

## **ALGORITMA DIJKSTRA DALAM PENETUAN RUTE TERPENDEK PADA JALAN RAYA ANTAR KOTA JAKARTA - TANGERANG**

**Niki Awalloedin<sup>1\*</sup>, Windu Gata<sup>2</sup>, dan Nurul Qomariyah<sup>3</sup>**

<sup>1,2,3</sup>Ilmu Komputer, STMIK Nusa Mandiri  
[niq.awal@gmail.com](mailto:niq.awal@gmail.com)\*

### **Abstrak**

Perubahan kondisi jalan saat ini semakin padat dengan meningkatnya perkembangan banyaknya kendaraan, pemerintah kota Jakarta telah melakukan beberapa cara untuk mengatasi masalah kemacetan seperti peraturan ganjil genap untuk kendaraan bermotor, pelebaran jalan, pembangunan *underpass* dan *flyovers*, tetapi hal tersebut masih dirasa kurang efektif, sehingga pemilihan rute / jarak waktu tercepat dari kota Jakarta ke kota Tangerang menjadi suatu permasalahan. Dengan perkembangan berbagai algoritma saat ini, penulis mencoba melakukan penelitian pencarian rute terpendek antar kota jakarta (titik awal Gedung Walikota Jakarta Pusat) ke Kota Tangerang (titik akhir Gedung Walikota Tangerang) dengan menggunakan algoritma djikstra, dengan harapan akan memberikan informasi rute tercepat atau jalur mana yang dapat ditempuh dengan jarak yang terpendek jika melalui jalan raya antara kota Jakarta dan kota Tangerang. Dimana hasil yang didapat bahwa algoritma djikstra dapat digunakan untuk mencari rute terpendek sesuai dengan perkiraan perhitungan pada tabel 4 rute 3, dengan jarak tempuh total 22,6km.

**Kata Kunci:** *Algoritma Dijkstra, Rute terpendek.*

### **Abstract**

Changes in road conditions are currently getting denser with the increasing development of many vehicles, the city government of Jakarta has taken several ways to solve congestion problems such as odd-even regulations for motor vehicles, road widening, construction of underpasses and flyovers, but this is still deemed ineffective, so the election the fastest route / distance from Jakarta to Tangerang is a problem. With the current development of various algorithms, the author tries to do research to find the shortest route between Jakarta cities (the starting point of the Central Jakarta Mayor's Building) to Tangerang City (the end point of the Tangerang Mayor's Building) using the djikstra algorithm, with the hope that it will provide information on the fastest route or which route. which can be reached with the shortest distance if via the highway between the city of Jakarta and the city of Tangerang. Where the results obtained show that the Dijkstra algorithm can be used to find the shortest route according to the estimated calculations in table 4 route 3, with a total distance of 22.6 km.

**Keywords:** *Dijkstra Algorithm, Shortest Path*

## 1. PENDAHULUAN

Perubahan kondisi jalan yang semakin hari semakin padat dan minimnya sarana dan prasana jalan yang mendukung mobilitas yang tinggi pengguna jalan kota besar seperti Jakarta dan Tangerang membuat sulit pengguna jalan dalam melakukan aktifitas sehari-hari, kemacetan menjadi pemandangan yang biasa terlihat pada pagi hari jam berangkat kerja dan pada waktu sore jam pulang kerja. Hal ini dirasa sangat merugikan untuk semua orang karena banyaknya waktu yang terbuang di jalan karena kemacetan, belum lagi biaya bahan bakar yang dihabiskan jika terjebak dalam kemacetan (Cindy, 2015).

Kota Tangerang memiliki andil penting dalam jalur transportasi darat dan aktivitas pendistribusian logistik dari Jakarta (Pelabuhan) maupun sebaliknya. penduduk kota Tangerang sebagian bekerja di kota Jakarta bekerja. dengan meningkatnya mobilitas penduduk akan menyebabkan peningkatan arus lalu lintas. Peningkatan arus lalu lintas menyebabkan masalah kemacetan.

Tangerang sebagai penyokong kota Jakarta memberikan mobilitas pergerakan orang dan kendaraan yang tinggi baik pekerja dan distribusi logistik atau barang yang melalui rute jalan raya antar kota Jakarta – Tangerang, hal ini penulis angkat menjadi topik penelitian dalam menentukan rute atau jalan terdekat yang dapat dilalui oleh pengguna jalan raya antar kota Jakarta – Tangerang.

Rute / jalan yang akan diteliti digambarkan dalam bentuk graf berarah dan berbobot untuk mempermudah interpretasi jarak antara ruas jalan yang satu dengan ruas jalan yang lainnya setelah dilakukan interpretasi ruas jalan dalam bentuk graf (Rifanti, 2017).

Graph adalah struktur diskrit yang terdiri dari adanya simpul (vertex) dan adanya sebuah sisi (edge), graph adalah pasangan himpunan  $(V, E)$  dimana  $V$  merupakan sebuah himpunan yang tidak kosong dari sebuah vertex dan  $E$  adalah himpunan sisi yang menghubungkan sepasang simpul dalam graph tersebut (Harahap & Khairina, 2017).

Algoritma Dijkstra berdasarkan teori graph dimana secara umum graf dibedakan

atas dua jenis sebagai berikut :

### 1. Graf tidak berarah (undirected graph)

Graf yang sisinya tidak memiliki orientasi arah disebut graf tidak berarah. Pada graf tidak berarah, urutan pasangan simpul yang dihubungkan oleh sisi tidak diperhatikan. Jadi,  $(u, v) = (v, u)$  adalah sisi yang sama.

### 2. Graf berarah (directed graph)

Graf yang setiap sisinya diberikan orientasi arah disebut graf berarah, pada graf berarah  $(u, v)$  dan  $(v, u)$  menyatakan dua buah sisi yang berbeda. Dengan kata lain dapat ditulis  $(u, v) \neq (v, u)$ .

Dalam mencari lintasan terpendek dari suatu simpul ke semua pasangan simpul algoritma Dijkstra melalui sejumlah langkah yang menggunakan prinsip greedy. Prinsip greedy pada algoritma Dijkstra menyatakan bahwa pada setiap langkah kita memilih sisi yang berbobot minimum dan memasukkannya dalam himpunan solusi (Rinaldi, 2012).

Algoritma ini bertujuan untuk menemukan jalur terpendek berdasarkan bobot terkecil dari satu titik ke titik lainnya. Misalkan titik menggambarkan gedung dan garis menggambarkan jalan, maka algoritma Dijkstra melakukan kalkulasi terhadap semua kemungkinan bobot terkecil dari setiap titik (Noviriandini & Safitri, 2017).

Selain itu penulis menggunakan Google Maps dalam mengambil data perhitungan jarak antar titik jalan yang sudah ditentukan. *Google maps* adalah aplikasi layanan gratis yang disediakan oleh perusahaan Google yang dikhususkan untuk melayani kebutuhan peta tidak hanya itu banyak sekali fitur-fitur yang terdapat pada peta yang diberikan seperti lokasi saat ini dengan (*GPS*), lokasi wisata, tempat makan dan juga perhitungan jarak antar tempat, untuk penelitian ini penulis menggunakan fitur perhitungan jarak menggunakan *Google Maps* yang dituangkan pada tabel rute dibawah.

## 2. METODE PENELITIAN

Penulis menggunakan dua buah metode dalam penelitian ini yaitu:

### a. Metode Pengamatan Langsung (*Observation*)

Untuk penelitian ini penulis mengambil data perhitungan dari google maps dan

menentukan titik-titik / node yang dilalui pada rute, data jarak yang di ambil pada *Google Maps* dalam satuan Meter disini penulis merubah satuan tersebut menjadi Kilometer (km). Jumlah rute yang diambil sebanyak 8 rute dengan jumlah total titik / node ada 20 titik.

- b. Metode Studi Pustaka (*Library Research*). Dalam penelitian penulis mengutip pendapat beberapa ahli dan nara sumber dalam artikel, buku / jurnal untuk memperkuat landasan teori.

### 3. HASIL DAN PEMBAHASAN

Langkah pertama yang dilakukan dalam penelitian ini adalah menentukan titik awal (Gedung Walikota Jakarta Pusat) dan titik akhir (Gedung Walikota Tangerang), selain itu tetukan node / titik lainnya yang akan dilalui darri titik awal ke titik akhir yang dijadikan dasar perhitungan rute seperti dalam tabel berikut:

Tabel 1. Koordinat Node/Titik

	Node / Titik	Koordinat
A	Gedung Walikota Jakarta Pusat	-6.172888715324132, 106.81973537875858
B	Pertigaan Jl. Tanah Abang 1	-6.173006048232146, 106.82155928076725
C	Perempatan Jl. Harmoni	-6.167768090476439, 106.82081149691832
D	Perempatan JL.Tomang	-6.179426701216781, 106.79647849675335
E	Tol JKT-TGR Fly Over Kb. Jeruk	-6.189899056784433, 106.76891695516191
F	Tol JKT-TGR Fly Over Meruya	-6.191498997564021, 106.73187028557426
G	Tol JKT-TGR Fly Over Metland	-6.203567726254221, 106.70435082441496
H	Jl. MH Thamrin, Kb. Nanas	-6.217683002838499, 106.63163550302913
I	Fly Over Jl. Jend. Sudirman - Jl. KH Hasyim Ashari	-6.185259201656089, 106.6430864892802
J	Pertigaan Jl. Jend. Sudirman - Jl. Benteng Betawi	-6.175604445395482, 106.64453659650975
K	Pertigaan Jl. Jend. Sudirman - Gerbang Walikota	-6.171242393255522, 106.64399512246715
L	Gedung Walikota Tangerang	-6.170551598784927, 106.64040501642599
M	Pertigaan Jl. KH. Hasyim Ashari	-6.1647493539112075, 106.81989804701848
N	Fly Over Grogol, Jl. Letjen S. Parman	-6.1668971214874615,

	Node / Titik	Koordinat
		106.78869353855056
O	Pertigaan Jl. Daan Mogot - Jl. Panjang	-6.158538983974811, 106.76303926800094
P	Fly Over Cengkareng, Jl. Daan Mogot	-6.154347541339528, 106.7280809797582
Q	Pertigaan Jl. Daan Mogot - Jl. Maulana Hasanudin	-6.161691049178048, 106.68140891853263
R	Pertigaan Jl. Jend. Sudirman - Jl.Daan Mogot	-6.167807957190466, 106.64383544984942
S	Pertigaan Jl. KH Hasyim Ashari, Situ Cipondoh	-6.193994894999654, 106.6754956203187
T	Pertigaan Jl. Benteng Betawi - Jl. Maulana Hasanudin	-6.170555391779686, 106.67950284087502

Dari node / titik tersebut dilakukan perhitungan jarak antara node / titik, data yang diambil dari *Google Maps* dalam satuan Meter dirubah menjadi satuan Kilometer, terlihat pada tabel dibawah:

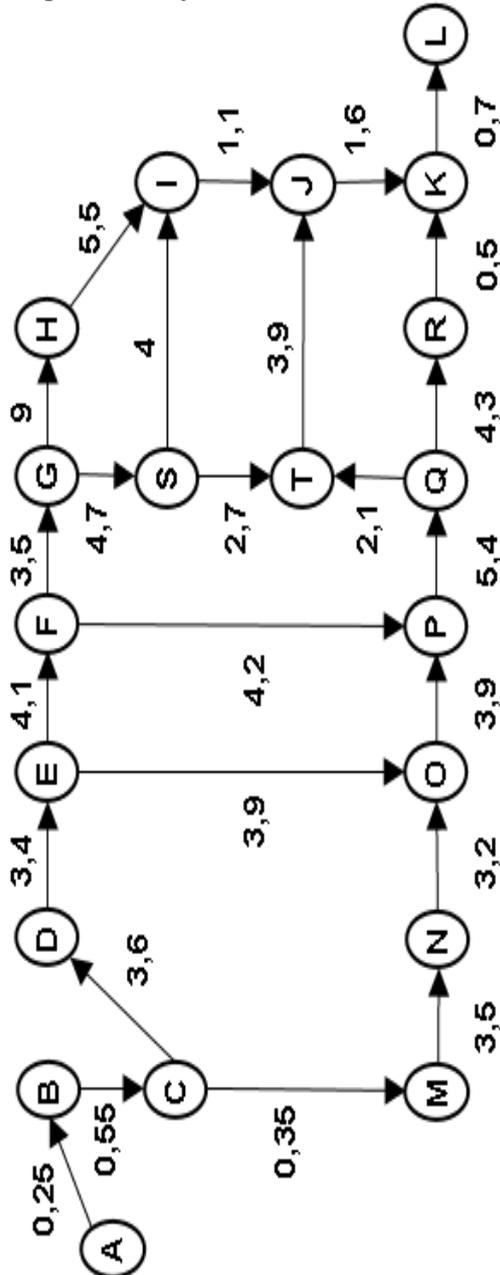
TABEL 2. Jarak Node/Titik

No	Node	Jarak	
1	A - B	250 meter	0,25 Km
2	B - C	550 meter	0,55 Km
3	C - D	3600 meter	3,6 Km
4	D - E	3400 meter	3,4 Km
5	E - F	4100 meter	4,1 Km
6	F - G	3500 meter	3,5 Km
7	G - H	9000 meter	9 Km
8	H - I	5500 meter	5,5 Km
9	I - J	1100 meter	1,1 Km
10	J - K	1600 meter	1,6 Km
11	K - L	700 meter	0,7 Km
12	C -M	350 meter	0,35 Km
13	M - N	3500 meter	3,5 Km
14	N - O	3200 meter	3,2 Km
15	O - P	3900 meter	3,9 Km
16	P - Q	5400 meter	5,4 Km
17	Q - R	4300 meter	4,3 Km
18	R - K	500 meter	0,5 Km
19	E - O	3900 meter	3,9 Km
20	F - P	4200 meter	4,2 Km
21	G - S	4700 meter	4,7 Km
22	S -T	2700 meter	2,7 Km
23	Q - T	2100 meter	2,1 Km
24	S - I	4000 meter	4 Km
25	T - J	3900 meter	3,9 Km

Setelah mengukur jarak antar titik koordinat langkah kedua adalah proses dari data rute antar kota yang telah dijabarkan pada tabel diatas dan dibuat graf berarah sesuai jalur jalan dan berikan bobot nilai pada masing-masing sisi sesuai dengan nilai jarak, seperti pada gambar 1.

Pada Gambar 1 dapat diinpertasikan dalam graf berarah dan mempunyai bobot

dengan titik merupakan titik persimpangan jalan dan sisi ruas jalan yang menghubungkan antar persimpangan, bobot pada sisi adalah interpretasi dari jarak dalam satuan Kilometer.



Gambar. 1. Graf rute Jakarta -Tangerang

Dari graf rute Jakarta -Tangerang diatas dapat kita jabarkan dalam bentuk matrik ketetanggaan seperti terlihat pada tabel 3.

Dari sini langkah selanjutnya berdasarkan gambar graf dan tabel matrik ketetanggaan, dapat kita simpulkan ada 8 rute yang bisa ditempuh masing-masing mempunyai bobot nilai jarak antar titik / node, terlihat pada tabel 4.

TABEL 3. Matrik Ketetanggaan

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
A	∞	0,25	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
B	0,25	∞	0,55	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
C	∞	0,55	∞	3,6	∞	∞	∞	∞	∞	∞	∞	∞	∞	0,35	∞	∞	∞	∞	∞	∞
D	∞	∞	3,6	∞	3,4	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
E	∞	∞	∞	3,4	∞	4,1	∞	∞	∞	∞	∞	∞	∞	∞	∞	3,9	∞	∞	∞	∞
F	∞	∞	∞	∞	4,1	∞	3,5	∞	∞	∞	∞	∞	∞	∞	∞	∞	4,2	∞	∞	∞
G	∞	∞	∞	∞	∞	3,5	∞	9	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	4,7
H	∞	∞	∞	∞	∞	9	∞	5,5	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
I	∞	∞	∞	∞	∞	∞	5,5	∞	1,1	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	4
J	∞	∞	∞	∞	∞	∞	∞	1,1	∞	1,6	∞	∞	∞	∞	∞	∞	∞	∞	∞	3,9
K	∞	∞	∞	∞	∞	∞	∞	∞	1,6	∞	0,7	∞	∞	∞	∞	∞	∞	∞	0,5	∞
L	∞	∞	∞	∞	∞	∞	∞	∞	∞	0,7	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
M	∞	∞	0,35	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	3,5	∞	∞	∞	∞	∞	∞
N	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	3,5	∞	3,2	∞	∞	∞	∞	∞
O	∞	∞	∞	∞	3,9	∞	∞	∞	∞	∞	∞	∞	∞	3,2	∞	3,9	∞	∞	∞	∞
P	∞	∞	∞	∞	∞	4,2	∞	∞	∞	∞	∞	∞	∞	∞	∞	3,9	5,4	∞	∞	∞
Q	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	5,4	4,3	∞	2,1
R	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0,5	∞	∞	∞	∞	∞	4,3	∞	∞	∞
S	∞	∞	∞	∞	∞	4,7	∞	∞	4	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	2,7
T	∞	∞	∞	∞	∞	∞	∞	∞	∞	3,9	∞	∞	∞	∞	∞	∞	∞	2,1	∞	2,7

TABEL 4. Jarak Rute

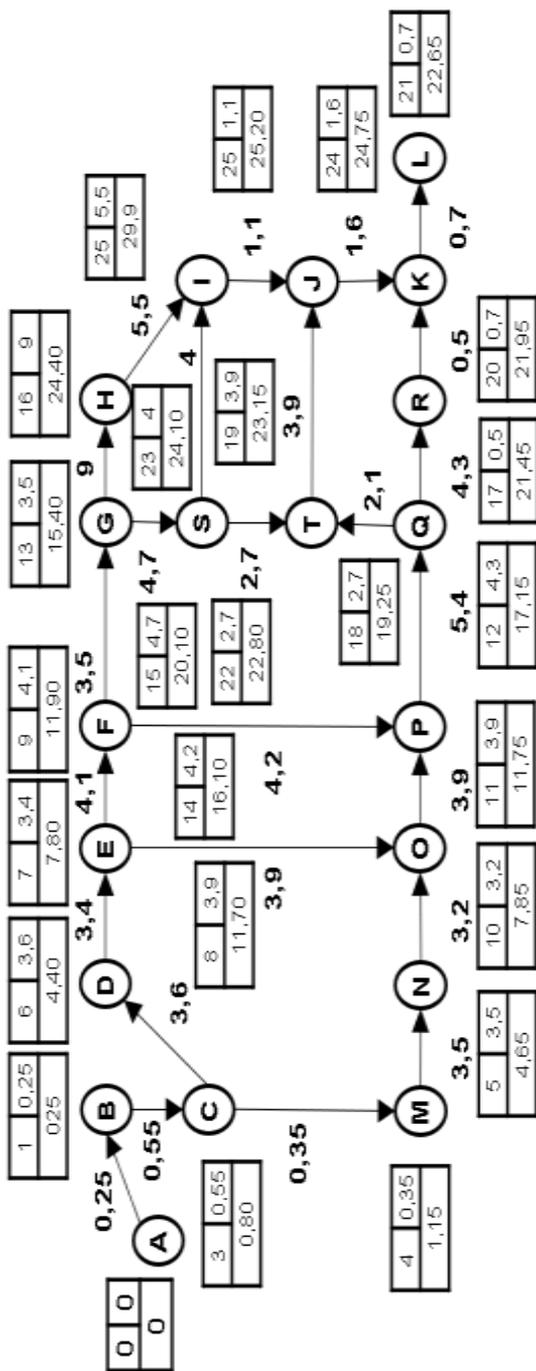
No	Rute	Jarak /Km
1	A-B-C-D-E-F-G-H-I-J-K-L	33,3
2	A-B-C-D-E-O-P-Q-T-J-K-L	29,3
3	A-B-C-D-E-O-P-Q-R-K-L	22,6
4	A-B-C-D-E-F-P-Q-T-J-K-L	29,8
5	A-B-C-D-E-F-P-Q-R-K-L	27
6	A-B-C-D-E-F-G-S-I-J-K-L	27,5
7	A-B-C-D-E-F-G-S-T-J-K-L	29
8	A-B-C-M-N-O-P-Q-R-K-L	22,65

Pada tabel diatas rute nomor 3 (A-B-C-D-E-O-P-Q-R-K-L) yaitu mempunyai jarak 22,6 Km, adalah merupakan rute terdekat.

Sementara rute nomor 1 (A-B-C-D-E-F-G-H-I-J-K-L) mempunyai jarak yang paling jauh yaitu 33,3 Km. Dan dapat kita urutkan sebagai alternatif rute terdekat ada adalah rute nomor 8 (A-B-C-M-N-O-P-Q-R-K-L) dengan jarak 22,65 dan selanjutnya rute nomor 5 (A-B-C-D-E-F-P-Q-R-K-L) dengan jarak 27 Km.

Langkah pertama adalah periksa jarak yang terdekat dari node A ke node yang paling dekat yaitu node B. jika sudah tidak ada node yang lain langkah dimulai dari node B.

Dari node B periksa kembali node yang paling dekat yaitu node C, jika tidak ada node lain selain node C maka langkah berlanjut dimulai dari node C.



Gambar. 2. Graf langkah proses algoritma dijkstra

Pada langkah yang ke 3 node C mempunyai 2 node yang dekat yaitu node D dan M, bandingkan nilai antara dua node tersebut dan ambil nilai yang terkecil, dalam hal ini langkah ke-4 dimulai dari node M.

Pada node M cek apakah mempunyai node terdekat dalam hal ini yang terdekat adalah node N, bandingkan nilai yang ada pada node N dan nilai pada node D.

Node D bernilai (4,40) lebih kecil dari nilai node N (4,65), maka untuk langkah selanjutnya dimulai dari node D, begitu seterusnya, sehingga semua node dikunjungi. Pada Gambar 2 graf langkah proses algoritma dijkstra, hasil nilai terkecil didapat pada langkah ke 21 dengan bobot nilai 22,65 Km hal tersebut membuktikan rute yang dipilih oleh algoritma dijkstra adalah rute ke 8 (A-B-C-M-N-O-P-Q-R-K-L) berbeda sedikit dengan jarak terdekat yang didaat dari perhitungan tabel 4 yaitu 22,6 Km.

#### 4. KESIMPULAN DAN SARAN

Dari pembahasan diatas dapat disimpulkan:

1. Dari tabel diatas dapat kita liat bahwa rute nomor 1 (A-B-C-D-E-F-G-H-I-J-K-L) mempunyai jarak yang paling jauh yaitu 33,3 Km, dan rute yang paling pendek atau dekat kita lihat ada pada rute nomor 3 (A-B-C-D-E-O-P-Q-R-K-L) yaitu mempunyai jarak 22,6 Km, dan dapat kita ambil alternatif kedua jarak terpendek didapat dari algoritma dijkstra pada rute nomor 8 (A-B-C-M-N-O-P-Q-R-K-L) yaitu mempunyai nilai 22,65 Km.
2. Dengan hasil perhitungan diatas diharapkan pengguna jalan dapat bijak dalam memilih rute jalan, karena hasil yang didapat tidak menjamin pengguna jalan akan sampai lebih cepat pada tujuan banyak faktor yang mempengaruhi seperti, lamanya lampu lalu lintas pada persimpangan dan tingkat kepadatan lalu lintas pada rute tertentu.
3. Dalam penelitian ini dirasa masih kurang karena selain jarak ada lagi faktor yang harus diperhatikan seperti tingkat kemacetan ada pada rute dan persimpangan jalan, hal ini juga dapat mempengaruhi hasil perhitungan. Untuk itu penulis menyarankan agar dalam penelitian selanjutnya penambahan faktor tingkat kemacetan dapat dimasukkan.

#### DAFTAR PUSTAKA

Cindy, N. (2015). Analisa dan Solusi Kemacetan Lalu Lintas di Ruas Jalan Kota (Studi Kasus Jalan Imam Bonjol - Jalan Sisingamangaraja). *Teknik Sipil*

*Fakultas Teknik Universitas Lampung,*  
4(ISSN:2303-0011), 153–162.

- Harahap, M. K., & Khairina, N. (2017). Pencarian Jalur Terpendek dengan Algoritma Dijkstra. *Sinkron*, 2(2), 18. <https://doi.org/10.33395/sinkron.v2i2.61>
- Noviriandini, A., & Safitri, M. (2017). Implementasi Algoritma Dijkstra Untuk Menentukan Jalur Terpendek Wilayah Pisangan Dan Kampus Nusa Mandiri Tangerang. *Jurnal Pilar Nusa Mandiri*, 13(2), 181–186.
- Rifanti, U. M. (2017). Pemilihan Rute Terbaik Menggunakan Algoritma Dijkstra Untuk Mengurangi Kemacetan Lalu Lintas di Purwokerto. *JMPM: Jurnal Matematika Dan Pendidikan Matematika*, 2(2), 90. <https://doi.org/10.26594/jmpm.v2i2.926>
- Rinaldi, M. (2012). *Matematika Diskrit*. Bandung: Informatika.