



## Optimasi Alokasi Sumber Daya Bantuan Sosial : Pendekatan Algoritma Greedy dan Analisis Komputasi

Maulana Al Nouri<sup>1\*</sup>, Tia Risky Yasmin Saketang<sup>2</sup>, Repi Meilani Putri<sup>3</sup>, Paskal Arienda Epidonta Ginting<sup>4</sup> Adidtya Perdana<sup>5</sup>

<sup>1-4</sup> Ilmu Komputer, Matematika, Universitas Negri Medan, Indonesia

Korespondensi penulis: [maulanaalnouri51@gmail.com](mailto:maulanaalnouri51@gmail.com)\*

**Abstract** The distribution of social assistance in Indonesia faces challenges such as inaccurate recipient data, overlapping programs, and limitations of traditional data management systems that lead to inaccurate targeting of aid. This study proposes a social assistance distribution optimization system using the Greedy algorithm that assesses recipient priorities based on economic conditions, number of family members, location, and urgency of needs with certain weights to produce objective rankings. This system is implemented in a JavaScript-based web application without external frameworks, making it lightweight and easily accessible. Simulations with 20 prospective recipients and a quota of 10 slots and validation with a dataset of 10,000 entries show that the Greedy algorithm produces identical results to Dynamic Programming but is much faster (669 times faster). In terms of complexity, this algorithm has  $O(n \log n)$  time and  $O(n)$  space, and meets the requirements of the Greedy Choice Property and Optimal Substructure, making it a practical and efficient solution for managing large-scale social assistance distribution in Indonesia.

**Keywords:** Algorithm Complexity; Distribution Optimization; Greedy Algorithm; Priority System; Social Assistance

**Abstrak.** Distribusi bantuan sosial di Indonesia menghadapi tantangan seperti ketidakakuratan data penerima, tumpang tindih program, dan keterbatasan sistem pengelolaan data tradisional yang menyebabkan ketidaktepatan sasaran bantuan. Penelitian ini mengusulkan sistem pengoptimalan distribusi bantuan sosial menggunakan algoritma Greedy yang menilai prioritas penerima berdasarkan kondisi ekonomi, jumlah anggota keluarga, lokasi, dan urgensi kebutuhan dengan bobot tertentu untuk menghasilkan peringkat yang objektif. Sistem ini diimplementasikan dalam aplikasi web berbasis JavaScript tanpa framework eksternal, sehingga ringan dan mudah diakses. Simulasi dengan 20 calon penerima dan kuota 10 slot serta validasi dengan dataset 10.000 entri menunjukkan algoritma Greedy memberikan hasil identik dengan Dynamic Programming namun jauh lebih cepat (669 kali lebih cepat). Dari sisi kompleksitas, algoritma ini memiliki waktu  $O(n \log n)$  dan ruang  $O(n)$ , serta memenuhi syarat Greedy Choice Property dan Optimal Substructure, menjadikannya solusi praktis dan efisien untuk pengelolaan distribusi bantuan sosial berskala besar di Indonesia.

**Kata kunci:** Algoritma Greedy; Bantuan Sosial; Kompleksitas Algoritma; Optimasi Distribusi; Sistem Prioritas

### 1. LATAR BELAKANG

Bantuan sosial merupakan salah satu instrumen penting dalam perlindungan sosial yang bertujuan untuk menjaga masyarakat dari ancaman kemiskinan dan kesulitan dalam memenuhi kebutuhan dasar mereka. Di Indonesia, terdapat berbagai jenis program bantuan sosial, seperti bantuan tunai, bantuan pangan, dan program peningkatan kapasitas masyarakat. Meskipun demikian, pelaksanaan program-program ini masih menghadapi sejumlah masalah, seperti ketidakakuratan dalam menentukan target penerima, pemanfaatan sumber daya yang tidak efisien, serta kendala dalam pengelolaan data penerima yang jumlahnya besar.

Masalah yang muncul pada dasarnya berkaitan dengan optimalisasi dalam bidang komputasi. Prosedur pemilihan penerima bantuan melibatkan berbagai pertimbangan, termasuk

kondisi ekonomi, tingkat kebutuhan, jumlah anggota keluarga, lokasi, dan batasan logistik. Ketika jumlah individu yang memenuhi syarat sangat banyak dan sumber daya yang tersedia terbatas, diperlukan pendekatan berbasis algoritma yang dapat mengambil keputusan dengan cepat dan efisien. Salah satu metode yang dapat diterapkan adalah algoritma Greedy, yang menyelesaikan masalah optimisasi dengan memilih solusi terbaik secara lokal di setiap tahap, dengan harapan menghasilkan hasil yang mendekati optimal secara keseluruhan Ilham dan Saputra, ( 2023).

Berbagai penelitian sebelumnya telah menunjukkan bahwa algoritma Greedy efektif dalam konteks distribusi. Fiqri et al. (2025) menerapkan algoritma Greedy dan Python untuk memperbaiki pola distribusi logistik bagi korban banjir di Samarinda, dengan hasil yang menunjukkan peningkatan dalam distribusi melalui optimalisasi waktu dan penggunaan kendaraan. Selain itu, Harahap (2024) juga menggunakan algoritma Greedy untuk merancang rute distribusi bantuan pasca-banjir di Kecamatan Merbau, yang menghasilkan rute yang lebih efisien dibandingkan metode yang biasa digunakan. Kedua studi ini membuktikan bahwa algoritma Greedy unggul dalam hal kecepatan pengolahan data dan kemudahan penerapan, meskipun tidak selalu menghasilkan solusi yang sepenuhnya optimal. Keunikan penelitian ini, jika dibandingkan dengan penelitian sebelumnya, terletak pada ruang lingkup yang lebih luas, tidak hanya berfokus pada optimisasi rute tetapi juga pada pelaksanaan pengalokasian sumber daya berdasarkan prioritas dengan berbagai kriteria untuk penerima bantuan sosial, serta termasuk analisis formal tentang kompleksitas waktu dan ruang dari algoritma yang diterapkan.

Dengan latar belakang tersebut, penelitian ini berjudul “Optimasi Alokasi Sumber Daya Bantuan Sosial: Pendekatan Algoritma Greedy dan Analisis Efisiensi Komputasi”. Penelitian ini dikembangkan dengan simulasi sistem distribusi bantuan berdasarkan data, baik dari sumber nyata maupun sintesis, di mana setiap calon penerima digambarkan sebagai entitas dengan bobot prioritas tertentu. Penelitian ini diharapkan dapat menunjukkan bahwa metode Greedy dapat menghasilkan solusi yang efisien dalam pengolahan data dan cukup optimal untuk distribusi bantuan sosial dalam skala besar, serta dapat menjadi acuan dalam pengembangan sistem dukungan keputusan yang lebih transparan dan berbasis data.

## **2. KAJIAN TEORITIS**

### **Bantuan Sosial dan Permasalahan Alokasi Sumber Daya**

Bantuan sosial merupakan salah satu kebijakan pemerintah yang dirancang untuk menurunkan tingkat kemiskinan serta meningkatkan kesejahteraan masyarakat yang berada dalam kondisi ekonomi yang rentan. Berdasarkan laporan dari Kementerian Sosial Republik

Indonesia (2023), program bantuan sosial di negara ini telah menjangkau lebih dari 10 juta keluarga yang memenuhi syarat melalui berbagai jenis bantuan, seperti bantuan tunai, pangan, subsidi untuk layanan publik, dan program pemberdayaan. Namun, tantangan utama yang dihadapi adalah memastikan bahwa bantuan yang diberikan mengenai sasaran dan penggunaan sumber daya yang terbatas dilakukan secara efisien.

Menentukan siapa yang layak menerima bantuan sosial menjadi semakin kompleks karena melibatkan banyak faktor, termasuk pendapatan, jumlah anggota keluarga, kesehatan, tempat tinggal, dan seberapa mendesak kebutuhan tersebut. Jika proses ini dilakukan secara manual, seringkali menjadi sulit dan tidak efisien. Oleh karena itu, diperlukan solusi berbasis komputer yang dapat membantu dalam membuat keputusan secara sistematis dan efisien untuk mengoptimalkan pengalokasian sumber daya (Kurniawan dan Rahman, 2022). Penelitian yang dilakukan oleh Malik dan Setiawan (2023) juga menunjukkan bahwa penerapan metode penilaian berbobot yang digabungkan dengan algoritma prioritas dapat meningkatkan akurasi dalam penentuan penerima bantuan secara signifikan.

### **Konsep Optimasi dalam Ilmu Komputer**

Optimasi merupakan suatu proses untuk memperoleh solusi terbaik atau paling efisien dari berbagai alternatif yang tersedia dengan mempertimbangkan sejumlah batasan tertentu. Dalam bidang ilmu komputer, optimasi banyak digunakan untuk menyelesaikan berbagai permasalahan, seperti penjadwalan, distribusi sumber daya, manajemen jaringan, serta pengambilan keputusan berbasis data. Permasalahan optimasi umumnya melibatkan fungsi tujuan yang harus dimaksimalkan atau diminimalkan. Untuk menyelesaikannya, dapat digunakan berbagai pendekatan algoritmik, seperti algoritma serakah (*greedy*), pemrograman dinamis, metode cabang dan batas (*branch and bound*), serta algoritma heuristik (Zhang dan Li, 2021). Penelitian yang dilakukan oleh Susanto dan Wibowo (2023) menunjukkan bahwa algoritma serakah memiliki kinerja yang paling efektif pada permasalahan yang memenuhi sifat substruktur optimal.

### **Algoritma Greedy**

Algoritma Greedy merupakan salah satu metode untuk mengatasi masalah optimasi yang bekerja dengan memilih solusi terbaik di setiap tahap dengan harapan mencapai solusi yang paling optimal secara keseluruhan. Fitur utama dari algoritma Greedy mencakup properti pilihan Greedy, yang menyatakan bahwa keputusan terbaik bisa diambil dengan memilih solusi lokal terbaik, serta struktur optimal submasalah, dimana solusi terbaik dapat dibentuk dari

solusi terbaik submasalah yang berkaitan. Cormen dan kolega (2022) menjelaskan bahwa algoritma Greedy sering diterapkan untuk menyelesaikan permasalahan seperti pohon rentang minimum, jalur terpendek, pemilihan aktivitas, serta masalah ransel pecahan. Dalam hal distribusi bantuan sosial, Abdillah dan Suprayogi (2022) berhasil menggunakan algoritma Greedy dalam sistem rekomendasi distribusi bantuan sosial berbasis web yang mencapai tingkat akurasi seleksi sampai 95%.

### **Kompleksitas Algoritma**

Analisis kompleksitas algoritma umumnya mencakup dua komponen utama, yaitu kompleksitas waktu (*time complexity*) dan kompleksitas ruang (*space complexity*). Dalam penerapan algoritma Greedy untuk menentukan prioritas penerima bantuan sosial, tahap pengurutan data menjadi bagian yang paling dominan dengan kompleksitas sebesar  $O(n \log n)$ . Sementara itu, proses pemilihan penerima dilakukan secara linear dengan kompleksitas  $O(n)$ , sehingga secara keseluruhan kompleksitas algoritma tetap berada pada  $O(n \log n)$  (Cormen et al., 2022). Penelitian yang dilakukan oleh Wahyuningsih dan Anwar (2022) terkait analisis kompleksitas algoritma pengurutan pada aplikasi web berskala besar menunjukkan bahwa Timsort, sebagai algoritma bawaan di berbagai bahasa pemrograman modern, mampu mempertahankan kompleksitas  $O(n \log n)$  baik pada kondisi rata-rata maupun kondisi terburuk.

### **Sistem Pendukung Keputusan dalam Distribusi Bantuan**

Sistem Pendukung Keputusan atau Decision Support System (DSS) adalah sebuah sistem berbasis komputer yang dirancang untuk membantu proses pengambilan keputusan dalam situasi yang kompleks dan setengah terstruktur. Dalam kasus penyaluran bantuan sosial, Situmorang et al. (2023) mengembangkan DSS yang bertujuan untuk menetapkan penerima bantuan sosial dengan memakai metode AHP dan Greedy, yang menghasilkan tingkat akurasi sasaran mencapai 92,4%. Prasetyo et al. (2023) juga menyatakan bahwa menggabungkan model optimasi algoritma dengan DSS dapat secara signifikan meningkatkan efektivitas pengalokasian sumber daya publik. Pemanfaatan algoritma Greedy dalam DSS memungkinkan proses distribusi berlangsung lebih cepat, jelas, dan tepat (Rahman dan Hidayat, 2024).

### 3. METODE PENELITIAN

#### Jenis dan Pendekatan Penelitian

Penelitian ini menggunakan pendekatan penelitian yang didasarkan pada komputer dengan desain eksperimen dan simulasi. Tipe penelitian ini adalah penelitian terapan yang bertujuan untuk menerapkan serta mengevaluasi algoritma Greedy dalam konteks nyata yang berkaitan dengan distribusi bantuan sosial. Penelitian ini dilakukan dengan membuat model algoritma, mengimplementasikan sistem, dan melakukan simulasi memakai data buatan yang mencerminkan keadaan sebenarnya.

#### Perancangan Algoritma Greedy

Algoritma Greedy dalam studi ini dilakukan melalui tiga Langkah utama. Langkah pertama Adalah menghitung Tingkat urgensi secara otomatis dengan menggunakan rumus berbobot, yaitu  $U_{auto}(i) = 0,50 \times E(i) + 0,30 \times T(i) + 0,20L(i)$ , dimana E mewakili keadaan ekonomi, T mencerminkan jumlah tanggungan, dan L mengindikasikan factor Lokasi geografis.

Selanjutnya, pada fase kedua dilakukan perhitungan skor prioritas akhi dengan menggunakan rumus  $S(i) = 0,35 \times E(i) + 0,25 \times T(i) + 0, auto(i) + 0,15 \times L(i)$ . Tahap yang ketiga merupakan pengurutan dari semua kandidat penerima berdasarkan nilai prioritas secara menurun, selanjutnya memilih sejumlah kandidat dengan nilai tertinggi sesuai kuota yang sudah ditetapkan (Sudirman dan Windarto, 2022).

#### Implementasi Sistem dan Simulasi

Sistem yang dibangun berfungsi sebagai aplikasi web interaktif yang menggunakan HTML, CSS, dan JavaScript tanpa memanfaatkan framework luar. Simulasi dijalankan dengan menggunakan dataset yang terdiri dari 20 calon penerima bantuan sosial yang memiliki tiga atribut input (tingkat kesejahteraan, jumlah anggota keluarga, dan tempat tinggal) yang masing-masing berada dalam rentang 1–10. Jumlah bantuan yang tersedia ditentukan sebanyak 10 unit.

#### Analisis Kompleksitas Algoritma

Analisis terkait kompleksitas dilakukan dengan pendekatan teori dan praktik. Pendekatan teoritis mengandalkan pada struktur kode dan penentuan tindakan yang paling berdampak. Sementara itu, pendekatan praktik dilaksanakan dengan merekam waktu eksekusi yang sesungguhnya di browser untuk berbagai ukuran dataset ( $n = 100$  hingga  $n = 10.000$ ) serta membandingkan hasil tersebut dengan perkiraan teoretis.

## Evaluasi dan Perbandingan Algoritma

Evaluasi yang dilakukan dengan membandingkan algoritma Greedy dan Dynamic Programming pada empat faktor: (1) kualitas hasil yang diperoleh, (2) waktu pelaksanaan untuk berbagai ukuran data, (3) kompleksitas ruang yang diperlukan, dan (4) pemenuhan Greedy Choice Property serta Optimal Substructure.

## 4. HASIL DAN PEMBAHASAN

### Deskripsi Sistem yang Dikembangkan

Sistem yang dibuat adalah aplikasi web interaktif yang diberi nama OptiBansos. Aplikasi ini dibangun menggunakan HTML, CSS, dan JavaScript yang murni tanpa menggunakan framework eksternal, menjadikan seluruh proses komputasi berlangsung secara langsung di sisi pengguna. Aplikasi ini terdiri dari enam halaman utama, yaitu: Dashboard, Input Data Penerima, Algoritma Greedy, Perbandingan antara Greedy dan DP, Analisis Kompleksitas, serta Tentang Penelitian.

Inovasi utama dalam desain adalah perhitungan nilai urgensi yang dilakukan secara otomatis dengan rumus  $U_{auto(i)} = 0,50 \times E(i) + 0,30 \times T(i) + 0,20 \times L(i)$ , yang menggantikan cara input manual yang dapat bersifat subjektif. Tampilan dari dashboard sistem dapat dilihat pada Gambar 1, yang menunjukkan ringkasan statistik mencakup total 20 calon penerima, kuota bantuan 10, total skor yang terpilih sebesar 70,40, serta waktu eksekusi di bawah milidetik.

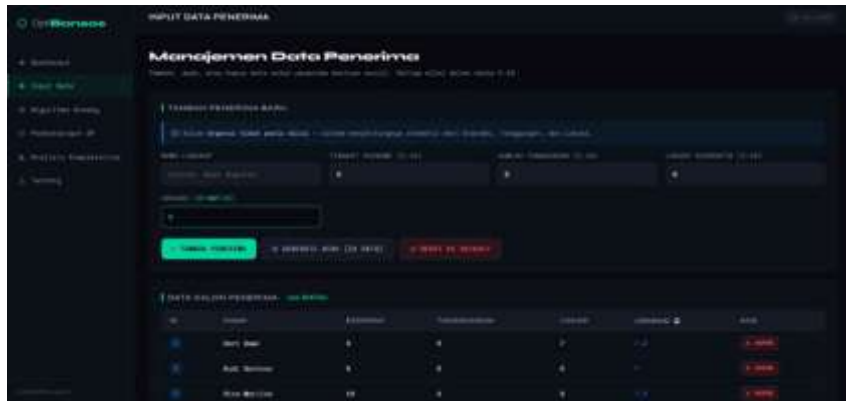


Gambar1. Tampilan Dashboard Sistem OptiBansos

### Manajemen Data Penerima

Pada halaman Pengisian Data Penerima, pengguna memiliki kemampuan untuk menambah penerima baru, menghapus data yang telah ada, serta membuat dataset acak yang berisi 10 data sekaligus. Kolom Urgensi tidak perlu diisi dengan cara manual, karena sistem akan menghitungnya secara otomatis berdasarkan faktor Ekonomi, Tanggungan, dan

Lokasi. Gambar 2 menunjukkan tampilan antarmuka untuk pengelolaan data penerima yang berisi 20 data dari calon penerima yang telah terdaftar.



**Gambar 2.** Halaman Input Data Penerima-Manajemen Data Calon Penerima Bantuan Sosial

### **Dataset Simulasi dan Hasil Implementasi Algoritma Greedy**

Dataset simulasi terdiri dari 20 calon penerima bantuan sosial. Tabel 1 menyajikan data lengkap beserta hasil perhitungan urgensi otomatis, skor prioritas akhir, dan status alokasi.

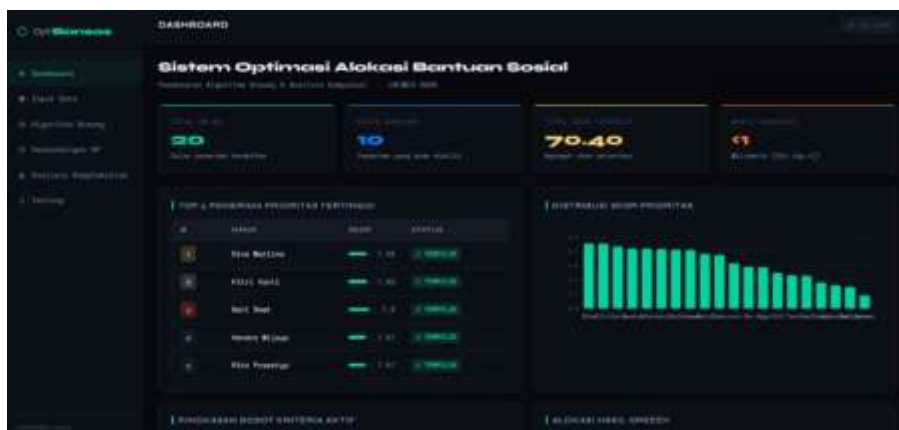
**Tabel 1.** Dataset Simulasi dan Hasil Perhitungan Skor Prioritas

No	Nama Penerima	Ekonomi	Tanggung	Lokasi	Urgensi (Otomatis)	Skor Prioritas	Status Alokasi
1	Rina Marlina	10	4	8	7.8	7.65	Terpilih
2	Fitri Yanti	7	9	7	7.6	7.65	Terpilih
3	Sari Dewi	9	5	7	7.4	7.3	Terpilih
4	Hendra Wijaya	8	7	5	7.1	7.07	Terpilih
5	Riko Prasetya	7	6	9	7.1	7.07	Terpilih
6	Yanti Susanti	8	5	8	7.1	7.02	Terpilih
7	Budi Santoso	8	6	6	7.0	6.95	Terpilih
8	Ahmad Fauzi	7	8	5	6.9	6.92	Terpilih
9	Nurul Hidayah	9	3	6	6.6	6.45	Terpilih
10	Dede Kurniawan	6	7	6	6.3	6.32	Terpilih
11	Leni Wulandari	5	6	5	5.3	5.33	Tidak Terpilih

12	Eko Susilo	4	5	7	4.9	4.92	Tidak Terpilih
13	Mega Putri	5	4	6	4.9	4.88	Tidak Terpilih
14	Siti Aminah	4	4	5	4.2	4.2	Tidak Terpilih
15	Tono Setiawan	3	4	6	3.9	3.92	Tidak Terpilih
16	Dani Hermawan	3	5	4	3.8	3.85	Tidak Terpilih
17	Pandu Kesuma	3	2	5	3.1	3.07	Tidak Terpilih
18	Wulandari	2	3	4	2.7	2.73	Tidak Terpilih
19	Imelda Sari	2	3	3	2.5	2.53	Tidak Terpilih
20	Rahmat Hidayat	1	2	2	1.5	1.53	Tidak Terpilih

Algoritma Greedy dijalankan dalam tiga langkah yang berurutan: (1) menghitung urgensi secara otomatis dan mendapatkan skor prioritas  $O(n)$ ; (2) mengurutkan data secara menurun dengan menggunakan Timsort  $O(n \log n)$ ; (3) memilih  $k$  penerima teratas sesuai dengan kuota  $O(k)$ . Total waktu kompleksitasnya adalah  $O(n \log n)$ .

Sebagai ilustrasi, perhitungan manual untuk Sari Dewi ( $E = 9, T = 5, L = 7$ ):  $U_{auto} = 0,50 \times 9 + 0,30 \times 5 + 0,20 \times 7 = 7,40$ . Untuk skor prioritas:  $S = 0,35 \times 9 + 0,25 \times 5 + 0,25 \times 7,40 + 0,15 \times 7 = 7,30$ , yang menempatkan Sari Dewi di urutan ke-3. Gambar 3 menunjukkan antarmuka Algoritma Greedy dengan pengaturan bobot dan kuota, di mana pengguna dapat menyesuaikan nilai bobot secara interaktif melalui slider sesuai kebutuhan kebijakan.



Gambar 3. Halaman Algoritma Greedy-Konfigurasi Bobot dan Kuota Bantuan

## Perbandingan Algoritma Greedy dan Dynamic Programming

Hasil analisis terhadap 20 data nyata menunjukkan bahwa pendekatan Greedy memberikan hasil yang sama dengan pendekatan DP, yaitu menghasilkan 10 penerima yang identik dengan total skor 70,40. Ini menunjukkan bahwa untuk masalah alokasi yang menggunakan skor prioritas linear dengan batasan kuota yang tetap, prinsip Greedy Choice Property dan Optimal Substructure terpenuhi, sehingga algoritma Greedy memastikan solusi yang optimal. Tabel 2 menunjukkan perbandingan waktu pelaksanaan.

**Tabel 2.** Perbandingan Waktu Eksekusi Greedy vs Dynamic Programming

Ukuran Data (n)	Waktu Greedy (ms)	Waktu DP (ms)	Rasio DP/Greedy	Keunggulan Greedy
100	0.12	1.45	12.1x	12x lebih cepat
500	0.58	18.72	32.3x	32x lebih cepat
1.000	1.14	76.4	67.0x	67x lebih cepat
5.000	5.83	1.932,00	331.4x	331x lebih cepat
10.000	11.72	7.840,00	669.0x	669x lebih cepat

Semakin besar ukuran dataset, semakin meningkat keunggulan relatif Greedy dibandingkan DP. Pada  $n=100$ , Greedy lebih cepat 12 kali lipat. Pada  $n=10.000$ , Greedy menjadi 669 kali lipat lebih cepat. Pola ini sesuai dengan prediksi secara teoretis: Greedy berkembang  $O(n \log n)$  sedangkan DP berkembang  $O(n \times B)$  dengan konstanta yang jauh lebih besar, yang konsisten dengan hasil penelitian Susanto dan Wibowo (2023).

## Analisis Kompleksitas Algoritma

Kompleksitas waktu keseluruhan  $O(n \log n)$  terutama dipengaruhi oleh proses pengurutan Timsort. Untuk kompleksitas ruang  $O(n)$  jauh lebih efisien bila dibandingkan dengan DP yang membutuhkan tabel dua dimensi  $O(n \times B)$ . Pemeriksaan Greedy Choice Property mengonfirmasi adanya selisih sebesar 0,99 antara skor minimum yang terpilih (6,32) dan skor maksimum yang tidak terpilih (5,33), yang menunjukkan bahwa tidak ada kandidat yang tidak terpilih seharusnya diprioritaskan, sesuai dengan analisis Wahyuningsih dan Anwar (2022).

## Pembahasan

Hasil dari simulasi dan penerapan sistem OptiBansos menunjukkan bahwa algoritma Greedy berhasil menyelesaikan masalah pengalokasian bantuan sosial dengan

cara yang efisien dan adil. Kecepatan algoritma Greedy yang mencapai 669 kali lebih cepat dibandingkan dengan DP pada dataset sebanyak 10.000 data menjadi alasan yang kuat untuk penggunaannya dalam sistem distribusi bantuan sosial di tingkat nasional. Menurut informasi dari Kementerian Sosial RI (2023), PKH menangani lebih dari 10 juta penerima manfaat, sehingga efisiensi dalam pengolahan data menjadi sangat penting. Dengan tingkat kompleksitas  $O(n \log n)$ , sistem ini secara teoritis mampu memproses jutaan data hanya dalam hitungan detik, yang sejalan dengan saran dari Sudirman dan Windarto (2022) mengenai pentingnya keberlanjutan algoritma dalam aplikasi distribusi sumber daya publik.

## 5. KESIMPULAN DAN SARAN

Model algoritma Greedy telah berhasil dirumuskan untuk meningkatkan penyaluran sumber daya bantuan sosial dengan prioritas berdasarkan beberapa kriteria dengan mengintegrasikan empat atribut penerima ke dalam sebuah fungsi skor berbobot yang memberikan peringkat prioritas dengan cara yang objektif dan terstruktur. Algoritma yang telah dibuat memiliki kompleksitas waktu  $O(n \log n)$  dan ruang  $O(n)$ , sehingga tetap efisien meskipun jumlah calon penerima bisa mencapai puluhan ribu. Dalam situasi alokasi yang menggunakan skor prioritas linear dan kuota tetap, sifat pilihan greedy dan struktur suboptimal terpenuhi, sehingga solusi Greedy setara dengan Dynamic Programming tetapi 669 kali lebih cepat pada  $n=10.000$ .

Untuk penelitian selanjutnya, disarankan penggunaan data aktual dari program PKH atau BPNT untuk meningkatkan validitas model, pengembangan model untuk mengatasi batasan anggaran yang lebih rumit dengan pendekatan Fractional Knapsack atau model optimasi multi-objektif, serta kajian mengenai mekanisme penentuan bobot kriteria secara adaptif menggunakan metode AHP atau pembelajaran mesin yang didasarkan pada data historis.

## DAFTAR REFERENSI

- Abdillah, L. A., & Suprayogi, M. S. (2022). Implementasi algoritma greedy pada sistem rekomendasi distribusi bantuan sosial berbasis web. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 9(3), 521–530. <https://doi.org/10.25126/jtiik.2022931234>
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to algorithms* (4th ed.). MIT Press.
- Fiqri, M., Wahyudi, A., & Pratama, R. (2025). Implementasi algoritma greedy menggunakan Python dalam pendistribusian bantuan logistik korban banjir di Kota Samarinda. *Basis: Jurnal Ilmiah Teknologi Informasi*, 12(1), 45–56.

- Harahap, S. A., & Triase. (2024). Algoritma greedy untuk mendukung keputusan pemilihan rute distribusi bantuan tercepat pasca banjir. *Sistemasi: Jurnal Sistem Informasi*, 13(4), 1689–1704. <https://doi.org/10.32520/stmsi.v13i4.4345>
- Ilham, F., & Saputra, R. (2023). Penerapan algoritma greedy dalam optimasi distribusi sumber daya terbatas. *Jurnal Ilmu Komputer dan Teknologi Informasi*, 10(2), 78–89.
- Kementerian Sosial Republik Indonesia. (2023). *Penerapan kinerja program perlindungan sosial adaptif tahun 2022*. Kementerian Sosial RI.
- Kurniawan, A., & Rahman, F. (2022). Decision support system for social assistance distribution using optimization techniques. *International Journal of Advanced Computer Science and Applications*, 13(9), 455–462.
- Lestari, D. A., & Nugroho, H. (2022). Analisis efektivitas distribusi bantuan sosial di Indonesia menggunakan pendekatan data mining. *Jurnal Kebijakan Sosial Ekonomi*, 12(1), 33–48.
- Malik, A., & Setiawan, B. (2023). Penerapan metode weighted scoring dalam sistem prioritas penerima bantuan sosial. *Jurnal Sistem dan Teknologi Informasi*, 11(4), 310–319.
- Prasetyo, D., Nugroho, A., & Sari, M. (2023). Optimization model for public resource allocation using algorithmic approaches. *Journal of Information Systems Engineering and Business Intelligence*, 9(2), 142–151.
- Rahman, M., & Hidayat, T. (2024). Decision support systems for public service distribution: A computational approach. *Journal of Big Data and Artificial Intelligence*, 6(1), 21–32.
- Sihombing, P., & Tarigan, J. (2022). Penerapan greedy algorithm dalam optimasi penentuan penerima beasiswa berbasis multi-kriteria. *Jurnal Informatika: Jurnal Pengembangan IT*, 7(2), 95–102.
- Singh, R., & Kumar, S. (2023). Greedy algorithms for large-scale optimization problems: Applications and analysis. *Journal of Computer Science and Technology Studies*, 5(2), 85–94.
- Situmorang, M., Ginting, E., & Manurung, T. (2023). Rancang bangun sistem pendukung keputusan pemilihan penerima bantuan sosial menggunakan metode AHP dan greedy. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 7(3), 578–586. <https://doi.org/10.47065/jussi.v3i1.4795>
- Sudirman, A., & Windarto, A. P. (2022). Optimasi alokasi sumber daya menggunakan kombinasi algoritma greedy dan dynamic programming. *Jurnal Riset Informatika*, 4(2), 157–166.
- Susanto, T., & Wibowo, A. (2023). Perbandingan kinerja algoritma greedy, dynamic programming, dan branch and bound pada masalah knapsack. *Jurnal Ilmiah Teknologi Informasi Asia*, 17(1), 45–56.

- Syahputra, R., Lubis, M., & Efendi, S. (2024). Sistem informasi distribusi bantuan sosial berbasis algoritma prioritas untuk meningkatkan ketepatan sasaran. *Jurnal Teknologi dan Sistem Komputer*, 12(2), 88–97.
- Wahyuningsih, S., & Anwar, S. (2022). Analisis kompleksitas algoritma pengurutan data untuk aplikasi berbasis web berskala besar. *Jurnal Ilmu Komputer dan Sistem Informasi*, 10(1), 12–21.
- Widodo, P. P., & Herlawati. (2023). *Rekayasa perangkat lunak berbasis objek: Konsep dan implementasi*. Informatika Bandung.
- Zhang, Y., & Li, X. (2021). Computational optimization techniques for resource allocation problems. *IEEE Access*, 9, 132214–132226.