

# Penerapan Algoritma *Greedy* dalam Menghitung Bandwith pada Jaringan Telepon

Muhammad Fikri Naufal (13519158)

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jalan Ganesha 10 Bandung

E-mail (gmail): 13519158@std.stei.itb.ac.id

**Abstrak**—Jaringan telepon adalah jaringan telekomunikasi yang menghubungkan telepon sehingga memungkinkan panggilan telepon dari dua atau lebih pihak. Tiap garis pada jaringan telepon memiliki nilai bandwith yaitu frekuensi tertinggi yang dapat didukung oleh garis tersebut, maka sebuah kota yang memiliki jaringan telepon dapat dimodelkan sebagai graf berbobot dengan bobot nilai bandwith tiap garis sehingga saat telepon melakukan sebuah panggilan ke telepon lain dapat dianggap sebagai persoalan routing. Persoalan ini dapat diselesaikan dengan lengkap dan optimal menggunakan salah satu algoritma greedy yaitu algoritma Dijkstra.

**Kata kunci**—Jaringan telepon, bandwith, graf berbobot, routing, algoritma greedy, algoritma Dijkstra

## I. PENDAHULUAN

Pada masa pandemi Covid-19, komunikasi jarak jauh menjadi jauh lebih penting dibandingkan tahun-tahun sebelumnya. Mulai dari proses belajar mengajar, jual beli, perkantoran, atau sekedar komunikasi tatap muka dilakukan menggunakan jaringan internet.

Sebelum munculnya jaringan internet, masyarakat menggunakan jaringan telepon sebagai alat untuk berkomunikasi jarak jauh. Telepon sendiri adalah pesawat dengan listrik dan kawat untuk bercakap-cakap antara dua orang yang berjauhan tempatnya dan beroperasi dengan menggunakan transmisi sinyal listrik dalam jaringan telepon sehingga memungkinkan pengguna telepon untuk berkomunikasi dengan pengguna lainnya.



Gambar 1. Telepon

Sumber: <https://www.pengertianilmu.com/2015/02/pengertian-telepon.html>

Jika dibahas melalui sejarahnya, sebelum mengenal telepon, manusia mengenal surat dan telegraf. Karena

pengiriman surat dianggap membutuhkan waktu yang panjang, orang-orang beralih pada teknologi telegraf, yaitu alat yang mengirimkan pesan dalam bentuk kode Morse menggunakan listrik. Sejak itu dimulailah banyak percobaan bunyi, listrik, dan telegraf oleh peneliti untuk menyempurnakan cara berkomunikasi. Salah satu peneliti tersebut yaitu Alexander Graham Bell, seorang guru yang melakukan percobaan tinggi-rendah nada dan getaran bunyi. Singkat cerita, Bell dan asistennya Thomas Watson melakukan uji coba pada telepon buatannya yang dapat melintasi jarak 8 mil pada tahun 1876. Setelah itu, telepon diperkenalkan ke masyarakat dan jaringan telepon diperluas di seluruh dunia.

Pada saat ini, jaringan telepon sudah berkembang menjadi beberapa jenis, yaitu PSTN, wireless network, dan PBX. PSTN atau public switched telephone network merupakan jaringan telepon yang langsung dihubungkan ke sistem sentral telepon menggunakan kabel yang ditanam dibawah tanah atau dengan tiang. Wireless network digunakan oleh telepon genggam dan dapat berpindah-pindah dalam area yang diliputi. PBX atau private branch exchange merupakan jaringan privat dimana beberapa telepon saling dihubungkan langsung dan dihubungkan juga dengan gateway untuk terhubung dengan jaringan luar.

Pada tiap jaring pada jaringan telepon memiliki sebuah nilai bandwith. Bandwith atau lebar pita adalah luas atau lebar cakupan frekuensi yang digunakan oleh sinyal dalam medium transmisi. Bandwith juga dapat diartikan sebagai perbedaan antara frekuensi terendah dan tertinggi dalam rentang tertentu. Misal line telepon memiliki bandwith 3000 Hz yang merupakan rentang frekuensi tertinggi yaitu 3300 Hz dan frekuensi terendah 300 Hz yang dapat dilewati line telepon tersebut.

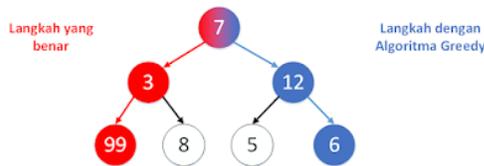
Jika dilihat secara luas, jaringan telepon dapat digambarkan dengan sebuah graf berbobot. Simpul dari graf menggambarkan switching station, sisi dari graf menggambarkan garis transmisi, dan bobot sisi graf menggunakan nilai bandwidth sehingga dapat dianggap sebagai persoalan routing saat sebuah telepon melakukan panggilan ke telepon lain. Ide dari pencarian jalur ini yaitu memilih jalur yang memiliki bobot sisi minimum terbesar. Persoalan tersebut dapat diselesaikan menggunakan salah satu algoritma greedy yaitu algoritma Dijkstra.

## II. TEORI DASAR

### A. Algoritma Greedy

Dalam memecahkan persoalan optimasi, algoritma greedy merupakan solusi paling populer. Dalam bahasa Indonesia greedy berarti rakus, tamak, atau serakah. Algoritma greedy membentuk solusi langkah per langkah sehingga pada tiap langkah harus dibuat keputusan terbaik untuk mengambil langkah selanjutnya. Keputusan yang telah dibuat tidak dapat diubah pada langkah-langkah selanjutnya.

#### Pencarian Nilai Terbesar



Gambar 2. Gambaran Solusi dengan Algoritma Greedy

Sumber: <https://www.it-jurnal.com/pengertian-algoritma-greedy/>

Algoritma greedy hanya dapat menyelesaikan persoalan optimasi yang terbagi menjadi dua yaitu maksimasi dan minimasi. Solusi optimum yang dihasilkan dari algoritma greedy merupakan solusi yang bernilai minimum atau maksimum dari kumpulan solusi yang dihasilkan. Bagian dari persoalan optimasi tersebut yaitu kendala (constraints) dan fungsi objektif/fungsi optimasi.

Solusi yang memenuhi seluruh constraints adalah solusi yang layak serta solusi layak yang bernilai optimum disebut solusi optimum.

### B. Algoritma Dijkstra

Algoritma Dijkstra merupakan sebuah algoritma yang digunakan untuk memecahkan shortest path pada sebuah graf berarah. Algoritma tersebut ditemukan oleh Edsger Dijkstra dan dipublikasikan pada tahun 1959 melalui jurnal *Numerische Mathematik* dengan judul "A Note on Two Problems in Connexion with Graphs". Algoritma Dijkstra termasuk salah satu algoritma greedy.



Gambar 3. Edsger Dijkstra

Sumber: <http://blog.magmalabs.io/2014/02/28/edsger-dijkstra.html>

Masalah shortest path merupakan masalah klasik optimasi yang banyak digunakan untuk menguji sebuah algoritma. Permasalahan tersebut banyak dipilih sebagai uji coba karena dianggap mudah dimengerti tetapi memiliki banyak solusi. Selain itu menurut Andrew Goldberg, seorang peneliti di Microsoft Research Silicon Valley, masalah shortest path adalah masalah relevan untuk berbagai macam aplikasi seperti network routing, game, desain sirkuit, dan pemetaan.

Secara matematis, graf dapat dituliskan dengan  $G = \{V, E\}$  dengan  $G$  yaitu graf yang didefinisikan oleh satu set simpul ( $V$ ) dan koleksi sisi-sisi yang menghubungkan simpul-simpul tersebut ( $E$ ).

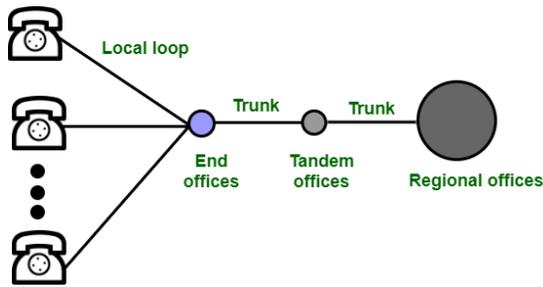
Cara kerja algoritma Dijkstra yaitu membuat jalur ke satu simpul optimal pada tiap langkah yang dipilih sehingga pada langkah ke  $n$  kita sudah tahu sejumlah  $n$  node sebagai jalur terpendeknya. Langkah-langkah algoritma Dijkstra secara rinci dijelaskan sebagai berikut.

1. Pilih node awal lalu beri bobot jarak pada node pertama ke node terdekat satu per satu.
2. Beri bobot (jarak) pada setiap titik ke titik lainnya lalu beri nilai 0 pada node awal dan nilai tak hingga pada node lain yang belum terisi.
3. Set semua node yang belum dilalui dan node awal sebagai node mulai
4. Dari node awal, pilih node tetangga yang belum dilalui dan hitung jaraknya dari titik awal. Hapus data jarak yang lama jika jarak lebih kecil dari data sebelumnya lalu simpan data jarak baru.
5. Setelah mempertimbangkan node tetangga, tandai node yang telah dilalui karena node tersebut tidak akan pernah dicek Kembali, jarak yang disimpan adalah jarak terakhir yang paling minimal bobotnya.
6. Set node yang belum dilewati dengan jarak terkecil sebagai node awal lalu ulangi langkah 5 sampai node awal merupakan node tujuan.

### C. Jaringan Telepon

Jaringan Telepon merupakan jaringan yang digunakan untuk menyediakan komunikasi suara jarak jauh. Jaringan telepon menggunakan sesuatu yang dinamakan Circuit Switching. Pada awalnya, seluruh jaringan telepon disebut sebagai Plain Old Telephone System (POTS) yang menggunakan sinyal analog. Dengan kemajuan teknologi, muncul sebuah fitur untuk membawa data selain dalam format suara. Jaringan yang dikenal sekarang sudah menggunakan sinyal analog dan digital.

Terdapat tiga komponen utama pada jaringan telepon, yaitu local loops, trunks, dan switching offices. Switching offices memiliki beberapa tingkatan, yaitu end offices, tandem offices, dan regional offices. Skema jaringan telepon tersebut dapat dilihat pada gambar berikut ini.



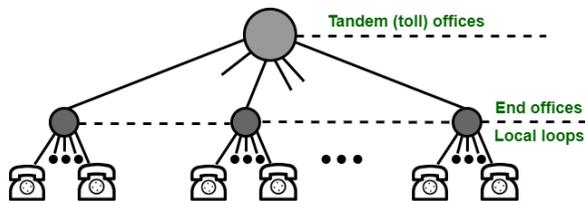
Gambar 4. Jaringan Telepon

Sumber: <https://www.geeksforgeeks.org/introduction-to-telephone-network/>

Local loops adalah sepasang kabel yang saling membelit untuk menghubungkan telepon pelanggan ke end office atau central office lokal terdekat. Kabel tersebut memiliki nilai bandwidth 4000 Hz untuk tujuan mengirim suara. Pada local loop, nomor telepon melambangkan beberapa bagian berbeda. Tiga digit pertama melambangkan office dan empat digit terakhir melambangkan nomor local loops.

Trunks adalah media transmisi yang digunakan untuk komunikasi antar office. Dengan multiplexing, trunks dapat menangani ratusan bahkan ribuan koneksi. Secara umum, transmisi dilakukan menggunakan optical fiber atau penghubung satelit.

Switching office menggunakan switch untuk menghubungkan loops yang bervariasi atau trunks dan memungkinkan koneksi antara pelanggan-pelanggan berbeda. Sebelum adanya sistem switch, dua pelanggan telepon memiliki link fisik yang permanen. Sistem switching office dapat digambarkan pada ilustrasi berikut.



Gambar 5. Switching Offices

Sumber: <https://www.geeksforgeeks.org/introduction-to-telephone-network/>

Kelebihan yang dimiliki jaringan telepon yaitu jaringan dengan sirkuit yang dapat berubah sehingga tidak ada jeda waktu transmisi karena penerima dapat dipilih. Harga dari penggunaan jaringan telepon juga dianggap murah karena jaringan tersebut tersebar luas.

Kelemahan yang dimiliki jaringan telepon yaitu membutuhkan waktu yang cukup lama untuk terkoneksi dan kecepatan transmisi yang relative lambat.

### III. PENERAPAN ALGORITMA DIJKSTRA PADA MASALAH PENGHITUNGAN BANDWITH JARINGAN TELEPON

#### A. Modifikasi Algoritma Dijkstra

Selain masalah shortest path, algoritma Dijkstra juga dapat dimodifikasi untuk menyelesaikan masalah penghitungan bandwidth dalam jaringan telepon. Komponen algoritma Dijkstra yang dapat dimodifikasi yaitu graf, priority queue, dan inialisasi jarak dan relaxation process.

Pada graf, simpul atau node, sisi atau edge, serta bobot tiap sisi dapat memiliki arti yang berbeda pada tiap kasus. Jarak dari node awal ke node akhir juga dapat diartikan berbeda serta dapat juga diubah artinya. Priority queue juga dapat memiliki aturan-aturan yang berbeda untuk menempatkan elemen-elemennya, dapat memprioritaskan yang maksimum atau minimum. Relaxation process juga akan berbeda pada tiap kasus yang ingin dipecahkan, serta akan membutuhkan inialisasi jarak yang berbeda pula.

Untuk memodifikasi algoritma Dijkstra yang telah dijelaskan pada bagian teori dasar agar dapat menyelesaikan permasalahan penghitungan bandwidth pada jaringan telepon, seperti yang telah disampaikan pada bagian pendahuluan, komponen-komponen graf yang digunakan memiliki arti berbeda. Simpul dari graf menggambarkan switching station, sisi dari graf menggambarkan garis transmisi, dan bobot sisi graf menggunakan nilai bandwidth.

Fungsi jarak pun diubah artinya. Dari yang sebelumnya jarak adalah total seluruh bobot terkecil sisi yang dilewati dari node awal hingga node akhir menjadi bandwidth minimum dari sisi yang dilewati node awal hingga node akhir yang maksimum pada graf tersebut. Sehingga bandwidth dari suatu path P adalah bandwidth minimum pada sisi dalam path P.

Aturan pada priority queue juga diubah. Untuk mencari shortest path, priority queue diset untuk meletakkan elemen dengan nilai terendah pada indeks yang lebih kecil. Saat persoalan diubah menjadi penghitungan bandwidth pada jaringan telepon, maka priority queue diset untuk meletakkan elemen dengan nilai terbesar karena diinginkan jalur dengan bandwidth terbesar.

Cara menginisialisasi nilai tiap node juga diubah. Untuk menyelesaikan persoalan shortest path, node awal diinisialisasi dengan nilai nol serta node lainnya diisi nilai tak hingga. Pada persoalan penghitungan bandwidth jaringan telepon, node awal diinisialisasi dengan nilai tak hingga dan node lainnya diisi nilai nol. Ini dilakukan karena jalur yang dipilih adalah jalur dengan bobot atau bandwidth terbesar.

Modifikasi terakhir pada algoritma Dijkstra ini yaitu relaxation process. Relaxation process adalah proses terkait perubahan pada tiap nilai node setelah memilih node tertentu. Node yang akan berubah nilainya adalah node yang bertetangga dengan node yang dipilih dengan syarat node tersebut belum dipilih sebelumnya. Pada kasus penghitungan bandwidth pada jaringan telepon, node tersebut berubah nilai sesuai dengan kondisi berikut

$$\text{If } \min(\text{bandwidth}(a, b), \text{bandwidth}(a)) > \text{bandwidth}(b) \text{ then} \\ \text{bandwidth}(b) = \min(\text{bandwidth}(a, b), \text{bandwidth}(a))$$

Dengan a adalah node yang dipilih dan b adalah tiap node yang bertetangga dengan a dengan syarat belum pernah dipilih sebelumnya.

**B. Tahap Algoritma Dijkstra pada Masalah Penghitungan Bandwith Jaringan Telepon**

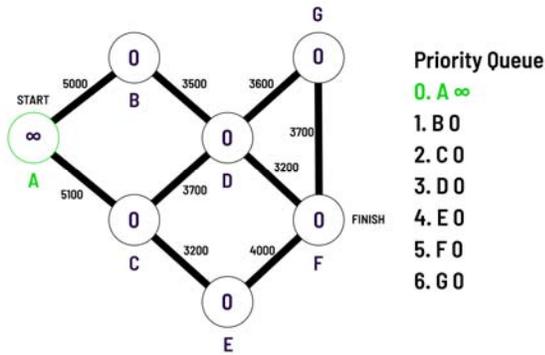
Tahap-tahap algoritma Dijkstra yang digunakan untuk menyelesaikan permasalahan penghitungan bandwith pada jaringan telepon akan dijelaskan secara terperinci sebagai berikut.

1. Menginisialisasi nilai node awal dengan tak hingga dan nilai node liannya dengan nol.
2. Memasukkan node-node tersebut ke dalam priority queue
3. Mengambil node dengan nilai terbesar dan memasukkannya ke dalam variabel a yaitu node yang dipilih lalu hapus node tersebut dari priority queue.
4. Untuk tiap node yang bertetangga dengan node a, ubah nilainya sesuai dengan kondisi yaitu:

```
If min(bandwith(a, b), bandwith(a)) > bandwith(b) then
    bandwith(b) = min(bandwith(a, b), bandwith(a))
```

5. Update nilai b pada priority queue
6. Ulangi proses 3-5 sampai dipilih node akhir.

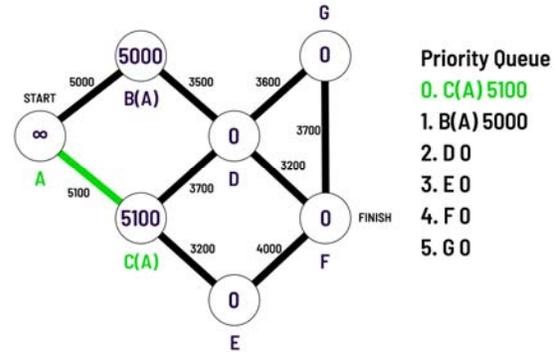
Untuk lebih memahami tahapan-tahapan dari algoritma tersebut diberikan contoh sebuah soal sebagai berikut. Terdapat catatan jika terdapat tulisan node A(CG E) maka berarti node A yang sebelumnya telah terpilih node C, lalu G, lalu E terlebih dahulu.



Gambar 6. Contoh Penyelesaian Persoalan Tahap 1

Sumber: Dokumen Pribadi

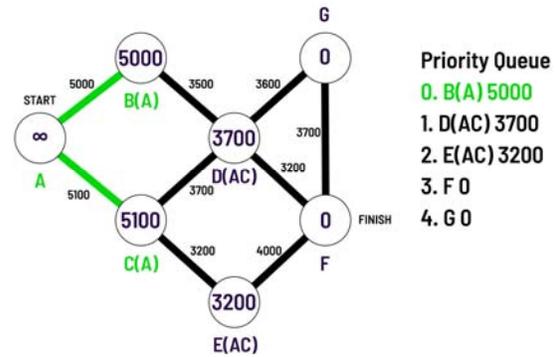
Diberikan sebuah graf jaringan telepon seperti pada gambar 6. Pada persoalan ini, node awal yaitu node A dan node tujuan yaitu node F. Mula-mula, node A dinisialisasi dengan nilai tak hingga, sementara node lainnya diinisialisasi dengan nilai nol. Karena node yang dipilih dari priority queue adalah node terbesar, maka node A yang bernilai tak hingga dipilih lalu dihapus dari priority queue.



Gambar 7. Contoh Penyelesaian Persoalan Tahap 2

Sumber: Dokumen Pribadi

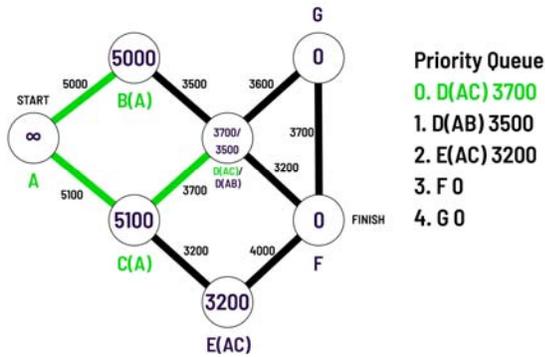
Tahap selanjutnya dapat dilihat pada gambar 7. Setelah node A dipilih, maka ganti nilai dari tetangga node A yang belum pernah dipilih dengan syarat relaxation process yang sudah dibahas sebelumnya, maka node B(A) bernilai 5000 dan node C(A) bernilai 5100 sehingga node C(A) dipilih dan dihapus dari priority queue.



Gambar 8. Contoh Penyelesaian Persoalan Tahap 3

Sumber: Dokumen Pribadi

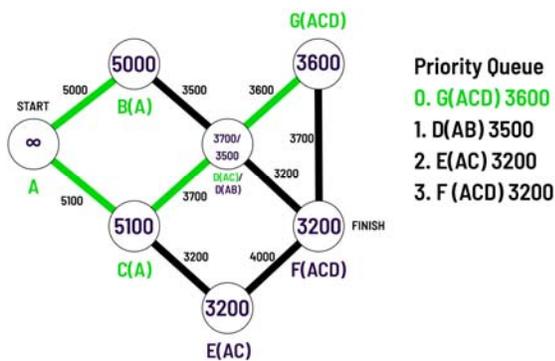
Lalu tahap ketiga dapat dilihat pada gambar 8. Akibat dari terpilihnya node C(A), maka nilai dari tetangga node C(A) yaitu node D(AC) dan node E(AC) berubah sesuai dengan kondisi relaxation process. Node D(AC) bernilai 3700 dan node E(AC) bernilai 3200. Tetapi pada priority queue, node B(A) masih merupakan node dengan nilai terbesar yaitu 5000 sehingga node B(A) dipilih dan dihapus dari priority queue.



Gambar 9. Contoh Penyelesaian Persoalan Tahap 4

Sumber: Dokumen Pribadi

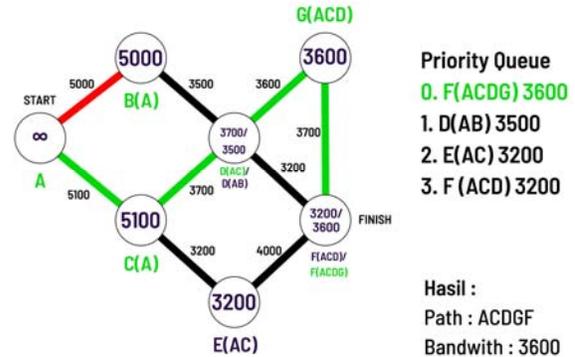
Tahap keempat pada algoritma dapat dilihat pada gambar 9. Akibat dari terpilihnya node B(A) pada tahap sebelumnya, maka node D(AB) berubah nilai menjadi 3500 dan disesuaikan penempatannya pada priority queue. Karena pada priority queue nilai terbesar dipegang oleh node D(AC) maka node tersebut dipilih dan dihapus dari priority queue.



Gambar 10. Contoh Penyelesaian Persoalan Tahap 5

Sumber: Dokumen Pribadi

Tahap selanjutnya yaitu tahap lima dijelaskan pada gambar 10. Karena terpilihnya node D(AC) pada tahap sebelumnya, maka nilai node G(ACD) berubah menjadi 3600 dan node F(ACD) berubah menjadi 3200. Kedua node tersebut ditempatkan pada priority queue sesuai aturannya. Dapat dilihat pada priority queue, node G(ACD) merupakan node dengan nilai terbesar, maka node tersebut dipilih dan dihapus dari priority queue.



Gambar 11. Contoh Penyelesaian Persoalan Tahap Terakhir

Sumber: Dokumen Pribadi

Tahap selanjutnya yaitu tahap terakhir pada penyelesaian persoalan ini dapat dilihat pada gambar 11. Setelah terpilihnya node G(ACD) pada tahap sebelumnya, maka nilai dari node tetangganya yaitu F(ACDG) berubah menjadi 3600. Pada priority queue, node F(ACDG) merupakan node dengan nilai terbesar sehingga node tersebut dipilih dan dihapus dari priority queue. Karena node tersebut adalah node tujuan, maka output dari algoritma tersebut adalah ACDGF sebagai rute dan bandwidth output yaitu bandwidth dari node tujuan atau bernilai 3600.

#### IV. PERCOBAAN

##### A. Teknis Percobaan

Percobaan dilakukan menggunakan persoalan yang sama dengan gambar 6. Program untuk menggambarkan algoritma Dijkstra dalam menyelesaikan permasalahan jaringan telepon dibuat menggunakan bahasa pemrograman Python. Salah satu input dari program yaitu text file berisi graf dengan format sebagai berikut.

```
A B 5000
A C 5100
B D 3500
C D 3700
C E 3200
D G 3600
D F 3200
E F 4000
F G 3700
```

Gambar 12. Format text file graf untuk input program

Sumber: Dokumen Pribadi

```

PROGRAM ALGORITMA DIJKSTRA UNTUK JARINGAN TELEPON
Masukkan nama file graph: graph.txt
Masukkan node awal: A
Masukkan node tujuan: F

Rute Jaringan: A > C > D > G > F
Bandwith: 3600
PS C:\Users\Fikri Naufal\Documents\sem_4\stima>

```

Gambar 13. Hasil Percobaan

Sumber: Dokumen Pribadi

Dapat dilihat bahwa program memiliki output yang sama dengan contoh penyelesaian masalah jaringan telepon dengan algoritma Dijkstra pada bagian sebelumnya.

### V. KESIMPULAN

Algoritma Dijkstra merupakan salah satu algoritma greedy yang dapat menyelesaikan berbagai macam masalah konkrit dengan cara memodifikasi komponen-komponen dari algoritma tersebut. Salah satu permasalahan konkrit yang dapat diselesaikan oleh algoritma Dijkstra yaitu routing dan penentuan bandwidth dari jaringan telepon. Merujuk pada bagian contoh penyelesaian masalah pada penerapan algoritma dijkstra pada masalah penghitungan bandwidth jaringan telepon dan percobaan, terbukti bahwa algoritma Dijkstra mampu memecahkan permasalahan tersebut dengan lengkap dan optimal.

### UCAPAN TERIMA KASIH

Penulis mengucapkan puji syukur kepada Tuhan Yang Maha Esa karena atas segala rahmat dan berkah-Nya, penulis dapat menyelesaikan makalah “Penerapan Algoritma Greedy dalam Menghitung Bandwith pada Jaringan Telepon” sebagai tugas mata kuliah IF2211 Strategi Algoritma Semester Genap tahun ajaran 2020/2021 ini. Penulis berterima kasih kepada Prof. Dwi Hendratmo Widiyantoro, Ph.D atas bimbingan beliau sebagai dosen pengajar mata kuliah Strategi Algoritma. Penulis juga berterima kasih kepada orang tua dan rekan-rekan atas dukungan dan arahan yang diberikan pada penulis hingga dapat menjalankan perkuliahan hingga saat ini.

- [1] GeeksforGeeks. (21 Agustus 2020). Applications of Dijkstra’s shortest path algorithm. Dikunjungi 9 Mei 2021 dari <https://www.geeksforgeeks.org/applications-of-dijkstras-shortest-path-algorithm/>
- [2] Author tanpa nama. Dijkstra’s Application. Michigan Tech. Dikunjungi 9 Mei 2021 dari [http://www.csl.mtu.edu/cs2321/www/newLectures/30\\_More\\_Dijkstra.htm](http://www.csl.mtu.edu/cs2321/www/newLectures/30_More_Dijkstra.htm)
- [3] Dr. rer. nat. Peter Bocker (1988). ISDN The Integrated Services Digital Network: Concepts, Methods, Systems. Springer Berlin Heidelberg. ISBN 978-3-662-08036-8.
- [4] M.A.M.Mirabito & B.L. Morgenstem, New Communication Technology: Applications, Policy, and Impact, Fifth Edition, 2004.
- [5] Jones, S., Kovac, R., & Groom F. M. (2009). Introduction to Communication Technologies: A Guide for Non Engineers. Bocaraton, FL:CRC Press.
- [6] Vektanova, Bandwidth Explanation
- [7] Efendi, Ilham. Pengertian Algoritma Greedy. IT-Jurnal. Dikunjungi 10 Mei 2021 dari <https://www.it-jurnal.com/pengertian-algoritma-greedy/>
- [8] Girsang, Abba Suganda. (28 November 2017). Algoritma Dijkstra. Binus. Dikunjungi 9 Mei 2021 dari <https://mti.binus.ac.id/2017/11/28/algoritma-dijkstra/>
- [9] GeeksforGeeks. (10 November 2020). Introduction to Telephone Network. Dikunjungi 9 Mei 2021 dari <https://www.geeksforgeeks.org/introduction-to-telephone-network/>

### 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, 11 Mei 2021



Muhammad Fikri Naufal (13519158)