**MAKALAH**

GRAPH DRAWING

Disusun untuk memenuhi salah satu tugas Mata Kuliah Pengantar Informatika



Disusun oleh :

1. Kemal Maulana Fauzi (19.14.1.0018)
2. Zenery Malik Nur Syamsi (19.14.1.0042)

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS MAJALENGKA**

**2019**

# 

# KATA PENGANTAR

Syukur Alhamdulillah senantiasa kami panjatakan kehadirar Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga kami dapat menyelesaikan makalah ini guna memenuhi tugas kelompok untuk mata kuliah Pengantar Informatika dengan judul : **Graph Drawing.**

Kami menyadari bahwa dalam penulisan makalah ini tidak terlepas dari bantuan banyak pihak yang dengan tulus memberikan bantuan serta saran sehingga makalah ini dapat terselesaikan.

Kami menyadari sepenuhnya bahwa makalah ini masih jauh dari sempurna dikarenakan terbatasnya pengalaman dan pengetahuan yang kami miliki. Oleh karena itu, kami mengharapkan segala bentuk saran dan masukan bahkan kritik yang membangun dari berbagai pihak. Akhirnya kami berharap semoga makalah ini dapat memberikan manfaat bagi pembaca.

Majalengka, 13 Oktober 2019

Penyusun

# DAFTAR ISI

[KATA PENGANTAR](#_Toc23689066) i

[DAFTAR ISI](#_Toc23689067) ii

[BAB I PENDAHULUAN 4](#_Toc23689068)

[1. LATAR BELAKANG 4](#_Toc23689069)

[2. RUMUSAN MASALAH 4](#_Toc23689070)

[3. TUJUAN PENULISAN 4](#_Toc23689071)

[BAB II MATERI 5](#_Toc23689072)

[1. Pengertian Graph Drawing 5](#_Toc23689073)

[1.1. Masalah di Königsberg (7 crossing point on progel river) 6](#_Toc23689074)

[1.2. Urban Planning Problem 7](#_Toc23689075)

[1.3. Pemecahan Oleh Euler 8](#_Toc23689076)

[2. Proses Kerja Graph Drawing 10](#_Toc23689077)

[3. Contoh Perhitungan dan Penerapan 11](#_Toc23689078)

[BAB III IMPLEMENTASI 13](#_Toc23689079)

[BAB IV PENUTUP 15](#_Toc23689080)

[1. Kesimpulan 15](#_Toc23689081)

[2. Saran 15](#_Toc23689082)

[DAFTAR PUSTAKA 16](#_Toc23689083)

[PAPER DATA SRUCTURE 16](#_Toc23689084)

# 

# BAB I

**PENDAHULUAN**

1. LATAR BELAKANG

Teori graf merupakan pokok bahasan yang sudah tua usainya namun memiliki banyak terapan sampai saat ini. Graf digunakan untuk mempresentasikan objek-objek dikrit dan hubungan antara objek-objek tersebut. Representasi visual dari graf adalah dengan menyatakan objek dinyatakan sebagai noktah, bulata, aatau titik, sedangkan hubungan antara objek dinyatakan garis.

Secara kasar, graf adalah suatu diagram yang memuat informasi tertentu jika diinterpretasikan secara tepat. Dalam kehidupan sehari-hari, graf digunakan untuk menggambarkan berbagai macam struktur yang ada. Tujuannya adalah sebagai visualisasi objek-objek agar lebih mudah dimengerti.

1. RUMUSAN MASALAH
2. Apa pengertian Graph Drawing
3. Bagaimana proses kerja Graph Drawing
4. Contoh penerapan dan perhitungan Graph Drawing
5. Bagaimana implementasi Graph Drawing
6. TUJUAN PENULISAN

Untuk mengetahui pengertian, proses kerja, implementasi serta penerapan Graph Drawing.

# 

# BAB II

**MATERI**

1. Pengertian Graph Drawing

Graph sering digunakan untuk mempresentasikan sebuah objek dan hubungannya dengan objek lain. Sejarah teori graph bermula saat ahli matematika Swiss Leonhard Euler memecahkan masalah jembatan Königsberg. Masalah jembatan Königsberg adalah teka-teki lama mengenai kemungkinan menemukan jalan setapak di tujuh jembatan yang membentang di sepanjang sebuah sungai bercabang yang melewati sebuah pulau tapi dengan tanpa melewati jembatan dua kali. Euler berpendapat bahwa tidak ada jalan semacam itu, buktinya hanya mengacu pada sususan fisik jembatan, namun intinya dia membuktikan teorema pertama dalam teori graph. (Carlson, 2017).

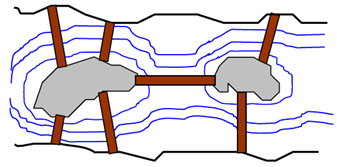
Seperti yang digunakan dalam teori graph, grafik istilah tidak mengacu pada grafik data, seperti grafik garis atau grafik batang. Sebaliknya, ini mengacu pada sekumpulan simpul (yaitu titik atau simpul) dan tepi (atau garis) yang menghubungkan simpul. Bila kedua simpul digabungkan lebih dari satu tepi, grafiknya disebut multi graph. Grafik tanpa loop dan paling banyak satu tepi antara dua simpul disebut grafik sederhana. Kecuali dinyatakan lain, grafik diasumsikan mengacu pada grafik sederhana. Bila setiap simpul dihubungkan oleh ujung ke setiap titik lainnya, grafik disebut grafik lengkap. Bila sesuai, arah dapat diberikan ke masing-masing ujung untuk menghasikan apa yang dikenal sebagai grafik terarah. (Carlson, 2017).

Secara umum istilah grafis berasal dari bahasa Inggris graph yang berarti membuat gambar dengan cara ditoreh atau digores, sedangkan Drawing adalah menggambar pada komputer tanpa menggunakan pensil dan lainnya. Gambar grafik atau graph drawing adalah representasi gambar dari simpul dan tepi grafik. Gambar ini tidak harus bingung dengan grafik itu sendiri: tata letak yang sangat beerbeda dapat sesuai dengan grafik yang sama. Dalam abstrak, yang terpenting adalah pasangan simpul mana yang dihubungkan oleh tepian. Namun, dalam beton, pengaturan simpul dan tepi dalam ini gambar memengaruhi pemahaman, kegunaan, biaya fabrikasi, dam estetika. Grafik sering digambarkan sebagai diagram simpul-tautan dimana simpul direpresentasikan sebagai disk, kotak, atau label tekstual dan ujung-ujungnya direpresentasikan sebagai segmen garis, polyline, atau kurva pada bidang Euclidean. Suatu **graph** didefinisikan oleh himpunan verteks dan himpunan sisi (edge). keterhubungan antara verteks. Biasanya untuk suatu graph G digunakan notasi matematis. Verteks menyatakan entitas-entitas data dan sisi menyatakan V adalah himpunan verteks dan E himpunan sisi yang terdefinisi antara pasangan-pasangan verteks. Dalam digraph didefinisikan juga terminologi-terminologi berikut ini. Predesesor dari suatu verteks x (ditulis Pred(x) adalah himpunan semua vertex yang adjacent ke x. Suksesor dari verteks x (ditulis Succ(x) adalah himpunan Pokok bahasan sebelumnya menjelaskan bahwa graf menampilkan visualisasi data dan hubungannya. Sedangkan jika berbicara masalah implementasi struktur data graf itu sendiri, isu utama yang dihadapi adalah bagaimana informasi itu disimpan dan dapat diakses dengan baik, ini yang dapat disebut dengan representasi internal

Secara umum terdapat dua macam representasi dari struktur data graf yang dapat diimplementasi. Pertama, disebut **adjacency list**, dan diimplementasi dengan menampilkan masing-masing simpul sebagai sebuah struktur data yang mengandung senarai dari semua simpul yang saling berhubungan. Yang kedua adalah representasi berupa **adjacency matrix** dimana baris dan kolom dari matriks (jika dalam konteks implementasi berupa senarai dua dimensi) tersebut. Merepresentasikan simpul awal dan simpul tujuan dan sebuah entri di dalam senarai yang menyatakan apakah terdapat sisi di antara kedua simpul tersebut.

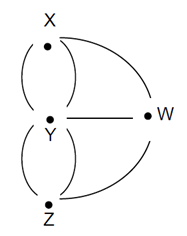
Graph pada dasarnya mempunyai komponen berupa simpul dan sisi dan pada graph tersebut sehingga membentuk graph terbuka dan graph tertutup sehingg membentuk sejumlah lintasan dan sirkuit. Sehingga pada teorema graph telah dapat menyelesaikan tanda tanya dalam penyelesaian teka-teki jembatan Königsberg dan dengan solusi masalah yang sama. (Widarsari, 2011).

1. Masalah di Königsberg (7 crossing point on progel river)

[](https://mti.binus.ac.id/files/2018/03/1-1.png)Euler adalah seorang ahli matematika yang mencoba untuk memecahkan teka-teki tersebut dan lebih dikenal dengan Jembatan Königsberg. (Widarsari, 2011). Terdapat 7 buah jembatan yang dapat menghubungkan 2 pulau dan juga sebuah sungai, seperti yang di tunjukan pada gambar 1.

**Gambar 1. Jembatan Königsberg** (Widarsari, 2011)**.**

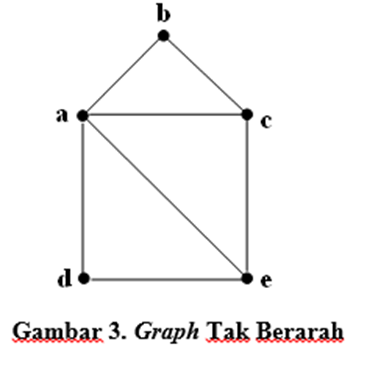
* 1. Urban Planning Problem

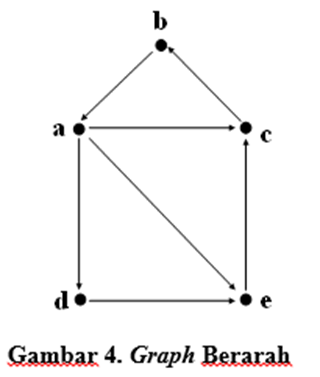
[](https://mti.binus.ac.id/files/2018/03/2.png)Dalam mencari solusi tersebut Euler mencoba metode dari masalah ini adalah dengan membentuk model dari jembatan Königsberg yang dikenal dengan multigraph, diperlihatkan pada gambar 2. Pada multigraph tersebut memliki 2 elemen yaitu himpunan verteks (titik/node) dan himpunan edge (garis) yang saling menghubungkan garis antar verteks. (Widarsari, 2011).

**Gambar 2. Representasi Multigraph Jembatan Königsberg** (Widarsari, 2011)**.**

Titik-titik yang diberi label X,Y,Z, dan W pada gambar 2 itulah yang disebut verteks dan dengan garis saling menghubungkan antar titik itulah yang disebut dengan edge.

Pada semua multigraph Euler telah membuat sebuah aturan yang dapat dipakai dalam mencari solusi pada jembatan Königsberg, sehingga aturan ini disebut dengan sebutan *Eulerian pat*h yang berbunyi : “*Andai kita mempunyai sebuah multigraph untuk beberapa pasang verteks sehingga akan terdapat sebuah path (lintasan) diantara verteks-verteks tersebut. Multigraph tersebut memiliki Eulerian path dan jika terdapat 0 atau 2 verteks tersebut maka banyak edge yang meninggalkan verteks tersebut akan berjumlah ganjil”.*

**Pada multigraph jembatan Königsberg tersebut memiliki 4 verteks dan pada ke-4 verteks tersebut memiliki edge sehingga meninggalkan verteks yang berjumlah ganjil. Maka Eulerian path tersebut tidak dimiliki pada multigraph jembatan Königsberg. Multigraph yang ditunjukan pada gambar 3. Tidak memiliki panah, sehingga disebut dengan *undirected graph* (graph tak berarah). Sehingga disebut dengan *directed graph* (graph berarah) adalah multigraph yang memiliki panah yang ditunjukan pada gambar 4.

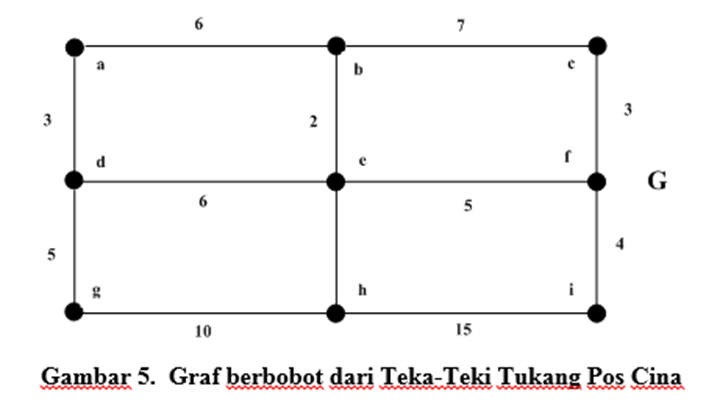


Sebuah *simple graph* (undirected graph) adalah pasangan dari G = (V , E) dimana :

1. V = Himpunan berhingga dari elemen yang disebut verteks
2. E = Sebuah relasi yang **irrefleksif** dan **simetri** pada V.

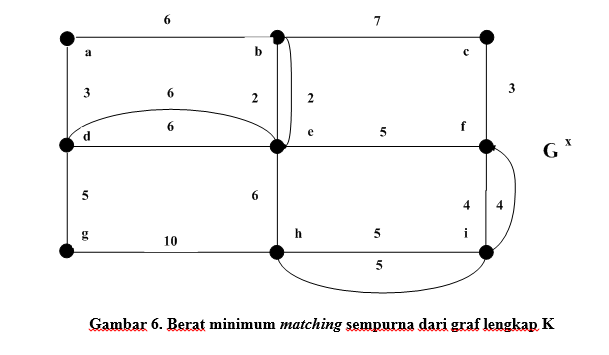
Pasangan berurutan pada E disebut edge dari graph yang berurutan. Lebih spesifik, jika e = (u,v) Î E, dikatakan bahwa edge e adalah antara u dan v (dan juga antara v dan u), dan dikatakan bahwa *u* ***adjacent ke*** *v.* Lebih jauh dapat dikatakan bahwa *e* ***incident ke*** *u* (dan juga ke v). Karena E simteri, maka kita dapat menotasikan e sebagai pasangan tak berurut {u,v}.

* 1. Pemecahan Oleh Euler

Hasil dari teka-teki jembatan Königsberg berdampak sungguh luar biasa terhadap ilmu pengetahuan. Dari teka-teki tersebut sangat berguna dan telah membuka jalan bagi terciptanya teorema baru yang disebut teorema baru yang disebut teorema graph. (Studi & Informatika).

Pada teka-teki tujuh jembatan Königsberg menghasilkan solusi permasalahan dengan dapat diperolehnya melalui dianalogikanya setiap jembatan sebagai sisi dan setiap daratan yang diperoleh sebagai simpul pada graph sehingga dapat terbentuknya graph yang lengkap. Dengan memperhitungkan derajat dalam graph dari setiap simpulnya maka dengan menggunakan metode seperti yang telah diungkapkan dalam pembuktian diatas, kita akan dapat mengetahui apakah graph tersebut merupakan suatu lintasan dimana setiap sisi dilalui hanya satu kali saja. (Studi & Informatika).

Fakta bahwa teorema graph yang dihasilkan oleh Euler telah menyelesaikan masalah berdasarkan teka-teki pada jembatan Königsberg yang menyatakan hubungan tersendiri antara jaringan spasial (seperti jalur transportasi) pada graph. Dalam memodelkan jaringan spasial, sebagai ada beberapa tambahan yaitu selain simpul dan sisi adalah biasanya diberikan sebuah nilai lainnya yang menyatakan kuantitas segmen jalan yang diwakili oleh sisi pada graph tersebut. Maka selanjutnya kita dapat mencari suatu rute pada jaringan spasial tersebut yaitu menggunakan sarana atau beban yang minimal seperti pada solusi teka-teki Tukang Pos Cina. (Studi & Informatika).

1. [](https://mti.binus.ac.id/files/2018/03/6.png)Proses Kerja Graph Drawing

Proses kerja graph drawing adalah dalam menentukan waktu tunggu total dan mengatur pergerakan arus lalu lintas. Lampu lalu lintas adalah lampu yang mengendalikan arus lalu lintas yang terpasang dipersimpangan jalan, tempat penyebrangan pejalan kaki (*zebra crooss*), dan tempat arus lalu lintas lainnya. Lampu ini menandakan waktu kendaraan harus berjalan dan berhenti secara bergantian dari berbagai arah. Lampu lalu lintas yang tersedia di persimpangan jalan mempunyai beberapa tujuan antara lain untuk menghindari hambatan karena adanya perbedaan arus bagi pergerakan kendaraan, memfasilitasi pejalan kaki agar dapat menyebrang dengan aman dan mengurangi tingkat kecelakaan yang diakibatkan oleh tabrakan karena perbedaan arus jalan. Banyak metode yang dapat digunakan untuk menentukan waktu total, salah satunya adalah graf kompatibel. Graf kompatibel digunakan secara luas dalam memecahkan masalah yang melibatkan peraturan data dalam urutan tertentu. Arus lalu lintas tertentu dapat disebut kompatibel jika kedua arus tersebut tidak akan menghasilkan kecelakaan yang disebabkan oleh kendaraan.

Masalah transportasi secara umum dan lalu lintas pada khususnya adalah merupakan fenomena yang terlihat sehari-hari dalam kehidupan manusia. Semakin tinggi tingkat mobilitas warga suatu kota, akan semakin tinggi juga tingkat perjalanannya. Peraturan arus lalu lintas pada dasarnya dimaksudkan untuk bagaimana pergerakan kendaraan pada masing-masing pergerakan kendaraan dapat bergerak secara bergantian sehingga tidak saling mengganggu antar arus yang ada. Sistem pengatur lalu lintas adaptif dan sinkron telah banyak digunakan di beberapa negara maju. Dengan adanya sistem pengaturan lalu lintas adaptif, durasi merah dan hijau disesuaikan dengan kepadatan kedatangan kendaraan. Dengan sistem ini, diharapkan durasi waktu tunggu kendaraan dari semua arah cenderung sama dan tidak akan melebihi satu siklus. Dengan kata lain tidak ada kendaraan yang mengalami isyarat merah dua kali. Adanya sistem pengaturan lalu lintas sinkron untuk mengurangi waktu tempuh jalan utama. Pengaturan saling disinkronkan satu dengan yang lain agar sebagian besar kendaraan di jalan utama tidak terlalu lama menanti isyarat hijau. Pengaturan lalu lintas sinkron digunakan untuk mengurangi durasi waktu tunggu kendaraan di jalan utama.

1. Contoh Perhitungan dan Penerapan
   1. Pengaplikasikan Graf Pada Jaringan Internet.

Teori graf dapat diimplementasikan dalam jaringan internet dan merupakan dasar teori yang sangat penting dalam jaringan internet, bak dalam mengatasi masalah-masalah dalam jaringan internet. Matematika diskrit sangat penting untuk mengembangkan ilmu pengetahuan, agar selanjutnya dapat membantu kehidupan masyarakat.

* 1. Pemanfaatan Graf Sebagai Alat Bantu Pemecah Masalah Rumus Molekul.

Penentuan rumus molekul senyawa kimia merupakan salah satu masalah yang sering dihadapi oleh para kimiawan. Dalam kaitan ini diperlukan bobot molekul dari senyawa yang dianalisis. Salah satu cara untuk menentukan bobot molekul ini adalah dengan menggunakan alat yang dikenal sebagai spectrometer masa berdaya pisah rendah yang menghasilkan data spectra dari senyawa yang dianalisis. Teori graf merupakan salah satu cabang ilmu matematika yang diterapkan dalam kehidupan sehari hari. Graf mempelajari himpunan titik, dengan hubungan antara titik objek yang dinyatakan dalam bentuk garis. Hal ini sangat realistik menghubungkan teori graf sebagai alat bantu pemecah masalah penentuan rumus molekul. Dalam suatu graf molekul, simpul-simpul digunakan untuk menyatakan atom-atom, sedangkan sisi-sisi menyatakan ikatan kimia antar atom-atom dalam molekul.

* 1. Teknologi Graph Database

Dalam komputasi, database graf adalah database yang menggunakan struktur grafik untuk query semantik dengan node, edge, dan properti untuk mewakili dan menyimpan data. Konsep kunci dari system adalah grafik (atau edge atau hubungan), yang secara langsung menghubungkan item data ditoko. Hubungan memungkinkan data ditoko dihubungkan secara langsung, dan dalam banyak kasus diambil dengan satu operasi.

Mengambil data dari database grafik memerlukan bahasa query selain SQL, yng dirancang untuk manipulasi data dalam sistem relasional dan oleh karena itu tidak dapat “secara elegan” menangani pelacakan grafik. Pada tahun 2017, tidak ada bahasa query grafik tunggal yang telah diadopsi secara universal dengan cara yang sama seperti SQL untuk basis data relasional, dan ada beragam sistem yang paling sering terkait erat dengan satu produk. Beberapa upaya standardisasi telah terjadi, yang mengarah ke bahasa query multi-vendor seperti SPARQL, dan Cypher. Selain memiliki antarmuka bahasa query, beberapa database grafik diakses melalui antarmuka pemrograman aplikasi (API).

# 

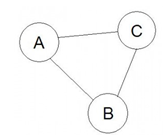
# BAB III

**PEMBAHASAN**

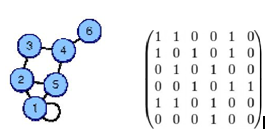
1. Implementasi Graph Pada Struktur Data

Secara umum terdapat dua macam representasi dari struktur data graf yang dapat diimplementasi. Pertama, disebut adjacency list, dan diimplementasi dengan menampilkan masing-masing simpul sebagai sebuah struktur data yang mengandung senarai dari semua simpul yang saling berhubungan. Yang kedua adalah representasi berupa adjacency matrix dimana baris dan kolom dari matriks (jika dalam konteks implementasi berupa senarai dua dimensi) tersebut merepresentasikan simpul awal dan simpul tujuan dan sebuah entri di dalam senarai yang menyatakan apakah terdapat sisi di antara kedua simpul tersebut.

Dalam teori graf, adjacency list merupakan bentuk representasi dari seluruh sisi atau busur dalam suatu graf sebagai suatu senarai. Simpul-simpul yang dihubungkan sisi atau busur tersebut dinyatakan sebagai simpul yang saling terkait. Dalam implementasinya, hash table digunakan untuk menghubungkan sebuah simpul dengan senarai berisi simpul-simpul yang saling terkait tersebut. Salah satu kekurangan dari teknik representasi ini adalah tidak adanya tempat untuk menyimpan nilai yang melekat pada sisi. Contoh nilai ini antara lain berupa jarak simpul, atau beban simpul.



Adjacency Matrix, Adjacency Matrix merupakan representasi matriks nxn yang menyatakan hubungan antar simpul dalam suatu graf. Kolom dan baris dari matriks ini merepresentasikan simpul-simpul, dan nilai entri dalam matriks ini menyatakan hubungan antar simpul, apakah terdapat sisi yang menghubungkan kedua simpul tersebut.Pada sebuah matriks nxn, entri non-diagonal aij merepresentasikan sisi dari simpul i dan simpul j. Sedangkan entri diagonal aii menyatakan sisiKelebihan dari adjacency matrix ini adalah elemen matriksnya dapat diakses langsung melalui indeks, dengan begitu hubungan ketetanggan antara kedua simpul dapat ditentukan dengan lan gsung. Sedangkan kekurangan pada representasi ini adalah bila graf memiliki jumlah sisi atau busur yang relative sedikit, karena matriksnya bersifat jarang yaitu hanya mengandung elemen bukan nol yang sedikit. Kasus seperti ini merugikan, karena kebutuhan ruang memori untuk matriks menjadi boros dan tidak efisien karena komputer menyimpan elemen 0 yang tidak perlu.



# 

# BAB IV

**PENUTUPAN**

1. Kesimpulan

Struktur data Graph merupakan salah satu bahan dasar pembuatan program. Pemakaian struktur data yang tepat di dalam proses pemrograman, akan menghasilkan algoritma yang jelas dan tepat sehingga menjadikan program secara keseluruhan lebih sederhana. Array merupakan bagian dari struktur data yaitu termasuk kedalam struktur data sederhana yang dapat di definisikan sebagai pemesanan alokasi memory sementara pada komputer. Apabila kita membuat program dengan data yang sudah kita ketahui batasnyamaka kita menggunakan Array (type data statis), namun apabila datanya belum kita ketahui batasnya maka gunakan pointer (type data dinamis)..

1. Saran

Dalam makalah singkat ini penulis ingin menyarankan kepada rekan mahasiswa hendaknya kita membuat tugas yang dibebankan oleh dosen pengasuh kita yang berupa makalah, kita membuat sendiri agar kedepannya kita menjadi mahasiswa yang benar-benar siap pakai di kalangan masyarakat maupun dunian kerja.

# DAFTAR PUSTAKA

<https://shafiqhusein.wordpress.com/2010/05/18/graph-dan-implementasi-graph-dalam-struktur-data/>

<https://dinda-dinho.blogspot.com/2013/11/pengertian-dan-konsep-graph-dalam.html>

[*https://mti.binus.ac.id/2018/03/05/teori-graph-sejarah-dan-manfaatnya/*](https://mti.binus.ac.id/2018/03/05/teori-graph-sejarah-dan-manfaatnya/)

# PAPER DATA SRUCTURE

[*https://scholar.google.com/scholar?hl=id&as\_sdt=0%2C5&q=graph+drawing&btnG=&oq=graph*](https://scholar.google.com/scholar?hl=id&as_sdt=0%2C5&q=graph+drawing&btnG=&oq=graph)*+*

[*https://scholar.google.com/scholar?hl=id&as\_sdt=0%2C5&q=heuristic+graph+drawing&oq=graph+drawing*](https://scholar.google.com/scholar?hl=id&as_sdt=0%2C5&q=heuristic+graph+drawing&oq=graph+drawing)

[*https://scholar.google.com/scholar?hl=id&as\_sdt=0%2C5&q=implementation+graph+drawing&btnG=&oq=graph+drawing*](https://scholar.google.com/scholar?hl=id&as_sdt=0%2C5&q=implementation+graph+drawing&btnG=&oq=graph+drawing)

[*https://scholar.google.com/scholar?hl=id&as\_sdt=0%2C5&q=graph+drawing+algorithms&oq=graph+drawing*](https://scholar.google.com/scholar?hl=id&as_sdt=0%2C5&q=graph+drawing+algorithms&oq=graph+drawing)

*https://scholar.google.com/scholar?hl=id&as\_sdt=0%2C5&q=visualization+graph+drawing+algorithms&oq=graph+drawing+algorithms*