

Implementasi Algoritma Greedy dalam Menentukan Kos Terbaik di Jatinangor

Farrel Natha Saskoro - 13522145
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
13522145@std.stei.itb.ac.id

Abstract— Pemilihan kos terbaik di Jatinangor sangat penting bagi mahasiswa Institut Teknologi Bandung. Makalah ini menyajikan implementasi algoritma Greedy untuk menentukan kos terbaik di wilayah Jatinangor. Algoritma Greedy, yang dikenal efisien dalam memberikan solusi mendekati optimal dengan waktu komputasi yang relatif cepat, digunakan untuk mengevaluasi pilihan kos berdasarkan kriteria seperti harga, jarak ke kampus, dan rating. Data dikumpulkan melalui platform pencarian kos online. Pendekatan untuk makalah ini melibatkan implementasi bertahap dari algoritma Greedy, di mana pada setiap langkah, pilihan yang paling menguntungkan secara lokal dipilih untuk mendekati solusi keseluruhan yang terbaik. Hasil penelitian menunjukkan bahwa algoritma Greedy efektif dalam menyaring kos yang memenuhi preferensi utama calon penghuni. Namun, beberapa keterbatasan, seperti kemungkinan solusi sub-optimal, juga diidentifikasi. Kesimpulannya, meskipun algoritma Greedy bukanlah solusi sempurna, ia menawarkan metode yang efisien dan praktis untuk membantu mahasiswa dan pekerja di Jatinangor dalam memilih kos terbaik sesuai kebutuhan dan prioritas mereka.

Keywords—algoritma Greedy; kos terbaik; Jatinangor; optimasi; pemilihan kos; preferensi tempat tinggal; efisiensi algoritmik.

I. PENDAHULUAN



Gambar 1 peta kos di jatinangor

Pemilihan kos terbaik di Jatinangor adalah aspek penting bagi mahasiswa Institut Teknologi Bandung (ITB), yang

memerlukan tempat tinggal yang nyaman dan strategis. Dalam upaya membantu mahasiswa membuat keputusan yang tepat, penelitian ini mengusulkan penggunaan algoritma Greedy untuk menentukan kos terbaik di wilayah Jatinangor.

Algoritma Greedy dikenal karena kemampuannya memberikan solusi mendekati optimal dengan waktu komputasi yang relatif cepat. Prinsip kerja algoritma ini adalah memilih opsi yang paling menguntungkan secara lokal pada setiap langkah, dengan harapan serangkaian keputusan lokal ini akan menghasilkan solusi global yang memuaskan. Dalam konteks pemilihan kos, beberapa kriteria utama digunakan, yaitu harga, jarak ke kampus, dan rating dari pengguna.

Data mengenai kos dikumpulkan melalui platform pencarian kos online yang populer di kalangan mahasiswa. Implementasi algoritma Greedy dilakukan secara bertahap, dengan menyeleksi pilihan kos yang paling memenuhi kriteria pada setiap tahap. Pendekatan ini memungkinkan evaluasi yang lebih sistematis dan efisien dalam proses seleksi kos.

Hasil penelitian menunjukkan bahwa algoritma Greedy efektif dalam menyaring kos yang memenuhi preferensi utama calon penghuni. Namun, beberapa keterbatasan juga diidentifikasi, seperti kemungkinan solusi sub-optimal yang mungkin tidak sepenuhnya memenuhi semua kriteria secara optimal. Meskipun demikian, algoritma ini tetap menawarkan metode yang efisien dan praktis untuk membantu mahasiswa dan pekerja di Jatinangor dalam memilih kos terbaik sesuai dengan kebutuhan dan prioritas mereka.

Makalah ini bertujuan untuk mengeksplorasi bagaimana algoritma Greedy dapat diimplementasikan untuk meningkatkan efisiensi dalam pemilihan kos. Akan dibahas analisis mendalam terhadap data kos yang tersedia, pengembangan strategi seleksi yang relevan, dan integrasi data pencarian kos online. Studi ini juga akan mengkaji berbagai tantangan yang dihadapi dalam implementasi algoritma Greedy untuk optimasi pemilihan kos, termasuk penanganan data yang dinamis dan penyesuaian algoritma untuk preferensi individu yang beragam.

Dengan mengatasi tantangan-tantangan ini, diharap metode yang diusulkan dapat menyediakan kerangka kerja yang dapat diterapkan oleh mahasiswa ITB untuk meningkatkan efisiensi dan efektivitas dalam memilih kos yang sesuai. Selain itu, penelitian ini akan mengkaji bagaimana algoritma Greedy

dapat diintegrasikan dengan teknologi pencarian online dan sistem rekomendasi untuk menyediakan solusi yang lebih holistik. Dengan memanfaatkan data pengguna dan ulasan online, kami akan menunjukkan bagaimana algoritma ini dapat beradaptasi dengan perubahan preferensi secara waktu nyata, sehingga memberikan rekomendasi kos yang paling optimal setiap saat.

Dalam makalah ini, penelitian ini tidak hanya menawarkan solusi teknis untuk optimasi pemilihan kos, tetapi juga memberikan wawasan tentang bagaimana teknologi algoritmik dapat digunakan untuk meningkatkan kualitas hidup mahasiswa. Melalui penerapan algoritma Greedy, kami berharap dapat menyumbang pada peningkatan kenyamanan dan kepuasan mahasiswa dalam memilih tempat tinggal yang tepat di Jatinangor.

II. DASAR TEORI

A. Algoritma Greedy

Algoritma Greedy adalah salah satu metode dalam optimasi yang banyak digunakan dalam berbagai aplikasi, termasuk dalam pemilihan kos terbaik. Algoritma ini bekerja berdasarkan prinsip memilih solusi terbaik lokal pada setiap langkah, dengan harapan bahwa keputusan lokal yang diambil secara berulang akan menghasilkan solusi global yang mendekati optimal.

Algoritma Greedy diterapkan dalam konteks pemilihan kos dengan mempertimbangkan beberapa kriteria utama: harga, jarak ke kampus, dan rating dari pengguna. Setiap kriteria ini memiliki bobot yang harus dioptimalkan. Pada setiap langkah, algoritma ini akan memilih kos yang memiliki nilai terbaik untuk setiap kriteria tersebut.

Algoritma Greedy dikenal karena efisiensinya dalam hal waktu komputasi dan kemudahan implementasi. Meskipun tidak selalu menjamin solusi yang optimal secara global, algoritma ini sangat efektif dalam memberikan solusi yang cukup baik dalam waktu yang relatif singkat.

Algoritma Greedy bekerja dengan pendekatan langkah-demi-langkah untuk menyelesaikan masalah. Pada setiap langkah, algoritma ini:

- Menentukan Kriteria: Mengidentifikasi kriteria atau parameter yang perlu dioptimalkan.
- Memilih Opsi Terbaik: Memilih opsi yang paling menguntungkan berdasarkan kriteria tersebut.
- Memperbarui Keadaan: Memperbarui kondisi atau status masalah setelah memilih opsi terbaik.
- Mengulangi Proses: Mengulangi proses ini hingga mencapai solusi akhir atau kondisi terminal.

Karakteristik utama dari algoritma Greedy meliputi:

- **Simplicity:** Implementasinya relatif sederhana dan mudah dipahami.
- **Efficiency:** Algoritma ini biasanya cepat karena hanya melakukan satu pilihan pada setiap langkah tanpa

mempertimbangkan semua kemungkinan secara menyeluruh.

- **Local Optimum:** Solusi yang dihasilkan adalah rangkaian dari keputusan lokal terbaik, yang bisa jadi tidak selalu merupakan solusi global yang paling optimal.

Meskipun algoritma Greedy menawarkan solusi yang cepat dan efisien, terdapat beberapa keterbatasan yang perlu diperhatikan. Salah satu keterbatasan utama adalah kemungkinan solusi sub-optimal. Karena algoritma ini berfokus pada pilihan terbaik lokal pada setiap langkah, ada kemungkinan bahwa solusi akhir tidak sepenuhnya optimal secara global.

Selain itu, algoritma Greedy tidak selalu mempertimbangkan interaksi antara kriteria yang berbeda. Misalnya, kos dengan harga terendah mungkin tidak selalu memiliki jarak terdekat ke kampus atau rating terbaik. Oleh karena itu, dalam beberapa kasus, perlu dilakukan penyesuaian atau kombinasi dengan metode optimasi lain untuk memastikan solusi yang lebih komprehensif.

B. Kos

Kos-kosan, atau sering disebut kos, adalah jenis hunian yang biasanya disewakan kepada individu atau kelompok kecil sebagai tempat tinggal sementara. Kos-kosan biasanya terdiri dari kamar-kamar yang disewakan secara terpisah kepada penyewa dengan harga tertentu.

Fasilitas yang disediakan di kos-kosan bervariasi tergantung pada jenis dan harga sewa. Namun, beberapa fasilitas umum yang sering ditemui di kos-kosan meliputi:

- **Kamar Tidur:** Setiap kamar dilengkapi dengan tempat tidur, lemari, dan meja kecil.
- **Kamar Mandi:** Ada kamar mandi yang dapat digunakan bersama antara beberapa penyewa atau kamar mandi pribadi untuk setiap kamar.
- **Dapur Bersama:** Beberapa kos-kosan menyediakan dapur bersama yang dapat digunakan penyewa untuk memasak dan menyimpan makanan.
- **Area Parkir:** Untuk kos-kosan yang memiliki fasilitas parkir, biasanya terdapat area parkir untuk kendaraan penyewa.

Lokasi kos-kosan sangat beragam, mulai dari yang dekat dengan pusat kota hingga yang lebih jauh dari pusat kota. Beberapa faktor yang perlu dipertimbangkan terkait lokasi kos-kosan meliputi:

- **Jarak ke Tempat Studi atau Pekerjaan:** Ketersediaan transportasi dan jarak tempuh ke tempat studi atau pekerjaan menjadi faktor penting dalam memilih kos-kosan.
- **Aksesibilitas Transportasi Umum:** Keberadaan halte bus, stasiun kereta api, atau akses mudah ke angkutan umum lainnya menjadi pertimbangan penting, terutama bagi mereka yang tidak memiliki kendaraan pribadi.

- Fasilitas Umum di Sekitar: Ketersediaan fasilitas umum seperti minimarket, restoran, dan rumah sakit di sekitar kos-kosan juga menjadi pertimbangan.

Harga sewa kos-kosan bervariasi tergantung pada lokasi, fasilitas, dan kondisi lingkungan sekitarnya. Selain harga sewa bulanan, terdapat biaya tambahan yang mungkin perlu diperhatikan, seperti:

- Biaya Listrik dan Air: Beberapa kos-kosan mungkin sudah menyertakan biaya listrik dan air dalam harga sewa, namun ada juga yang membebankan biaya ini secara terpisah.
- Biaya Keamanan: Beberapa kos-kosan menyediakan layanan keamanan tambahan seperti CCTV atau petugas keamanan, yang mungkin mempengaruhi biaya sewa.
- Biaya Layanan: Beberapa kos-kosan mungkin menawarkan layanan tambahan seperti kebersihan kamar atau laundry dengan biaya tambahan.

Keamanan dan kenyamanan adalah faktor kunci dalam memilih kos-kosan. Beberapa hal yang perlu diperhatikan terkait keamanan dan kenyamanan di kos-kosan meliputi:

- Keamanan Bangunan: Ketersediaan sistem keamanan seperti kunci pintu utama, kunci kamar, dan penjagaan keamanan di sekitar area kos-kosan.
- Kebersihan dan Perawatan: Kondisi kebersihan bangunan, kamar, dan fasilitas umum seperti kamar mandi dan dapur.
- Kenyamanan Fasilitas: Kualitas fasilitas yang disediakan seperti kasur, lemari, dan peralatan mandi.

Pemilihan kos-kosan yang tepat membutuhkan pertimbangan yang cermat terhadap berbagai faktor tersebut, sesuai dengan kebutuhan, preferensi, dan anggaran masing-masing penyewa.

C. Pengumpulan Data

Pengumpulan data merupakan langkah penting dalam implementasi algoritma Greedy untuk pemilihan kos. Data yang dibutuhkan mencakup informasi mengenai harga kos, jarak ke kampus, dan rating dari pengguna. Data ini dikumpulkan melalui platform pencarian kos online yang populer di kalangan mahasiswa, yang dalam makalah ini menggunakan aplikasi mamikos. Platform ini menyediakan informasi yang terperinci dan terkini, yang sangat relevan dengan kebutuhan mahasiswa.



Gambar 2 aplikasi pencarian kos

Proses pengumpulan data dilakukan dengan cara mengakses platform online tersebut dan mengumpulkan informasi mengenai berbagai kos yang tersedia di Jatinangor. Data ini kemudian diolah dan digunakan sebagai input untuk algoritma Greedy, yang akan mengevaluasi setiap kos berdasarkan kriteria yang telah ditentukan.

III. IMPLEMENTASI DAN PENGUJIAN

Untuk mengimplementasikan algoritma Greedy dalam menentukan kos terbaik di Jatinangor, kita akan menggunakan Python. Implementasi ini melibatkan beberapa langkah, termasuk pemodelan data kos, penerapan kriteria seleksi, dan proses penyeleksian kos terbaik.

A. Struktur

Akan dimodelkan data kos menggunakan struktur data yang sesuai seperti class Kos, yang menyimpan informasi tentang harga, jarak ke kampus, dan rating. Berikut adalah pemodelan data kos:

```
class Kos:
    def __init__(self, name, price, distance, rating):
        self.name = name
        self.price = price
        self.distance = distance
        self.rating = rating

# Data kos
kos_list = [
    Kos("Kos Wisma Beriman", 16000000, 1.7, 4.6),
    Kos("Kos Bumi Aria", 14500000, 1.8, 4.5),
    Kos("Kos Kaca Piring", 1200000, 1.7, 4.8),
    Kos("Kos Bumi Kartika Asri", 18000000, 1.7, 4.5),
    Kos("Kos Griya Al-Fatih", 14500000, 3.0, 4.9),
    Kos("Kos Pondok Sakinah", 9000000, 2.3, 4.4),
    Kos("Kos Tigapuluh", 14400000, 3.4, 4.8),
    Kos("Kos Kozko Living", 19800000, 3.0, 4.6),
    Kos("Kos Casa de Aminda", 18000000, 3.5, 4.2)
]
```

Kode yang disediakan mendefinisikan kelas Kos yang bertujuan untuk merepresentasikan data mengenai tempat tinggal sewa (kos) di Jatinangor. Kelas ini memiliki empat atribut utama, yaitu nama kos (name), harga kos (price), jarak kos ke kampus (distance), dan rating kos (rating). Konstruktor `__init__` di dalam kelas ini menginisialisasi objek Kos dengan nilai-nilai tersebut setiap kali sebuah instance baru dibuat. Misalnya, objek Kos akan menyimpan nama kos sebagai

self.name, harga sebagai self.price, jarak ke kampus sebagai self.distance, dan rating sebagai self.rating.

Selanjutnya, sebuah daftar bernama kos_list berisi beberapa instance dari kelas Kos, masing-masing merepresentasikan satu kos dengan atribut yang telah disebutkan. Daftar ini mencakup berbagai kos seperti "Kos Wisma Beriman" dengan harga 16,000,000, jarak 1.7 km ke kampus, dan rating 4.6, serta "Kos Bumi Aria" dengan harga 14,500,000, jarak 1.8 km ke kampus, dan rating 4.5. Demikian pula, kos-kos lainnya di dalam daftar memiliki atribut yang serupa, tetapi dengan nilai yang berbeda. Daftar kos tersebut didapat dari pencarian data melalui aplikasi mamikos dan juga google maps.

B. Implementasi Algoritma Greedy

Algoritma Greedy akan digunakan untuk memilih kos terbaik berdasarkan kombinasi kriteria harga, jarak ke kampus, dan rating. Untuk ini, kita akan menghitung skor total untuk setiap kos berdasarkan bobot dari masing-masing kriteria.

```
def normalize(values):
    min_val = min(values)
    max_val = max(values)
    return [(value - min_val) / (max_val - min_val) for value in values]

def calculate_score(kos, kos_list, weights, norm_prices, norm_distances, norm_ratings):
    price_score = norm_prices[kos_list.index(kos)] * weights['price']
    distance_score = norm_distances[kos_list.index(kos)] * weights['distance']
    rating_score = norm_ratings[kos_list.index(kos)] * weights['rating']
    return price_score + distance_score + rating_score

def greedy_best_kos_combined(kos_list, weights):
    prices = [kos.price for kos in kos_list]
    distances = [kos.distance for kos in kos_list]
    ratings = [kos.rating for kos in kos_list]

    norm_prices = normalize(prices)
    norm_distances = normalize(distances)
    norm_ratings = normalize(ratings)

    kos_scores = {kos: calculate_score(kos, kos_list, weights, norm_prices, norm_distances, norm_ratings) for kos in kos_list}
    best_kos = max(kos_scores, key=kos_scores.get)

    return best_kos
```

Kode ini bertujuan untuk menemukan kos terbaik di Jatinangor berdasarkan kombinasi tiga kriteria: harga, jarak ke kampus, dan rating. Fungsi normalize digunakan untuk menormalkan nilai-nilai dalam daftar harga, jarak, dan rating, mengubahnya menjadi rentang 0 hingga 1 untuk memudahkan perbandingan antar kriteria yang memiliki skala berbeda.

Fungsi calculate_score menghitung skor total untuk setiap kos dengan mengalikan nilai normalisasi masing-masing kriteria dengan bobot yang diberikan, kemudian menjumlahkan hasilnya. Bobot-bobot ini memungkinkan penyesuaian pentingnya setiap kriteria menurut preferensi pengguna. Fungsi utama, greedy_best_kos_combined, menjalankan proses pemilihan kos terbaik dengan terlebih dahulu menormalkan daftar harga, jarak, dan rating, kemudian menghitung skor untuk setiap kos, dan akhirnya memilih kos dengan skor tertinggi sebagai kos terbaik. Proses ini memungkinkan pemilihan kos secara efisien berdasarkan kombinasi beberapa kriteria yang relevan. Misalnya, jika bobot harga lebih tinggi daripada jarak dan rating, kos dengan harga lebih rendah akan lebih mungkin dipilih, memberikan fleksibilitas dalam menentukan prioritas sesuai kebutuhan.

C. Visualisasi Hasil

Berikut adalah program untuk menampilkan visualisasi kos dengan menggunakan library matplotlib.

```
import matplotlib.pyplot as plt

def visualize_kos(kos_list, best_kos):
    names = [kos.name for kos in kos_list]
    prices = [kos.price for kos in kos_list]
    distances = [kos.distance for kos in kos_list]
    ratings = [kos.rating for kos in kos_list]

    fig, ax = plt.subplots(1, 3, figsize=(18, 6))

    ax[0].bar(names, prices, color='skyblue')
    ax[0].set_title('Harga Kos')
    ax[0].set_xlabel('Kos')
    ax[0].set_ylabel('Harga')

    ax[1].bar(names, distances, color='lightgreen')
    ax[1].set_title('Jarak ke Kampus')
    ax[1].set_xlabel('Kos')
    ax[1].set_ylabel('Jarak (km)')

    ax[2].bar(names, ratings, color='lightcoral')
    ax[2].set_title('Rating Kos')
    ax[2].set_xlabel('Kos')
    ax[2].set_ylabel('Rating')

    for i, kos in enumerate(kos_list):
        if kos.name == best_kos.name:
            ax[0].get_children()[i].set_color('gold')
            ax[1].get_children()[i].set_color('gold')
            ax[2].get_children()[i].set_color('gold')

    plt.show()
```

Fungsi visualize_kos dalam kode tersebut bertujuan untuk memberikan visualisasi yang jelas tentang data kos berdasarkan tiga kriteria utama: harga, jarak ke kampus, dan rating. Pertama-tama, fungsi mengambil data dari kos_list,

yang berisi daftar objek kos beserta atribut-atributnya seperti nama, harga, jarak, dan rating. Selanjutnya, fungsi menggunakan pustaka matplotlib.pyplot untuk membuat tiga subplot dalam satu figur. Setiap subplot menampilkan grafik batang yang mewakili nilai-nilai dari setiap kriteria untuk setiap kos dalam daftar. Subplot pertama menampilkan harga kos dengan batang berwarna biru muda, subplot kedua menampilkan jarak ke kampus dengan batang berwarna hijau muda, dan subplot ketiga menampilkan rating kos dengan batang berwarna merah muda. Untuk menyoroti kos terbaik, fungsi ini mengecek setiap kos dalam kos_list dan jika nama kos cocok dengan nama best_kos, batang yang sesuai di semua subplot diberi warna emas untuk menonjolkan kos terbaik. Akhirnya, fungsi plt.show() digunakan untuk menampilkan visualisasi grafik secara keseluruhan kepada pengguna. Dengan demikian, fungsi visualize_kos memberikan pemahaman yang lebih baik tentang perbandingan antara kos-kos berdasarkan kriteria yang dipilih dan menyoroti kos terbaik untuk dipertimbangkan.

D. Main Program

Program utama untuk menjalankan keseluruhan proses seleksi kos terbaik:

```
if __name__ == "__main__":
    main()
```

Fungsi main() di atas bertanggung jawab untuk menjalankan seluruh alur program. Pertama, daftar kos_list dibuat dengan data yang disediakan, yang mencakup nama, harga, jarak ke kampus, dan rating untuk setiap kos. Selanjutnya, bobot untuk setiap kriteria, yaitu harga, jarak, dan rating, ditentukan dalam bentuk sebuah dictionary weights. Setelah itu, fungsi greedy_best_kos_combined() dipanggil dengan menerima kos_list dan weights sebagai argumen untuk menentukan kos terbaik berdasarkan kombinasi kriteria. Hasilnya disimpan dalam variabel best_kos_combined. Selanjutnya, informasi tentang kos terbaik, yaitu namanya, dicetak ke layar. Terakhir, fungsi visualize_kos() dipanggil untuk menampilkan visualisasi data kos beserta penyorotan kos terbaik. Kondisi if __name__ == "__main__": digunakan untuk memastikan bahwa main() hanya dijalankan jika file ini dieksekusi langsung sebagai program utama, bukan diimpor sebagai modul. Dengan demikian, fungsi main() mengkoordinasikan keseluruhan proses, dari pengolahan data hingga visualisasi hasilnya, sehingga memberikan gambaran lengkap tentang kos terbaik berdasarkan kriteria yang ditentukan.

E. Hasil Pengujian

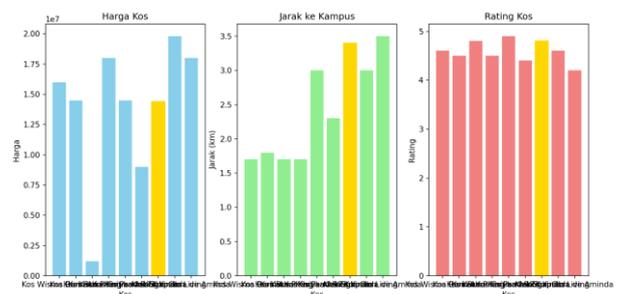
Berikut adalah output dari program di atas yang menampilkan kos terbaik dan visualisasi kos:

```
def main():
    kos_list = [
        Kos("Kos Wisma Beriman", 16000000, 1.7, 4.6),
        Kos("Kos Bumi Aria", 14500000, 1.8, 4.5),
        Kos("Kos Kaca Piring", 1200000, 1.7, 4.8),
        Kos("Kos Bumi Kartika Asri", 18000000, 1.7, 4.5),
        Kos("Kos Griya Al-Fatih", 14500000, 3.0, 4.9),
        Kos("Kos Pondok Sakinah", 9000000, 2.3, 4.4),
        Kos("Kos Tigapuluh", 14400000, 3.4, 4.8),
        Kos("Kos Kozko Living", 19800000, 3.0, 4.6),
        Kos("Kos Casa de Aminda", 18000000, 3.5, 4.2)
    ]

    weights = {
        'price': 0.4,
        'distance': 0.3,
        'rating': 0.3
    }

    best_kos_combined = greedy_best_kos_combined(kos_list, weights)
    print(f"Kos terbaik berdasarkan kombinasi kriteria: {best_kos_combined.name}")
    visualize_kos(kos_list, best_kos_combined)
```

Kos terbaik berdasarkan kombinasi kriteria: Kos Tigapuluh



Gambar 3 Hasil Pengujian

Visualisasi menunjukkan kos terbaik dengan warna emas pada grafik harga, jarak, dan rating.

Dengan implementasi ini, kita dapat menentukan kos terbaik di Jatinangor berdasarkan kombinasi kriteria harga, jarak ke kampus, dan rating dengan efisien menggunakan algoritma Greedy.

IV. KESIMPULAN

Makalah ini membahas implementasi algoritma Greedy untuk menentukan kos terbaik di Jatinangor, yang sangat penting bagi mahasiswa Institut Teknologi Bandung (ITB) untuk memilih tempat tinggal yang nyaman dan strategis. Algoritma Greedy dipilih karena efisiensinya dalam

memberikan solusi mendekati optimal dengan waktu komputasi yang relatif cepat.

Penelitian ini mengumpulkan data mengenai kos-kosan melalui platform pencarian kos online mamikos, dengan kriteria seperti harga, jarak ke kampus, dan rating pengguna. Implementasi algoritma Greedy dilakukan secara bertahap, dengan menyeleksi pilihan kos yang paling memenuhi kriteria pada setiap tahap.

Hasil penelitian menunjukkan bahwa algoritma Greedy efektif dalam menyaring kos yang memenuhi preferensi utama calon penghuni. Namun, beberapa keterbatasan juga diidentifikasi, seperti kemungkinan solusi sub-optimal yang tidak sepenuhnya memenuhi semua kriteria secara optimal.

Meskipun demikian, algoritma Greedy tetap menawarkan metode yang efisien dan praktis untuk membantu mahasiswa dan pekerja di Jatinangor dalam memilih kos terbaik sesuai dengan kebutuhan dan prioritas mereka. Dengan penyesuaian bobot dan pengembangan lebih lanjut, algoritma ini dapat memberikan solusi yang lebih komprehensif dan memuaskan dalam pemilihan kos terbaik.

Melalui implementasi ini, penelitian ini tidak hanya menawarkan solusi teknis untuk optimasi pemilihan kos, tetapi juga memberikan wawasan tentang bagaimana teknologi algoritmik dapat digunakan untuk meningkatkan kualitas hidup mahasiswa. Diharapkan, penelitian ini dapat memberikan kontribusi pada peningkatan kenyamanan dan kepuasan mahasiswa dalam memilih tempat tinggal yang tepat di Jatinangor.

V. UCAPAN TERIMA KASIH

Penulis ingin mengungkapkan rasa terima kasih kepada Tuhan Yang Maha Esa atas anugerah-Nya yang memungkinkan penyelesaian makalah berjudul "Penerapan Algoritma Greedy untuk Pemilihan Kos Terbaik di Jatinangor" dengan sukses dan tepat waktu. Terima kasih juga disampaikan kepada para dosen pengajar Mata Kuliah Strategi Algoritma, terutama kepada Bapak Monterico Adrian, S.T., M.T., Bapak Ir. Rila Mandala, M.Eng., Ph.D., dan Bapak Dr. Rinaldi Munir, atas arahan, ilmu, dan dorongan yang diberikan selama proses pembelajaran.

Tak lupa, penulis ingin mengucapkan terima kasih kepada para asisten dosen, rekan-rekan seangkatan, dan keluarga atas dukungan yang amat berarti selama masa kuliah dan penulisan makalah ini. Penghargaan juga disampaikan kepada semua sumber referensi dan literatur yang menjadi pijakan penulisan, karena kontribusi mereka sangat berharga dalam memperkaya substansi dan kualitas penelitian ini. Penulis berharap bahwa makalah ini dapat memberikan manfaat serta berperan positif dalam pengembangan ilmu pengetahuan, terutama dalam konteks algoritma Greedy.

REFERENCES

- [1] R. Munir, "Algoritma Greedy (Bag.1)" Diakses dari [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag1.pdf) pada 11 Juni 2024.
- [2] R. Munir, "Algoritma Greedy (Bag.2)" Diakses dari [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag2.pdf) pada 11 Juni 2024.
- [3] R. Munir, "Algoritma Greedy (Bag.3)" Diakses dari [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Greedy-\(2022\)-Bag3.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Greedy-(2022)-Bag3.pdf) pada 11 Juni 2024.
- [4] Suryawan, B. Definisi Kos. Diakses dari <https://id.scribd.com/doc/288003775/Definisi-Kos> pada 11 Juni 2024

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 12 Juni 2024



13522145 Farrel Natha Saskoro