BAB IV

PERANCANGAN

4.1 Deskripsi Simulasi Pencarian Rute Terpendek

Simulasi pencarian rute terpendek menggunakan algoritma *greedy* menjadi permasalahan yang akan dipecahkan dan dibahas pada perancangan sistem ini, simulasi pencarian rute terpendek berdasarkan bobot jarak antar node dan bobot waktu yang dihitung dari node asal ke node tujuan.

4.2 Perancangan Sistem

Dalam membangun suatu sistem yang akan kita buat diperlukan adanya suatu perancangan sistem. Perancangan sistem diperlukan agar kita dapat melakukan semua kegiatan sesuai dengan prosedur. Pada perancangan sistem ini akan dibuat suatu program yang dapat dijalankan pada PC maupun laptop, yang digunakan untuk simulasi pencarian rute terpendek, rute yang terpendek dilihat dari jarak dan waktu tempuh.

4.2.1 Tujuan Perancangan Sistem

Tujuan perancangan program simulasi ini adalah untuk menghasilkan jalur terpendek dari rute jalan yang akan ditempuh. Dengan demikian pembuatan program simulasi ini diharapkan dapat membantu mencari rute terpendek dan menghasilkan bobot jarak dan bobot waktu.

4.2.2 Gambaran Umum Sistem

Gambaran umum sistem ini bertujuan untuk menghasilkan perancangan program pencarian rute terpendek menggunakan algoritma *greedy* (simulasi rute angkot Cicaheum Ciroyom) dengan tujuan dari terminal Cicaheum ke jalan Dipatiukur. Perancangan program simulasi ini akan menampilkan jarak terpendek dari node awal ke node tujuan yang akan dicari. Jarak terpendek rute diambil dari bobot jarak dan bobot waktu terkecil.

4.3 Implementasi Algoritma Greedy Pada Simulasi Pencarian Rute Terpendek Angkot Cicaheum Ciroyom

Kegunaan utama pada simulasi pencarian rute terpendek angkot Cicaheum – Ciroyom adalah untuk mengetahui jarak terpendek dari jalur yang akan ditempuh. Perancangan simulasi pencarian rute terpendek ini tidak terlepas dari peran algoritma sebagai salah satu bentuk pemecahan masalah dari pencarian rute terpendek.

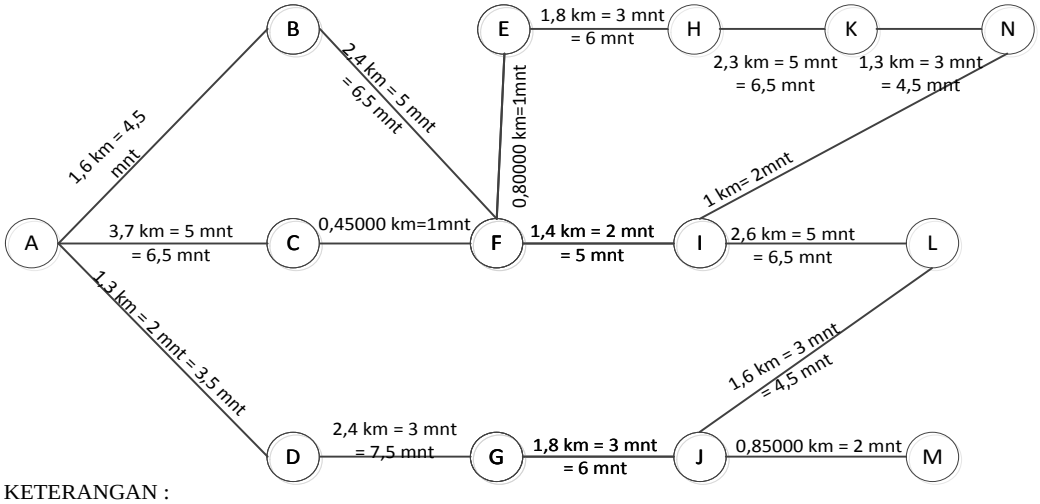
Algoritma yang di implementasikan dalam pencarian rute terpendek ini adalah algoritma *greedy* . Algoritma ini merupakan algoritma yang membentuk solusi langkah per langkah. Pada langkah tersebut akan dipilih keputusan yang paling optimal. Keputusan tersebut tidak perlu memperhatikan keputusan selanjutnya yang akan diambil, dan keputusan tersebut tidak dapat diubah lagi pada langkah selanjutnya. Pada setiap langkah dalam algoritma greedy diambil keputusan yang paling optimal untuk langkah tersebut tanpa memperhatikan konsekuensi pada langkah selanjutnya.

4.3.1 Pengembangan Cara Kerja Umum Algorima Greedy

Dalam pencarian rute terpendek algoritma greedy akan mencari solusi langkah per langkah. Algoritma greedy akan membangkitkan simpul berikutnya dari sebuah simpul (yang sejauh ini terbaik diantara semua pasangan node) yang pernah dibangkitkan. Penentuan simpul terbaik dilakukan dengan menggunakan informasi berupa waktu perkiraan dari suatu simpul menuju ke tujuan. Pada setiap langkah akan diambil bobot waktu yang paling minimum dengan membandingkan simpul yang terhubung dari node awal ke node tujuan.

4.3.2 Data Dalam Graf

Gambar 4.1 merupakan penerapan graf tidak berarah dengan penempatan node beserta bobotnya.



KETERANGAN :

A : Terminal Cicaheum
B : Antapani
C : Jl. Ahmad Yani
D : Jl. K.H Hasan Mustopa
E : Jl. Sukabumi
F : Jl. Jakarta
G: Jl. Surapati
H: Jl. Laswi
I : Jl. Supratman
J : Jl. Panata Yuda
K : Jl. Riau
L : Jl. Dipenogoro
M: Jl. Dipatiukur
N: Jl. Taman Pramuka

Gambar 4.1 Rute jalan dari terminal Cicaheum ke Dipatiukur

Dalam tabel 4.1 menunjukan bobot antar node yang tersambung ke node lainnya dengan mengacu pada gambar 4.1

Tabel 4.1

Nilai bobot

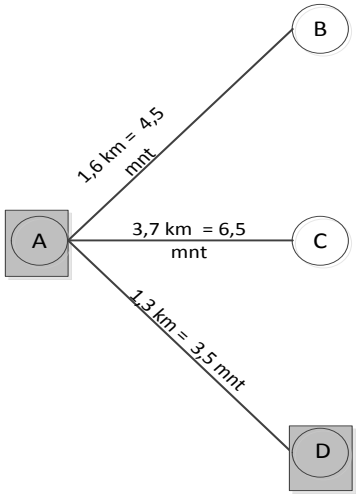
Node	Link ke Node	Bobot Jarak	Bobot Waktu
A (Terminal Cicaheum)	B (Antapani)	1,6 km	4,5 mnt
A (Terminal Cicaheum)	C (Jl. Ahmad Yani)	3,7 km	5 mnt
A (Terminal Cicaheum)	D (Jl. KH. Hasan Mustofa)	1,3 km	3,5 mnt
B (Antapani)	F (Jl. Jakarta)	2,4 km	6,5 mnt
C (Jl. Ahmad Yani)	F (Jl. Jakarta)	0,45000 km	2,5 mnt
D (Jl. KH. Hasan Mustofa)	G (Jl. Surapati)	2,4 km	7,5 mnt
E (Jl. Sukabumi)	F (Jl. Jakarta)	0,80000 km	1 mnt
E (Jl. Sukabumi)	H (Jl. Laswi)	1,8 km	6 mnt
F (Jl. Jakarta)	E (Jl. Sukabumi)	0,80000 km	1 mnt
F (Jl. Jakarta)	I (Jl. Supratman)	1,4 km	5 mnt
G (Jl. Surapati)	J (Jl. Panata Yuda)	1,8 km	4,5 mnt
H (Jl. Laswi)	K (Jl. Riau)	2,3 km	6,5 mnt
I (Jl. Supratman)	N (Jl. Taman Pramuka)	1 km	2 mnt
I (Jl. Supratman)	L (Jl. Dipenogoro)	2,6 km	6,5 mnt
J (Jl. Panata Yuda)	L (Jl. Dipenogoro)	1,6 km	4,5 mnt
J (Jl. Panata Yuda)	M (Jl. Dipatiukur)	0,85000 km	2 mnt
K (Jl. Riau)	N (Jl. Taman Pramuka)	1,3 km	4,5 mnt
L (Jl. Dipenogoro)	I (Jl. Supratman)	2,6 km	6,5 mnt
L (Jl. Dipenogoro)	J (Jl. Panata Yuda)	1,6 km	4,5 mnt
M (Jl. Dipatiukur)	J (Jl. Panata Yuda)	0,85000 km	2 mnt
N (Jl. Taman Pramuka)	K (Jl. Riau)	1,3 km	4,5 mnt
N (Jl. Taman Pramuka)	I (Jl. Supratman)	1 km	2 mnt

(*Sumber : Google Maps*)

Berikut adalah langkah-langkah dalam menyeleseikan masalah jalur angkot Cicaheum Ciroyom yang terdapat pada gambar 4.1 dengan menggunakan algoritma greedy untuk mencari rute terpendek. Dengan rute awal Terminal Cicaheum ke jalan Dipatiukur.

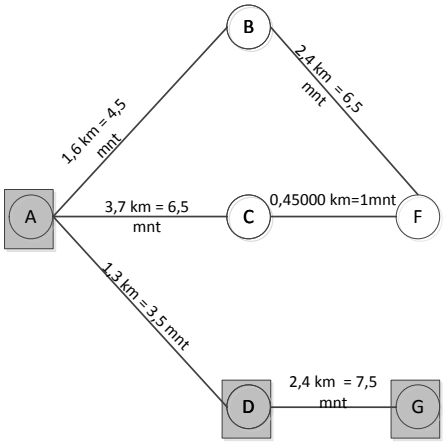
a. *Greedy* Terhadap Jarak

Pada langkah pertama, ada 3 simpul yang bisa dipilih dari simpul awal A, yaitu B,C, dan D. Pada saat menemui persimpangan algoritma greedy akan mengecek simpul manakah yang paling menguntungkan. Ternyata simpul yang paling kecil adalah B,C, dan D. Algoritma akan memilih D.



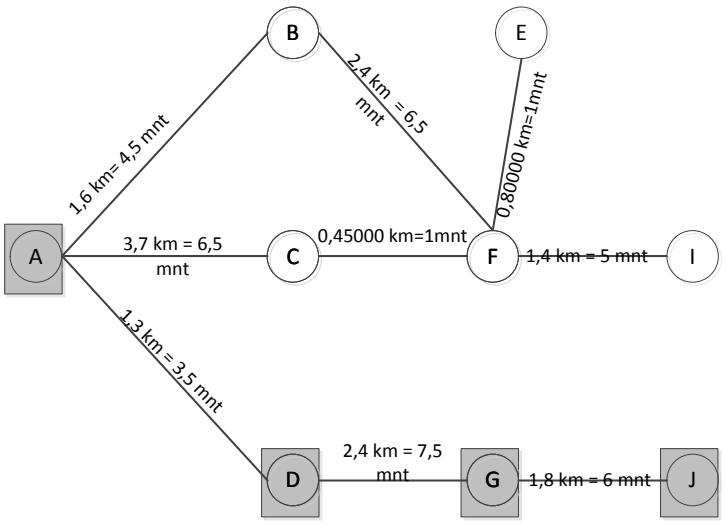
Gambar 4.2 Langkah 1 *Greedy* terhadap jarak

Pada langkah ke-2, setelah mengambil simpul D, maka kemudian algoritma akan memeriksa, simpul apakah yang ada di D. Ternyata simpul yang terhubung dengan simpul D hanya simpul G, maka simpul G terpilih sebagai simpul selanjutnya.



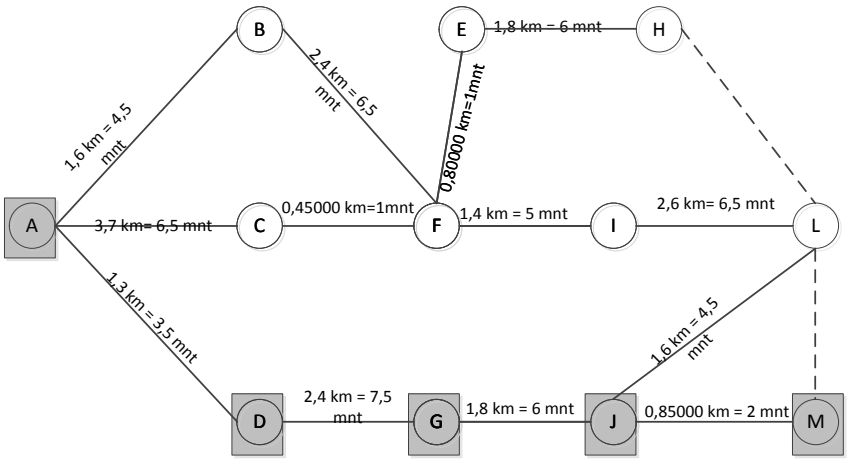
Gambar 4.3 Langkah ke-2 *Greedy* terhadap jarak

Pada langkah ke-3, sama seperti sebelumnya, dilakukan pengecekan. Ternyata simpul G hanya terhubung ke simpul J, maka simpul J terpilih sebagai simpul selanjutnya.



Gambar 4.4 Langkah ke-3 *Greedy* terhadap jarak

Pada langkah ke-4 setelah mengambil simpul J, algoritma akan memeriksa, simpul apakah yang ada di J. Setelah diperiksa simpul M dipilih, maka sampai pada tujuan.



Gambar 4.5 Langkah ke-4 *Greedy* terhadap jarak

Jadi dengan menggunakan algoritma greedy terhadap jarak didapatkan jalur A→D→G→J→M. Jalur yang ditempuh adalah Terminal Cicaheum → Jl. K.H. Hasan Mustopa → Jl. Surapati → Jl. Panata Yuda → Jl. Dipatiukur.

Estimasi jarak dan waktu angkotnya adalah :

Jarak = AD + DG + GJ +JM

= 1,3 + 2,4 + 1,8 + 0,85000

= 6,35 km

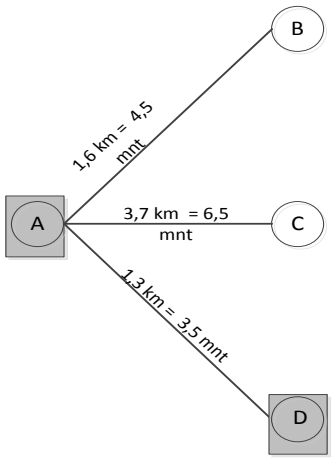
Waktu = AD + DG + GJ +JM

= 3,5 + 7,5 + 6 + 2

= 19 menit

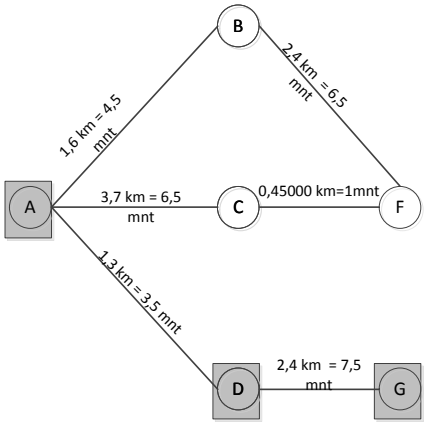
b. *Greedy* terhadap waktu

Dalam mencari rute terpendek terhadap waktu langkah-langkahnya sama dengan menentukan greedy terhadap jarak diatas, dipilih simpul yang paling menguntungkan. Pada langkah ke-1, ada 3 simpul yang bisa dipilih dari simpul awal A, yaitu B,C, dan D. Pada saat menemui persimpangan algoritma greedy akan mengecek simpul manakah yang paling menguntungkan. Ternyata simpul yang paling kecil adalah B,C, dan D. Algoritma akan memilih D.



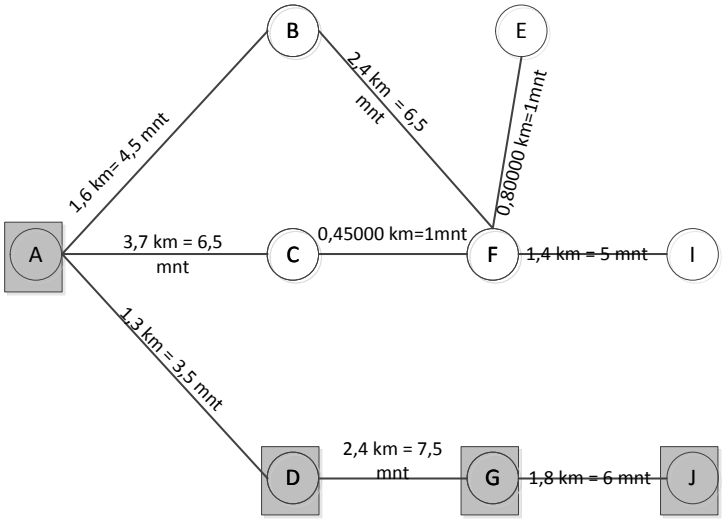
Gambar 4.6 Langkah ke-1 *greedy* terhadap waktu

Pada langkah ke-2, setelah mengambil simpul D, maka kemudian algoritma akan memeriksa, simpul apakah yang ada di D. Ternyata simpul yang terhubung dengan simpul D hanya simpul G, maka simpul G terpilih sebagai simpul selanjutnya.



Gambar 4.7 Langkah ke-2 *greedy* terhadap waktu

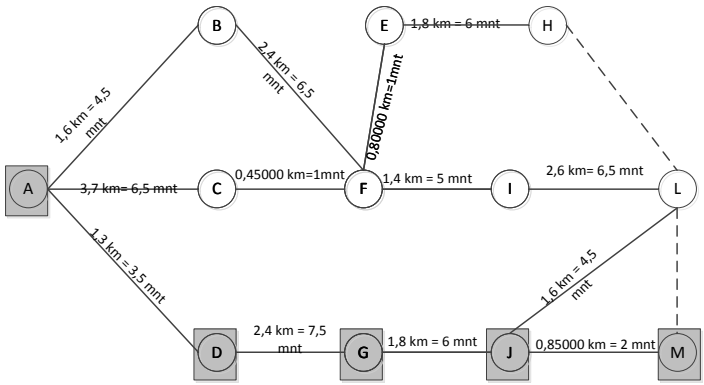
Pada langkah ke-3, sama seperti sebelumnya, dilakukan pengecekan bobot waktu yang paling minimum. Ternyata simpul G hanya terhubung ke simpul J, maka simpul J terpilih sebagai simpul selanjutnya.



Gambar 4.8 Langkah ke-3 *greedy* terhadap waktu

Pada langkah ke-4 setelah mengambil simpul J, algoritma akan memeriksa, simpul apakah yang ada di J. Setelah diperiksa simpul M dipilih,

karena memiliki bobot waktu yang paling dibandingkan simpil L maka sampai pada tujuan.



Gambar 4.9 Langkah ke-4 *greedy* terhadap jarak

Jadi dengan menggunakan algoritma greedy terhadap waktu didapatkan jalur A→D→G→J→M. Jalur yang ditempuh adalah Terminal Cicaheum → Jl. K.H. Hasan Mustopa → Jl. Surapati → Jl. Panata Yuda → Jl. Dipatiukur.

Estimasi waktu dan jarak angkotnya adalah :

Waktu = AD + DG + GJ +JM

= 3,5 + 7,5 + 6 + 2

= 19 menit

Jarak = AD + DG + GJ +JM

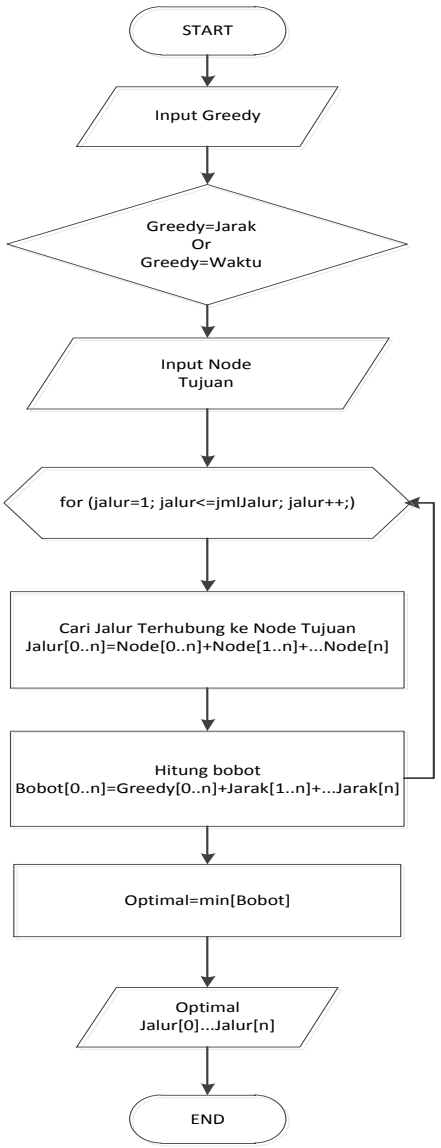
= 1,3 + 2,4 + 1,8 + 0,85000

= 6,35 km

4.4 Perancangan Prosedur

Proses perancangan ini merupakan tahap awal dari perancangan simulasi pencarian rute terpendek sebagai pemecahan masalah untuk mencari rute terpendek terhadap jarak dan waktu pada angkot Cicaheum-Ciroyom Kota Bandung menggunakan algoritma greedy.

4.4.1 Flowchart



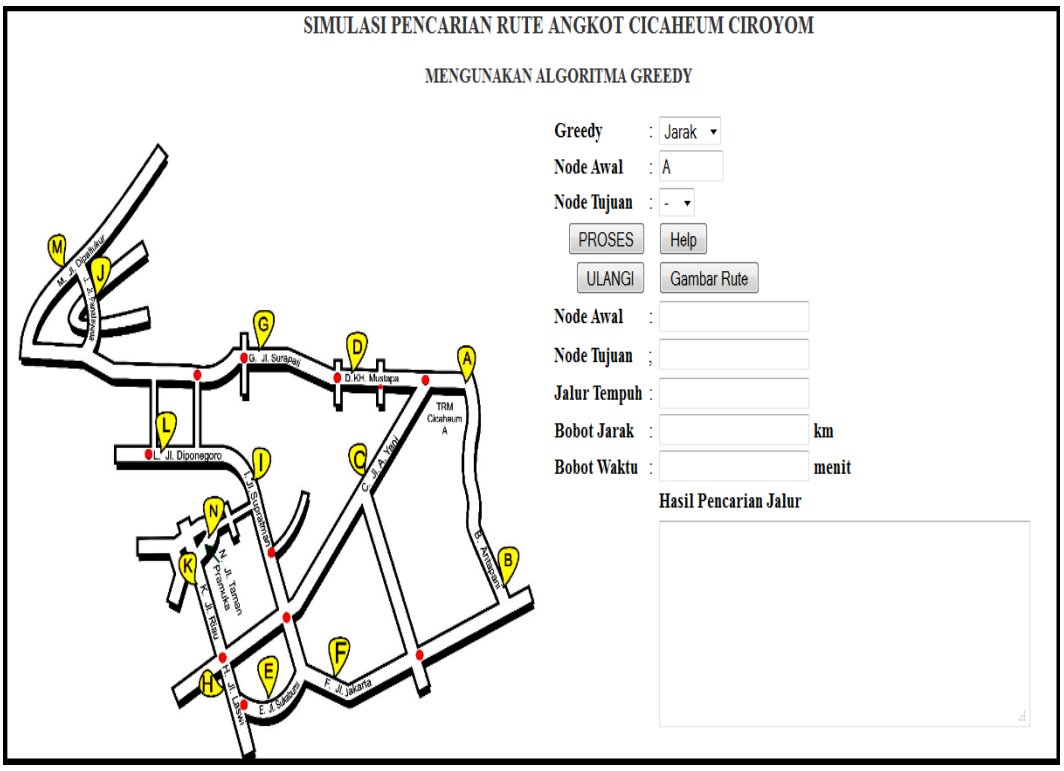
Gambar 4.10 Flowchart Algoritma untuk mencari jarak dan waktu yang optimal

4.5 Perancangan Antar Muka

Program dapat didefinisikan sebagai kumpulan dari intruksi-intruksi atau perintah-perintah terperinci yang sudah disiapkan oleh komputer sehingga dapat melakukan fungsi sesuai dengan yang telah ditentukan. Tujuan dari pembuatan program ini adalah untuk mencari rute terpendek dengan bobot jarak dan waktu.

4.5.1 Perancangan Input dan Output Sistem

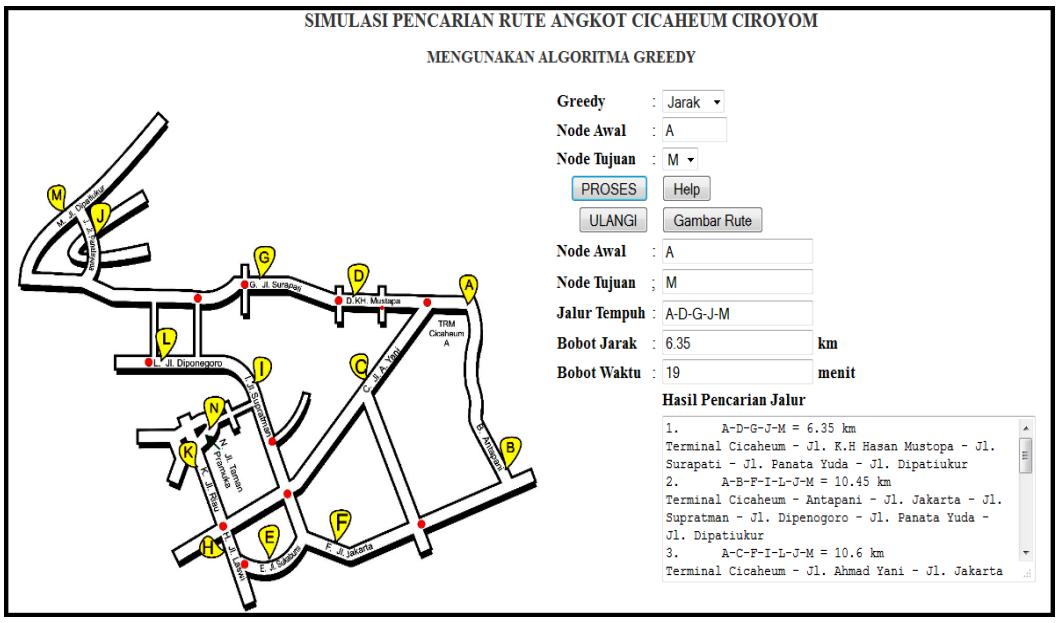
Perancangan input merupakan dimulainya suatu proses informasi. Dalam perancangan input ini, data yang dimasukan akan mempengaruhi hasil yang ditampilkan. Adapun perancangan-perancangan input yang ada dalam perancangan simulasi pencarian rute terpendek ini terdapat pada gambar 4.11



Gambar 4.11 Tampilan Input Pencarian Rute Terpendek

Pada gambar 4.11 terlihat tampilan input simulasi pencarian rute terpendek yang dirancang oleh penulis, pada tampilan di atas pun di implementasikan pada perangkat lunak macromedia dreamwaver agar perancangan tersebut sesuai dengan apa yang telah dirancangan sebelumnya.

Sedangkan untuk output dari simulasi pencarian rute terpendek ini nantinya akan didapatkan informasi rute terpendek berdasarkan greedy terhadap jarak atau waktu, beserta informasi node awal, node tujuan, jalur tempuh , bobot jarak, dan bobot waktu. Dari output simulasi pencarian rute terpendek dapat dilihat hasil jalur pencarian dan gambar graph rute tujuan. Adapun tampilannnya seperti gambar 4.12 :



Gambar 4.12 Output Simulasi Pencarian Rute Terpendek