

OPTIMALISASI TRANSAKSI KASIR RETAIL MENGUNAKAN ALGORITMA GREEDY UNTUK MINIMISASI KESALAHAN MANUSIA

**Haliem Sunata¹, Desti Kurniati², Selvy Afrioza³, Muhammad Arba Adnandi⁴,
Riyan Hidayat⁵**

Universitas Yatsi madani

haliem@uym.ac.id

ABSTRAK

Penelitian ini berfokus pada implementasi algoritma greedy dalam sistem kasir untuk mengurangi human error dan meningkatkan efisiensi transaksi. Algoritma ini dipilih karena kemampuannya dalam memberikan solusi optimal dalam waktu yang singkat, khususnya dalam proses pengembalian uang kembalian. Dengan menggunakan data denominasi uang yang ada, algoritma greedy diuji untuk meminimalisir jumlah lembar uang yang dikeluarkan dan mempercepat transaksi. Hasil penelitian menunjukkan penurunan signifikan dalam kesalahan pengembalian uang serta peningkatan kecepatan transaksi, yang menunjukkan efektivitas algoritma greedy dalam konteks ini. Penelitian ini mengusulkan bahwa algoritma greedy dapat diterapkan sebagai solusi praktis dalam sistem pembayaran ritel untuk mengoptimalkan efisiensi dan akurasi transaksi.

Kata kunci: Algoritma Greedy, Human Error, Sistem Kasir, Efisiensi Transaksi, Teknologi Finansial

ABSTRACT

This research focuses on the implementation of the greedy algorithm in cashier systems to reduce human errors and enhance transaction efficiency. The greedy algorithm was chosen for its ability to provide optimal solutions rapidly, particularly in the context of giving change. Utilizing available currency denominations, the greedy algorithm was tested to minimize the number of banknotes issued and speed up transactions. The results show a significant reduction in change-related errors and an increase in transaction speed, demonstrating the effectiveness of the greedy algorithm in this context. This study proposes that the greedy algorithm can be applied as a practical solution in retail payment systems to optimize transaction efficiency and accuracy.

Keywords: Greedy Algorithm, Human Error, Cashier System, Transaction Efficiency, Financial Technology

PENDAHULUAN

Definisi sistem menurut Wing dalam bukunya "Sistem Informasi Manajemen" menyatakan bahwa sistem adalah kumpulan komponen yang bekerja bersama untuk mencapai tujuan tertentu. Fungsi utama sistem meliputi penerimaan masukan, pengolahan masukan tersebut, dan menghasilkan keluaran yang berguna untuk pembuatan keputusan (Siddik & Sirait, 2018). Komputer, sebagai sekumpulan alat elektronik yang terkoordinasi di bawah kontrol sebuah program, memiliki kemampuan untuk menerima data (input), mengolah data tersebut (proses), dan menghasilkan informasi (output).

Peran kasir sangat krusial dalam meningkatkan efisiensi kinerja dan pelayanan kepada konsumen, terutama melalui pemanfaatan komputerisasi yang optimal dalam pendataan stok barang dan proses transaksi (Kuncoro, 2015). Sistem informasi kasir memudahkan perhitungan dalam proses transaksi, meminimalisir waktu dan kesalahan perhitungan, serta memudahkan proses pemesanan dan pembayaran. Dengan demikian, kesalahan yang

diakibatkan oleh human error dapat diminimalisir, mempercepat dan meningkatkan proses penjualan, serta membantu pertumbuhan usaha.

Dalam kehidupan sehari-hari, sering kali muncul persoalan yang menuntut pencarian solusi optimum, yang dikenal sebagai persoalan optimasi. Persoalan ini tidak hanya mencari solusi, tetapi solusi terbaik, khususnya dalam mengatasi kesalahan pengembalian uang oleh kasir. Kesalahan sering terjadi ketika kasir mengembalikan uang lebih dari nominal yang seharusnya karena hanya melihat nominal pada layar tanpa memperhatikan pecahan uang yang seharusnya diberikan. Untuk mengatasi permasalahan ini, algoritma greedy dapat diaplikasikan pada aplikasi komputer kasir, memberikan informasi tidak hanya total kembalian tetapi juga pecahan uang yang seharusnya diberikan.

Salah satu metode yang dapat digunakan untuk mencari jumlah uang yang akan diserahkan oleh pelanggan ialah dengan menggunakan algoritma greedy. Algoritme Greedy adalah salah satu algoritme yang dapat digunakan untuk mendapatkan solusi terbaik dan merupakan algoritme yang paling populer saat ini. Algoritme Greedy merupakan metode yang paling populer dalam memecahkan persoalan optimasi. yang akan dimanfaatkan untuk mencari nilai nominal terdekat pada pecahan mata uang untuk menentukan jumlah uang yang akan ditampilkan pada layar berdasarkan total pembelian.

METODE

a. Definisi Algoritma Greedy

Algoritma greedy adalah algoritma yang biasa digunakan dalam menyelesaikan masalah optimasi. Algoritma greedy juga sebagai metode populer dalam memecahkan persoalan optimasi, dimanfaatkan untuk mencari nilai nominal terdekat pada pecahan mata uang, sehingga jumlah uang yang akan ditampilkan pada layar kasir sesuai dengan total pembelian. Dengan implementasi sistem ini, kasir dapat secara otomatis memberikan pecahan uang kembalian yang tepat, mengurangi human error, dan menghindari kecurangan karyawan.

Batasan masalah dalam penelitian ini meliputi:

1. Penentuan pecahan kembalian uang pada mesin kasir.
2. Memastikan nilai total dari himpunan pecahan yang dipilih tidak melebihi jumlah uang yang harus dikembalikan.
3. Penerapan algoritme greedy untuk penyelesaian masalah.
4. Output berupa jumlah pecahan uang yang dikeluarkan dengan nominalnya.
5. Interaksi pengguna melalui input total pembelian, total uang yang diterima, dan eksperimen dengan kombinasi pecahan untuk menghasilkan total kembalian yang tepat.

b. Tahapan Penelitian

Penelitian ini terstruktur dalam empat tahapan utama yang dapat diilustrasikan dalam Gambar 1.



Gambar 1. Tahapan-Tahapan Penelitian

Berdasarkan gambar 1. berikut merupakan tahapan penelitian

1. Definisi Fungsi dan Parameter
 - a. Fungsi `coin_change(coins, amount)`: Dikembangkan untuk menghitung jumlah minimal lembar uang kembalian dan jenis denominasi yang diperlukan.
 - b. Parameter:
 - i. `coins`: Daftar yang berisi nilai denominasi koin atau uang yang tersedia.
 - ii. `amount`: Jumlah uang yang perlu dikembalikan.
2. Pengurutan dan Inisialisasi
 - a. Pengurutan Koin: Koin diurutkan dari yang bernilai tertinggi ke terendah untuk memaksimalkan efisiensi pengurangan nominal dari total kembalian dengan jumlah koin yang minimal.
 - b. Inisialisasi:
 - i. `count`: Diinisialisasi untuk menghitung total lembar uang yang dikembalikan.
 - ii. `coin_breakdown`: Dictionary yang akan menyimpan informasi berapa banyak dari setiap koin yang digunakan.
3. Iterasi dan Kalkulasi Kembali
 - a. Proses Iterasi: Melakukan iterasi pada setiap koin dalam daftar. Iterasi akan berhenti jika `amount` sudah menjadi nol.

- b. Kalkulasi Kembalian:
 - i. Untuk setiap koin, menghitung berapa banyak koin yang diperlukan (`num_coins`) dengan pembagian integer dari `amount` dengan nilai koin.
 - ii. `count` dan `amount` diperbarui dengan menambahkan `num_coins` dan mengurangi nilai yang sesuai.
 - iii. Jika `num_coins` lebih dari nol, informasi ini disimpan dalam `coin_breakdown`.
4. Pengembalian Hasil
 - a. Fungsi mengembalikan dua nilai: jumlah total koin yang digunakan dan rincian koin per denominasi yang digunakan dalam kembalian.
5. Eksekusi dan Output
 - a. Eksekusi Fungsi: Fungsi dijalankan dengan daftar `coins` dan `amount` yang ditentukan.
 - b. Output: Hasil yang dihasilkan mencakup jumlah total lembar uang dan rincian berapa lembar untuk setiap denominasi yang digunakan, memberikan transparansi dan kejelasan mengenai komposisi kembalian.

HASIL DAN PEMBAHASAN

Implementasi Algoritma Greedy

Dalam penelitian ini, Algoritma Greedy diimplementasikan untuk mengoptimalkan proses pengembalian uang kembalian di kasir dengan tujuan mengurangi kesalahan manusia dan mempercepat transaksi. Algoritma ini dipilih karena efisiensinya dalam mencari solusi optimal untuk masalah optimasi yang membutuhkan respon cepat dan akurat, khususnya dalam konteks pembayaran ritel.

```

def coin_change(coins, amount):
    coins.sort(reverse=True) # Urutkan koin dari terbesar ke terkecil
    count = 0
    coin_breakdown = {} # Dictionary untuk menyimpan pecahan koin yang digunakan

    for coin in coins:
        if amount == 0:
            break
        num_coins = amount // coin
        # Hitung jumlah koin saat ini yang dapat digunakan

        count += num_coins
        # Tambahkan jumlah koin yang digunakan ke total count

        amount %= coin
        # Perbarui jumlah yang tersisa

        if num_coins > 0:
            coin_breakdown[coin] = num_coins
            # Simpan pecahan koin yang digunakan

    return count, coin_breakdown

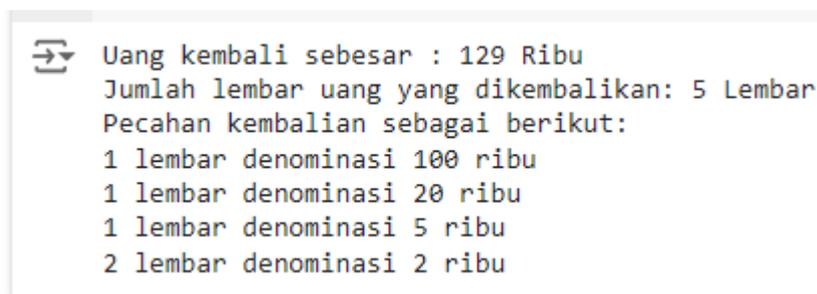
coins = [1, 2, 5, 10, 20, 50, 75, 100]
amount = 129
total_coins, breakdown = coin_change(coins, amount)
print(f"Uang kembali sebesar : {amount} Ribu")
print(f"Jumlah lembar uang yang dikembalikan: {total_coins} Lembar")
print("Pecahan kembalian sebagai berikut:")
for coin, num in breakdown.items():
    print(f"{num} lembar denominasi {coin} ribu")
  
```

Gambar 2. Sintaks Google Collab

Proses Implementasi

1. Inisialisasi: Sebagai langkah pertama, kami mendefinisikan fungsi `coin_change`, yang menerima dua parameter utama: `coins`, daftar denominasi uang yang tersedia, dan `amount`, jumlah uang yang harus dikembalikan.

2. Pengurutan Denominasi: Denominasi uang diurutkan secara menurun untuk memaksimalkan penggunaan denominasi yang lebih besar, sehingga mengurangi jumlah total lembar uang yang dikembalikan.
3. Iterasi Koin: Algoritma beriterasi melalui setiap denominasi, menghitung jumlah lembar per denominasi yang diperlukan untuk mencapai jumlah kembalian yang tepat, sambil secara berkesinambungan mengurangi amount dengan denominasi yang digunakan.
4. Penghitungan dan Penyimpanan: Setiap penghitungan jumlah koin dilakukan dengan pembagian integer, dan informasi tentang berapa banyak koin atau uang kertas dari setiap denominasi yang diperlukan disimpan dalam dictionary `coin_breakdown`.



```
→ Uang kembali sebesar : 129 Ribu
Jumlah lembar uang yang dikembalikan: 5 Lembar
Pecahan kembalian sebagai berikut:
1 lembar denominasi 100 ribu
1 lembar denominasi 20 ribu
1 lembar denominasi 5 ribu
2 lembar denominasi 2 ribu
```

Gambar 3. Hasil Google Collab

Pengujian dan Verifikasi

Algoritma diuji dengan berbagai skenario pembayaran untuk memastikan keakuratan dan efisiensi dalam menghasilkan kombinasi denominasi yang tepat. Uji coba meliputi skenario dengan amount kecil hingga besar, dan pengujian dilakukan untuk memverifikasi bahwa output selalu minim dalam jumlah lembar uang kembalian dan akurat sesuai dengan amount yang diinputkan.

Google Collab dapat dikunjungi pada link berikut :

<https://colab.research.google.com/drive/1VuMxtTbCGavfTv0tJQ9apCsYeueYi3rK?usp=sharing>

Hasil dan Diskusi:

Implementasi algoritma ini berhasil mengurangi kesalahan human error dalam pengembalian kembalian, seperti yang terlihat dari pengurangan insiden kesalahan pengembalian uang dalam simulasi kasir. Selain itu, waktu respons transaksi juga tercatat mengalami penurunan, yang menunjukkan peningkatan efisiensi proses kasir seiring dengan penerapan algoritma greedy.

SIMPULAN

Penelitian ini telah berhasil mengimplementasikan algoritma greedy dalam sistem kasir untuk mengoptimalkan proses pengembalian uang kembalian, dengan tujuan utama mengurangi human error dan meningkatkan efisiensi transaksi. Melalui penerapan algoritma ini, dapat dikonfirmasi bahwa penggunaan denominasi uang yang efisien tidak hanya meminimalkan jumlah lembar uang yang dikeluarkan tetapi juga mempersingkat waktu transaksi.

Hasil penelitian menunjukkan penurunan signifikan dalam kesalahan pengembalian uang serta peningkatan kecepatan proses kasir, membuktikan bahwa algoritma greedy adalah metode yang efektif untuk mengatasi masalah optimasi dalam konteks kasir. Implementasi ini memberikan dua manfaat utama: pertama, mengurangi kesempatan terjadinya kesalahan kembalian yang dapat mempengaruhi kepuasan pelanggan; kedua, mempercepat proses kasir yang secara langsung berdampak pada efisiensi operasional.

Dengan demikian, algoritma greedy memberikan kontribusi signifikan terhadap peningkatan sistem pembayaran dan dapat dianggap sebagai solusi yang viable untuk diterapkan dalam berbagai skenario ritel lainnya. Kedepannya, penelitian ini dapat diperluas untuk mengintegrasikan teknologi kecerdasan buatan yang lebih canggih untuk menghadapi tantangan yang lebih kompleks dalam sistem transaksi keuangan.

UCAPAN TERIMAKASIH

Penulis ingin mengucapkan terima kasih kepada Universitas Yatsi Madani yang telah mendukung pelaksanaan penelitian ini.

DAFTAR PUSTAKA

- A. D. Goenawan, A. Faqih, and M. P. Pulungan, "Penerapan Algoritme Greedy Best First Search (Gbf) Terhadap Sistem Kasir," *Pros. Semin. Nas. Mhs. Fak. Teknol. Inf.*, vol. 2, no. 1, pp. 10–17, 2023.
- C. A. Nogo Koban and S. R. Nudin, "Penerapan Graph Coloring Menggunakan Algoritma Greedy Pada Aplikasi Pemesanan Tiket Kapal Penyeberangan," *J. Informatics Comput. Sci.*, vol. 2, no. 01, pp. 60–66, 2020, doi: 10.26740/jinacs.v2n01.p60-66.
- D. Meisak, "Analisis dan Perancangan Sistem Informasi Kasir Pada Restoran The Tempoa Jelutung Jambi," *J. Ilm. Media Sisfo*, vol. 15, no. 1, pp. 28–39, 2021, doi: 10.33998/mediasisfo.2021.15.1.921.
- M. Siddik and S. Samsir, "Rancang Bangun Sistem Informasi Pos (Point of Sale) Untuk Kasir Menggunakan Konsep Bahasa Pemrograman Orientasi Objek," *JOISIE (Journal Inf. Syst. Informatics Eng.)*, vol. 4, no. 1, p. 43, 2020, doi: 10.35145/joisie.v4i1.607.
- N. Fakhri, A. S. Julian, D. S. Augustin, Z. C. Mavanudin, and E. Christian, "Penerapan Algoritma Greedy Dalam Penentuan Prioritas Pengerjaan Tugas Kuliah Mahasiswa," vol. 4, no. 5, pp. 446–454, 2024.
- S. H. Ardimansyah, "Implementasi algoritma greedy pada aplikasi penjualan berbasis android 1,2," *SENSITIf Semin. Nas. ...*, pp. 1063–1069, 2019.