

Penerapan Algoritma Greedy untuk Menentukan Prioritas Pengantaran Paket

Johannes Winson Sukiatmodjo - 13520123
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
13520123@std.stei.itb.ac.id

Abstrak—Algoritma adalah metode atau langkah yang direncanakan secara tersusun dan berurutan untuk menyelesaikan atau memecahkan permasalahan dengan sebuah instruksi atau kegiatan. Pada dasarnya fungsi utama dari algoritma adalah untuk memecahkan suatu masalah. Suatu algoritma pemrograman membawa keuntungan serta fungsi penting dalam aktivitas pembuatan program. Algoritma pada umumnya digunakan untuk membantu setiap orang yang ingin mengkonversikan sebuah permasalahan ke bahasa pemrograman. Terdapat banyak jenis algoritma yang tersedia, salah satunya algoritma Greedy. Dalam makalah ini, algoritma tersebut akan digunakan untuk membantu para kurir dalam menentukan prioritas pengantaran suatu paket.

Kata Kunci—Algoritma Greedy, Prioritas, Rute, Waktu, Kurir Paket

I. PENDAHULUAN

Pada zaman modern saat ini, algoritma sudah banyak digunakan dalam kehidupan kita sehari-hari tanpa kita sadari. Saat ini, penyelesaian suatu masalah sering kali menggunakan bantuan algoritma agar metode atau langkah yang direncanakan bisa tersusun secara berurutan dan sekuensial dengan baik. Jika kita berbicara mengenai jenisnya, algoritma memiliki jenis yang bermacam-macam, salah satunya algoritma Greedy. Algoritma ini sering kali digunakan untuk menyelesaikan masalah-masalah seperti persoalan penukaran uang, persoalan memilih aktivitas, minimisasi waktu di dalam sistem, persoalan knapsack, penjadwalan job dengan tenggat waktu, pohon merentang minimum, lintasan terpendek, kode huffman, serta pecahan mesir. Pada dasarnya, algoritma ini bertujuan untuk memaksimalkan atau meminimumkan suatu target yang hendak dicapai.

Permasalahan muncul setiap kali seorang kurir mendapatkan tugas untuk mengirimkan sejumlah paket ke masing-masing alamat yang lokasinya berjauhan. Mereka sering kali kebingungan untuk menentukan lokasi mana yang harus dikunjungi terlebih dahulu untuk mengantarkan suatu paket. Karena ketidaktahuan mereka, alhasil kegiatan pengantaran suatu paket menjadi tidak efektif dalam segala aspek, mulai dari waktu, tenaga, dan biaya. Biasanya, mereka mengambil keputusan berdasarkan region suatu lokasi tertentu terlebih dahulu tanpa menganalisis jarak antar dua lokasi pengantaran yang hendak dikunjungi.

Dalam makalah ini, akan dijelaskan secara rinci terkait cara pengambilan keputusan dalam menentukan prioritas rute pengantaran paket yang hendak dipilih dengan menggunakan algoritma Greedy agar dapat meminimumkan total waktu pengiriman, sekaligus menghemat tenaga serta biaya yang dikeluarkan. Cara menentukan jarak antar dua lokasi dapat menggunakan bantuan sebuah aplikasi bernama Google Maps agar dapat memudahkan kita dalam mendapatkan informasi terkait rute terpendek dan waktu tersingkat untuk sampai ke lokasi pengantaran dengan akurat.

II. LANDASAN TEORI

A. Algoritma Greedy

Algoritma Greedy merupakan metode yang paling populer dan sederhana untuk memecahkan persoalan optimasi. Persoalan optimasi (optimization problems) yang dimaksud adalah persoalan mencari solusi optimal. Hanya ada dua macam persoalan optimasi, yaitu

1. maksimasi (maximization)
2. minimasi (minimization)

Kata greedy berasal dari Bahasa Inggris yang memiliki arti tamak, serakah, atau rakus. Prinsip dari algoritma Greedy adalah “take what you can get now!”. Algoritma Greedy membentuk solusi langkah per langkah (step by step). Pada setiap langkah, terdapat banyak pilihan yang perlu dievaluasi. Oleh karena itu, pada setiap langkah harus dibuat keputusan yang terbaik dalam menentukan pilihan. Tidak bisa mundur lagi (kembali) ke langkah sebelumnya. Jadi pada setiap langkah, kita memilih optimum lokal (local optimum) dengan harapan bahwa langkah sisanya mengarah ke solusi optimum global (global optimum).

Jadi, definisi dari algoritma Greedy adalah algoritma yang memecahkan persoalan secara langkah per langkah (step by step) sedemikian sehingga pada setiap langkah:

1. mengambil pilihan yang terbaik yang dapat diperoleh pada saat itu tanpa memperhatikan konsekuensi ke depan (prinsip “take what you can get now!”)
2. “berharap” bahwa dengan memilih optimum lokal pada setiap langkah akan berakhir dengan optimum global.

Algoritma Greedy terdiri dari beberapa elemen, diantaranya sebagai berikut.

1. Himpunan kandidat, C : berisi kandidat yang akan dipilih pada setiap langkah (misal: simpul/sisi di dalam graf, job, task, koin, benda, karakter, dan sebagainya)
2. Himpunan solusi, S : berisi kandidat yang sudah dipilih
3. Fungsi solusi: menentukan apakah himpunan kandidat yang dipilih sudah memberikan solusi
4. Fungsi seleksi (selection function): memilih kandidat berdasarkan strategi greedy tertentu. Strategi greedy ini bersifat heuristik
5. Fungsi kelayakan (feasible): memeriksa apakah kandidat yang dipilih dapat dimasukkan ke dalam himpunan solusi (layak atau tidak)
6. Fungsi obyektif: memaksimumkan atau meminimumkan

Dengan menggunakan elemen-elemen di atas, maka dapat dikatakan bahwa algoritma Greedy melibatkan pencarian sebuah himpunan bagian S , dari himpunan kandidat C , yang dalam hal ini, S harus memenuhi beberapa kriteria yang ditentukan, yaitu S menyatakan suatu solusi dan S dioptimisasi oleh fungsi obyektif.

Algoritma Greedy memiliki skema umum seperti di bawah ini.

```
function greedy(C: himpunan_kandidat) → himpunan_solusi
{ Mengembalikan solusi dari persoalan optimasi dengan algoritma greedy }
Deklarasi
x: kandidat
S: himpunan_solusi
Algoritma:
S ← {} { inisialisasi S dengan kosong }
while (not SOLUSI(S) and (C ≠ {})) do
x ← SELEKSI(C) { pilih sebuah kandidat dari C }
C ← C - {x} { buang x dari C karena sudah dipilih }
if LAYAK(S ∪ {x}) then { x memenuhi kelayakan untuk dimasukkan ke dalam himpunan solusi }
S ← S ∪ {x} { masukkan x ke dalam himpunan solusi }
endif
endwhile
{ SOLUSI(S) or C = {} }
if SOLUSI(S) then { solusi sudah lengkap }
return S
else
write("tidak ada solusi")
endif
```

Pada akhir setiap lelaran (iterasi), solusi yang terbentuk adalah optimum lokal. Pada akhir kalang while-do diperoleh optimum global (jika ada).

Sebagai contoh persoalan optimasi, terdapat soal sebagai berikut. "Diberikan uang senilai A . Tersedia uang koin-koin dalam jumlah yang banyak. Tukar A dengan koin-koin uang yang ada. Berapa jumlah minimum koin yang diperlukan untuk penukaran tersebut?". Contoh soal tersebut merupakan salah satu dari contoh persoalan optimasi, yaitu persoalan penukaran uang (coin exchange problem) dengan jenis persoalan minimasi.

Jika ditinjau lebih mendalam terkait persoalan tersebut, strategi greedy dalam persoalan tersebut adalah pilihlah koin dengan nilai terbesar dari himpunan koin yang tersisa pada

setiap langkahnya. Misal: $A = 32$, koin yang tersedia: 1, 5, 10, dan 25 (dalam jumlah banyak)

- Langkah 1: pilih 1 buah koin 25 (Total = 25)
- Langkah 2: pilih 1 buah koin 5 (Total = 25+5 = 30)
- Langkah 3: pilih 1 buah koin 1 (Total = 25+5+1 = 31)
- Langkah 4: pilih 1 buah koin 1 (Total = 25+5+1+1 = 32)

Solusi: Jumlah koin minimum = 4 (solusi optimal!)

Elemen-elemen algoritma Greedy dalam persoalan tersebut adalah sebagai berikut.

- Himpunan kandidat: himpunan koin atau pecahan uang yang merepresentasikan nilai 1, 5, 10, 25, paling sedikit mengandung satu koin untuk setiap nilai.
- Himpunan solusi: koin-koin yang terpilih.
- Fungsi solusi: memeriksa apakah total nilai koin yang dipilih tepat sama jumlahnya dengan nilai uang yang ditukarkan.
- Fungsi seleksi: pilihlah koin yang bernilai tertinggi dari himpunan koin yang tersisa.
- Fungsi kelayakan: memeriksa apakah koin yang baru dipilih apabila dijumlahkan dengan semua koin yang sudah berada di dalam himpunan tidak melebihi jumlah uang yang harus dibayar.
- Fungsi obyektif: jumlah koin yang digunakan minimum.

Perlu diingat bahwa optimum global belum tentu merupakan solusi optimum (terbaik), bisa jadi merupakan solusi sub-optimum atau pseudo-optimum. Alasannya adalah sebagai berikut.

1. Algoritma Greedy tidak beroperasi secara menyeluruh terhadap semua kemungkinan solusi yang ada (sebagaimana pada metode exhaustive search).
2. Terdapat beberapa fungsi SELEKSI yang berbeda, sehingga kita harus memilih fungsi yang tepat jika kita ingin algoritma menghasilkan solusi optimal.

Jadi, pada sebagian persoalan, algoritma Greedy tidak selalu berhasil memberikan solusi yang optimal, namun sub-optimal. Perhatikan contoh-contoh kasus berikut.

- Koin: 5, 4, 3, dan 1 (dalam jumlah yang banyak)
Uang yang ditukar $A: 7$
Solusi greedy: $7 = 5+1+1$ (3 koin) → tidak optimal
Solusi optimal: $7 = 4+3$ (2 koin)
- Koin: 10, 7, 1 (dalam jumlah yang banyak)
Uang yang ditukar: 15

Solusi greedy: $15 = 10+1+1+1+1+1$ (6 koin) → tidak optimal

Solusi optimal: $15 = 7+7+1$ (3 koin)

- Koin: 15, 10, dan 1 (dalam jumlah yang banyak)

Uang yang ditukar: 20

Solusi greedy: $20 = 15+1+1+1+1+1$ (6 koin) → tidak optimal

Solusi optimal: $20 = 10+10$ (2 koin)

Jika solusi terbaik mutlak tidak terlalu diperlukan, maka algoritma Greedy dapat digunakan untuk menghasilkan solusi hampiran (approximation), daripada menggunakan algoritma yang kebutuhan waktunya eksponensial untuk menghasilkan solusi yang eksak. Misalnya mencari tur dengan bobot minimal pada persoalan TSP untuk jumlah simpul (n) yang banyak dengan algoritma Brute Force dibutuhkan waktu komputasi yang lama untuk menemukannya. Dengan algoritma Greedy, meskipun tur dengan berbobot minimal tidak dapat ditemukan, namun solusi dengan algoritma Greedy dianggap sebagai hampiran solusi optimal.

Namun bila algoritma Greedy dapat menghasilkan solusi optimal, maka keoptimalannya itu harus dapat dibuktikan secara matematis. Membuktikan optimalitas algoritma Greedy secara matematis adalah tantangan tersendiri. Lebih mudah memperlihatkan algoritma Greedy tidak selalu optimal dengan menunjukkan counterexample (contoh kasus yang menunjukkan solusi yang diperoleh tidak optimal), misalnya pada persoalan penukaran uang di atas (kadang-kadang solusinya optimal kadang kadang tidak).

Berikut disajikan beberapa contoh persoalan yang diselesaikan dengan algoritma Greedy.

1. Persoalan penukaran uang (coin exchange problem)
2. Persoalan memilih aktivitas (activity selection problem)
3. Minimisasi waktu di dalam sistem
4. Persoalan knapsack (knapsack problem)
5. Penjadwalan Job dengan tenggat waktu (job scheduling with deadlines)
6. Pohon merentang minimum (minimum spanning tree)
7. Lintasan terpendek (shortest path)
8. Kode Huffman (Huffman code)
9. Pecahan Mesir (Egyptian fraction)

B. Kurir

Kurir adalah sebuah aktivitas pengiriman barang yang dilakukan secara langsung. Sebenarnya pekerjaan menjadi kurir sudah ada sejak zaman dahulu kala, di mana seseorang akan berlari untuk menyampaikan pesan atau barang. Dari situlah awalnya muncul kata courier dalam bahasa Inggris yang berasal dari bahasa Yunani "curere" yang berarti lari.

Masyarakat Indonesia sendiri lebih mengenal kurir sebagai sebuah pekerjaan untuk mengantarkan barang. Biasanya kurir akan mengendarai sepeda atau motor agar lebih cepat dalam proses pengirimannya. Walaupun sebenarnya ada juga kurir yang menggunakan mobil, namun masyarakat Indonesia belum terlalu familiar.

Pekerjaan menjadi seorang kurir, sekarang menjadi incaran oleh banyak orang. Berbeda dengan dulu, pekerjaan menjadi kurir sering kali diremehkan. Hal itu dipengaruhi dengan berkembangnya industri e-commerce yang turut membawa angin segar bagi jasa kurir di Indonesia. Banyaknya orang yang gemar berbelanja secara online juga berdampak pada banyaknya orang yang ingin menjadi kurir.

Seorang kurir bertugas untuk mengantarkan dokumen dan paket untuk individu, bisnis, institusi, dan lembaga pemerintah. Kurir biasanya bepergian dengan berjalan kaki, sepeda, sepeda motor, mobil, truk pengiriman, atau angkutan umum.

Seorang kurir menyediakan layanan pengiriman door-to-door untuk berbagai klien, termasuk kantor hukum, bank, dan rumah sakit. Sebagian besar kurir memiliki spesialisasi untuk pengiriman lokal, sering kali di daerah perkotaan besar. Mereka menawarkan layanan pengiriman satu hari atau satu jam.

Kurir menerima instruksi baik secara langsung atau melalui telepon atau radio dua arah. Dengan informasi ini, mereka merencanakan rute dan jadwal pengiriman yang paling efisien. Beberapa kurir membawa barang hanya untuk klien tertentu, seperti firma hukum, lembaga keuangan, atau laboratorium medis.

Pekerjaan kurir adalah mulai dari mengambil paket yang masuk ke kantor dan mengantarnya ke penerima. Di samping itu, seorang kurir juga harus memastikan bahwa konsumen menerima pesannya tepat waktu, melakukan pengecekan serta pencocokan data dengan dokumen yang disediakan terhadap paket. Berikut ini disajikan rincian jobdesk seorang kurir.

- Melakukan pengiriman barang ke alamat dan penerima yang tepat
- Melakukan pendataan status pengiriman barang agar dapat dipantau
- Memastikan barang tetap dalam keadaan aman hingga sampai tujuan
- Menjalankan dan menaati SOP perusahaan delivery yang berlaku
- Menjaga data pengirim dan penerima barang tetap aman pada label
- Menerima pembayaran dan transaksi COD
- Menyetorkan data dan laporan hasil pengiriman kepada perusahaan

C. Google Maps

Google Maps adalah layanan pemetaan web yang dikembangkan Google. Dalam Google Maps, kita dapat

melihat citra satelit, foto udara, peta jalan, pemandangan jalan panorama interaktif 360° (Street View), kondisi lalu lintas dalam waktu nyata, dan perencanaan rute untuk bepergian dengan berjalan kaki, mobil, sepeda, udara (dalam versi beta), dan transportasi umum. Pada tahun 2020 yang lalu, Google Maps telah digunakan oleh lebih dari 1 miliar orang setiap bulannya.

Google Maps awalnya adalah program C++ yang dirancang oleh dua bersaudara asal Denmark, Lars dan Jens Eilstrup Rasmussen, serta Noel Gordon dan Stephen Ma, di perusahaan Where 2 Technologies yang berbasis di Sydney. Google Maps pertama kali dirancang untuk diunduh secara terpisah oleh pengguna, tetapi perusahaan kemudian mengajukan ide untuk produk berbasis web murni ke manajemen Google, mengubah metode distribusinya.

Berikut beberapa manfaat yang bisa kita dapatkan dari penggunaan layanan Google Maps secara umum, yaitu:

- Mencari tahu arah

Setiap kali kita akan pergi ke suatu tempat untuk pertama kalinya, kita dapat menggunakan Google Maps untuk mendapatkan petunjuk arah dan jalan yang dapat kita lalui untuk ke sana. Cukup buka maps.google.com, pilih Petunjuk Arah, dan masukkan tujuan kita.

- Menyediakan jalan pintas

Selain arah jalan utama, Google Maps juga dapat memberikan petunjuk arah belokan demi belokan secara gratis. Smartphone kita akan secara otomatis bergerak dan memberi tahu kapan tepatnya harus berbelok ke kiri, memutar balik, atau terus mengemudi. Google Maps bahkan dapat memberi informasi saat ada banyak lalu lintas di jalan yang sedang atau akan kita lalui dan akan menyarankan rute alternatif atau jalan pintas.

- Menemukan informasi dengan cepat

Google Maps lebih dari sekadar alat untuk membantu menemukan arah jalan. Ada banyak informasi tentang berbagai tempat yang dimasukkan langsung ke dalam layanan peta ini. Misalnya, kita dapat mencari alamat kedai kopi lokal hanya dengan menelusuri namanya. Kita juga dapat melihat sampai kapan kedai tersebut buka dan juga tersedia tautan ke situs webnya melalui Google Maps.

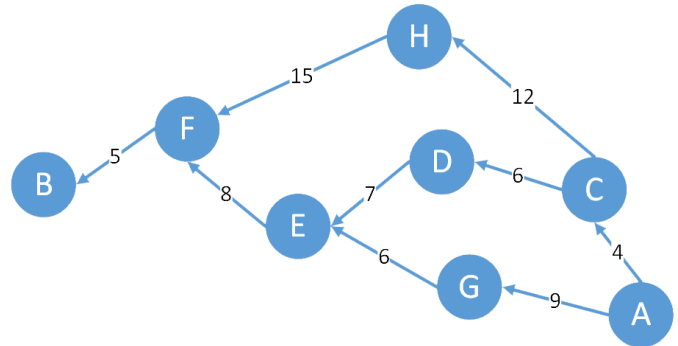
- Pergi menjelajah

Google Maps adalah alat yang sangat berguna. Layanan ini dapat membantu kita mencapai tujuan dan mempelajari apa yang perlu kita ketahui sebelum sampai di sana. Tidak seperti peta kertas, Google Maps tidak terbatas pada satu negara bagian atau benua. Faktanya, Google Maps menyediakan peta untuk hampir setiap negara yang ada di dunia. Dikombinasikan dengan citra satelit, foto kiriman pengguna, dan fitur tampilan jalan Google yang canggih, Google Maps lebih dari sekadar peta. Ini

adalah atlas interaktif dan cara untuk mempelajari berbagai tempat dan orang di seluruh dunia.

III. PENERAPAN ALGORITMA GREEDY UNTUK MENENTUKAN PRIORITAS PENGANTARAN PAKET

Pertama-tama, untuk menentukan jarak antara 2 lokasi juga dapat menggunakan algoritma Greedy. Misalnya diberikan sebuah peta jalur perjalanan yang sudah direpresentasikan dengan menggunakan graf, spesifiknya Directed Graph (graf berarah) seperti di bawah ini.



Gambar 1. Graf berarah dari titik A ke B

Sumber: <http://dev.bertzzie.com/knowledge/analisis-algoritma/Greedy.html>

Untuk mencari jarak terpendek dari A ke B, sebuah algoritma Greedy akan menjalankan langkah-langkah seperti berikut.

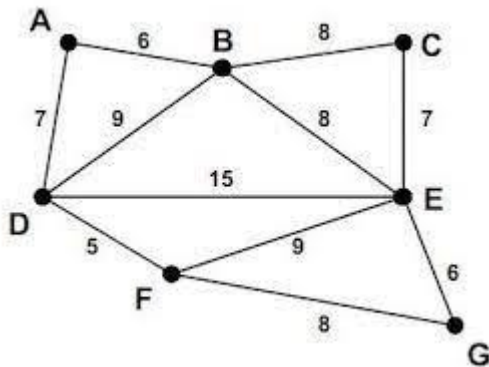
1. Kunjungi satu titik pada graf dan ambil seluruh titik yang dapat dikunjungi dari titik sekarang.
2. Cari minimum lokal ke titik selanjutnya.
3. Tandai graf sekarang sebagai graf yang telah dikunjungi dan pindah ke minimum lokal yang telah ditentukan.
4. Kembali ke langkah 1 sampai titik tujuan didapatkan.

Jika mengaplikasikan langkah-langkah di atas, maka kita akan mendapatkan pergerakan seperti berikut.

1. Mulai dari titik awal (A), ambil seluruh titik yang dapat dikunjungi.
2. Minimum lokal adalah ke C karena jarak ke C adalah yang paling dekat.
3. Tandai A sebagai titik yang telah dikunjungi, lalu pindah ke C.
4. Ambil seluruh titik yang dapat dikunjungi dari C.
5. Minimum lokal adalah ke D dengan jarak 6.
6. Tandai C sebagai titik yang telah dikunjungi, lalu pindah ke D.
7. Setelah itu, pindah ke E karena hanya ada 1 titik yang dapat dikunjungi dari titik D.
8. Dengan cara yang sama seperti langkah 7, pindah ke F lalu pindah ke B.

Dengan menggunakan algoritma Greedy pada graf di atas, hasil akhir yang akan didapatkan sebagai jarak terpendek adalah A-C-D-E-F-B. Hasil jarak terpendek yang didapatkan ini tidak tepat dengan jarak terpendek yang sebenarnya (A-G-E-F-B). Sesuai dengan landasan teori yang diberikan, algoritma Greedy memang tidak selamanya memberikan solusi yang optimal dikarenakan pencarian minimum lokal pada setiap langkahnya tanpa memperhatikan solusi secara keseluruhan. Oleh karena itu, untuk ke depannya akan digunakan aplikasi Google Maps untuk mendapatkan data jarak antara 2 lokasi dengan akurat.

Kembali ke topik bahasan dalam makalah ini, misalkan terdapat sebuah studi kasus seorang kurir yang memiliki tugas untuk mengantarkan paket di 6 lokasi yang berbeda. Peta lokasi untuk setiap rumah direpresentasikan dalam graf berbobot seperti di bawah ini.



Gambar 2. Graf berbobot peta pengiriman paket

Sumber: <https://eprints.uny.ac.id/28787/2/c.BAB%20II.pdf>

Dalam studi kasus ini, kita misalkan titik B sebagai tempat bekerja si kurir sekaligus tempat untuk mengambil paket-paket yang hendak diantarkan ke masing-masing rumah, sedangkan titik A, C, D, E, F, dan G merupakan lokasi-lokasi rumah yang hendak dikunjungi si kurir.

Sebelum lanjut ke penyelesaian solusinya, kita harus menganalisis elemen-elemen algoritma Greedy-nya terlebih dahulu yang terdapat dalam studi kasus ini. Untuk detail penjelasannya diberikan di bawah ini.

- Himpunan kandidat: himpunan titik yang merepresentasikan rumah-rumah yang hendak dikunjungi si kurir, yaitu titik A, C, D, E, F, dan G.
- Himpunan solusi: rute-rute yang terpilih berdasarkan prioritas jalur terpendek.
- Fungsi solusi: memeriksa apakah semua lokasi tujuan sudah dikunjungi atau belum.
- Fungsi seleksi: pilihlah rute dengan jarak terpendek dari himpunan titik yang belum dikunjungi.
- Fungsi kelayakan: memeriksa apakah titik yang hendak dikunjungi sudah pernah dilalui sebelumnya atau belum.
- Fungsi obyektif: total panjang rute yang dilalui minimum.

Setelah menganalisis elemen-elemen algoritma Greedy dalam studi kasus ini, kita akan langsung menyelesaikan solusinya dengan cara yang sama seperti menentukan jarak antara 2 lokasi yang berbeda. Langkah-langkahnya secara rinci dipaparkan di bawah ini.

- Langkah 1: Dari titik asal (B), titik-titik yang dapat dikunjungi adalah titik A, C, D, dan E. Titik yang merupakan minimum lokal adalah titik A karena memiliki panjang rute yang terpendek, yaitu 6. Tandai titik B sebagai titik yang telah dikunjungi, lalu pindah ke titik A.
- Langkah 2: Dari titik A, titik-titik yang dapat dikunjungi adalah titik B dan D. Karena titik B sudah pernah dikunjungi sebelumnya, maka titik yang akan dikunjungi selanjutnya adalah titik D. Tandai titik A sebagai titik yang telah dikunjungi, lalu pindah ke titik D.
- Langkah 3: Dari titik D, titik-titik yang dapat dikunjungi adalah titik A, B, E, dan F. Karena titik A dan B sudah pernah dikunjungi sebelumnya, maka titik-titik tersebut dapat diabaikan. Oleh karena itu, titik yang merupakan minimum lokal adalah titik F karena memiliki panjang rute yang terpendek, yaitu 5. Tandai titik D sebagai titik yang telah dikunjungi, lalu pindah ke titik F.
- Langkah 4: Dari titik F, titik-titik yang dapat dikunjungi adalah titik D, E, dan G. Karena titik D sudah pernah dikunjungi sebelumnya, maka titik tersebut dapat diabaikan. Oleh karena itu, titik yang merupakan minimum lokal adalah titik G karena memiliki panjang rute yang terpendek, yaitu 8. Tandai titik F sebagai titik yang telah dikunjungi, lalu pindah ke titik G.
- Langkah 5: Dari titik G, titik-titik yang dapat dikunjungi adalah titik E dan F. Karena titik F sudah pernah dikunjungi sebelumnya, maka titik yang akan dikunjungi selanjutnya adalah titik E. Tandai titik G sebagai titik yang telah dikunjungi, lalu pindah ke titik E.
- Langkah 6: Dari titik E, titik-titik yang dapat dikunjungi adalah titik C, D, F, dan G. Karena titik D, F, dan G sudah pernah dikunjungi sebelumnya, maka titik yang akan dikunjungi selanjutnya adalah titik C. Tandai titik E sebagai titik yang telah dikunjungi, lalu pindah ke titik C.
- Langkah 7: Karena semua titik yang menjadi tujuan pengantaran sudah dikunjungi, maka si kurir kembali ke titik asal, yaitu titik B dengan panjang rute 8.

Jadi, rute terpendek yang bisa ditempuh oleh si kurir adalah B-A-D-F-G-E-C-B dengan panjang rute sebesar $6+7+5+8+6+7+8 = 47$. Solusi yang didapat dari penyelesaian di atas juga merupakan solusi optimum dari studi kasus ini.

IV. KESIMPULAN

Algoritma Greedy memiliki banyak manfaat dalam mengambil keputusan optimasi, salah satunya dalam proses penentuan prioritas rute terpendek dalam pengantaran suatu paket. Berdasarkan pemaparan yang disajikan pada bab III, dapat disimpulkan bahwa pada studi kasus tersebut solusi yang didapat merupakan solusi optimal. Sesuai dengan landasan teori yang diberikan, algoritma Greedy memang tidak selamanya memberikan solusi yang optimal, hanya saja solusi yang diberikan mendekati optimal. Namun, yang terpenting adalah dengan ini akan membantu para kurir untuk bisa lebih efektif dalam mengantarkan pesanan customernya agar bisa sampai tujuan secepat mungkin.

V. UCAPAN TERIMA KASIH

Pertama-tama, penulis mengucapkan terima kasih kepada Tuhan Yang Maha Esa atas berkat dan rahmat-Nya, penulis bisa menyelesaikan tugas makalah ini dengan baik. Penulis juga mengucapkan terima kasih kepada Bapak Rinaldi selaku dosen dari mata kuliah IF2211 Strategi Algoritma, yang selama ini telah membimbing penulis dalam pembelajaran Strategi Algoritma dan selalu menyediakan materi perkuliahan dalam bentuk ppt, latihan-latihan soal untuk UTS maupun UAS, dan lain sebagainya di website milik beliau yang tentunya sangat berguna dalam proses pembelajaran Strategi Algoritma ini. Penulis juga berterima kasih atas insight-insight yang telah diberikan oleh beliau selama mengajar mata kuliah Strategi Algoritma ini sehingga penulis tidak hanya mendapatkan ilmu yang bermanfaat, namun juga bekal untuk menjalani kehidupan pasca perkuliahan.

VIDEO LINK AT YOUTUBE

<https://youtu.be/PiD7MJ2t6Mw>

REFERENSI

- [1] <https://www.unida.ac.id/teknologi/artikel/pengertian-algoritma.html>. Diakses pada 20 Mei 2022 pukul 19:01.
- [2] <https://www.ekrut.com/media/algoritma-adalah>. Diakses pada 20 Mei 2022 pukul 19:01.
- [3] [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). Diakses pada 20 Mei 2022 pukul 21:30.
- [4] <http://www.parselday.com/blog/apa-itu-kurir-2/>. Diakses pada 21 Mei 2022 pukul 02:02.
- [5] <https://campus.quipper.com/careers/kurir>. Diakses pada 21 Mei 2022 pukul 02:03.
- [6] <https://blog.lokerpalangka.net/2021/04/tugas-kurir.html>. Diakses pada 21 Mei 2022 pukul 02:03.
- [7] <https://www.merdeka.com/jatim/google-maps-adalah-layanan-pemetaan-wilayah-melalui-web-ketahui-selengkapnyakln.html>. Diakses pada 21 Mei 2022 pukul 04:16.
- [8] <http://dev.bertzze.com/knowledge/analisis-algoritma/Greedy.html>. Diakses pada 21 Mei 2022 pukul 22:11.

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, 23 Mei 2022



Johannes Winson Sukiatmodjo
13520123