

Penjadwalan Pekerjaan pada Manajemen Waktu dan Sumber Daya Menggunakan Algoritma *Greedy*

Hayatun Nufus¹, Taufiq²

¹Program Studi Teknik Informatika, Fakultas Teknik, Universitas Malikussaleh

²Program Studi Teknologi Informasi, Fakultas Teknik, Universitas Malikussaleh

Email: hayatun.220170035@mhs.unimal.ac.id, taufiq@unimal.ac.id

ABSTRAK

Penjadwalan pekerjaan adalah proses krusial dalam manajemen waktu dan sumber daya, khususnya dalam konteks manufaktur dan produksi. Artikel ini mengeksplorasi penerapan algoritma greedy untuk penjadwalan pekerjaan dengan tujuan memaksimalkan keuntungan, dengan mempertimbangkan tenggang waktu, tingkat kesulitan, dan durasi pekerjaan. Studi kasus yang digunakan mencakup lima pekerjaan berbeda: membuat kemeja, topi, syal, kaos, dan jeans. Setiap pekerjaan memiliki karakteristik unik seperti tenggang waktu, kesulitan, durasi, dan benefit yang bervariasi. Algoritma greedy diimplementasikan dengan mengutamakan pekerjaan yang memiliki rasio benefit per jam tertinggi. Hasil penelitian menunjukkan bahwa metode ini efektif dalam memberikan solusi penjadwalan yang optimal, dengan total keuntungan mencapai Rp 580.000 dengan pengalokasian jobs. Penelitian ini menegaskan bahwa algoritma greedy dapat memberikan solusi yang efisien dan optimal dalam penjadwalan pekerjaan. Meskipun algoritma greedy tidak selalu menjamin solusi optimal untuk semua kasus, dalam konteks studi kasus ini, hasil yang didapat sangat memuaskan. Diharapkan hasil ini akan menunjukkan peningkatan efisiensi dan keuntungan dalam penjadwalan pekerjaan.

Kata kunci: penjadwalan pekerjaan, algoritma *greedy*, optimasi waktu, keuntungan

Penulis koresponden : Hayatun Nufus

Tanggal terbit : 15 Juni 2024

Tautan : <https://jurnal.komputasi.org/index.php/jst/article/view/28>

1. PENDAHULUAN

Penjadwalan kerja atau tugas merupakan salah satu masalah mendasar di berbagai bidang, termasuk riset operasi, ilmu komputer, dan manajemen. Ini melibatkan pengalokasian pekerjaan ke slot waktu atau sumber daya tertentu untuk mengoptimalkan kriteria tertentu, seperti meminimalkan penundaan, memaksimalkan efisiensi, atau mengurangi biaya. Jadwal adalah salah satu hal yang sangat penting dalam setiap kegiatan, kata penjadwalan berasal dari jadwal yang mempunyai arti pembagian waktu berdasarkan rencana alokasi perintah kerja, daftar pekerjaan, tabel, atau rencana tindakan dengan pembagian waktu yang terperinci[3].

Secara umum, kerja adalah suatu kegiatan atau serangkaian tugas yang dilakukan oleh dilakukan oleh seseorang atau sekelompok orang untuk mencapai suatu tujuan tertentu. Seorang individu atau kelompok manusia untuk mencapai tujuan tertentu. Pekerjaan dapat menjadi fisik dan mental, dan itutan mental, dan biasanya dilakukan biasanya dilakukan mengimbangi kerugian seperti pajak, denda, atau kompensasi lainnya. Untuk mengimbangi kerugian seperti pajak, denda, atau kompensasi lainnya. Selain itu, pekerjaan juga dapat juga melibatkan melibatkan berbagai aktivitas atau tugas senggang yang diselesaikan tanpa memerlukan bantuan keuangan segera. Berbagai aktivitas atau tugas waktu luang yang diselesaikan tanpa memerlukan bantuan keuangan segera[2][4].

Menurut Baker (1974), tujuan penjadwalan adalah untuk meningkatkan produktivitas mesin dengan cara mengurangi waktu mesin menganggur, mengurangi persediaan barang setengah jadi, dan mengurangi jumlah rata-rata pekerjaan yang menunggu dalam antrean karena mesin tersebut sibuk. Selain itu, penjadwalan bertujuan untuk mengurangi keterlambatan penyelesaian pekerjaan. Setiap pekerjaan memiliki batas waktu penyelesaian (due date), dan jika pekerjaan selesai melewati batas waktu tersebut, maka dianggap terlambat. Metode penjadwalan bertujuan untuk mengurangi keterlambatan ini, baik dalam hal waktu maupun frekuensi[5].[5]

Narasimhan (1985) menyatakan bahwa penjadwalan yang efektif seharusnya simpel, mudah dipahami, dan dapat diimplementasikan oleh manajemen dan pengguna lainnya. Aturan penjadwalan harus cukup kuat namun realistik, sehingga fleksibel dalam mengatasi masalah tak terduga dan memungkinkan perencanaan ulang[5].

Salah satu metode untuk mengelola beban kerja yang besar adalah dengan menyusun penjadwalannya[6]. Hal ini dikarenakan tidak semua individu mampu hanya dengan mencatat dan mengetahui seluruh tugas yang ada dan secara intuitif memahami prioritas pengerjaan. Apalagi, menyelesaikan daftar panjang kerja yang harus diselesaikan dapat menimbulkan perasaan terbebani[7].

Dalam beberapa tahun terakhir, pengembangan algoritma penjadwalan yang efisien menjadi semakin penting karena semakin kompleksnya sistem modern dan kebutuhan akan pemanfaatan sumber daya yang optimal[8]. Salah satu pendekatan populer untuk memecahkan masalah penjadwalan adalah dengan menggunakan algoritma greedy, yang membuat pilihan optimal lokal pada setiap langkah dengan harapan menemukan optimal global. Algoritma greedy telah banyak digunakan dalam berbagai aplikasi, termasuk penjadwalan tugas, karena kesederhanaan, efisiensi, dan kemampuannya menangani kumpulan data yang besar. Oleh karena itu, penyusunan jadwal kerja merupakan langkah lebih lanjut yang lebih sistematis dibandingkan hanya mencatat seluruh kerjaan.

Tujuan utama dari penjadwalan adalah untuk mengalokasikan sumber daya secara efisien guna mencapai hasil optimal. Berbagai metode telah dikembangkan untuk menyelesaikan masalah penjadwalan, salah satunya adalah algoritma greedy. Algoritma greedy merupakan metode yang mengutamakan pilihan terbaik pada setiap langkah untuk mencapai solusi optimal secara keseluruhan. Pada artikel ini, kami akan mengaplikasikan algoritma greedy dalam penjadwalan pekerjaan untuk memaksimalkan keuntungan dengan mempertimbangkan tenggat waktu, kesulitan, dan durasi pekerjaan[9][10].

Algoritma greedy atau algoritma serakah adalah salah satu metode yang dapat digunakan untuk menyelesaikan masalah dengan cara langkah demi langkah. Pada setiap langkah, algoritma ini memilih pilihan yang terbaik yang tersedia saat itu (optimum lokal) tanpa mempertimbangkan akibat jangka panjang, dengan harapan pilihan tersebut akan membawa pada solusi terbaik secara keseluruhan (optimum global). Algoritma greedy memiliki beberapa elemen-elemen penting, yaitu:

1. **Himpunan kandidat (c)** : kumpulan pilihan yang dapat diambil dari setiap langkah algoritmanya. Contohnya seperti simpul pada graf, pekerjaan, tugas, koin, benda, karakter, dan lain sebagainya.

2. **Himpunan solusi (S)** : Kumpulan elemen dari himpunan kandidat yang telah dipilih untuk membentuk solusi akhir. Misalnya, kumpulan pekerjaan yang dipilih dalam masalah penjadwalan.

3. **Fungsi solusi** : Fungsi ini digunakan untuk memastikan apakah elemen-elemen yang telah dipilih dari himpunan kandidat sudah membentuk solusi yang valid untuk masalah yang dihadapi.

4. **Fungsi seleksi** : Fungsi ini memilih elemen dari himpunan kandidat pada setiap langkah berdasarkan strategi serakah yang diterapkan.

5. **Fungsi kelayakan** : Fungsi ini memeriksa apakah kandidat yang dipilih oleh fungsi seleksi dapat dimasukkan ke dalam himpunan solusi.

6. **Fungsi objektif** : Fungsi ini bertujuan untuk memaksimalkan atau meminimalkan nilai dari kumpulan solusi yang terbentuk[11].

Sejarah algoritma Greedy dapat ditelusuri kembali ke awal perkembangan teori algoritma. Salah satu contoh awal penerapan pendekatan Greedy adalah dalam algoritma Dijkstra untuk mencari jalur terpendek dalam graf. Algoritma ini diusulkan oleh Edsger W. Dijkstra pada tahun 1956 dan menjadi salah satu fondasi dalam pengembangan algoritma Greedy modern. Sejak saat itu, pendekatan Greedy telah banyak diterapkan dalam berbagai konteks, termasuk optimasi kombinatorial, jaringan, dan kecerdasan buatan.[12]

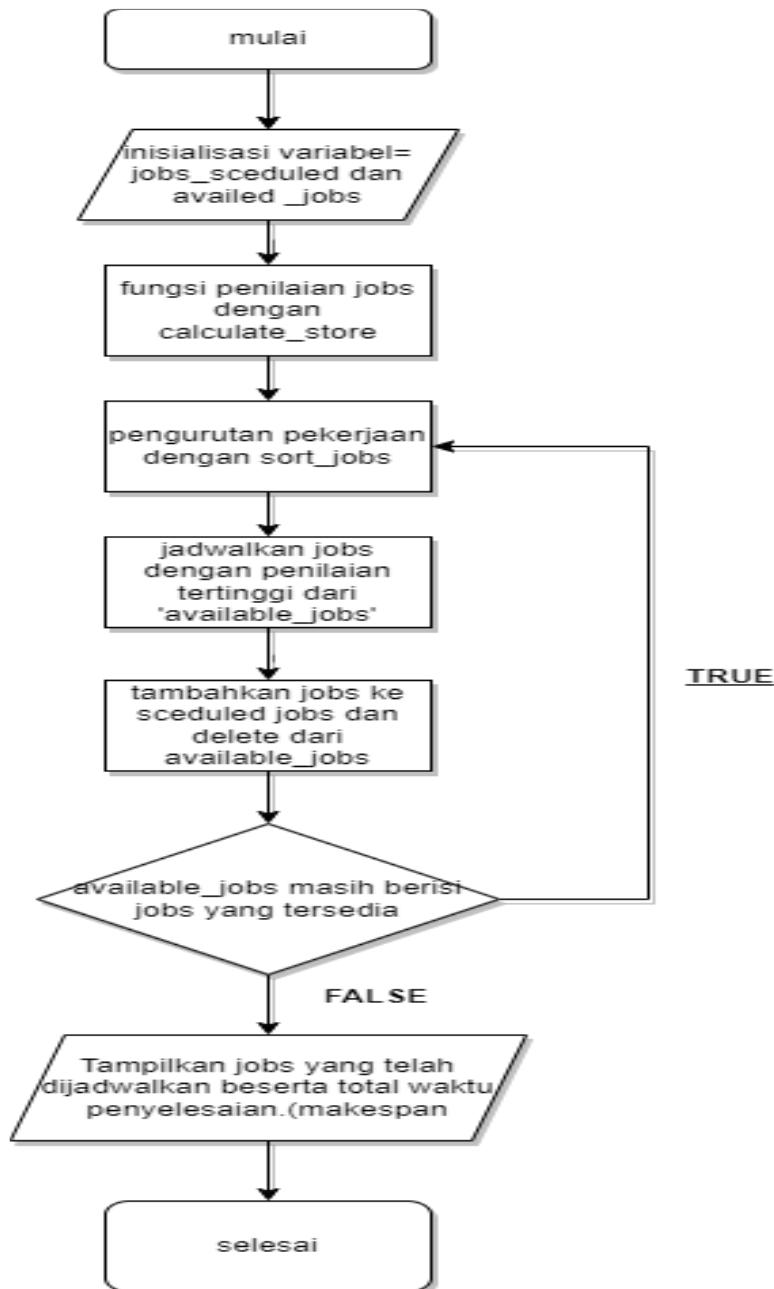
Prinsip dasar dari algoritma Greedy adalah membuat keputusan lokal yang tampak paling baik pada setiap langkah dengan harapan bahwa solusi yang dihasilkan secara keseluruhan juga akan optimal. Langkah-langkah dalam algoritma Greedy tidak mempertimbangkan konsekuensi jangka panjang dari keputusan tersebut, melainkan hanya fokus pada keadaan saat ini[9][13].

Dalam kasus ini, saya mengusulkan algoritma greedy untuk penjadwalan tugas, dengan fokus pada meminimalkan total penundaan dalam suatu sistem. Algoritma yang diusulkan dirancang untuk mengalokasikan kerjaan secara efisien ke sumber daya yang tersedia, dengan mempertimbangkan waktu pemrosesan, ketergantungan, dan prioritasnya. Dengan memanfaatkan pendekatan greedy, algoritma ini bertujuan untuk mengurangi penundaan keseluruhan dalam sistem, sehingga meningkatkan kinerja.

2. METODELOGI

Dalam kasus ini data yang digunakan adalah data yang terdiri dari sejumlah daftar pekerjaan, yang memiliki kaitannya dengan masalah yang akan dipecahkan yaitu jumlah mesin yang akan digunakan untuk operasi, jumlah pekerjaan yang akan dikerjakan, waktu proses pada setiap pekerjaan dan mesin yang ada[15]. Pada masalah ini, terdapat n pekerjaan yang harus diselesaikan. Pekerjaannya hanya dapat diselesaikan pada satu waktu, dan setiap pekerjaan memiliki tenggat waktu (deadline) d , dengan $d \geq 0$. Sebuah pekerjaan i akan memberikan keuntungan sebesar p_i , jika dan hanya jika pekerjaan tersebut diselesaikan tepat waktu. Tujuan dari masalah ini adalah menentukan cara memilih pekerjaan-pekerjaan yang akan dikerjakan oleh mesin sehingga keuntungan yang diperoleh maksimal[11]. Pendekatan dari algoritma greedy untuk menyelesaikan permasalahan ini adalah:

1. **Kumpulan Kandidat:** Semua pekerjaan yang bisa dipilih untuk dikerjakan.
2. **Kumpulan Solusi:** Daftar pekerjaan yang dipilih untuk dikerjakan dengan tujuan mendapatkan keuntungan maksimal.
3. **Fungsi Solusi:** Memastikan apakah masih ada pekerjaan yang bisa dipilih oleh fungsi seleksi. Jika tidak ada lagi, maka solusi sudah ditemukan.
4. **Fungsi Seleksi:** Memilih pekerjaan yang memberikan keuntungan terbesar.
5. **Fungsi Kelayakan:** Memastikan bahwa pekerjaan yang dipilih oleh fungsi seleksi bisa dimasukkan ke dalam kumpulan solusi. Pekerjaan ini tidak boleh memiliki waktu penggerjaan yang bertabrakan dengan pekerjaan lain yang sudah dipilih dan harus dapat diselesaikan sebelum batas waktu yang ditentukan.
6. **Fungsi Objektif:** Menghitung total keuntungan dari kumpulan solusi yang telah dipilih.



Gambar 1.langkah menjadwalkan pekerjaan menggunakan algoritma greedy

```
isialisasi pekerjaan:
jobs = [
    {"Job": "Membuat kemeja", "Deadline": "24-05-2024", "Kesulitan": 4,
     "Durasi": 3, "Benefit": 150000},
    {"Job": "Membuat topi", "Deadline": "20-05-2024", "Kesulitan": 3, "Durasi": 2,
     "Benefit": 80000},
    {"Job": "Membuat syal", "Deadline": "22-05-2024", "Kesulitan": 1, "Durasi": 1,
     "Benefit": 50000},
    {"Job": "Membuat kaos", "Deadline": "21-05-2024", "Kesulitan": 2, "Durasi": 2,
     "Benefit": 100000},
    {"Job": "Membuat jeans", "Deadline": "22-05-2024", "Kesulitan": 5,
     "Durasi": 4, "Benefit": 200000}
]
Function konversi_tanggal(tanggal_str):
    Return datetime.strptime(tanggal_str, "%d-%m-%Y")
Function calculate_score(job):
    deadline = konversi_tanggal(job["Deadline"])
    hari_ini = datetime.now()
    kesulitan = job["Kesulitan"]
    benefit = job["Benefit"]
    waktu_tersisa = (deadline - hari_ini).days
    Return benefit / (kesulitan * waktu_tersisa)
Function select_best_job(available_jobs):
    Return job with max(calculate_score(job)) from available_jobs
Function greedy_job_scheduling(jobs):
    scheduled_jobs = []
    available_jobs = copy of jobs
    While available_jobs is not empty:
        best_job = select_best_job(available_jobs)
        Append best_job to scheduled_jobs
        Remove best_job from available_jobs
    Return scheduled_jobs
Run greedy_job_scheduling(jobs):
    scheduled_jobs = greedy_job_scheduling(jobs)
    Print "Hasil Penjadwalan Pekerjaan:"
    For each job in scheduled_jobs:
        Print job["Job"]
        total_benefit = sum(job["Benefit"] for job in scheduled_jobs)
    Print "Total Manfaat:", total_benefit
```

Gambar 2.Pseudocode algoritma greedy

3. HASIL DAN PEMBAHASAN

Pekerjaan yang ada terdiri dari 5 jenis pekerjaan yang berbeda yang memiliki komponen nya masing-masing. Data pekerjaan yang diperoleh dapat dilihat pada tabel berikut:

Tabel 1.Daftar pekerjaan

No	Jobs	Deadline	Kesulitan	Durasi(jam)	Benefit
1	Membuat kemeja	24-05-2024	4	3	Rp.150.000
2	Membuat topi	20-05-2024	3	2	Rp.80.000

3	Membuat syal	22-05-2024	1	1	Rp.50.000
4	Membuat kaos	21-05-2024	2	2	Rp.100.000
5	Membuat jeans	22-05-2024	5	4	Rp.200.000

Dari tabel diatas terlihat bahwa beberapa jobs yang memiliki deadline yang berdekatan dan tingkat kesulitan dan durasi serta benefit yang berbeda-beda. Berikut adalah langkah-langkah dalam penjadwalan menggunakan algoritma greedy:

1. Kumpulan Kandidat
 - Membuat kemeja (Deadline: 24-05-2024, Durasi: 3 jam, Benefit: Rp. 150.000)
 - Membuat topi (Deadline: 20-05-2024, Durasi: 2 jam, Benefit: Rp. 80.000)
 - Membuat syal (Deadline: 22-05-2024, Durasi: 1 jam, Benefit: Rp. 50.000)
 - Membuat kaos (Deadline: 21-05-2024, Durasi: 2 jam, Benefit: Rp. 100.000)
 - Membuat jeans (Deadline: 22-05-2024, Durasi: 4 jam, Benefit: Rp. 200.000)
2. Kumpulan Solusi: Kosong pada awalnya.
3. Fungsi Solusi: Masih ada pekerjaan yang bisa dipilih.
4. Fungsi Seleksi:
 - Memilih pekerjaan dengan keuntungan terbesar.
 - Urutkan pekerjaan berdasarkan benefit:
 - Membuat jeans - Rp. 200.000
 - Membuat kemeja - Rp. 150.000
 - Membuat kaos - Rp. 100.000
 - Membuat topi - Rp. 80.000
 - Membuat syal - Rp. 50.000
5. Fungsi Kelayakan: Pastikan pekerjaan yang dipilih bisa dimasukkan ke dalam kumpulan solusi tanpa bertabrakan dan dapat diselesaikan sebelum deadline.
6. Fungsi Objektif: Menghitung total keuntungan dari kumpulan solusi yang telah dipilih.
 - Proses Seleksi dan Penjadwalan:
 - Pilih pekerjaan "Membuat jeans" (Benefit: Rp. 200.000, Durasi: 4 jam):
 - Deadline: 22-05-2024
 - Jadwal: 18-22 Mei 2024
 - Kelayakan: Bisa dijadwalkan.
 - Pilih pekerjaan "Membuat kemeja" (Benefit: Rp. 150.000, Durasi: 3 jam):
 - Deadline: 24-05-2024
 - Jadwal: 23-24 Mei 2024
 - Kelayakan: Bisa dijadwalkan.
 - Pilih pekerjaan "Membuat kaos" (Benefit: Rp. 100.000, Durasi: 2 jam):
 - Deadline: 21-05-2024
 - Jadwal: 16-17 Mei 2024
 - Kelayakan: Bisa dijadwalkan.
 - Pilih pekerjaan "Membuat topi" (Benefit: Rp. 80.000, Durasi: 2 jam):
 - Deadline: 20-05-2024
 - Jadwal: 14-15 Mei 2024
 - Kelayakan: Bisa dijadwalkan.
 - Pilih pekerjaan "Membuat syal" (Benefit: Rp. 50.000, Durasi: 1 jam):
 - Deadline: 22-05-2024
 - Jadwal: 13 Mei 2024
 - Kelayakan: Bisa dijadwalkan.
 - Jadwal Akhir:
 - Membuat syal: 13 Mei 2024
 - Membuat topi: 14-15 Mei 2024
 - Membuat kaos: 16-17 Mei 2024
 - Membuat jeans: 18-22 Mei 2024
 - Membuat kemeja: 23-24 Mei 2024
 - Total Keuntungan: Rp. 200.000 (jeans) + Rp. 150.000 (kemeja) + Rp. 100.000 (kaos) + Rp. 80.000 (topi) + Rp. 50.000 (syal) = Rp. 580.000

Gambar 3.Tabel hasil penjadwalan pekerjaan

No	Jobs	Deadline	Kesulitan	Durasi(jam)	Benefit	Waktu pengerjaan
3	Membuat syal	22 Mei 2024	1	1	Rp.50.000	13 Mei 2024
2	Membuat topi	20 Mei 2024	3	2	Rp.80.000	14-15 Mei 2024
4	Membuat kaos	21 Mei 2024	2	2	Rp.100.000	16-17 Mei 2024
5	Membuat jeans	22 Mei 2024	5	4	Rp.200.000	18-22 Mei 2024
1	Membuat kemeja	24 Mei 2024	4	3	Rp.150.000	23-24 Mei 2024
total						

Hasil dari pada kasus ini menunjukkan bahwa algoritma greedy dapat digunakan untuk menyelesaikan masalah penjadwalan pekerjaan dengan hasil yang optimal. Dalam contoh yang digunakan algoritma greedy dapat menyelesaikan penjadwalan dengan memberikan benefit mencapai Rp 580.000 dengan pengalokasian waktu pengerjaan jobs.

4. KESIMPULAN

Dalam menyelesaikan masalah penjadwalan pekerjaan menggunakan algoritma greedy, hal yang dapat memaksimalkan keuntungan dengan mempertimbangkan berbagai faktor seperti deadline, durasi, kesulitan, dan benefit dari setiap pekerjaan. Langkah pertama adalah menghitung rasio benefit per jam untuk setiap pekerjaan dan mengurutkan pekerjaan berdasarkan rasio ini serta deadline. Ini memungkinkan kita untuk memilih pekerjaan dengan keuntungan terbesar yang bisa diselesaikan dalam waktu yang tersedia.

Setelah mengurutkan pekerjaan, kita menjadwalkan pekerjaan secara berurutan, memastikan bahwa setiap pekerjaan diselesaikan tepat waktu tanpa melanggar deadline. Pekerjaan yang tidak bisa diselesaikan dalam waktu yang tersedia berdasarkan prioritas rasio benefit per jam akan diabaikan untuk mencapai optimasi global. Hasil dari pendekatan ini adalah serangkaian pekerjaan yang memberikan total benefit maksimal dalam kerangka waktu yang ada.

Dengan menggunakan algoritma greedy, jadwal optimal yang diperoleh menunjukkan bahwa pekerjaan dengan rasio benefit per jam tertinggi yang dapat diselesaikan tepat waktu memberikan total benefit yang signifikan. Solusi ini menunjukkan efisiensi algoritma greedy dalam menangani masalah penjadwalan pekerjaan, menghasilkan keuntungan total yang lebih besar dibandingkan dengan metode penjadwalan yang kurang terstruktur.

Penerapan algoritma greedy dalam penjadwalan pekerjaan menunjukkan bahwa metode ini efektif dalam memaksimalkan keuntungan dengan mempertimbangkan berbagai faktor seperti tenggat waktu, kesulitan, dan durasi pekerjaan. Meskipun algoritma greedy tidak selalu menjamin solusi optimal untuk semua kasus, dalam konteks studi kasus ini, hasil yang didapat sangat memuaskan dan efisien. Penelitian lebih lanjut dapat dilakukan untuk membandingkan algoritma greedy dengan metode penjadwalan lainnya guna mengevaluasi performa dan efektivitasnya dalam berbagai situasi.

5. DAFTAR PUSTAKA

- [1] P. B. Steel, "Sudi dan Implementasi Algoritma Optimasi," *Order A J. Theory Ordered Sets Its Appl.*
- [2] Patel, "Yayasan Lembaga Pendidikan Islam Riau Universitas Islam Riau Fakultas Ilmu Sosial Dan Ilmu Politik," pp. 9–25, 2019.
- [3] H. Ardiansyah and M. B. S. Junianto, "Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran," *J. Media Inform. Budidarma*, vol. 6, no. 1, p. 329, 2022, doi: 10.30865/mib.v6i1.3418.
- [4] R. Febrian Ginting, A. Situmorang, S. Manurung, and H. Artikel, "Sistem Informasi Gografis Pencarian Lokasi Sekolah Terdekat Pada Kota Medan Dengan Algoritma Greedy Berbasis Web," *Methotika J. Ilm. Tek. Inform.*, vol. 2, no. 2, pp. 85–91, 2022, [Online]. Available: <http://ojs.fikom-methodist.net/index.php/methotika>
- [5] H. Irwan, "Optimasi Penjadwalan Job Shop dengan Metode Algoritma Greedy," *PROFISIENSI J. Progr. Stud. Tek. Ind.*, vol. 8, no. 2, pp. 164–176, 2020, doi: 10.33373/profis.v8i2.2810.
- [6] S. Ginting, RF, Situmorang, A., & Manurung, "Sistem Informasi Gografis Pencarian Lokasi Sekolah Terdekat Pada Kota

- Medan Dengan Algoritma Greedy Berbasis Web.,” *J. Ilm. Tek. Inform.*, 2022.
- [7] S. C. Abdurasyid, “Penerapan Algoritma Greedy dalam Membantu Penjadwalan Pengerjaan Tugas,” 2021.
- [8] D. Audit and E. R. P. Systems, *Audit sistem informasi*.
- [9] R. Fikria and R. P. Sari, “Sistem Penjadwalan Piket Pegawai di Dinas Pencegahan dan Penyelamatan Kota Medan,” *Innov. J. Soc. Sci. Res.*, vol. 3, no. 6, pp. 5744–5757, 2023.
- [10] W. M. Hidayatullah, “Solusi Capacitated Vehicle Routing Problem with Time Windows Sistem Distribusi LNG Papua Dengan Algoritma Ant Colony Optimization,” *Tesis*, 2019.
- [11] F. A. W. Dharmasatya, “Implementasi Algoritma Greedy Dalam Menjadwalkan Pengerjaan Tugas,” 2021.
- [12] D. K. Di Indonesia, “Abstrak,” vol. 2, no. 2, 2024.
- [13] dan devi budianti santi, indah hartami, “penerapan algoritma greedy dalam mencari rute terdekat lokasi spbu berbasis web,” *J. Penelit. multidisiplin ilmu* 2, pp. 1225–1234, 2023.
- [14] V. Syifana, A., Meylani, L., & Prabowo, “(2021). Alokasi Sumber Daya Radio Pada Komunikasi Underlay Perangkat Ke Perangkat Menggunakan Algoritma Lelang Dua Bertahap Berbasis Alokasi Sumber Daya Yang Adil Dan Interferensi. eProsiding Teknik , 8 (5).” 2021.
- [15] R. Rachman, “Penjadwalan Produksi Garment Menggunakan Algoritma Heuristic Pour,” *J. Inform.*, vol. 5, no. 1, pp. 81–89, 2018, doi: 10.31311/ji.v5i1.2743.