

Penerapan Algoritma Dijkstra Dalam Menentukan Jalur Terpendek dari Jalan Tubagus Ismail ke Kampus Ganesha Institut Teknologi Bandung

Vincent Christian Siregar- 13520136
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13520136@std.stei.itb.ac.id

Abstract—Algoritma Dijkstra merupakan salah satu algoritma untuk mencari jalur terpendek dari sebuah graf. Di kehidupan nyata, algoritma Dijkstra dapat digunakan sebagai metode pengambilan keputusan dalam mengambil jalur terpendek untuk mencapai tujuan. Salah satu aplikasi dari algoritma ini adalah mencari jalur terpendek menuju kampus Ganesha Institut Teknologi Bandung dari tempat tinggal mahasiswa. Salah satu daerah tempat tinggal yang ditempati oleh mahasiswa Institut Teknologi Bandung adalah daerah Tubagus Ismail.

Keywords—*Institut Teknologi Bandung; Tubagus Ismail; jalur terpendek; Dijkstra; algoritma*

I. PENDAHULUAN

A. Institut Teknologi Bandung (ITB)

Institut Teknologi Bandung atau biasa disingkat ITB adalah sekolah tinggi teknik pertama di Indonesia yang didirikan pada tanggal 2 Maret 1959 di Bandung, Jawa Barat. ITB merupakan salah satu dari 122 unit Perguruan Tinggi Negeri di Indonesia. Perguruan Tinggi Negeri adalah perguruan tinggi yang diselenggarakan oleh pemerintah.

ITB memiliki tiga kampus di Indonesia yaitu kampus Ganesha, kampus Jatinangor, dan kampus Cirebon. Kampus ITB Ganesha berada di Kota Bandung, tepatnya di Jalan Ganesha 10.

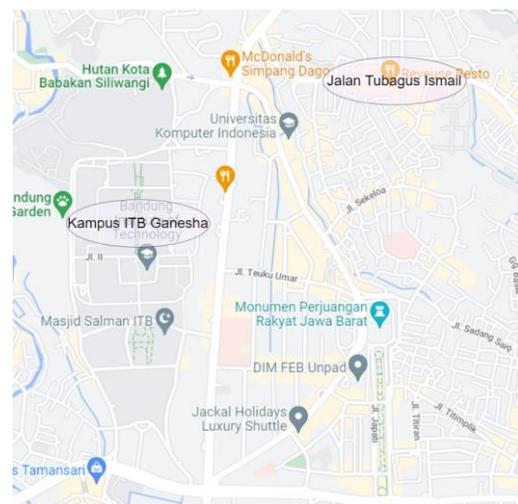


Gambar 1 Kampus Ganesha ITB
Sumber: www.itb.ac.id

Institut Teknologi Bandung sangat diminati oleh pelajar dari seluruh penjuru Indonesia. Terdapat tiga jalur yang dapat dilalui oleh siswa yang ingin menempuh pendidikan di ITB. Ketiga jalur tersebut adalah jalur SNMPTN, jalur SBMPTN, dan jalur mandiri. Jumlah mahasiswa ITB yang diterima pada tahun 2021 mencapai 7277 untuk program S1, S2, S3, dan Profesi.

B. Jalan Tubagus Ismail

Jalan Tubagus Ismail berada di timur laut dari kampus Ganesha Institut Teknologi Bandung. Terdapat banyak indekos yang banyak ditinggali oleh Mahasiswa ITB. Letaknya yang tidak terlalu jauh dari kampus Ganesha ITB menjadi daya tarik mahasiswa untuk menetap di daerah Jalan Tubagus Ismail.



Gambar 2 Peta Letak Jalan Tubagus Ismail
Sumber: Google Maps

C. Alternatif Jalan Menuju Institut Teknologi Bandung

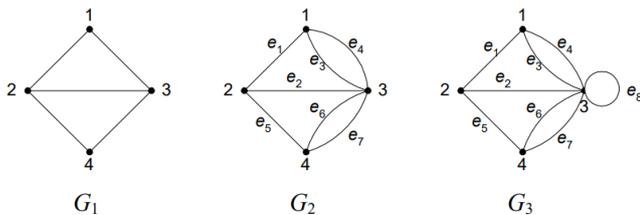
Untuk mencapai kampus Ganesha dari Jalan Tubagus Ismail, mahasiswa dapat memilih beberapa jalur yang tersedia. Setiap jalur memiliki akses yang berbeda-beda dan memiliki jarak tempuh total yang berbeda-beda.

Salah satu cara pengambilan keputusan jalan yang dapat digunakan oleh mahasiswa adalah menggunakan algoritma djikstra. Algoritma djikstra juga dapat digunakan untuk mengetahui urutan jalan yang terpendek yang bisa dipilih oleh mahasiswa untuk sampai ke kampus Ganesha.

II. DASAR TEORI

A. Graf

Graf adalah representasi objek – objek diskrit dan hubungan antara objek-objek tersebut. Graf dinyatakan dalam sebuah tupel (V,E) . V (Vertex) merupakan himpunan tidak-kosong dari simpul-simpul. E (Edges) merupakan himpunan sisi yang menghubungkan sepasang simpul. Vertex juga sering disebut sebagai node.

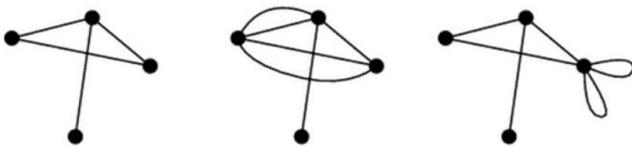


Gambar 3. Contoh graf
Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf> Diakses pada tanggal 20 mei 2022

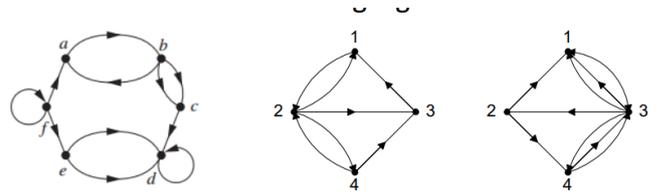
B. Graf Berarah

Berdasarkan orientasi arah pada sisi, graf dibedakan menjadi graf tak-berarah dan graf berarah. Graf tak-berarah adalah graf yang tidak memiliki orientasi arah. Graf berarah adalah graf yang memiliki orientasi arah. Dalam penerapan algoritma Dijkstra dalam menentukan jalur terpendek, jenis graf yang digunakan adalah graf berarah. Node yang ditunjuk oleh panah merupakan node tujuan.



Gambar 4 Contoh Graf Tak Berarah
Sumber:

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf> diakses pada tanggal 20 mei 2022

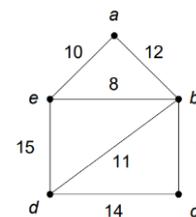


Gambar 5 Contoh Graf Berarah
Sumber:

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf> diakses pada tanggal 20 mei 2022

C. Graf Berbobot

Graf berbobot adalah salah satu jenis graf setiap sisinya diberi sebuah harga, Pada penerapan algoritma Dijkstra, graf yang digunakan adalah graf berbobot. Dalam penentuan jalur terpendek, bobot merepresentasikan jarak yang ditempuh dari suatu node ke node lainnya.



Gambar 4 Contoh Graf Berbobot
Sumber:

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf> diakses pada tanggal 20 mei 2022

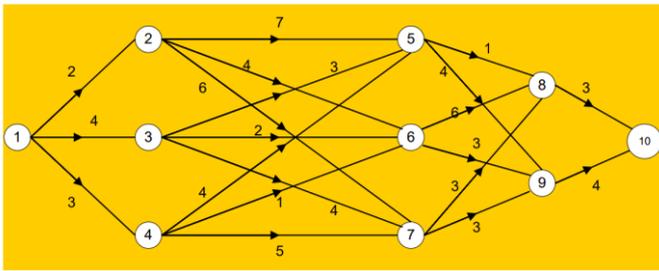
D. Algoritma Dijkstra

Algoritma djikstra adalah algoritma yang dibuat oleh Edsger W. Dijkstra untuk menemukan jalur terpendek dari setiap node pada graf. Algoritma ini dibuat pada Tahun 1959 dan di rilis tiga tahun setelahnya.

Terdapat beberapa varian dari algoritma Dijkstra. Awalnya algoritma Dijkstra hanya mencari jalur terpendek dari dua node. Variasi lainnya adalah menetapkan suatu node sebagai node asal dan mencari jalur terpendek ke seluruh node yang dapat dicapai.

E. Dynamic Pogramming

Algoritma Dijkstra merupakan salah satu contoh dynamic programming. Dynamic programming atau program dinamis adalah metode pemecahan masalah dengan membentuk tahapan atau *stage* sedemikian sehingga solusi persoalan dapat dipandang sebagai serangkaian keputusan yang saling berkaitan.



Gambar 6. Contoh mapping persoalan program dinamis
Sumber:

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Program-Dinamis-2020-Bagian1.pdf> Diakses pada tanggal 20 mei 2022

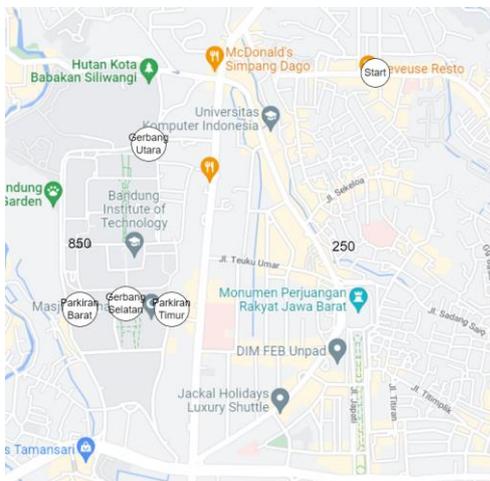
Rangkaian keputusan yang dibuat pada program dinamis dibuat dengan prinsip optimalitas. Prinsip optimalitas adalah bagian solusi sampai ke-k optimal jika solusi total optimal. Artinya adalah jika kita bekerja dari tahap k ke tahap k+1, nilai optimal k dapat digunakan tanpa kembali ke tahap sebelumnya.

III. PEMBENTUKAN GRAF

A. Penentuan Titik Awal dan Tujuan Akhir

Tahap pertama dalam penentuan jalur terpendek dari jalan Tubagus Ismail ke kampus Ganesha ITB adalah menentukan titik awal dan titik akhir pada peta. Tujuannya agar dapat diketahui goal yang diinginkan.

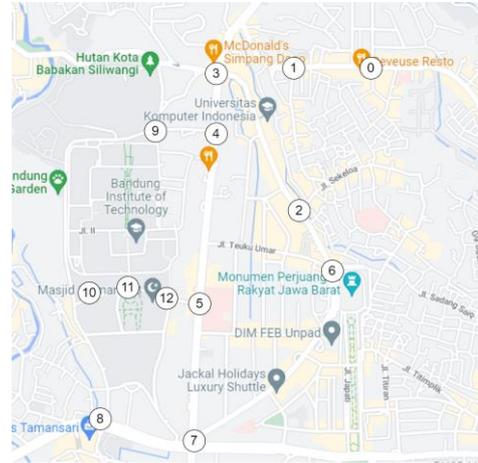
Kampus ITB memiliki beberapa gerbang masuk yang dapat ditetapkan sebagai titik akhir. Gerbang utara berada di bagian utara dekat jalan Dayang Sumbi. Gerbang Selatan berada di bagian selatan tepat di depan jalan Ganesha. Gerbang Utara dan Gerbang Selatan adalah gerbang yang umum digunakan mahasiswa untuk masuk ke kampus Ganesha. Jika menggunakan kendaraan pribadi, mahasiswa dapat masuk melalui prakiran barat atau melalui parkir timur.



Gambar 7 Penentuan titik awal dan titik akhir
Sumber: Google Maps

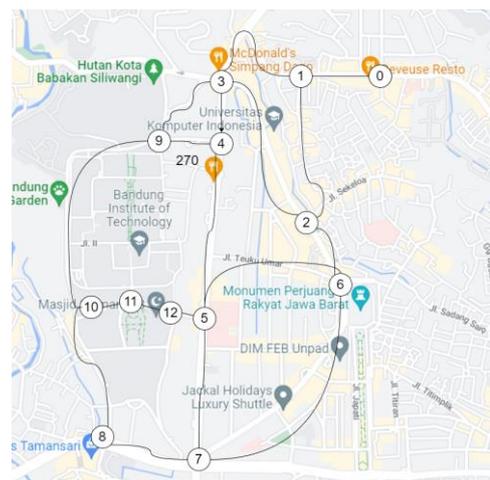
B. Pembentukan Graf Tak Berarah

Terdapat banyak alternatif jalan yang dapat dilalui untuk sampai ke titik akhir. Alternatif jalan terbentuk jika terdapat pilihan jalur yang berbeda tetapi sama-sama dapat mengarah ke kampus Ganesha. Untuk setiap jalan yang memberi pilihan jalur, diberikan tanda sebuah node. Berikut adalah gambar node yang dapat dibentuk.



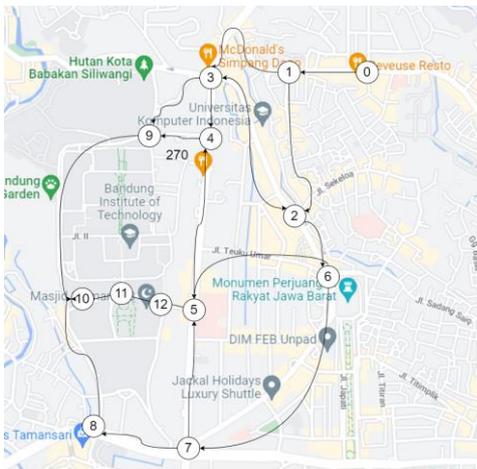
Gambar 8 Penambahan node pada peta
Sumber: Google Maps

Setiap node memiliki pilihan jalur ke node lainnya. Satu node dapat memiliki jalur ke lebih dari satu node. Kondisi ini timbul jika terdapat persimpangan yang memberikan berapa alternatif jalur. Jalur di representasikan dengan garis yang menghubungkan node pada gambar. Garis yang terbentuk menandakan mahasiswa dapat melewati jalur tersebut untuk berpindah dari satu node ke node lainnya.



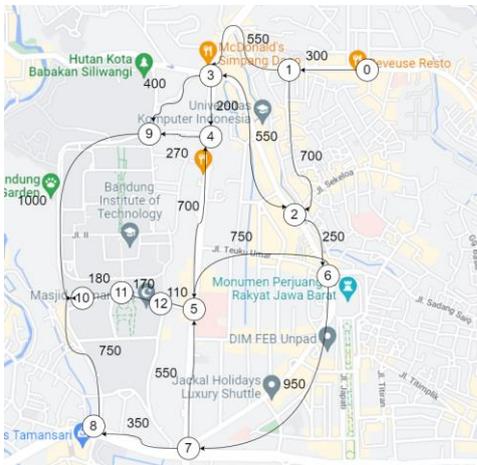
Gambar 9 Penambahan edge pada peta
Sumber: Google Maps

Graf yang digunakan pada program adalah graf berarah. Oleh karena itu, arah dari setiap edge perlu ditambahkan. Panah dapat berupa panah satu arah atau panah dua arah tergantung jalan yang dilalui. Berikut adalah gambar graf yang sudah diberi arah.



Gambar 10 Penambahan edge pada peta
Sumber: Google Maps

Dari graf berarah yang sudah dibuat, bobot setiap jalur diisi dengan jarak yang ditempuh pada jalur tersebut. Jarak yang digunakan pada makalah ini menggunakan satuan meter (m). Pengambilan jarak dilakukan dengan menggunakan bantuan google maps. Berikut adalah graf berbobot yang dihasilkan.



Gambar 11 Penambahan edge pada peta
Sumber: Google Maps

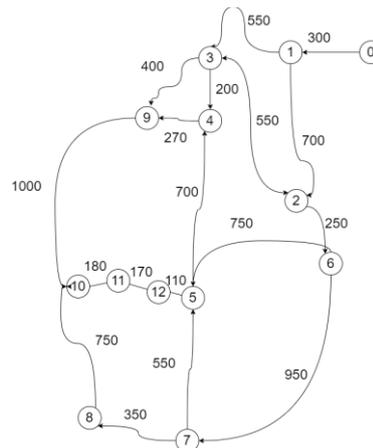
IV. PENERAPAN ALGORITMA DIJKSTRA

A. Adjacency Matrix

Terdapat beberapa alternatif struktur data yang dapat digunakan untuk melakukan kalkulasi terhadap graf yang sudah dibuat pada bab sebelumnya. Pada pembahasan ini, struktur data yang digunakan adalah matriks atau *Adjacency Matrix*. Matriks ini menyimpan relasi setiap node dalam sebuah array dua dimensi. Sumbu-y dari matriks merepresentasikan node asal. Sumbu-x dari matriks merepresentasikan node tujuan yang ditunjuk oleh panah dari node asal.

karena graf yang digunakan adalah graf berbobot, maka relasi setiap node diisi dengan bobot sisi tersebut. Contoh pada gambar 12, node 0 memiliki panah menuju node 1 dengan bobot

300. Maka, isi matriks pada baris 0 dan kolom 1 diberi nilai 300. Berikut adalah gambar graf dan *adjacency matrix* yang dihasilkan.



Gambar 12 Graf yang dihasilkan

Sumber: Penulis

0	300	0	0	0	0	0	0	0	0	0	0	0
0	0	700	550	0	0	0	0	0	0	0	0	0
0	0	0	550	0	0	250	0	0	0	0	0	0
0	0	550	0	200	0	0	0	0	400	0	0	0
0	0	0	0	0	700	0	0	0	0	270	0	0
0	0	0	0	700	0	0	550	0	0	0	0	110
0	0	0	0	0	750	0	950	0	0	0	0	0
0	0	0	0	0	0	0	0	350	0	0	0	0
0	0	0	0	0	0	0	0	0	0	750	0	0
0	0	0	0	0	0	0	0	0	0	0	1000	0
0	0	0	0	0	0	0	0	0	0	0	0	180
0	0	0	0	0	0	0	0	0	0	0	180	0
0	0	0	0	0	110	0	0	0	0	0	0	170

Tabel 1 *Adjacency Matrix* yang dihasilkan

B. Pseudocode

Berikut adalah pseudocode dari algoritma djikstra dalam penentuan jalur terpendek dari graf.

```
// inisialisasi array dan variable
Graph <- array dua dimensi berisi Adjacency Matrix

Row <- Jumlah baris pada matriks
col <- Jumlah kolom pada matriks

Jarak <- array bertipe Integer berukuran Row dengan isi default infinity
Parent <- array bertipe Integer berukuran Row dengan isi default -1
```

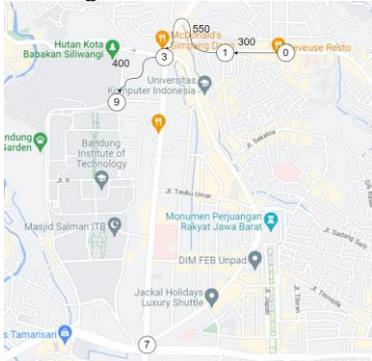
```
// Jarak node awal ke node itu sendiri adalah 0
Jarak[0] <- 0

Queue <- array dinamis

// Algoritma
Masukkan setiap node ke dalam array queue (0..12)
while queue is not empty:
    u <- node terdekat dari Start yang terdapat
    pada queue
    hapus u dari Queue
    for i in range(Col):
        if Graph[u][i] != 0 && i ada di Queue:
            if Jarak[u] + Graph[u][i] < Jarak[i]:
                Jarak[i] = Jarak[u] + Graph[u][i]
                Parent[i] = u
```

Dari Tabel 2 terlihat bahwa algoritma Dijkstra dapat menemukan seluruh rute terpendek dari node awal ke setiap node yang dapat dilalui dan jarak terpendek dari setiap node.

Dari hasil tersebut, rute terpendek dari Jalan Tubagus Ismail ke kampus Ganesha melalui gerbang utara (node 9) adalah melalui rute 0-1-3-9 dengan jarak ±1250 meter. Tampilan rute pada peta adalah sebagai berikut.



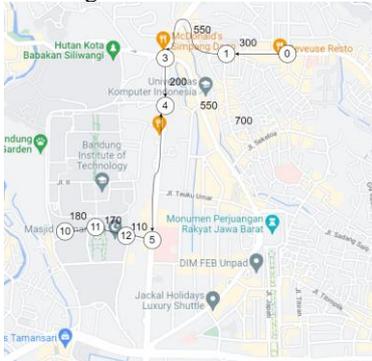
Gambar 13 Jarak terpendek ke gerbang utara

Rute terpendek dari Jalan Tubagus Ismail ke kampus Ganesha melalui gerbang selatan (node 11) adalah melalui rute 0-1-3-4-5-12-11 dengan jarak ±2030 meter. Tampilan rute pada peta adalah sebagai berikut.



Gambar 14 Jarak terpendek ke gerbang selatan
Sumber: Google Maps

Rute terpendek dari Jalan Tubagus Ismail ke kampus Ganesha melalui parkir barat (node 10) adalah melalui rute 0-1-3-4-5-12-11-10 dengan jarak ±2210 meter. Tampilan rute pada peta adalah sebagai berikut.



Gambar 15 Jarak terpendek ke parkir barat
Sumber: Google Maps

Pseudocode diadaptasi dari kode yang di terbitkan pada situs [GeeksforGeeks](https://www.geeksforgeeks.org/) yang diakses pada tanggal 21 mei 2022. Hasil dari program adalah array Jarak dan array Parent. Array Jarak menyimpan jarak terdekat dari node start ke setiap node yang dapat dicapai. Array parent menyimpan node asal terdekat yang dapat mengarah ke node awal. Array parent digunakan untuk menentukan urutan jalur dari node awal sampai node yang diinginkan.

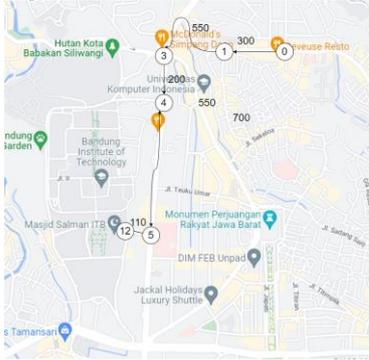
C. Hasil Perhitungan Program

Implementasi algoritma menggunakan bahasa pemrograman Python3. Kode tidak disertakan pada makalah ini. Hasil perhitungan jarak dari node awal ke setiap node dapat dilihat pada tabel dibawah ini

Node Tujuan	Jarak ± (dalam meter)	Rute terpendek
1	300	0-1
2	1000	0-1-2
3	850	0-1-3
4	1050	0-1-3-4
5	1750	0-1-3-4-5
6	1250	0-1-2-6
7	2200	0-1-2-6-7
8	2550	0-1-2-6-7-8
9	1250	0-1-3-9
10	2210	0-1-3-4-5-12-11-10
11	2030	0-1-3-4-5-12-11
12	1860	0-1-3-4-5-12

Tabel 2 Hasil perhitungan rute terpendek setiap node

Rute terpendek dari Jalan Tubagus Ismail ke kampus Ganesha melalui parkir timur (node 12) adalah melalui rute 0-1-3-4-5-12 dengan jarak ± 1860 meter. Tampilan rute pada peta adalah sebagai berikut.



Gambar 15 Jarak terpendek ke parkir timur
Sumber: Google Maps

V. KESIMPULAN

Algoritma Dijkstra dapat digunakan untuk menghitung rute terpendek yang dapat ditempuh dari suatu node ke node tertentu pada sebuah graf berarah. Salah satu aplikasi nyata dari algoritma Dijkstra adalah mencari rute terpendek untuk sampai ke lokasi tujuan dari lokasi awal.

Algoritma Dijkstra mampu menghasilkan solusi yang selalu optimal untuk setiap kasus. Algoritma ini lebih baik dari algoritma exhaustive search dalam segi kompleksitas waktu karena data jalur terpendek setiap node dicatat dan tidak dihitung ulang pada iterasi berikutnya.

VI. UCAPAN TERIMA KASIH

Puji syukur kepada Tuhan Yang Maha Esa karena rahmat dan karunia-Nya makalah ini dapat terselesaikan.

Terima kasih juga kepada dosen pengampu yang sudah senantiasa membimbing saya selama pembelajaran saya dalam mata kuliah IF2211 Strategi Algoritma Institut Teknologi Bandung.

VIDEO LINK AT YOUTUBE

<https://youtu.be/FEFs2iuI9G0>

REFERENCES

- [1] <https://databoks.katadata.co.id/datapublish/2017/05/05/berapa-jumlah-perguruan-tinggi-di-indonesia> diakses pada tanggal 20 mei 2022
- [2] www.itb.ac.id diakses pada tanggal 20 mei 2022
- [3] <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf> diakses pada tanggal 20 mei 2022
- [4] <https://informatika.stei.itb.ac.id/~rinaldi.munir/Smik/2020-2021/Program-Dinamis-2020-Bagian1.pdf> Diakses pada tanggal 20 mei 2022
- [5] <https://www.geeksforgeeks.org/dijkstras-shortest-path-algorithm-greedy-algo-7/> diakses pada tanggal 21 mei 2022

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, 20 Mei 2022

Vincent Christian Siregar 13520136