semua sisinya. Jika solusi belum didapatkan dan tidak ada sisi yang dapat dilewati, maka dilakukan pencarian mundur satu langkah ke node induk. Setiap sisi tidak akan dilewati lebih dari satu kali selama proses pencarian. Hasil penelusuran lengkap DFS (dengan titik D sebagai titik/simpul keberangkatan) adalah sebagai berikut:

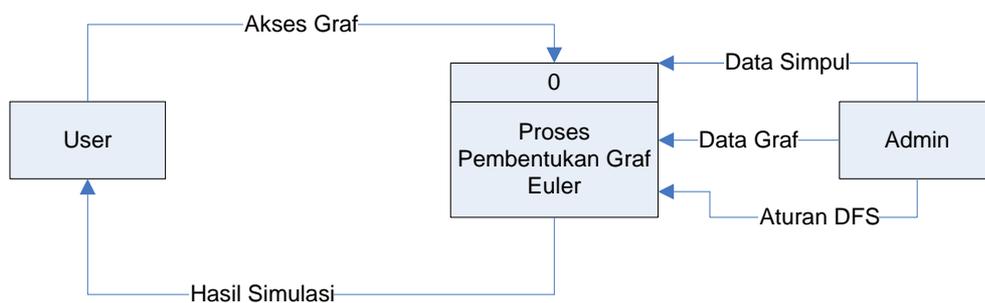
- 1) Keadaan 1, (D)
- 2) Keadaan 2, tarik garis ke simpul B (D-B)
- 3) Keadaan 3, tarik garis ke simpul A (D-B-A)
- 4) Keadaan 4, tarik garis ke simpul C (D-B-A-C)
- 5) Keadaan 5, tarik garis ke simpul E (D-B-A-C-E)
- 6) Keadaan 6, tarik garis ke simpul B (D-B-A-C-E-B)
- 7) Keadaan 7, tarik garis ke simpul C (D-B-A-C-E-B-C)
- 8) Keadaan 8, tarik garis ke simpul D (D-B-A-C-E-B-C-D)
- 9) Keadaan 9, tarik garis ke simpul E (D-B-A-C-E-B-C-D-E)
- 10) Selesai.

Alur penyelesaian atau proses yang dilakukan dalam penyelesaian lintasan dan sirkuit *Euler* disajikan seperti pada Gambar 5.



Gambar 5. Proses penyelesaian lintasan dan sirkuit *Euler*

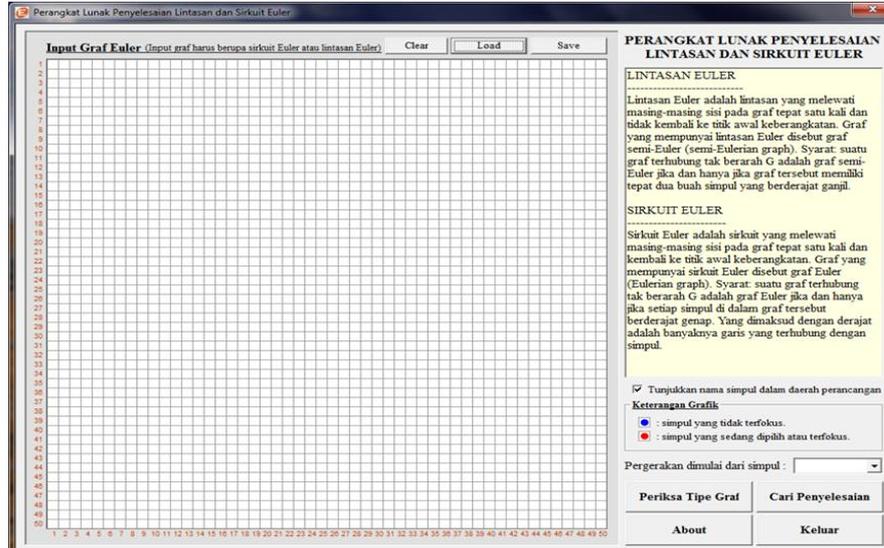
Diagram konteks dari proses penyelesaian pembentukan sirkuit dan lintasan Euler ini dapat dilihat pada Gambar 6. Yang menjadi entitas adalah *user* yang melakukan input graf dan mencari solusi pergerakan dengan algoritma DFS.



Gambar 6. Diagram Konteks Penyelesaian Pembentukan sirkuit dan lintasan *Euler*

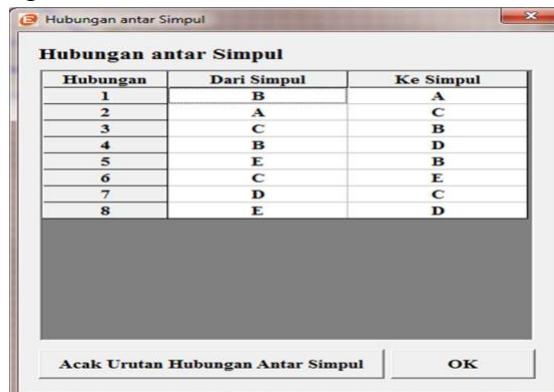
Implementasi algoritma DFS dalam penyelesaian permasalahan lintasan dan sirkuit *Euler* dibuat dalam sebuah aplikasi. Yang pertama *form input* graf, form ini berfungsi sebagai

tempat untuk merancang sirkuit atau lintasan *Euler* yang akan dicari penyelesaiannya menggunakan DFS. Dalam *form* ini berisi menu hapus, *load*, simpan, periksa tipe graf, cari penyelesaian, *about*, dan keluar disajikan seperti pada gambar 7.



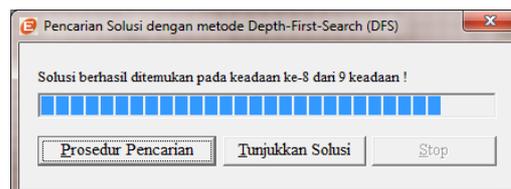
Gambar 7. Tampilan *Form* Utama

*Form* hubungan antar simpul akan muncul ketika mengklik tombol cari penyelesaian. Dalam *form* hubungan antar simpul terdapat dua menu yaitu acak urutan hubungan simpul dan ok seperti disajikan pada gambar 8



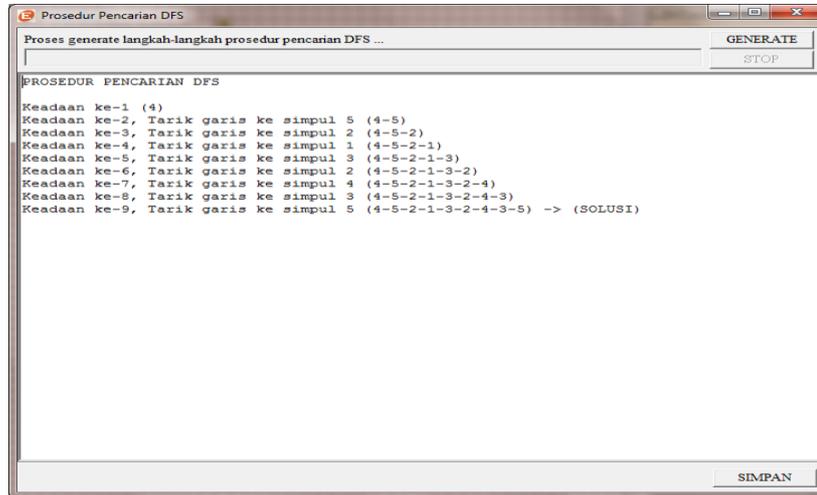
Gambar 8. Tampilan *Form* Hubungan Antar Simpul

*Form* pencarian DFS terdapat menu prosedur pencarian, tunjukkan solusi dan *stop* disajikan seperti pada gambar 9.



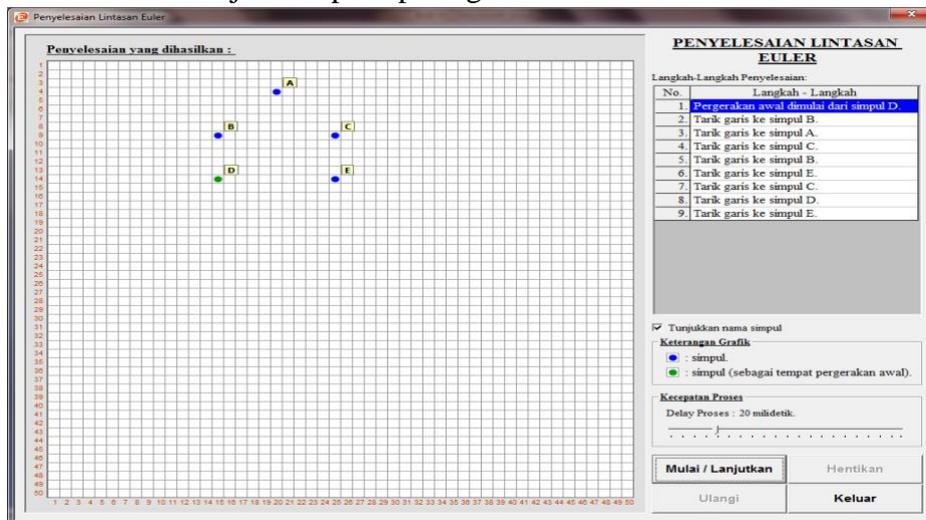
Gambar 9. Tampilan *Form* Pencarian DFS

*Form* prosedur pencarian DFS ini berfungsi untuk menampilkan langkah-langkah pencarian DFS untuk mencari solusi. Pada *form* ini terdapat menu *generate*, *stop* dan simpan disajikan seperti pada gambar 10.

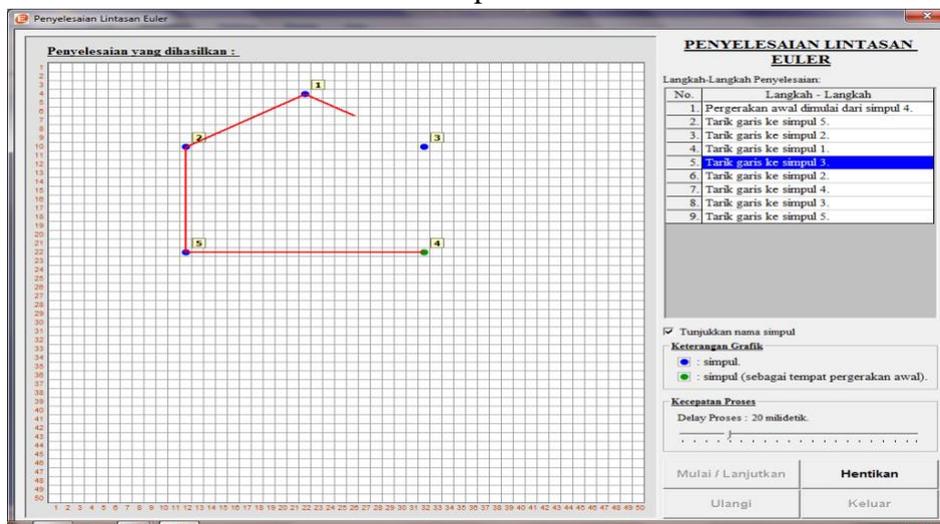


Gambar 10. Tampilan *Form* Prosedur Pencarian DFS

*Form* solusi berfungsi untuk menampilkan dan mensimulasikan langkah-langkah penyelesaian sesuai dengan solusi yang ditemukan. Pada *form* solusi terdapat menu mulai, ulangi, berhenti dan keluar disajikan seperti pada gambar 11 dan 12.



Gambar 11. Tampilan *Form* Solusi



Gambar 12. Tampilan *Form* Solusi

## KESIMPULAN

Setelah melakukan analisis perangkat lunak penyelesaian lintasan dan sirkuit *Euler*, maka dapat diambil kesimpulan bahwa aplikasi mampu mencari solusi penggambaran graf terhadap sirkuit dan lintasan *Euler* serta dapat mensimulasikan langkah-langkah penyelesaian yang telah diproses menggunakan algoritma DFS. Kemudian algoritma DFS relatif cepat dan efektif dalam mencari solusi penyelesaian lintasan dan sirkuit *Euler*.

## UCAPAN TERIMA KASIH

Ucapan terima kasih diucapkan kepada Program Studi Sistem Informasi serta Lembaga Penelitian dan Pengabdian kepada Masyarakat (LPPM) Universitas Victory Sorong yang telah memberikan kesempatan untuk melakukan penelitian serta dorongan dan motivasi selama penelitian dilakukan.

## DAFTAR PUSTAKA

- [1] D. Wilantara, N. B. Mulyono, And D. Winarso, "Automate Short Cyclic Well Job Candidacy Using Artificial Neural Networks-Enabled Lean Six Sigma Approach: A Case Study In Oil And Gas Company," *Int. J. Adv. Sci. Eng. Inf. Technol.*, Vol. 12, No. 4, Pp. 1639–1649, 2022, Doi: 10.18517/Ijaseit.12.4.12845.
- [2] B. Mehdi, C. Hasna, E. K. Ahmed, And O. Tayeb, "Intelligent Credit Scoring System Using Knowledge Management," *Iaes Int. J. Artif. Intell.*, Vol. 8, No. 4, Pp. 391–398, 2019, Doi: 10.11591/Ijai.V8i4.Pp391-398.
- [3] T. N. Lina *Et Al.*, "Comparison Analysis Of Breadth First Search And Depth Limited Search Algorithms In Sudoku Game," *Bull. Comput. Sci. Electr. Eng.*, Vol. 2, No. 2, Pp. 74–83, 2021, Doi: 10.25008/Bcsee.V2i2.1146.
- [4] M. J. Habibie, "Mengidentifikasi Tanaman Beracun Pada Pola Daun Dengan Jaringan Syaraf Tiruan Learning Vector Quantification," *J. Jtik (Jurnal Teknol. Inf. Dan Komunikasi)*, Vol. 3, No. 1, P. 7, 2019, Doi: 10.35870/Jtik.V3i1.47.
- [5] D. Hariyadi, C. B. Setiawan, And B. Setiyadi, "Pengembangan Sistem Pemantauan Dan Deteksi Serangan Pada Ekosistem Rumah Cerdas," *J. Komtika (Komputasi Dan Inform.)*, Vol. 5, No. 2, Pp. 132–139, 2021, Doi: 10.31603/Komtika.V5i2.5861.
- [6] R. Bramasta And L. Fitriana Masitoh, "Rancang Bangun Sistem Penilaian Produk Virtual Reality Berbasis Website Di Pt.Shinta Vr," *J. Komtika (Komputasi Dan Inform.)*, Vol. 5, No. 2, Pp. 103–111, 2021, Doi: 10.31603/Komtika.V5i2.6060.
- [7] M. Supriyanto Rumetna *Et Al.*, "Pengetahuan Serta Peran Auditor Secara Komprehensif Dalam Menghadapi Dampak Perkembangan Teknologi Informasi," *Komtika (Komputasi Dan Inform.)*, Vol. 6, No. 1, Pp. 26–38, 2022, Doi: 10.31603/Komtika.V6i1.6776.
- [8] A. Azlan, Y. Yusof, And M. F. M. Mohsin, "Univariate Financial Time Series Prediction Using Clonal Selection Algorithm," *Int. J. Adv. Sci. Eng. Inf. Technol.*, Vol. 10, No. 1, Pp. 151–156, 2020, Doi: 10.18517/Ijaseit.10.1.10235.
- [9] F. Arci, J. Reilly, P. Li, K. Curran, And A. Belatreche, "Forecasting Short-Term Wholesale Prices On The Irish Single Electricity Market," *Int. J. Electr. Comput. Eng.*, Vol. 8, No. 6, Pp. 4060–4078, 2018, Doi: 10.11591/Ijece.V8i6.Pp4060-4078.
- [10] S. Supriyadi, A. F. Wijaya, And R. G. Mayopu, "Perancangan Dan Implementasi Sistem Informasi Desa Wisata Kandri Berbasis Web," *Ccit J.*, Vol. 9, No. 3, Pp. 276–289, 2016, Doi: 10.33050/Ccit.V9i3.459.
- [11] A. S. M. Lumenta, "Perbandingan Metode Pencarian Depth-First Search, Breadth-First Search Dan Best-First Search Pada Permainan 8-Puzzle," *E-Journal Tek. Elektro Dan Komput.*, Pp. 1–

- 6, 2014.
- [12] N. Salleh, S. S. Yuhaniz, S. F. Sabri, And N. F. M. Azmi, “Enhancing Prediction Method Of Ionosphere For Space Weather Monitoring Using Machine Learning Approaches: A Review,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, Vol. 10, No. 1, Pp. 9–15, 2020, Doi: 10.18517/Ijaseit.10.1.10163.
- [13] A. Wibowo, . B., . L., And F. Fathurrahman, “Implementasi Algoritma Breadth First Search Dan Obstacle Detection Dalam Penelusuran Labirin Dinamis Menggunakan Robot Lego,” *Ilmu Komput. Dan Inf.*, Vol. 4, No. 1, Pp. 15–22, 2011, Doi: 10.21609/Jiki.V4i1.153.
- [14] J. Purwadi, A. R. C, And P. Solitaire, “Implementasi Algoritma Depth Limited Search Pada Permainan Peg Solitaire,” No. 1.
- [15] S. Lailiyah, A. Yusnita, And T. A. Panotogomo, “Penerapan Algoritma Depth First Search Pada Sistem Pencarian Dokumen,” In *Snitt*, 2017, Pp. 174–179.
- [16] B. Prasetyo And M. R. Hidayah, “Penggunaan Metode Depth First Search (DFS) Dan Breadth First Search (BFS) Pada Strategi Game Kamen Rider Decade Versi 0.3,” *Sci. J. Informatics*, Vol. 1, No. 2, Pp. 161–167, 2014, Doi: 10.15294/Sji.V1i2.4022.
- [17] D. Hariyadi And F. E. Nastiti, “Analisis Keamanan Sistem Informasi Menggunakan Sudomy Dan Owasp Zap Di Universitas Duta Bangsa Surakarta,” *J. Komtika (Komputasi Dan Inform.*, Vol. 5, No. 1, Pp. 35–42, 2021, Doi: 10.31603/Komtika.V5i1.5134.
- [18] M. S. Rumetna, “Pemanfaatan Cloud Computing Pada Dunia Bisnis: Studi Literatur,” *J. Teknol. Inf. Dan Ilmu Komput.*, Vol. 5, No. 3, Pp. 305–314, 2018, Doi: 10.25126/Jtiik.201853595.
- [19] M. S. Rumetna, T. N. Lina, W. Ferdinandus, F. Matahelumual, M. Pattiwael, And K. Sorong, “Optimasi Hasil Produksi Lemon Cina Dan Daun Jeruk Purut Dengan Memanfaatkan Teknologi Informasi,” *Selaparang J. Pengabd. Masy. Berkemajuan*, Vol. 6, No. 2, Pp. 733–740, 2022.
- [20] M. S. Rumetna And T. N. Lina, “Dampak Teknologi Informasi Bagi Generasi Milenial,” *Abdimas Unwahas*, Vol. 7, No. 1, Pp. 45–52, 2022.
- [21] M. S. Rumetna And T. N. Lina, “Forecasting Number Of Covid-19 Positive Patients In Sorong City Using The Moving Average And Exponential Smoothing Methods,” *IJICS (International J. Informatics Comput. Sci.*, Vol. 5, No. 1, Pp. 37–43, 2021, Doi: 10.30865/Ijics.V5i1.2908.
- [22] M. S. Rumetna, T. N. Lina, I. S. Rajagukguk, F. S. Pormes, And A. B. Santoso, “Payroll Information System Design Using Waterfall Method,” *Int. J. Adv. Data Inf. Syst.*, Vol. 3, No. 1, Pp. 1–10, 2022, Doi: 10.25008/Ijadis.V3i1.1227.
- [23] M. S. Rumetna, T. N. Lina, T. Aponno, A. Palisoa, And F. Singgir, “Penerapan Metode Simpleks Dan Software Pom- Qm Untuk Optimalisasi Hasil Penjualan Pentolan Bakso,” *Ilm. Manaj. Inform. Dan Komput.*, Vol. 02, No. 03, Pp. 143–149, 2018.
- [24] M. S. Rumetna, E. E. Renny, And T. N. Lina, “Designing An Information System For Inventory Forecasting,” *Int. J. Adv. Data Inf. Syst.*, Vol. 1, No. 2, Pp. 80–88, 2020, Doi: 10.25008/Ijadis.V1i2.187.
- [25] M. S. Rumetna, “Audit Lingkungan Dan Pengendalian Teknologi Informasi Pada PT. XYZ,” *Simetris J. Tek. Mesin, Elektro Dan Ilmu Komput.*, Vol. 9, No. 2, Pp. 753–768, 2018, Doi: 10.24176/Simet.V9i2.2294.
- [26] M. S. Rumetna, T. N. Lina, A. B. Santoso, J. Karay, R. Komansilan, And B. G. Kaitelapatay, “Pengetahuan Serta Peran Auditor Secara Komprehensif Dalam Menghadapi Dampak Perkembangan Teknologi Informasi,” *J. KOMTIKA (Komputasi Dan Inform.*, Vol. 6, No. 1, Pp. 26–38, 2022, Doi: <https://doi.org/10.31603/Komtika.V6i1.6776>.
- [27] M. S. Rumetna, “Kombinasi Gnu Privacy Guard Dan Hamming Distance Untuk Keamanan Email Serta Jalur Sertifikasi Combination Of Gnu Privacy Guard And Hamming Distance For Email Security And Certification Paths,” *Elektro Luceat [November]*, Vol. 7, No. 2, Pp. 151–160, 2021.

- [28] M. S. Rumetna *Et Al.*, “Berbasis Website Pada Perusahaan Cendrawasih Wiputra Mandiri Kota Sorong Design Of A Website-Based Demand Information System In Cendrawasih Wiputra Mandiri Company,” *Elektro Luceat*, Vol. 7, No. 1, Pp. 10–19, 2021.
- [29] M. S. Rumetna, T. N. Lina, And A. B. Santoso, “Rancang Bangun Aplikasi Koperasi Simpan Pinjam Menggunakan Metode Research And Development,” *Tek. Mesin, Elektro Dan Ilmu Komput.*, Vol. 11, No. 1, Pp. 119–128, 2020.



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)

---