

Penerapan Program Dinamis untuk Optimasi Waktu *Pizza Hut Delivery* (30 Menit Dijamin Tiba)

Riska - 13512062
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13512062@std.stei.itb.ac.id

Abstract—*Pizza Hut* merupakan salah satu restoran yang menyediakan layanan pesan – antar yang biasa disebut *Pizza Hut Delivery* (PHD). Layanan ini memberikan jaminan pesanan tiba dalam waktu 30 menit. Adanya layanan ini tentu akan menarik para calon konsumen dan akan menambah keuntungan bagi *Pizza Hut*. Akan tetapi, bagaimana jika dalam waktu 30 menit pesanan belum tiba? Hal ini akan membuat konsumen kecewa dan restoran kehilangan pelanggan. Keterlambatan tersebut dapat disebabkan oleh banyak faktor, salah satunya karena pemilihan rute yang tidak tepat. Persoalan seperti ini dapat diselesaikan dengan program dinamis. Makalah ini akan membahas tentang optimasi waktu *Pizza Hut Delivery* dengan metode pemecahan masalah menggunakan program dinamis.

Index Terms—program dinamis, *Pizza Hut*, *Delivery*, optimasi, 30 menit.

I. PENDAHULUAN

Pizza Hut Delivery adalah salah satu layanan yang disediakan oleh restoran *Pizza Hut*. Restoran *Pizza Hut* sendiri merupakan salah satu restoran terkenal di dunia dengan cabang hampir 12.000 restoran di berbagai negara. Restoran *Pizza Hut* dibuka pertama kali pada tahun 1958 di Wichita, Kansas, Amerika Serikat oleh James McLamore dan David Edgerton, dan hadir di Indonesia pada tahun 1984.

Adanya jasa pesan antar ini tentu saja memberikan keuntungan baik bagi konsumen maupun restoran itu sendiri. Konsumen tidak harus datang ke restoran tersebut untuk menikmati berbagai macam menu yang disediakan *Pizza Hut*. Konsumen hanya cukup memesan via telepon ke 500600 dan menunggu pesanan datang. Dengan begitu, konsumen *Pizza Hut* pun akan bertambah banyak.

Nah, demi bertambahnya konsumen dan keuntungan yang akan didapat, *Pizza Hut* menambahkan fitur pada layanan pesan-antar ini dengan menjamin pesanan tiba hanya dalam waktu 30 menit.



Gambar 1 : Kotak pesanan PHD

Tawaran tersebut tentu akan menarik perhatian para calon konsumen. Tetapi masalahnya adalah bagaimana jika pesanan tidak tiba dalam waktu 30 menit? Hal ini akan membuat konsumen kecewa dan *Pizza Hut* akan kehilangan pelanggan.

Terkadang, konsumen tidak menerima alasan apapun ketika keterlambatan itu terjadi. Apalagi konsumen yang merasa bahwa alamat tujuannya dapat ditempuh dalam waktu kurang dari atau sama dengan 30 menit. Padahal, keterlambatan tersebut dapat disebabkan oleh berbagai faktor. Salah satunya adalah pemilihan rute yang salah.

Pemilihan tur yang tepat adalah hal yang perlu diperhatikan dalam persoalan optimasi.

Sebagai restoran terkenal di dunia, *Pizza Hut* tentu saja tidak ingin kehilangan kepercayaan dari pelanggannya hanya karena hal – hal kecil seperti keterlambatan pengiriman.

Oleh karena itu, dalam makalah ini akan dibahas mengenai optimasi waktu *Pizza Hut Delivery* dengan menggunakan program dinamis sebagai salah satu metode pemecahan masalah yang dapat digunakan untuk menyelesaikan persoalan optimasi.

II. DASAR TEORI

A. Pengenalan Program Dinamis

Program dinamis (*Dynamic Programming*) adalah salah satu metode pemecahan masalah dengan cara menguraikan solusi menjadi sekumpulan langkah atau tahapan sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan.

Menurut Dimayati (1992), program dinamis adalah suatu teknik matematis yang biasanya digunakan untuk membuat suatu keputusan dari serangkaian keputusan yang saling berkaitan. Tujuan utama model ini adalah untuk mempermudah penyelesaian persoalan optimasi yang mempunyai karakteristik tertentu. Istilah program dinamis, pertama kali diperkenalkan pada era 1950-an oleh Richard Bellman seorang professor di Universitas Princeton dan juga bekerja pada RAND corporation, perlu diketahui bahwa RAND corporation pada era itu merupakan suatu perusahaan yang dibentuk untuk menawarkan analisis dan riset untuk angkatan bersenjata Amerika Serikat. Saat itu, Bellman bekerja di bidang pengambilan keputusan multi tahap (*multistage desicion process*) dan mengerjakan beberapa metode matematis, beberapa tahun kemudian setelah Bellman berada di RAND, lahirlah istilah pemrograman dinamis. Istilah ini tidak secara langsung berhubungan dengan pemrograman, melainkan digunakan sebagai judul *project* yang kemudian yang diusulkan RAND corporation pada Angkatan Udara Amerika Serikat.

Ada beberapa hal mendasar yang perlu diketahui pada penyelesaian persoalan dengan program dinamis ini, yaitu:

1. Terdapat sejumlah berhingga pilihan yang mungkin,
2. Solusi pada setiap tahap dibangun dari hasil solusi tahap sebelumnya,
3. Menggunakan persyaratan optimasi dan kendala untuk membatasi sejumlah pilihan yang harus dipertimbangkan pada suatu tahap.

Penyelesaian persoalan dengan metode program dinamis ini memiliki kemiripan dengan metode *greedy* yang juga membentuk solusi secara bertahap. Pada metode *greedy*, pengambilan keputusan pada setiap tahap dilakukan dengan cara mengambil pilihan yang paling menarik (memenuhi ukuran optimasi yang digunakan), didasarkan hanya pada informasi lokal, dan pada setiap tahap itu tidak pernah diambil keputusan yang salah. Sebagai contoh, pada persoalan minimisasi waktu di dalam sistem, dimana pelanggan berikutnya yang akan dimasukkan ke dalam antrian adalah pelanggan yang membutuhkan waktu pelayanan terkecil diantara pelanggan lain yang belum dilayani. Pada persoalan tertentu, algoritma *greedy* bekerja dengan baik, namun pada kebanyakan persoalan yang lain tidak. Hal tersebut

karena pengambilan keputusan pada setiap langkah *greedy* tidak pernah mempertimbangkan lebih jauh apakah pilihan tersebut merupakan pilihan yang tepat pada langkah-langkah selanjutnya.

Berbeda dengan *greedy*, program dinamis menggunakan Prinsip Optimalitas untuk membuat rangkaian keputusan yang optimal. Prinsip tersebut berbunyi : *jika solusi total optimal, maka bagian solusi sampai tahap ke-k juga optimal*. Prinsip optimalitas berarti bahwa jika kita bekerja dari tahap k ke tahap $k+1$, maka kita dapat menggunakan hasil optimal dari tahap k tanpa harus kembali ke tahap awal. Jika ongkos (*cost*) pada setiap tahap dihitung, maka dapat dirumuskan bahwa

Ongkos pada tahap $k+1$ = (ongkos yang dihasilkan pada tahap k) + (ongkos dari tahap k ke tahap $k+1$)

Dengan prinsip optimalitas, bisa dijamin bahwa pengambilan keputusan pada suatu tahap adalah keputusan yang benar untuk tahap-tahap selanjutnya. Jadi, perbedaan mendasar antara *greedy* dengan program dinamis adalah bahwa hanya ada satu rangkaian keputusan yang pernah dihasilkan pada metode *greedy*, sedangkan pada program dinamis ada lebih dari satu rangkaian keputusan, dan hanya rangkaian keputusan yang memenuhi prinsip optimalitas yang akan dihasilkan.

Nama program dinamis sendiri mengacu pada perhitungan dengan menggunakan tabel. Seperti halnya algoritma *greedy*, program dinamis digunakan untuk memecahkan masalah optimasi.

B. Karakteristik Persoalan Program Dinamis

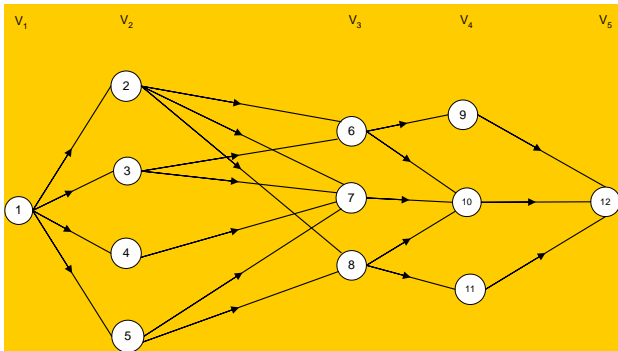
Program dinamis diterapkan pada persoalan yang memiliki karakteristik sebagai berikut :

1. Persoalan dapat dibagi menjadi beberapa tahap (*stage*), dan pada setiap tahap hanya diambil satu keputusan.
2. Masing - masing tahap terdiri dari sejumlah status (*state*) yang berhubungan dengan tahap tersebut. Status merupakan berbagai kemungkinan masukan yang ada pada tahap tersebut. Jumlah status bisa berhingga atau tidak berhingga.
3. Hasil dari keputusan yang diambil pada setiap tahap ditransformasikan dari status yang bersangkutan ke status berikutnya pada tahap berikutnya.
4. Ongkos (*cost*) pada suatu tahap meningkat secara teratur dengan bertambahnya jumlah tahapan.
5. Ongkos pada suatu tahap bergantung pada ongkos tahap – tahap yang sudah berjalan dan ongkos pada tahap tersebut.
6. Keputusan terbaik pada suatu tahap bersifat independen terhadap keputusan yang dilakukan pada tahap sebelumnya.
7. Adanya hubungan rekursif yang mengidentifikasi keputusan terbaik untuk setiap

status pada tahap k memberikan keputusan terbaik untuk setiap status pada tahap $k+1$.

- Prinsip optimalitas berlaku pada persoalan tersebut.

Graf multistage (*multistage graph*) adalah contoh persoalan yang dapat diselesaikan dengan program dinamis. Tiap simpul di dalam graf tersebut menyatakan status, sedangkan V_1, V_2, \dots menyatakan tahap.



Gambar 2 : Graf multistage (*multistage graph*)

Dalam menyelesaikan persoalan dengan program dinamis, dapat digunakan dua ancangan (*approach*) berbeda yaitu maju (*forward* atau *up-down*) dan mundur (*backward* atau *bottom-up*). Kedua ancangan ini ekuivalen dan menghasilkan solusi optimum yang sama. Pada umumnya, penyelesaian dengan program dinamis mundur lebih mangkus.

Ada empat langkah yang dilakukan dalam mengembangkan algoritma program dinamis, yaitu :

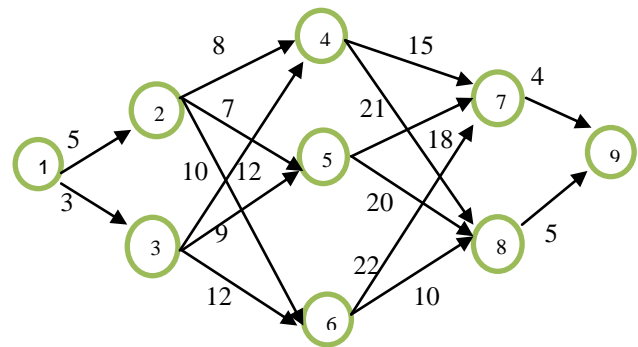
- Karakteristikan struktur solusi optimal.
- Definisikan secara rekursif nilai solusi optimal.
- Hitung nilai solusi optimal secara maju atau mundur.
- Konstruksi solusi optimal.

III. ISI

Penggunaan program dinamis pada persoalan ini adalah untuk menentukan rute dengan waktu optimum dari layanan pesan – antar *Pizza Hut* yang menjamin pesanan tiba dalam waktu 30 menit.

Sebagai contoh, misalkan *Pizza Hut* menerima sejumlah pesanan dengan alamat tujuan X. Untuk mencapai tujuan tersebut, tentukan rute dengan waktu paling optimal (kurang dari atau sama dengan 30 menit).

Persoalan tersebut dapat digambarkan sebagai graf berarah di bawah ini. Simpul 1 menyatakan lokasi restoran *Pizza Hut* dan simpul 9 menyatakan alamat tujuan pengiriman. Bobot pada tiap sisi menyatakan estimasi waktu (satuan menit) yang dibutuhkan untuk mencapai simpul tetangga (tujuan berikutnya). Dengan asumsi kecepatan yang digunakan konstan, sehingga waktu berbanding lurus dengan jarak.



Gambar 3 : Graf pengiriman pesanan

Pada contoh diatas, pencarian solusi dapat dilakukan dengan metode pemecahan masalah program dinamis mundur maupun maju. Dalam makalah ini, akan dibahas penyelesaian dengan program dinamis mundur dan maju.

Program dinamis membagi persoalan diatas menjadi empat tahap dan akan mengambil satu keputusan pada setiap tahapnya..

A. Penyelesaian dengan Program Dinamis Mundur

Misalkan x_1, x_2, x_3 , dan x_4 adalah simpul – simpul yang dikunjungi pada tahap k , dengan $k=1,2,3,4$. Maka rute yang dilalui adalah $1 \rightarrow x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4$, yang dalam hal ini $x_4 = 9$.

Pada contoh persoalan ini :

- Tahap (k) adalah proses memilih simpul tujuan berikutnya
- Status (s) yang berhubungan dengan masing – masing tahap adalah simpul – simpul di dalam graf

Relasi rekurens berikut menyatakan rute dengan waktu optimal dari status s ke x_4 pada tahap k :

$$f_k(s) = c_{sx_4} \quad (\text{basis})$$

$$f_k(s) = \min \{ c_{sx_k} + f_{k-1}(x_k) \}; k=1,2,3 \quad (\text{rekurens})$$

$$= f_k(s, x_k)$$

Keterangan :

- X_k : peubah keputusan pada tahap k
- C_{sx_k} : bobot (cost) sisi dari s ke x_k
- $f_k(s, x_k)$: total bobot lintasan dari s ke x_k
- $f_k(s)$: nilai minimum dari $f_k(s, x_k)$

Program dinamis mundur bertujuan untuk mendapatkan $f_1(1)$ dengan cara mencari $f_4(s), f_3(s), f_2(s)$ terlebih dahulu.

Tahap 4 :

$$f_4(s) = c_{sx_4}$$

| s | Solusi Optimum | |
|---|----------------|---------|
| | $f_4(s)$ | x_4^* |
| 7 | 4 | 9 |
| 8 | 5 | 9 |

Tahap 3 :

$$f_k(s) = \min \{ c_{s x_k} + f_{k-1}(x_k) \}; k=1,2,3$$

$$= f_k(s, x_k)$$

| x ₃ s | f ₃ (s, x ₃) | | Solusi Optimum | |
|---------------------|-------------------------------------|----|--------------------|------------------|
| | 7 | 8 | f ₃ (s) | x ₃ * |
| 4 | 19 | 26 | 19 | 7 |
| 5 | 22 | 25 | 22 | 7 |
| 6 | 26 | 15 | 15 | 8 |

Tahap 2 :

| x ₂ s | f ₂ (s, x ₂) | | | Solusi Optimum | |
|---------------------|-------------------------------------|----|----|--------------------|------------------|
| | 4 | 5 | 6 | f ₂ (s) | x ₂ * |
| 2 | 27 | 29 | 25 | 25 | 6 |
| 3 | 31 | 31 | 27 | 27 | 6 |

Tahap 1 :

| x ₁ s | f ₁ (s, x ₁) | | Solusi Optimum | |
|---------------------|-------------------------------------|----|--------------------|------------------|
| | 2 | 3 | f ₁ (s) | x ₁ * |
| 1 | 30 | 30 | 30 | 2 atau 3 |

Solusi optimum dengan program dinamis mundur dapat dilihat pada table di bawah ini :

| | x1 | x2 | x3 | x4 | Waktu optimum |
|---|----|----|----|----|---------------|
| 1 | 2 | 6 | 8 | 9 | 30 |
| | 3 | | | | 30 |

Jadi, ada dua rute dengan waktu yang optimal, yaitu :

$$1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 9$$

$$1 \rightarrow 3 \rightarrow 6 \rightarrow 8 \rightarrow 9$$

Waktu yang dibutuhkan untuk mencapai tujuan melalui kedua rute tersebut sama, yaitu 30 menit.

B. Penyelesaian dengan Program Dinamis Maju

Misalkan x₁, x₂, x₃, dan x₄ adalah simpul – simpul yang dikunjungi pada tahap k, dengan k=1,2,3,4. Maka rute yang dilalui adalah x₁ → x₂ → x₃ → x₄ → 9, yang dalam hal ini x₁ = 1.

Relasi rekurens berikut menyatakan rute dengan waktu optimal dari status x₁ ke s pada tahap k :

$$f_1(s) = c_{s x_1} \quad (\text{basis})$$

$$f_k(s) = \min \{ c_{s x_k} + f_{k-1}(x_k) \}; k=2,3,4 \quad (\text{rekurens})$$

$$= f_k(s, x_k)$$

Program dinamis mundur bertujuan untuk mendapatkan f₄(9) dengan cara mencari f₁(s), f₂(s), f₃(s) terlebih dahulu.

Tahap 1 :

$$f_1(s) = c_{s x_1}$$

| s | Solusi Optimum | |
|---|--------------------|------------------|
| | f ₁ (s) | x ₁ * |
| 2 | 5 | 1 |
| 3 | 3 | 1 |

Tahap 2 :

$$f_k(s) = \min \{ c_{s x_k} + f_{k-1}(x_k) \}; k=2,3,4$$

$$= f_k(s, x_k)$$

| x ₂ s | f ₂ (s, x ₂) | | Solusi Optimum | |
|---------------------|-------------------------------------|----|--------------------|------------------|
| | 2 | 3 | f ₂ (s) | x ₂ * |
| 4 | 13 | 15 | 13 | 2 |
| 5 | 12 | 12 | 12 | 2 atau 3 |
| 6 | 15 | 15 | 15 | 2 atau 3 |

Tahap 3 :

| x ₃ s | f ₃ (s, x ₃) | | | Solusi Optimum | |
|---------------------|-------------------------------------|----|----|--------------------|------------------|
| | 4 | 5 | 6 | f ₃ (s) | x ₃ * |
| 7 | 28 | 30 | 37 | 28 | 4 |
| 8 | 34 | 32 | 25 | 25 | 6 |

Tahap 4 :

| x ₄ s | f ₄ (s, x ₄) | | Solusi Optimum | |
|---------------------|-------------------------------------|----|--------------------|------------------|
| | 7 | 8 | f ₄ (s) | x ₄ * |
| 9 | 32 | 30 | 30 | 8 |

Solusi optimum dengan program dinamis maju dapat dilihat pada table di bawah ini :

| | x4 | x3 | x2 | x1 | Waktu optimum |
|---|----|----|----|----|---------------|
| 9 | 8 | 6 | 2 | 1 | 30 |
| | | | 3 | 1 | 30 |

Jadi, ada dua rute dengan waktu yang optimal, yaitu :

$$1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 9$$

$$1 \rightarrow 3 \rightarrow 6 \rightarrow 8 \rightarrow 9$$

Waktu yang dibutuhkan untuk mencapai tujuan melalui kedua rute tersebut sama, yaitu 30 menit.

IV. ANALISIS

Hasil pencarian solusi dengan program dinamis mundur maupun maju memberikan solusi yang sama. Namun pada umumnya, metode yang banyak digunakan untuk menyelesaikan persoalan – persoalan seperti persoalan

diatas adalah metode program dinamis mundur.

Dengan program dinamis mundur, solusi $f_4(s)$, $f_3(s)$, dan $f_2(s)$ dicari terlebih dahulu untuk mendapatkan $f_1(s)$. Berdasarkan perhitungan diatas, pada tahap 4 dihasilkan solusi waktu optimum 4 dari status 7 dan 5 dari status 8 dengan simpul yang dikunjungi berikutnya adalah simpul 9. Pada tahap 3 dihasilkan solusi waktu optimum 19 dari status 4 dengan simpul yang dikunjungi adalah simpul 7, 22 dari status 5 dengan simpul yang dikunjungi adalah simpul 7, dan 15 dari status 6 dengan simpul yang dikunjungi adalah simpul 8. Pada tahap 2 dihasilkan solusi waktu optimum 25 dari status 2 dengan simpul yang dikunjungi adalah simpul 6 dan 27 dari status 3 dengan simpul yang dikunjungi adalah simpul 6. Terakhir, pada tahap 1 dihasilkan solusi waktu optimum 30 dari status 1 dengan simpul yang dikunjungi adalah simpul 2 atau 3.

Untuk mendapatkan rute dengan solusi waktu optimum, tinjau kembali solusi – solusi optimum yang dihasilkan dari setiap tahap. Karena rute yang dilalui adalah $1 \rightarrow x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4$, maka setelah simpul 1, simpul yang dikunjungi selanjutnya adalah x_1 dengan dua kemungkinan simpul yaitu simpul 2 dan 3 yang merupakan solusi optimum pada tahap 1. Selanjutnya, simpul yang dikunjungi adalah $x_2=6$ yang merupakan solusi optimum pada tahap 2 dan dihasilkan dari status 2 dan 3. Setelah dari simpul 6, simpul yang dikunjungi berikutnya adalah $x_3=8$ yang merupakan solusi optimum pada tahap 3 dan dihasilkan dari status 6. Terakhir, simpul yang dikunjungi adalah $x_4=9$ yang merupakan solusi optimum pada tahap 4 dan dihasilkan dari status 8. Sehingga diperoleh rute $1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 9$ dan $1 \rightarrow 3 \rightarrow 6 \rightarrow 8 \rightarrow 9$ dengan waktu optimum 30 menit sebagai solusi dari persoalan *Pizza Hut Delivery*.

Dengan program dinamis maju, solusi $f_1(s)$, $f_2(s)$, dan $f_3(s)$ dicari terlebih dahulu untuk mendapatkan $f_4(s)$. Berdasarkan perhitungan yang telah dilakukan, pada tahap 1 dihasilkan solusi waktu optimum 5 dari status 1 dan 3 dari status 1 dengan simpul yang dikunjungi berikutnya adalah simpul 2 dan 3. Pada tahap 2 dihasilkan solusi waktu optimum 13 dari status 2 ke status 4, 12 dari status 2 atau 3 ke status 5, dan 15 dari status 2 atau 3 ke status 6. Pada tahap 3 dihasilkan solusi waktu optimum 28 dari status 4 ke status 7 dan 25 dari status 6 ke status 8. Pada tahap 4 dihasilkan solusi waktu optimum 30 dari status 8 ke status 9.

Untuk mendapatkan rute dengan solusi waktu optimum, tinjau kembali solusi – solusi optimum yang dihasilkan dari setiap tahap. Karena rute yang dilalui adalah $x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \rightarrow 9$, maka untuk mencapai simpul 9, simpul yang dikunjungi pertama adalah $x_1=1$ yang merupakan solusi optimum pada tahap 1. Karena simpul 1 dapat dicapai dari simpul 2 atau 3, maka simpul selanjutnya yang dikunjungi adalah $x_2=2$ atau $x_2=3$ yang merupakan solusi optimum pada tahap 2. Simpul 2 dan 3 dapat dicapai dari simpul 5 atau 6. Keputusan yang diambil adalah $x_3=6$ karena 6 merupakan solusi optimum pada tahap 3. Dari simpul 6 dapat dicapai simpul 8 yang

merupakan solusi optimum pada tahap 4, sehingga $x_4=8$. Dan dari simpul 8 dapat dicapai simpul 9. Maka diperoleh rute $1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 9$ dan $1 \rightarrow 3 \rightarrow 6 \rightarrow 8 \rightarrow 9$ dengan waktu optimum 30 menit sebagai solusi dari persoalan *Pizza Hut Delivery*.

Dengan asumsi kecepatan yang digunakan konstan, persoalan optimasi waktu *Pizza Hut Delivery* ini dapat diselesaikan dengan cukup mudah menggunakan program dinamis. Penyelesaian masalah yang dilakukan mirip dengan penyelesaian persoalan mencari lintasan terpendek. Akan tetapi, pada kenyataannya kecepatan yang digunakan tidak selalu konstan. Sehingga jika jarak tempuh ke alamat pengiriman cukup jauh dan waktu yang dibutuhkan mungkin lebih dari 30 menit, maka untuk mencapai waktu optimum, kecepatan yang digunakan harus ditambah. Oleh karena itu, pembahasan dalam makalah ini lebih cenderung pada pemilihan rute yang tepat dengan mengasumsikan kecepatan yang digunakan konstan. Dengan begitu, restoran *Pizza Hut* dapat menggunakan metode pemecahan masalah program dinamis untuk mengoptimalkan layanan *Pizza Hut Delivery*. Sehingga harapannya tidak ada konsumen yang kecewa karena keterlambatan pengiriman dan *Pizza Hut* tidak mengalami kerugian karena kehilangan pelanggan..

V. KESIMPULAN

1. Program dinamis dapat digunakan untuk menyelesaikan berbagai persoalan optimasi.
2. Persoalan optimasi waktu *Pizza Hut Delivery* yang menjamin pesanan tiba dalam waktu 30 menit dapat diselesaikan baik dengan program dinamis mundur maupun maju.
3. Penyelesaian persoalan dengan program dinamis mundur dan program dinamis maju menghasilkan solusi yang sama, yaitu rute $1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 9$ dan $1 \rightarrow 3 \rightarrow 6 \rightarrow 8 \rightarrow 9$ dengan waktu optimal 30 menit.
4. Program dinamis dapat menghasilkan solusi lebih dari satu.

REFERENSI

- [1] Rinaldi Munir, "Diktat Kuliah IF2211 Strategi Algoritma", Program Studi Teknik Informatika STEI ITB, 2009.
- [2] www.phd.co.id
Waktu akses : 17/5/2014, 09:47 AM
- [3] <http://sutrisnoadityo.wordpress.com/2013/10/12/program-dinamis/>
Waktu akses : 18/5/2014, 10:55 AM

PERNYATAAN

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

Bandung, 19 Mei 2014

A handwritten signature in black ink, appearing to be 'Riska', written in a cursive style.

Riska
13512062