APLIKASI GRAF BERARAH UNTUK MENGETAHUI HUBUNGAN ANTAR UNSUR-UNSUR PADA SEBUAH HIMPUNAN

OLEH :

A S N I A H

H 111 98 004

JURUSAN MATEMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS HASANUDDIN

MAKASSAR

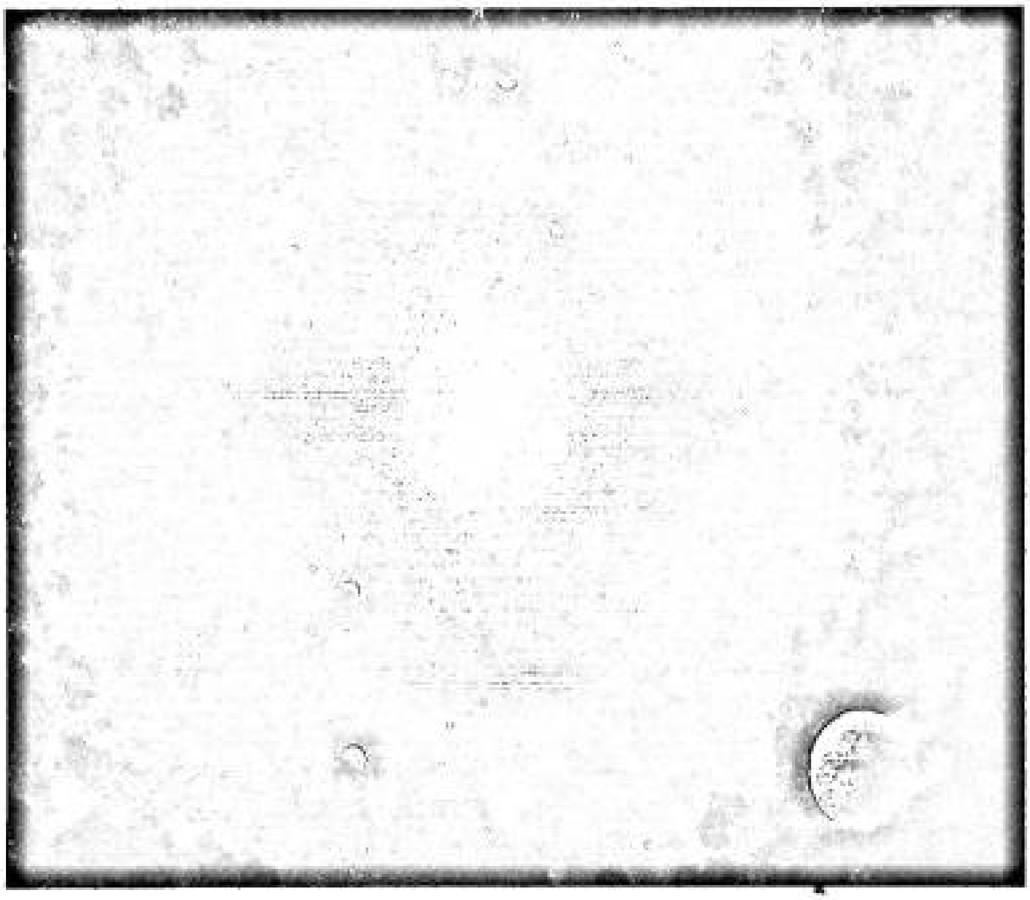
2004

KUPERSEMBAHKAN TULISAN INI UNTUK:

KAKEKKU, ALMARHUM AYAHKU ,IBUNDA

&

TANTEKU TERCINTA



AKU TAK MAMPU MEMPERSAMAKAN ENGKAU DENGAN BULAN PURNAMA YANG HANYA BERSINAR SEMALAM SAJA TAPI KALIAN ADALAH DARAH MERAH YANG MENGALIR DALAM URAT NADI KU

UNTUK
ORANG TERKASIHKU

Aku ingin mencintaimu secara sederhana
seperti yang tak sempat terisyaratkan oleh mendung
kepada awan yang menjadikanya hujan
seperti yang tak sempat terisyaratkan oleh kayu
kepada api yang menjadikannya abu

(for some one: I love you forever)

APLIKASI GRAF BERARAH UNTUK MENGETAHUI HUBUNGAN ANTAR UNSUR-UNSUR PADA SEBUAH HIMPUNAN

Skripsi ini diajukan sebagai salah satu syarat untuk
Mencapai gelar Sarjana Matematika pada Fakultas M IPA
Universitas Hasanuddin
Makassar

OLEH

A S N I A H

H 111 98 004

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN

2004

APLIKASI GRAF BERARAH UNTUK MENGETAHUI HUBUNGAN ANTAR UNSUR-UNSUR PADA SEBUAH HIMPUNAN

Disetujui oleh:

Pembimbing Utama	Pembimbing Pertama
Nurdin, S.Si, M.Si	**Drs. M. Saleh AF**
NIP. 132 259 080	**NIP. 130 675 575**

Pada tanggal, ...Februari 2004

KATA PENGANTAR

بسم الله الرحمن الرحيم

Alhamdulillah, penulis panjatkan kehadirat ALLAH SWT, karena atas Rahmat dan Hidayah-Nya sehingga penulis diberikan kekuatan untuk menyelesaikan skripsi dengan judul " Aplikasi Graf Berarah Dalam Mengetahui Hubungan Antar Unsur-unsur Pada Sebuah Himpunan", yang disusun sebagai salah satu syarat untuk mencapai gelar Sarjana Sains pada Jurusan Matematika Fakultas MIPA Universitas Hasanuddin.

Dalam penyusunan skripsi ini, penulis menyadari adanya banyak kekurangan dan kelemahan yang penulis miliki. Dengan segala keterbatasan itu, bantuan dan dorongan dari berbagai pihak sangatlah membantu penulis untuk menyelesaikan skripsi ini.

Sembah sujud Ananda haturkan pada Ayahanda **Mappeasse** (almarhum) dan Ibunda **Itihang** , terima kasih sebesar-besarnya atas bimbingan Ayah dan Ibunda sejak Ananda kecil hingga dewasa seperti saat ini.

Dalam kesempatan ini pula, penulis ingin mengucapkan terima kasih sebesar-besarnya kepada yang terhormat :

1. Kakek H. Karatte, tante Hj. Sumaenah, tante Hj. Maedar tercinta; tante Hj. Musdalifa Om' Alwi, Om' Rusdi&tante Hj. Rawasia, Om' Bahtiar& tante Nunu , Bapak Abd.Hamid & tante Hj. Hadia, Kak Rukma & Kak tajud, Kak Yuli tersayang ; Adinda tersayang : Rahmawati, Asirah, Syam-Syam, Afwan, Andien ; Kanda tercinta; Kak Ullah , Kak Afdhal, Kak Faizal, Kak Anto , Kak Dani , dan tersayang Tante Hj. Naidah dan tante Hj. Maryam . Terima kasih atas dukungan dan do'anya.
2. Bapak Nurdin S.Si M.Si selaku Pembimbing Utama dan Bapak Drs. Muh. Saleh AF selaku Pembimbing Pertama.

3. Bapak Drs. Muh. Zakir, MSi. selaku Ketua Jurusan Matematika maupun serta Bapak Drs. Syamsuddin Toaha, M.Sc. selaku Sekretaris Jurusan Matematika.
4. Bapak dan Ibu Dosen Jurusan Matematika FMIPA Unhas serta seluruh staff pegawai Jurusan Matematika FMIPA Unhas.
5. Teruntuk sahabat-sahabatku tercinta Dalmi SSi., Pathrayuna, SSi. Kamariah SSi; Rosnawati SSi beserta orang tuanya, . Terima kasih atas dukungan, nasehat dan motivasi kalian selama ini.
6. Teruntuk adik-adikku tercinta : Arham, Ismail, Firsam, Nini, Fitrah,Wawan Iyan, Imma, . Terima kasih atas dukungannya.
7. Rekan-rekan Tercinta Angkatan '98 : Marwati Majid, SSi; Rasmi Abdullah, SSi. ; Fitriyah Hidayati, SSi Mbak Yanti Dwi Indra Wahyuni,, S.Si & Zulkhaersyam, S.Si; Lukman Syafie, SSi.; Ibrahim;Yuni; Nanna; Anca; Maria, SSi; Sandra; Nita, SSi; Ida, SSi; Adi; Ichal; Edy; Darnah, SSi ; Ikas, SSi; Faika, SSi; Asli; Hendra SSi; Iwan; Ayu SSi; Lina SSi; Eva SSi; Rini SSi; Tuti SSi; Astri, SSi; Vera, SSi; Ode'; Nurarfiah SSi; Fatihiyah; Robiatul; Nino SSi'; Pepen; Azis; Aspiah (Alm.); Ronald; Hamka; Rahman SSi; Rahmatiah; Arman SSi ; Chery, SSi; Tina, SSi; Ina; Asra; Dedi; Cully; Bram; dan Idris serta rekan HIMATIKA '99, '00, '01.
8. Teruntuk kakak-kakakku tersayang : Kak Adi, Kak Marhatang, Kak Nawir Muh. Fitrah, Muh.Farid semua crew Pondok Sofa.
9. Buat A. Surgawati & Pak' Hamza terimah kasih atas bantuannya selama ini.
10. Ma' tina & tante lenteng yang banyak mengulurkan bantuannya demi kelancaran skripsi ini.
11. Tak lupa pula saya Ucapkan Banyak terima kasih kepada Ibu Hj. Rosdiana, Ibu Hj. Nurfaisah, Ibu Nurhayati, Ibu Samri, Pak Nasir, Ibu Hj. Murni, Ibu Nurhaedah serta seluruh Staf pengajar SMP Panincong yang senangtiasa memberiku nasehat, semangat, maupun bantuan materi dalam menjalani masa masa perkuliahan

12. Tak lupa pula saya Ucapkan Banyak terima kasih kepada Pak' Semmauna, Pak' budi, Pak' Udil, Pak' Hatta, Pak H. Hasbi Ibu Hj. Rosmania serta seluruh staf pengajar SMU negeri Cangadi yang senangtiasa memberikan nasehat, memotifasiku dalam mengarungi titik jenuh dalam kuliah.
13. Khusus buat Some One (NAF) Terima kasih atas segalanya Semoga kau bisa menyusul secepatnya, senantiasa dalam lindungan Allah Swt dan Sukses Selalu.

Terima kasih atas segala bantuan, saran dan do'a yang telah diberikan kepada penulis, semoga Allah SWT memberikan balasan yang setimpal, Amin.

Penulis menyadari bahwa skripsi ini masih terdapat banyak kekurangan, oleh karena itu saran dan kritik yang sifatnya membangun sangat penulis harapkan untuk penyempurnaan selanjutnya. Semoga skripsi ini dapat bermanfaat bagi kita semua, Amin.

Akhirnya, untuk yang tercinta Kakekku, Ibunda,dan Tanteku kudoakan semoga senantiasa dalam lindungan Allah Swt serta selamat di Dunia dan lebih-lebih di Akhirat. Amiin Ya Rabball Alamiin.. Almarhum Ayahanda semoga Allah, SWT memberikan tempat yang suci di sisi-Nya .

Makassar, Februari 2004

Penulis

ABSTRAK

Graf berarah merupakan salah satu graf khusus pada teori graf. Representasi graf berarah dari suatu masalah akan memudahkan dalam mencari solusinya.

Dalam tulisan ini akan dibahas penggunaan matriks ketetanggaan untuk merepresentasikan objek-objek diskrit dan hubungan objek-objek tersebut. Disamping itu, juga akan diperlihatkan salah satu aplikasi graf berarah pada penjadwalan kegiatan proyek dan rute perjalanan antar kota .

Algoritma BFS digunakan untuk menentukan waktu minimum pada suatu proyek dan penggunaan algoritma jalur terpendek adalah salah satu alternatif yang digunakan untuk mencari lintasan terpendek pada rute perjalanan antar kota.

ABSTRACT

Digraph is one of special graph on graph theory. Represention digraph of one problem with be easy to find its solution.

In this paper we study about application to representation of discreatobjects and its relation. On the other side, we study about application of digraf in scheduling of project activities and journey route intercities

A BFS algorithm used to determine the minimum time on the project and the shortest path algorithm to find the shortest route in intercities.

DAFTAR ISI

DAFTAR SIMBOL

LAMBANG	ARTI
$+$	Tambah
$=$	Sama dengan
$-$	Kurang
$>$	Lebih besar dari
$<$	Lebih kecil dari
$\sum$	Sigma
$\forall$	Untuk setiap
$\exists$	Terdapat
$\ni$	Sehingga
$\rightarrow$	Menuju
$\subseteq$	Bagian
$\cup$	Gabung
∞	Tak hingga
ω	Omega
ϕ	Kosong

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1 Latar Belakang

Di dunia ini tak terhitung banyaknya contoh himpunan yang terdiri dari sejumlah berhingga anggota di mana terdapat suatu hubungan di antara anggota-anggota himpunan tersebut. Misalnya himpunan itu terdiri dari sekumpulan orang, binatang, negara, perusahaan, tim olah raga dan hubungan diantara dua anggota A dan B dari himpunan seperti binatang A memberi makan pada binatang B, negara A secara militer mendukung negara B, perusahaan A menjual produksinya ke perusahaan B, tim olah raga A secara konsisten mengalahkan tim olah raga B, atau kota A mempunyai penerbangan langsung ke kota B.

Dengan munculnya berbagai macam problem maka untuk menghadapi dan memecahkan problem tersebut, komputer merupakan salah satu alat bantu yang ampuh. Namun terlebih dahulu kita cari model yang tepat. Misalnya problem numerik dibawa ke model matematika, problem menentukan kuat arus dalam suatu rangkaian listrik dibawa ke model persamaan linear, problem meramalkan pertumbuhan populasi dibawa ke model persamaan deferensial.

Problem optimasi misalnya mencari pola penjadwalan yang baik dapat dibawa ke model kombinasi dan graf. Selanjutnya apabila kita telah berhasil menentukan model yang sesuai untuk problem tersebut, kita mencari dan menentukan algoritma untuk menyelesaikannya. Dari algoritma ini, kita membuat suatu program dalam suatu bahasa pemrograman tertentu yang siap untuk dijalankan sehingga diperoleh penyelesaian yang diharapkan.

Dalam tulisan ini, pembahasan dititik beratkan pada penggunaan graf berarah dalam menganalisis unsur-unsur pada himpunan.

Berdasarkan uraian di atas maka penulis tertarik untuk mengangkat dan mengajukan masalah tersebut sebagai tugas akhir dengan judul :

APLIKASI GRAF BERARAH UNTUK MENGETAHUI HUBUNGAN ANTAR UNSUR-UNSUR PADA SEBUAH HIMPUNAN

1.2 Tujuan dan Manfaat Penulisan

a. Tujuan

1. Menyatakan hubungan antara unsur-unsur sebuah himpunan dalam bentuk matriks.
2. Mencari banyaknya jalan dengan panjang r yang mungkin terjadi dari satu simpul ke simpul yang lain dari suatu graf berarah. Misalnya kita ingin mengetahui apakah suatu posisi tertentu dapat dicapai dari suatu posisi lain dengan serangkaian langkah yang dibenarkan.
3. Memeriksa keterhubungan graf dengan menggunakan Algoritma Keterhubungan.
4. Menentukan Waktu minimum yang digunakan untuk menyelesaikan suatu proyek dan menggunakan algoritma BFS untuk mengecek kebenaranya.
5. Menentukan lintasan terpendek antara dua simpul pada graf berarah dan menggunakan Algoritma lintasan terpendek untuk mengecek kebenarannya.

b. Manfaat

Penulisan ini diharapkan memberikan manfaat dalam membuat model graf berarah .Misalnya Penjadwalan (scheduling) sejumlah tugas pada suatu sistem.

Representasi graf berarah seringkali merupakan bentuk penyelesaian yang memudahkan.

1.3 Rumusan Masalah

Adapun rumusan masalah yang akan dibahas sebagai berikut:

1. Membuat matriks ketetanggaan dari suatu Graf berarah.
2. Bagaimana menggunakan matriks ketetanggaan untuk menghitung banyaknya jalan dengan panjang r dari dua simpul pada graf berarah.
3. Bagaimana menggunakan Algoritma keterhubungan (simulasi) untuk memeriksa apakah graf tersebut terhubung atau tidak.
4. Bagaimana menentukan waktu minimal pada suatu proyek supaya selesai dan membuat jadwal kegiatannya.
5. Bagaimana menentukan lintasan terpendek antar dua kota dan menggunakan Algoritma lintasan terpendek (simulasi) untuk mengecek kebenaranya.

1.4 Sistematika Pembahasan

Secara garis besar sistematika pembahasan dibagi menjadi:

- Bab I membahas latar belakang masalah, tujuan penulisan, rumusan masalah , tinjauan pustaka serta sistematika pembahasan.
- Bab II berisi teori pendukung yang didalamnya membahas definisi graf, jenis-jenis graf, terminologi graf , representasi graf.
- Bab III membahas penggunaan matriks ketetanggaan dalam mengetahui banyaknya jalan Penggunaan matriks ketetanggaan pada graf berarah untuk mengetahui banyaknya jalan dengan panjang *r* yang mungkin ($r = 1, 2, 3, \ldots$)dari simpul v_i ke simpul v_j

dan aplikasi graf berarah dalam mengetahui hubungan antara unsur-unsur pada sebuah himpunan .

Penyelesaian suatu masalah dengan mengaplikasikan graf berarah sebagai suatu cara untuk mencari penyelesaian.

- Bab IV kesimpulan dan saran.

BAB II

KONSEP DASAR GRAF

Dalam bab ini, akan diperkenalkan *konsep dasar, definisi, dan notasi suatu graf.*

2.1 Definisi Graf

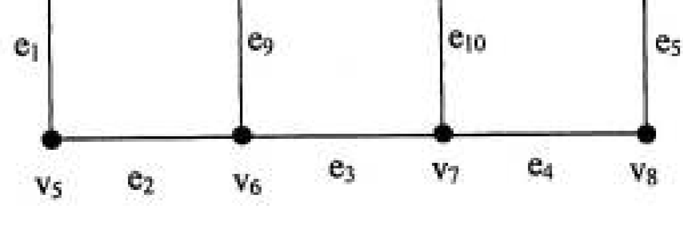
Definisi *graf* G akan disusun dengan menggunakan notasi-notasi himpunan.

Definisi 2.1

Graf G adalah pasangan himpunan (V,E) dimana V adalah himpunan tak kosong dari elemen yang disebut simpul ; dan E adalah himpunan dari pasangan tak terurut $\{u,v\}$ yang disebut rusuk.(Slamet Sumantri dan Makalive hendrik, 1992)

Jika sebuah *graf* G mempunyai n *simpul* dan m *rusuk* maka himpunan V dan E dapat ditulis sebagai $V = \{v_1, v_2, v_3, ... v_n\}$dan $E = \{e_1, e_2, ... e_m\}$dimana setiap e_k berbentuk $e_k = \{v_i, v_j\}$ $v_i, v_j \in V$.

Contoh :



gambar 1

Graf $G=(V,E)$ seperti pada gambar 1 terdiri dari 8 simpul dan 10 rusuk yaitu $V=\{v_1,v_2,v_3,v_4,v_5,v_6,v_7,v_8\}$ dan $E=\{e_1,e_2,e_3,e_4,e_5,e_6,e_7,e_8,e_9,e_{10}\}$ dimana

$e_1=\{v_1,v_5\}, e_2=\{v_5,v_6\}, e_3=\{v_6,v_7\}, e_4=\{v_7,v_8\}, e_5=\{v_8,v_4\}, e_6=\{v_4,v_3\},$

$e_7=\{v_3,v_2\}, e_8=\{v_2,v_1\}, e_9=\{v_2,v_6\}, e_{10}=\{v_3,v_7\}$

2. 2 Jenis-jenis Graf

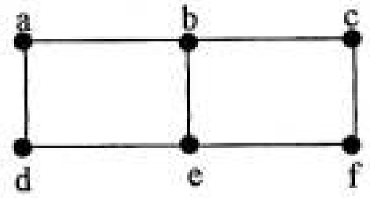
Graf dapat dikelompokkan menjadi beberapa jenis bergantung pada sudut pandang pengelompokannya. Pengelompokan graf dapat dipandang berdasarkan ada tidaknya rusuk ganda atau loop, berdasarkan simpul, atau berdasarkan arah pada sisi.

Berdasarkan ada tidaknya loop atau rusuk ganda pada suatu graf maka secara umum dapat digolongkan menjadi 2 jenis:

1. Graf sederhana (simple graf)

Definisi 2.2

Suatu *graf G* disebut *graf sederhana* jika G tidak mempunyai *loop* dan tidak mempunyai *rusuk ganda*.(Robin J. Wilson, 1979)



gambar 2

2. *Graf tak sederhana*

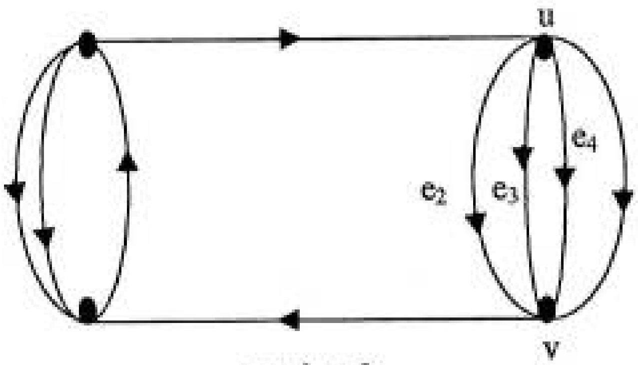
Definisi 2.3

Graf yang mengandung rusuk ganda atau loop dinamakan graf tak sederhana.(Munir rinaldi , 2001)

Ada dua macam graf tak sederhana yaitu **graf ganda** dan **graf semu.**

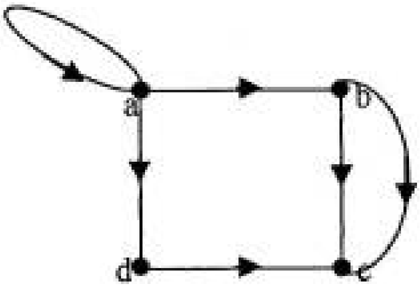
Graf ganda adalah graf yang mengandung rusuk ganda.(Munir rinaldi , 2001)

Jika $e_1 = (u, v) = e_2$, maka e_1 *dan* e_2 disebut *rusuk ganda.*



gambar 3

Graf semu adalah graf yang mengandung rusuk ganda dan loop.(Munir rinaldi , 2001)



gambar 4

Berdasarkan jumlah simpul pada suatu graf, maka secara umum graf dapat digolongkan menjadi 2 jenis :

1. *Graf berhingga*

Definisi 2.4

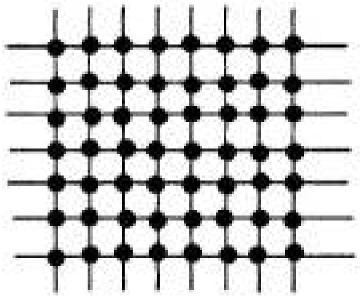
Graf berhingga adalah graf yang jumlah simpulnya berhingga.(Munir rinaldi , 2001) Contoh pada gambar 4 yang mempunyai simpul 4.

2. *Graf tak berhingga*

Definisi 2.5

Graf tak berhingga adalah graf yang jumlah simpulnya tidak berhingga. (Munir rinaldi , 2001)

Contohnya :



gambar 5

Berdasarkan orientasi arah pada rusuk, maka secara umum graf dibedakan atas 2 jenis :

1. Graf tak berarah

Definisi 2.6

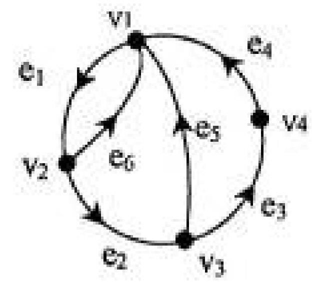
Graf yang rusuknya tidak mempunyai arah. Pada graf tak berarah urutan pasangan simpul tidak diperhatikan. Jadi $\{v_i, v_j\} = \{v_j, v_i\}$.(Munir rinaldi , 2001)

Seperti pada gambar 1.

2. Graf berarah .

Definisi 2.7

Graf berarah G adalah graf G dimana setiap rusuk-rusuknya memiliki arah.(Anthony Ralston dan Stephen B. Maurer,1991)



gambar 6

Graf $G=(V,E)$ seperti pada gambar 6 adalah graf berarah yang terdiri dari 4 simpul dan 6 rusuk $V=\{v_1,v_2,v_3,v_4\}$ dan $E=\{e_1,e_2,e_3,e_4,e_5,e_6\}$ dimana $e_1=(v_1,v_2), e_2=(v_2,v_3), e_3=(v_3,v_4), e_4=(v_4,v_1), e_5=(v_3,v_1), e_6=(v_2,v_1)$.

2.3 Terminologi Graf

Dalam pembahasan mengenai graf, kita akan sering menggunakan terminologi (istilah) yang berkaitan dengan graf. Beberapa terminologi yang digunakan seperti berikut.

Definisi 2.8

Misal u dan v simpul di graf G. Simpul v dikatakan *tetangga* u jika terdapat rusuk e sehingga $e=\{u,v\}$.(Slamet Sumantri dan Mekalive hendrik , 1992) (Untuk graf tak berarah)

Contohnya pada gambar 1 v_2 bertetangga dengan $v_1, v_3, dan\ v_6$. Misalkan $e_1=\{u,v\}$ maka u disebut bertetangga dengan v dan e disebut *insidensi* dengan $u\ dan\ v$.

Dua simpul v_i $dan\, v_j$ dikatakan bertetangga jika $(v_i, v_j) \in$ E atau $(v_j, v_i) \in$ E.(Anthony Ralston dan Stephen B. Maurer , 1991)

Definisi 2.9

Derajat dari simpul $v\, di\, G$ *adalah* banyaknya simpul yang bertetangga dengan v. Jika simpul v mempunyai derajat *nol*, maka v disebut *simpul terpencil* atau *simpul terisolasi*.(Slamet Sumantri dan Mekalive hendrik , 1992)

Definisi 2.10

Jalan atau walk dari v_1 *ke* v_n pada suatu graf G adalah barisan simpul dan rusuk berganti-ganti:

$$v_1, e_1, v_2, e_2 .... e_{n-1}, v_n$$

dimana $e_i = \{v_i, v_{i+1}\}$ untuk graf tak berarah dan (v_i, v_{i+1}) untuk digraf.(H. S., Suryadi , 1994)

Definisi 2.11

Lintasan adalah jalan yang setiap simpulnya berbeda-beda. (Slamet Sumantri dan Makalive hendrik,1992)

Macam-macam Lintasan :

Lintasan Hamilton yaitu lintasan yang setiap rusuknya berbeda-beda / hanya sekali dilewati simpulnya.(Anthony Ralston dan Stephen B. Maurer, 1991)

Lintasan Eulerian yaitu lintasan yang setiap rusuknya dilewati sekali namun simpulnya boleh dilewati lebih dari satu kali.(Anthony Ralston dan Stephen B. Maurer, 1991)

Definisi 2.12

Jalan disebut *jalan tertutup* bila $v_1 = v_n$. Dalam hal lain disebut *jalan terbuka* yang menghubungkan $v_1 \; dan \, v_n$.(H.S., Suryadi,1994)

Definisi 2.13

Jalan tertutup yang semua simpulnya berbeda dinamakan lingkaran.(Slamet Sumantri dan Makalive hendrik , 1992)

Lingkaran Hamilton yaitu lintasan tertutup yang hanya sekali dilewati simpulnya.(Anthony Ralston dan Stephen B. Maurer ,1991)

Lingkaran Eulerian yaitu lintasan tertutup yang hanya sekali dilewati rusuknya.(Anthony Ralston dan Stephen B. Maurer ,1991)

Definisi 2.14

Suatu graf G disebut *terhubung*, bila untuk setiap dua simpul dari graf G selalu terdapat lintasan yang menghubungkan kedua simpul tersebut.(H. S., Suryadi ,1994)

Definisi 2.15

Jarak antara dua simpul $u \; dan \; v$ dalam graf G, ditulis $d(u,v)$, adalah panjang lintasan terpendek antara kedua simpul $u \, dan \, v$ tersebut. (H. S., Suryadi ,1994)

Definisi 2.16

Panjang lintasan adalah banyaknya rusuk dalam barisan yang membentuk lintasan tersebut.(Anthony Ralston dan Stephen B. Maurer ,1991)

Definisi 2.17

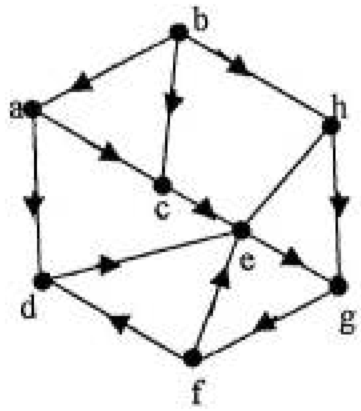
Rusuk e yang menghubungkan dua simpul yang sama, yaitu $e=\{u,u\}$ disebut *loop* pada u.(Anthony Ralston dan Stephen B. Maurer ,1991)

Definisi 2.18

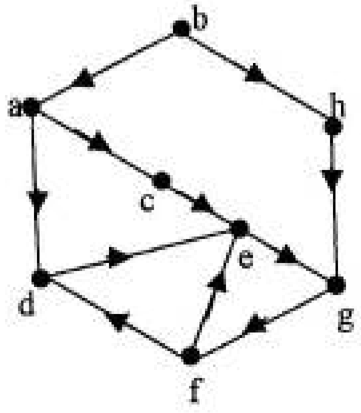
Misalkan $G=(V,E)$ sebuah graf .Suatu graf $G'=(V',E')$ dinamakan graf bagian dari G jika $E'\subseteq E$ dan $V'\subseteq V$.(Anthony Ralston dan Stephen B. Maurer ,1991)

<u>*Contoh*</u>

$G=(V,E)$



$G'=(V',E')$

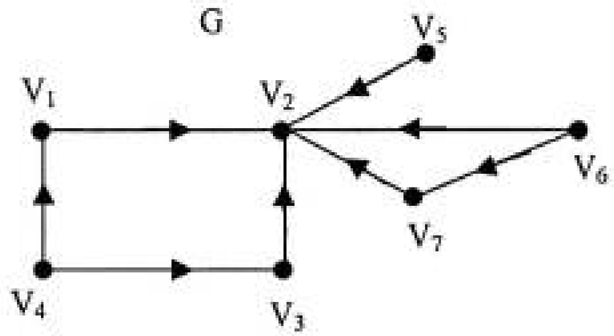


gambar 7

Definisi 2.19

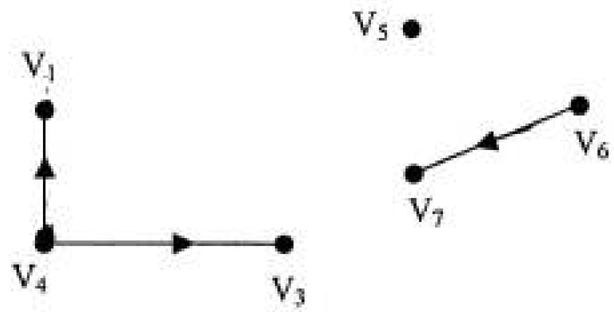
Penghapusan simpul v dalam G adalah menghasilkan Graf G – { v } yang merupakan suatu subgraf dari G .(H. S., Suryadi , 1994)

Contoh



gambar 8

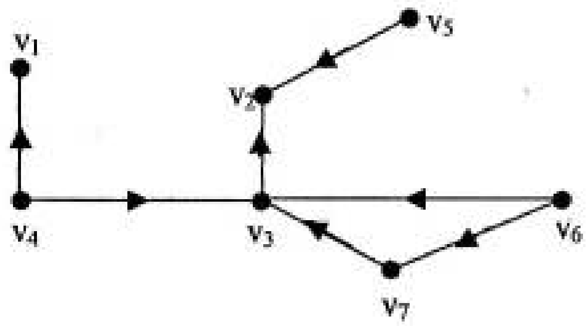
Penghapusan V_2 dari G menghasilkan :



gambar 8a

Penghapusan simpul biasanya diikuti dengan penghapusan rusuk. Namun penghapusan rusuk dari suatu Graf G, kita tulis G – {e} tidak menyebabkan terhapusnya simpul ujung dari rusuk tersebut.

Pada gambar 8 diatas, penghapusan rusuk e = (v_1, v_2) akan menghasilkan Graf seperti pada gambar 8b .



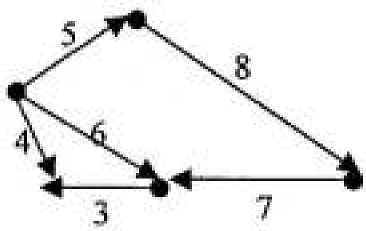
gambar 8b

Definisi 2.20

Suatu simpul *u* dikatakan dapat dicapai dari simpul *v* bila terdapat jalan dari *u* ke *v*.(H. S., Suryadi , 1994)

Definisi 2.21

Graf berarah terboboti adalah graf berarah dimana setiap rusuk atau simpulnya diberi bobot.(H. S., Suryadi , 1994)

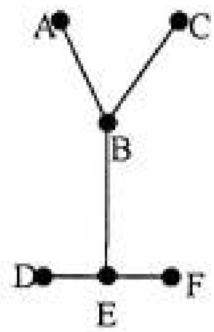


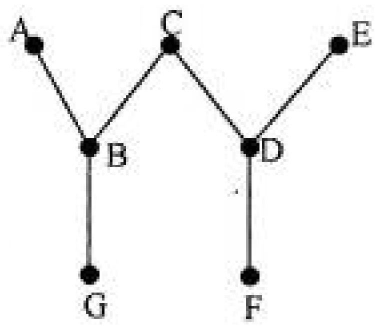
gambar 9

Definisi 2.22

Suatu graf *G* disebut graf pohon jika G tidak memuat lingkaran.

Contoh Pohon:





2. 4 Matriks dan Graf

Misal G suatu graf sederhana n *simpul*

Matriks ketetanggaan dari *graf* $G(V, E)$ berukuran $n\ x\ n$ adalah A= (a_{ij}),

dimana

$$a_{ij} = \begin{cases} 1 & \text{bila } (v_i, v_j) \in E \text{ untuk } v_i, v_j \in V \\ 0 & \text{dalam hal lain} \end{cases}$$

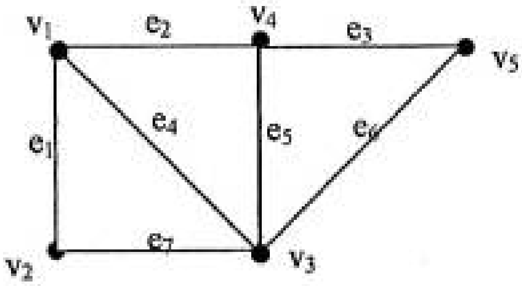
matriks ketetanggaan merupakan matriks simetris yaitu $A^t = A$

Matriks Insidensi $M = (m_{ij})$ dari graf G, didefinisikan sebagai matriks

berukuran $n\ x\ m$ dimana :

$$m_{ij} = \begin{cases} 1 & \text{bila rusuk } e_j \text{ berinsidensi simpul } v_i \\ 0 & \text{dalam hal lain} \end{cases}$$

- *Untuk graf tak berarah*



gambar 11

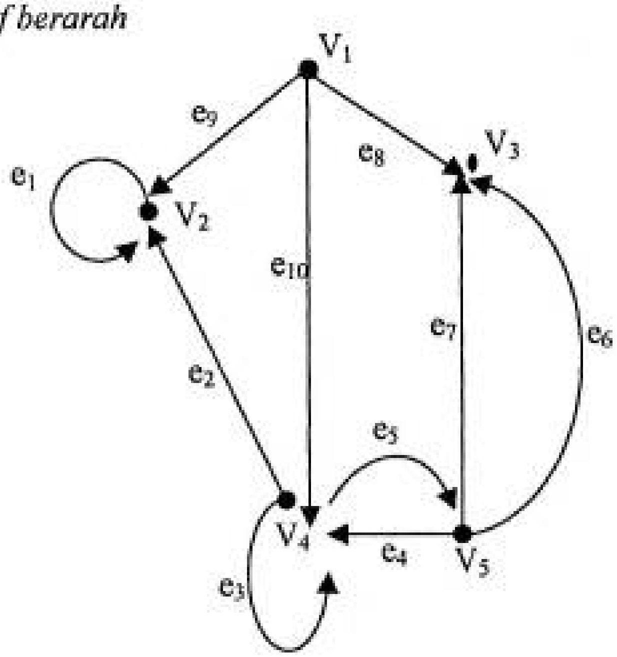
Matriks ketetanggaannya adalah A= $\begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$

Matriks Insidensi $M = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$

Untuk graf dengan rusuk ganda, maka matriks ketetanggaan didefinisikan sebagai berikut:

$$a_{ij} = \begin{cases} p\,, \textit{bila ada p rusuk menghubungkan} \left(v_i, v_j\right) \textit{di mana } p \geq 1 \\ 0\,, \textit{dalam hal lain} \end{cases}$$

- *Untuk graf berarah*



gambar 12

Matriks ketetanggaannya adalah $A=\begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 2 & 1 & 0 \end{bmatrix}$

Matriks Insidensi $M=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$

BAB III

APLIKASI GRAF BERARAH UNTUK MENGETAHUI HUBUNGAN ANTAR UNSUR-UNSUR PADA SEBUAH HIMPUNAN

Pembahasan terakhir ini akan dibatasi pada aplikasi graf berarah dalam kehidupan sehari – hari . Sehingga dengan adanya graf berarah ini segala masalah menjadi lebih mudah untuk mencari solusinya. Kebenaran solusi dengan menggunakan Algoritma yang bisa digunakan untuk menyelesaikan problem tersebut.

Algoritma adalah rangkaian langkah yang terstruktur dan berhingga untuk memecahkan suatu masalah.

3.1 Penggunaan Matriks Ketetanggaan

Salah satu kegunaan matriks ketetanggaan pada graf berarah adalah untuk mengetahui banyaknya jalan dengan panjang r yang mungkin ($r = 1, 2, 3, \ldots$)dari simpul v_i ke simpul v_j .

Teorema

Misalnya $A=(a_{ij})$ adalah matriks ketetanggaan dari suatu graf G berorde N (N>1). Maka elemen c_{ij} dari matriks A^r menyatakan banyaknya jalan dengan panjang r dari simpul v_i ke simpul v_j.

Bukti :

Dengan Induksi.

- Untuk $r=1$

  Dari definisi matriks A,

  Entri-entri dari matriks A adalah 0 atau 1. Angka 0 pada baris *ke*-i dan kolom *ke-j* jika v_i tidak terhubung dengan v_j. Ini berarti tidak ada jalan dari simpul v_i ke simpul v_j yang panjangnya 1. Angka 1 pada baris *ke*-i dan kolom *ke-j* jika v_i terhubung dengan v_j. Ini berarti ada jalan/ cara dari simpul v_i ke simpul v_j yang panjangnya 1.

  Jadi untuk $r=1$ a_{ij} menyatakan banyaknya jalan dengan panjang 1 dari simpul v_i ke simpul v_j.

- Untuk $r > 1$

  Misalkan $A^{r-1} = (a_{ik})$

  $A = (b_{kj})$

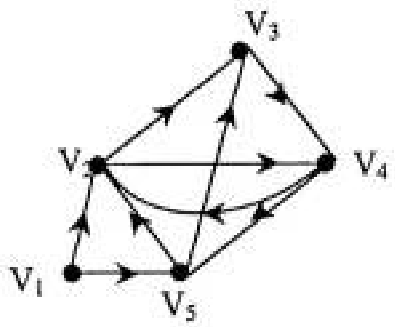
  $A^r = (a_{ik})(b_{kj}) = (c_{ij})$ $\qquad i, j = 1, 2, \ldots n$

  $$\text{dimana } c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \qquad (1)$$

Perhatikan bahwa a_{ik} adalah banyaknya jalan dengan panjang $r-1$ dari simpul v_i *ke* v_k dan b_{kj} adalah banyaknya jalan dari *simpul* v_k *ke simpul* v_j yang panjangnya 1.

Maka $a_{ik}b_{kj}$ banyaknya jalan dari simpul v_i *ke* v_j dengan panjang r yang melalui simpul v_k. Jadi dengan demikian $\sum_{k=1}^{n} a_{ik} b_{kj}$ menunjukkan banyaknya jalan dari simpul v_i *ke* v_k yang panjangnya $r-1+1$. Dengan demikian elemen $c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$ dari A^r menunjukkan banyaknya jalan yang panjangnya r dari simpul v_i ke simpul v_j.

Misal order dari *graf G adalah N* maka jalan dari simpul v_i *ke simpul* v_j mempunyai jalan *maksimum* $N-1$. Sementara itu, matriks $A+A^2+A^3+\dots\dots+A^{N-1}$ bisa mempunyai unsur 0 pada baris ke i kolom ke j yang berarti tidak ada lintasan antara simpul v_i *dan* v_j.



gambar 13

Gambar 13 adalah peta rute dari sebuah perusahaan yang melayani lima kota. Disini sebuah perusahaan itu ditinjau sebagai sebuah himpunan, sedangkan lima kota yang dia layani dimisalkan sebagai unsur-unsur himpunan dinyatakan sebagai simpul yaitu $v_1, v_2, v_3, v_4,$ *dan* v_5 sedangkan garis berarah dari satu simpul ke simpul lainnya menyatakan kerterhubungan langsung antara simpul tersebut.

Dari gambar 13 dapat dibuat matrik ketetanggaanya yaitu

$$A=\begin{bmatrix}0&1&0&0&1\\0&0&1&1&0\\0&0&0&1&0\\0&1&0&0&1\\0&1&1&0&0\end{bmatrix}$$

Kita memperoleh bahwa $A^2=\begin{bmatrix}0&1&2&1&0\\0&1&0&1&1\\0&1&0&0&1\\0&1&2&1&0\\0&0&1&2&0\end{bmatrix}$

$$A^3=\begin{bmatrix}0&1&1&3&1\\0&2&2&1&1\\0&1&2&1&0\\0&1&1&3&1\\0&2&0&1&2\end{bmatrix}$$

$$A^4=\begin{bmatrix}0&4&2&2&3\\0&2&3&4&1\\0&1&1&3&1\\0&4&2&2&3\\0&3&4&2&1\end{bmatrix}$$

Untuk mencari berapa banyak jalan dari kota v_4 *ke kota* v_2. Dapat menggunakan bentuk umum dari banyaknya jalan dari simpul v_i ke simpul v_j ditentukan oleh elemen (c_{ij}) yaitu $c_{ij}=\sum_{k=1}^{n} a_{ik}b_{kj}$.

Dalam hal ini $c_{42} = \sum_{k=1}^{5} a_{4k} b_{k2}$

- Jalan dengan panjang 1 dari v_4 *ke* v_2 yaitu $v_4 \rightarrow v_2$ atau (v_4, v_2)
- Jalan dengan panjang 2 dari v_4 *ke* v_2 yaitu :

  $v_4 \rightarrow v_5 \rightarrow v_2$ atau ditulis (v_4, v_5, v_2).

$$c_{42}^{(2)} = \sum_{k=1}^{5} a_{4k} b_{k2}$$

$$= a_{41}b_{12} + a_{42}b_{22} + a_{43}b_{32} + a_{44}b_{42} + a_{45}b_{52}$$

$$= 0.1 \quad + 1.0 \quad + \quad 0.0 \quad + \quad 0.1 \quad + 1.1$$

$$= 1$$

Artinya ada 1 jalan dari *simpul* v_4 *ke simpul* v_2 yang panjangnya 2 yang melalui simpul v_5 . Hal ini dapat dilihat pada saat $a_{45} = 1$ artinya ada jalan dari *simpul* v_4 *ke simpul* v_5 yang panjangnya 1 yaitu $v_4 \rightarrow v_5$ demikian halnya $b_{52} = 1$ artinya ada jalan dari *simpul* v_5 *ke simpul* v_2 yang

Panjangnya 1 yaitu $v_5 \rightarrow v_2$. Karena

simpul v_4 *terhubung dengan simpul* v_5 *dan*
simpul v_5 *terhubung dengan simpul* v_2 *maka*
simpul v_4 *terhubung dengan simpul* v_2

Sehingga jalan dengan panjang 2 dari v_4 *ke* v_2 yaitu $v_4 \rightarrow v_5 \rightarrow v_2$ atau ditulis (v_4, v_5, v_2).

Jalan dengan panjang 3 dari v_4 *ke* v_2 yaitu

$$c_{42}^{(3)} = \sum_{k=1}^{5} a_{4k} b_{k2}$$

$= a_{41}b_{12} + a_{42}b_{22} + a_{43}b_{32} + a_{44}b_{42} + a_{45}b_{52}$

$= 0.1 + 1.0 + 1.0 + 1.1 + 0.1$

$= 1$

Artinya ada 1 jalan dari *simpul* v_4 *ke simpul* v_2 yang panjangnya 3 yang melalui simpul V_4. $a_{44}b_{42}$ adalah perjalanan *simpul* v_4 *ke simpul* v_2 yang melalui V_4. a_{44} adalah banyaknya jalan yang panjangnya 2 dari *simpul* v_4 *ke simpul* v_4. Diperoleh 1 jalan yaitu $v_4 \to v_2 \to v_4$ b_{42} adalah banyaknya jalan yang panjangnya 1 dari *simpul* v_4 *ke simpul* v_2 yaitu $v_4 \to v_2$ sehingga diperoleh jalan dari *simpul* v_4 *ke simpul* v_2 yang panjangnya 3 Yaitu $v_4 \to v_2 \to v_4 \to v_2$.

- Jalan dengan panjang 4 dari v_4 *ke* v_2:

$$c_{42}^{(4)} = \sum_{k=1}^{5} a_{4k}b_{k2}$$

$= a_{41}b_{12} + a_{42}b_{22} + a_{43}b_{32} + a_{44}b_{42} + a_{45}b_{52}$

$= 0.1 + 1.0 + 1.0 + 3.1 + 1.1$

$= 4$

Artinya ada 4 jalan dari *simpul* v_4 *ke simpul* v_2 yang panjangnya 4 yang melalui V_4 dan V_5. a_{44} adalah banyaknya jalan yang panjangnya 3 dari *simpul* v_4 *ke simpul* v_4. Diperoleh 3 jalan yaitu

$v_4 \to v_5 \to v_2 \to v_4$

$v_4 \to v_5 \to v_3 \to v_4$

$v_4 \to v_2 \to v_3 \to v_4$

Dan b_{42} adalah banyaknya jalan yang panjangnya 1 dari *simpul* v_4 *ke simpul* v_2 yaitu $v_4 \to v_2$ sehingga bila dihubungkan *simpul* v_4 *ke simpul* v_2 diperoleh

$v_4 \to v_5 \to v_3 \to v_4 \to v_2$

$v_4 \to v_2 \to v_3 \to v_4 \to v_2$.

$v_4 \to v_5 \to v_2 \to v_4 \to v_2$

Dan untuk a_{45} adalah banyaknya jalan yang panjangnya 3 dari *simpul* v_4 *ke simpul* v_5 . Diperoleh 1 jalan yaitu $v_4 \to v_2 \to v_4 \to v_5$ dan b_{42} adalah banyaknya jalan yang panjangnya 1 dari *simpul* v_4 *ke simpul* v_2 yaitu $v_4 \to v_2$ sehingga bila dihubungkan diperoleh

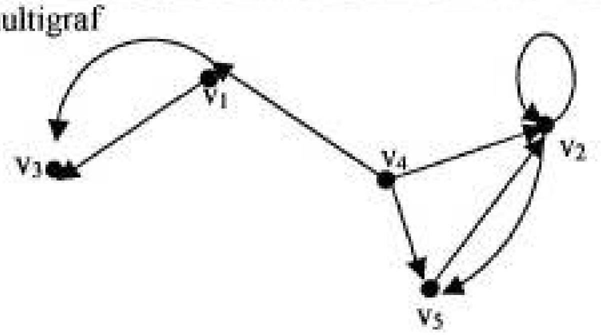
$v_4 \to v_2 \to v_4 \to v_5 \to v_2$.

Sehingga banyaknya semua jalan yang panjangnya 4 dari *simpul* v_4 *ke simpul* v_2 ada sebanyak 4.

Jadi banyaknya jalan dari simpul *simpul* v_4 *ke simpul* v_2 secara keseluruhan ada 7 cara.

Dengan melihat perkalian matriks ketetanggaan diperoleh $a_{42} = 1$, maka ada 1 jalan dengan panjang 2, karena $a_{42}^{(2)} = 1$ maka ada satu jalan dengan panjang 2, karena $a_{42}^{(3)} = 1$ maka ada satu jalan dengan panjang 3 dan karena $a_{42}^{(4)} = 4$ maka ada empat jalan dengan panjang 4.

Contoh : untuk multigraf



gambar 14

Matriks ketetanggaan dari graf diatas adalah

$$A = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$A^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 2 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$$A^3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 2 \\ 0 & 2 & 0 & 0 & 1 \end{bmatrix}$$

Untuk memeriksa keterhubungan graf dari pembuktian teorema dapat dibuat algoritma keterhubungan yaitu

Algoritma keterhubungan sebagai berikut:

$Langkah(0)$: $y = A, A =$ Matriks ketetanggaan, $k = 1$ [inisialisasi]

$Langkah(1)$: $jika\ y$ tidak mempunyai $elemen = 0$, maka kerjakan (3)

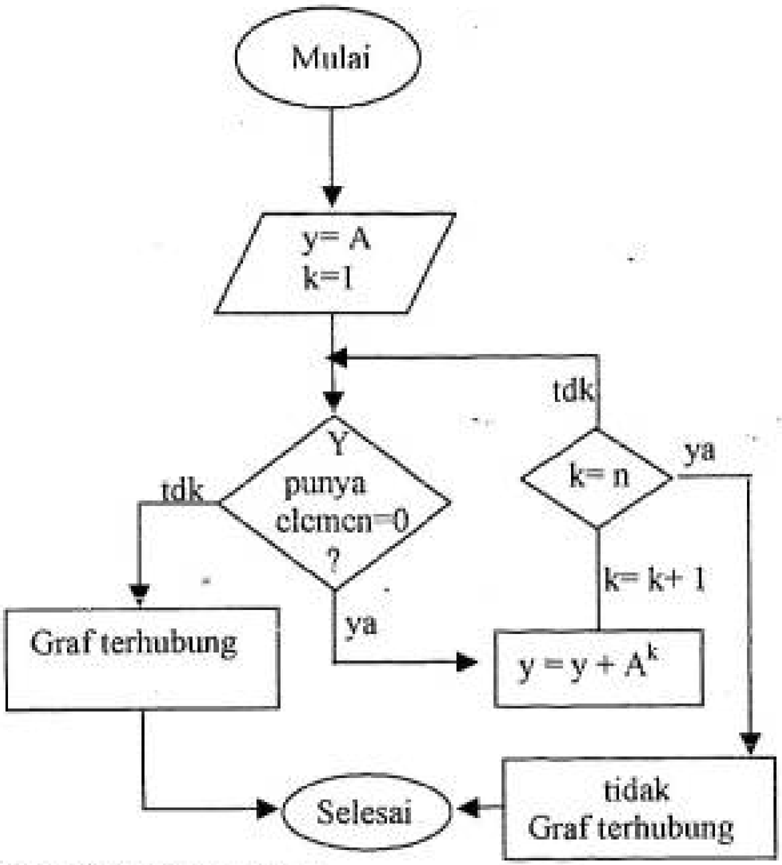
$Langkah(2)$: $k = k + 1$

$jika\ k = n$ maka kerjakan (4)

$y = y + A^k$, kerjakan lagi (1)

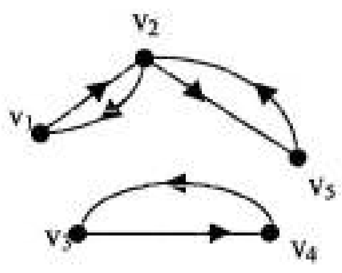
$Langkah(3)$: Graf terhubung .Selesai

$Langkah(4)$: Graf tidak terhubung. Selesai (H. S., Suryadi, 1994)



Simulasi algoritma keterhubungan

Diketahui graf G



gambar 15

dengan matriks ketetanggaan

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$Langkah(0)$: $y = A$

$k = 1$

$Langkah\ (1)$: y mengandung $elemen\ nol$? Ya, kerjakan $Langkah\ (2)$

$Langkah(2)$: $k = 2 \neq n(=5)$

$$A^2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$y = y + A^2 = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 2 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$, kerjakan $langkah(1)$

$Langkah(1)$: y mengandung $elemen\ nol$? Ya, kerjakan $langkah(2)$

$Langkah(2)$: $k = 2 + 1 = 3 \neq n(=5)$

$$A^3 = \begin{bmatrix} 0 & 2 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \end{bmatrix},$$

$$y = y + A^3 = \begin{bmatrix} 1 & 3 & 0 & 0 & 1 \\ 3 & 2 & 0 & 0 & 3 \\ 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 2 & 1 & 0 \\ 1 & 3 & 0 & 0 & 1 \end{bmatrix}, \text{ kerjakan } \textit{langkah}(1)$$

Langkah (1): y mengandung *elemen nol* ? Ya, kerjakan *Langkah* (2)

Langkah(2): $k = 4 \neq n(=5)$

$$A^4 = \begin{bmatrix} 2 & 0 & 0 & 0 & 2 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 0 & 2 \end{bmatrix}$$

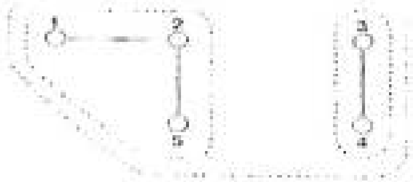
$$y = y + A^4 = \begin{bmatrix} 3 & 3 & 0 & 0 & 3 \\ 3 & 6 & 0 & 0 & 3 \\ 0 & 0 & 2 & 2 & 0 \\ 0 & 0 & 2 & 2 & 0 \\ 3 & 3 & 0 & 0 & 3 \end{bmatrix}, \text{ kerjakan } \textit{langkah}(1)$$

Langkah (1): y mengandung *elemen nol* ? Ya

Langkah(1): $k = 5 = n$, dilanjutkan ke *langkah*(4)

Langkah(4): Graf tidak terhubung. Selesai

Dari matriks y yang terakhir dapat dilihat bahwa simpul i berhubungan dengan simpul j apabila elemen $(i, j) \neq 0$. Jadi simpul 1 berhubungan dengan simpul 2 dan simpul 5 .Pembagian komponen dapat dilihat dari setiap baris pada matriks y yang terakhir. Komponen pertama ditunjukkan pada baris 1, 2 dan 5 yang terdiri atas 3 simpul.(lihat matriks ketetanggaan). Sedangkan komponen kedua ditunjukkan pada baris 3 dan 4 yang terdiri atas 2 simpul .Sehingga dapat dibuat graf keterhubungannya seperti dibawah ini



gambar 16

2. Algoritma fusi untuk keterhubungan

Langkah (0): Siapkan matriks ketetanggaan A, $i=1$, $c=0$

Langkah (1): Cari $a_{ij} \neq 0$ untuk $i \langle j$

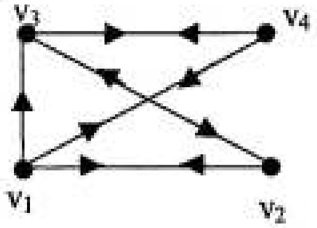
jumlah baris j ke baris i, kolom j ke kolom i (jumlah OR) .

Hapus j .

Langkah (2): $c=c+1$

cari $k>i$ dimana baris k belum dihapus bila tidak ada berhenti, *jumlah komponen sebanyak* c dalam hal ini masukkan $i=k$, lanjutkan *langkah* (1) . (H. S., Suryadi, 1994)

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$



gambar 17

Misalnya dilakukan fusi *simpul* 1 *dan* 2 menghasilkan *simpul* 1 :

- Tambah *baris* 1 dengan *baris* 2 ($c_{1j} + c_{2j}$) j =1,2,3,4

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 1 & 1 & 2 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

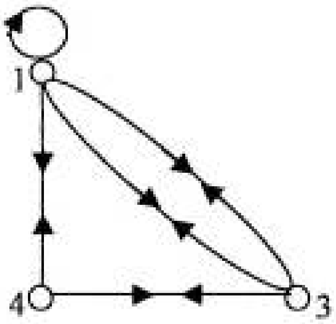
- Buat *kolom* 1=*baris* 1(*yang baru*) ($c_{1j}= c_{i1}$)

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 1 & 1 & 2 & 1 \\ 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

- Hapus *baris* 2, *dan kolom* 2

$$\begin{array}{c} \\ v_1 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{ccc} v_1 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \end{array}$$

Graf :



gambar 18

3.2 Aplikasi Graf Berarah

Disini akan dibahas dua aplikasi graf berarah yaitu menentukan waktu minimal pada manajemen proyek dan mencari lintasan terpendek pada rute perjalanan antar dua kota.

Untuk menentukan waktu digunakan algoritma di bawah ini:

Algoritma BFS

1. Untuk setiap titik $\omega \neq u$, $e(V)=\infty$ lebih lanjut $e(u)=0$ dan $Q=\{u\}$
2. Jika $Q\neq \phi$ maka hapus X dari G jika tidak stop karena tidak ada lintasan dari u ke v
3. Untuk setiap y bertetangga dengan $X \ni e(y)=\infty$ Parent $(y)=X$ $e(y)=e(x)+1$ dan masukkan y ke dalam Q
4. Jika $e(\omega)=\infty$ maka menuju ke langkah 2 jika tidak ke langkah 5
5. 5.1 $k=e(V)$ *dan* $u_k=v$

   5.2 Jika $k\neq 0$ maka $u_{k-1}=$ Parent u_k jika tidak ke 5.1

   5.3 $k=k-1$ dan menuju ke 5.2

5.4 Output $u_0, u_1, \ldots u_k$ yang merupakan lintasan terpendek /waktu minimal

KETERANGAN

$e(y)$ merupakan suatu bilangan yang menunjukkan urutan label simpul pada graf G

$e(\omega)$ adalah simpul awal pada suatu graf G yang berlabel 0

V adalah simpul pada suatu graf G

$e(V)$ adalah simpul pada suatu graf yang berlabel ∞

Q adalah suatu himpunan yang anggotanya merupakan simpul yang bertetangga dengan simpul yang berlabel ∞

X adalah simpul yang merupakan unsur/anggota dari Q

y adalah simpul yang bertetangga dengan X

Sedangkan untuk menentukan lintasan terpendek pada rute perjalanan antar dua kota digunakan algoritma sebagai berikut :

ALGORITMA LINTASAN TERPENDEK

L. 1 $i=0$ $U=\{v_0\}$ $\bar{U}=V(G)-\{v_0\}$

$d(v_0)=0$ dan $d(V)=\infty\ \forall\ V\in\bar{U}$

Jika $p=1$ maka berhenti jika tidak lanjut ke L.2

L. 2 $\forall\ V\in\bar{U}\ \ni(u_i,V)\in E(G)$

Jika $d(V)\leq d(u_i)+\omega(u_i,V)$ maka lanjut

Jika tidak $d(V)\leq d(u_i)+\omega(u_i,V)$ dan Parent $(V)=u_i$

L. 3 $m = \min \{d(V) \mid V \in U\}$

jika $v_i \in \bar{U} \quad \ni d(v_i) = m$

m output yang merupakan jarak u_0 *dan* v_i *dan* $u_{i+1} = v_i$

L. 4 $U = U \cup \{u_{i+1}\}$ dan $\bar{U} = \bar{U} - \{v_{i+1}\}$

L. 5 $i = i + 1$

Jika $i = p - 1$ maka berhenti jika tidak kembali ke L.2.

(Anthony Ralston dan Stephen B. Maurer,1991)

KETERANGAN

p adalah banyaknya simpul

V adalah simpul-simpul pada graf G

$d(v_0)$ adalah jarak awal pada graf G

U adalah himpunan simpul pada lintasan terpendek

$\bar{U}$ adalah himpunan simpul awal

$\omega(u_i V)$ adalah bobot yang menghubungkan antara simpul u_i ke simpul V yang

Simulasi algoritma BFS dapat dilihat pada manajemen proyek

3.2.1. Manejemen Proyek

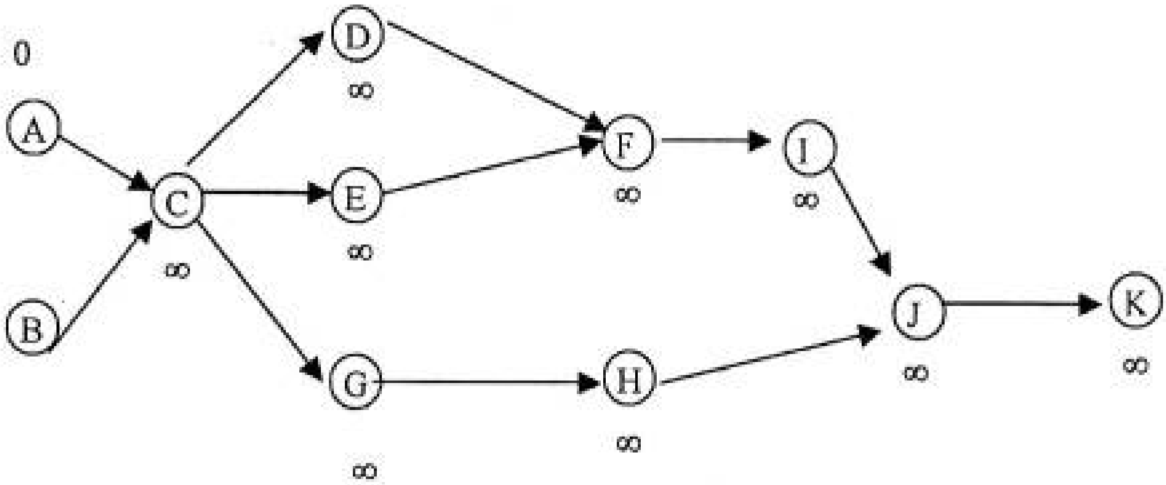
Misalkan dalam proyek terdapat jenis kegiatan yang berbeda-beda misalnya $k_1, k_2, k_3, \ldots k_n$ yang masing-masing kegiatan tersebut memerlukan waktu $T_1, T_2, T_3, \ldots . T_n$. Jenis kegiatan tersebut ada yang tergantung pada kegiatan lainnya. Yang menjadi masalah bagaimana menentukan waktu minimal untuk menyelesaikan proyek tersebut agar proyek itu bisa selesai dengan tepat waktu.

Tabel 1 Jenis kegiatan dan waktu pelaksanaan setiap kegiatan

Jenis Kegiatan	Waktu/ (minggu)	Syarat
A	3	-
B	2	-
C	2	A, B
D	4	C
E	3	C
F	2	D, E
G	3	C
H	1	G
I	5	F
J	2	H, I
K	10	J

Dari tabel 1 dibuat grafnya seperti di bawah ini:

Adapun syarat pemberian arah dari graf berdasarkan kolom ke tiga dari tabel diatas dimana ada kegiatan yang bergantung pada kegiatan lain. Misalnya kegiatan C baru bisa dilaksanakan bila kegiatan A dan B terpenuhi demikian pula kegiatan D, E, G baru bisa terlaksana bila kegiatan C telah terpenuhi demikian seterusnya.



Gambar19

Untuk menentukan lintasan terpendek/waktu minimal pada graf diatas maka digunakan Algoritma BFS sebagai berikut :

Sebelumnya diasumsikan bahwa simpul awal berlabel 0 dan simpul yang lain berlabel ∞

1 . Untuk setiap simpul $\omega \neq A$, $e(\omega)=\infty$ lebih lanjut $e(A)=0$ dan $Q=\{A\}$

2. $Q=\{A\} \neq \phi$ hapus A dari G maka diperoleh $Q=\{\ \}$

3. Simpul yang bertetangga dengan A yang labelnya ∞ adalah simpul C

Parent $(C)=A$; $e(C)=e(A)+1$

$=0+1$

$=1$

$Q=\{C\}$

4. $e(V)=\infty$ ke langkah 2

2. $Q=\{C\} \neq \phi$, hapus C dari G maka diperoleh $Q=\{\ \}$

3. Simpul yang bertetangga dengan C yang labelnya ∞ adalah simpul D,E,G

Parent $(D)=C$; $e(D)=e(C)+1$

$=1+1$

$=2$

Parent $(E)=C$; $e(E)=e(C)+1$

$=1+1$

$=2$

Parent $(G)=C$; $e(G)=e(C)+1$

$=1+1$

$=2$

$Q=\{D,E,G\}$

4. $e(V)=\infty$ ke langkah 2

2. $Q=\{D,E,G\} \neq \phi$, hapus D,E,G dari G maka diperoleh $Q=\{\ \}$

3. Simpul yang bertetangga dengan D yang labelnya ∞ adalah simpul F

Parent $(F)=D$; $e(F)=e(D)+1$

$=2+1$

$=3$

Simpul yang bertetangga dengan E yang labelnya ∞ adalah simpul F

Parent $(F)=E$; $e(F)=e(E)+1$

$=2+1$

$=3$

Simpul yang bertetangga dengan G yang labelnya ∞ adalah simpul H

Parent $(H)=G$; $e(H)=e(G)+1$

$=2+1$

$=3$

$Q=\{F,H\}$

4. $e(V)=\infty$ ke langkah 2

2. $Q=\{F,H\}\neq\phi$, hapus F,H dari G maka diperoleh $Q=\{\ \}$

3. Simpul yang bertetangga dengan F yang labelnya ∞ adalah simpul I

Parent $(I)=F$; $e(I)=e(F)+1$

$=3+1$

$=4$

Simpul yang bertetangga dengan H yang labelnya ∞ adalah simpul J

Parent $(J)=H$; $e(J)=e(H)+1$

$=3+1$

$=4$

$Q=\{I,J\}$

4. $e(V)=\infty$ ke langkah 2

2. $Q=\{I,J\}\neq\phi$, hapus I,J dari G maka diperoleh $Q=\{\ \}$

3. Simpul yang bertetangga dengan I yang labelnya ∞ adalah simpul J

Parent $(J)=I$; $e(I)=e(J)+1$

$=4+1$

$=5$

Simpul yang bertetangga dengan J yang labelnya ∞ adalah simpul K

Parent $(K)=J$; $e(K)=e(J)+1$

$=4+1$

$=5$

$Q=\{I,K\}$

4. $e(V)=\infty$ ke langkah 2

2. $Q=\{I,K\}\neq\phi$, hapus I, K dari G maka diperoleh $Q=\{\ \}$

3. Simpul yang bertetangga dengan I yang labelnya ∞ adalah simpul J

Parent $(J)=I$; $e(I)=e(J)+1$

$=5+1$

5.4 Output $u_0, u_1, u_2, u_3, u_4, u_5, u_6 = (A, C, D, F, I, J, K)$

Simpul $A, C, D, F, I, J,$ *dan* K adalah lintasan terpendek pada gambar 19. Dengan menjumlah bobot pada setiap simpul pada lintasan terpendek tersebut diperoleh 28 maka hal ini menunjukkan bahwa waktu minimum yang digunakan agar proyek tersebut dapat selesai dengan tepat waktu digunakan waktu 28 minggu. Sehingga bisa dibuat tabel jadwal kegiatannya seperti di belakang ,dimana tanda silang yang berwarna merah menunjukkan jenis kegiatan yang tidak bisa ditunda waktu pelaksanaannya . Sehingga dalam gambar 19 di atas simpul yang membentuk lintasan yang menunjukkan kegiatan yang tidak bisa ditunda disebut lintasan kritis. Kapan kegiatan itu ditunda maka timbul kerugian dalam penyelesaian proyek.

$=6$

Simpul yang bertetangga dengan K yang labelnya ∞ adalah tidak ada lanjut ke langkah 5

5 5.1 $k = e(K) = 6$ *dan* $u_6 = K$

5.2 $k \neq 0$ maka u_5 Parent $(u_6 = K) = J$

5.3 $k = k-1$ dan menuju ke 5.2

5.2 $k = 4 \neq 0$ maka u_4 Parent $(u_5 = J) = I$

5.3 $k = k-1$ dan menuju ke 5.2

5.2 $k = 3 \neq 0$ maka u_3 Parent $(u_4 = I) = F$

5.3 $k = k-1$ dan menuju ke 5.2

5.2 $k = 2 \neq 0$ maka u_2 Parent $(u_3 = F) = D$

5.3 $k = k-1$ dan menuju ke 5.2

5.2 $k = 1 \neq 0$ maka u_1 Parent $(u_2 = D) = C$

5.3 $k = k-1$ dan menuju ke 5.2

5.2 $k = 1 \neq 0$ maka u_0 Parent $(u_1 = C) = A$

5.3 $k = 0$ maka menuju ke 5.4

Tabel 2 Jadwal Kegiatan

KEGIATAN	MINGGU																											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
A	X	X	X																									
B	X		X																									
C				X	X																							
D						X	X	X	X																			
E						X	X																					
F										X	X																	
G									X	X	X																	
H														X														
I												X	X	X	X	X												
J																	X	X										
K																			X	X	X	X	X	X	X	X	X	X

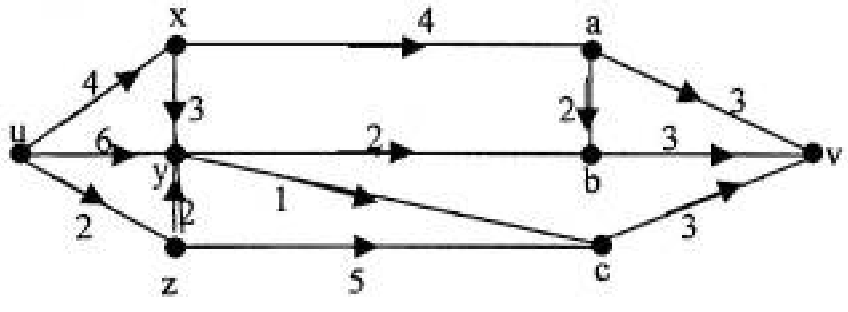
Simulasi algoritma lintasan terpendek dapat dilihat pada rute perjalanan antar 2 kota yang melewati beberapa kota yang

3.2.2 Rute Perjalanan Antar Dua Kota

Pandang G suatu graf berarah yang berhingga dengan setiap rusuk mempunyai bobot., dan diasumsikan graf G tidak mengandung loop. Jika hendak menentukan jarak lintasan terpendek antara 2 simpul yang melalui beberapa simpul tertentu digunakan algoritma lintasan terpendek yang telah dibahas sebelumnya

<u>Contoh</u>

Gambar berikut merupakan suatu rute perjalanan dari kota *u* ke kota *v*. Untuk menempuh kota *u* ke kota *v* ada beberapa jalan yang bisa ditempuh dan sebelum tiba ke tujuan ada beberapa kota yang akan dilewati sebelumnya. Di sini kita akan mencari lintasan terpendek untuk menempuh kota *u* ke kota *v*. Anggaplah setiap kota yang dilewati adalah sebuah simpul. Kota *u* disebut simpul *u* (simpul awal) kota v disebut simpul v(simpul akhir).Kita hendak menghitung lintasan terpendek dari simpul *u* ke simpul *v*.



gambar 20

Untuk menentukan lintasan terpendek pada gambar diatas digunakan algoritma lintasan terpendek diatas.

Misalkan : p adalah banyaknya simpul

U adalah himpunan simpul pada lintasan terpendek

$\bar{U}$ adalah himpunan simpul awal

Diketahui : $V(G) = \{u, x, y, z, a, b, c, v\}$

$$E(G) = \{e_1 = (ux), e_2 = (uy), e_3 = (uz), e_4 = (xy), e_5 = (zy), e_6 = (xa), \\ e_7 = (yb), e_8 = (yc), e_9 = (zc), e_{10} = (cv), e_{11} = (ab), \\ e_{12} = (bv), e_{13} = (av)\}$$

$$d(e_1) = 4\ ,\ d(e_2) = 6\ ,\ d(e_3) = 2\ ,\ d(e_4) = 3 \\ d(e_5) = 2\ ,\ d(e_6) = 2\ ,\ d(e_7) = 2,\ d(e_8) = 1 \\ d(e_9) = 5\ ,\ d(e_{10}) = 3\ ,\ d(e_{11}) = 2,\ d(e_{12}) = 3,\ d(e_{13}) = 3$$

L.1 $i = 0$ $U = \{u_0 = a\}$ $\bar{U} = V(G) - \{a\} = \{x, y, z, a, b, v\}$

$d(a) = 0$ $d(V) = \infty$ jika $p = 8$ maka lanjut ke L.2

L.2 $\forall\ V \in \bar{U} \ni (u_0 a) \in E(G)$ diperoleh $x, y\ dan\ z$

Apakah $d(x) \leq d(u_0) + \omega(u_0 x)$?

$\infty \leq 0 + 4$

$= 4$ (tidak)

PARENT $(x) = u_0$

Apakah $d(y) \leq d(u_0) + \omega(u_0 y)$?

$\infty \leq 0 + 6$

$= 6$ (tidak)

PARENT $(y) = u_0$

Apakah $d(z) \leq d(u_0) + \omega(u_0 z)$?

$\infty \leq 0 + 2$

$= 2$ (tidak)

PARENT $(z) = u_0$

L.3 $m = \min\{d(V) \mid V \in U\}$

jika $z \in \bar{U} \quad \ni d(z) = m = 2$

m output I

yang merupakan jarak u_0 *dan* z *dan* $\boxed{u_{0+1} = u_1 = x}$

L. 4 $U = U \cup \{u_{i+1}\}$

$= U \cup \{u_1\}$

$= \{u_0, u_1\}$

dan $\bar{U} = \bar{U} - \{z\}$

$= \{x, y, z, a, b, c, v\} - \{z\}$

$= \{x, y, a, b, c, v\}$

L. 5 $i = i + 1$

$= 0 + 1$

$= 1$

Jika $i \neq p - 1$ maka kembali ke L.2

L.2 $\forall\ V \in \bar{U} \ni (u_1 z) \in E(G)$ diperoleh $y\ dan\ c$

Apakah $d(y) \leq d(u_1) + \omega(u_1 y)$?

$6 \leq 2 + 2$

$= 4$ (tidak)

PARENT $(y) = u_1$

Apakah $d(c) \leq d(u_1) + \omega(u_1 c)$?

$\infty \leq 2 + 5$

$= 7$ (tidak)

PARENT $(c) = u_1$

L.3 $m = \min\{d(V) \mid V \in U\}$

jika $y \in \bar{U} \ni d(y) = m = 4$

m output II

yang merupakan jarak $u_1\ dan\ y$ dan $\boxed{u_{1+1} = u_2 = y}$

L. 4 $U = U \cup \{u_{i+1}\}$

$= U \cup \{u_2\}$

$= \{u_0, u_1, u_2\}$

dan $\bar{U} = \bar{U} - \{z\}$

$= \{x, y, a, b, c, v\} - \{y\}$

$= \{x, a, b, c, v\}$

L. 5 $i = i + 1$

$= 1 + 1$

$= 2$

Jika $i \neq p - 1$ maka kembali ke L.2

L.2 $\forall\ V \in \bar{U} \ni (u_2 y) \in E(G)$ diperoleh $b\ dan\ c$

Apakah $d(b) \leq d(u_2) + \omega(u_2 b)$?

$\infty \leq 4 + 2$

$= 6$ (tidak)

PARENT $(b) = u_2$

Apakah $d(c) \leq d(u_2) + \omega(u_2 c)$?

$\infty \leq 4 + 1$

$= 5$ (tidak)

PARENT $(c) = u_2$

L.3 $m = \min\{d(V) \mid V \in U\}$

jika $c \in \bar{U} \ni d(c) = m = 5$

m output III

yang merupakan jarak u_2 *dan c* *dan* $\boxed{u_{2+1} = u_3 = c}$

L. 4 $U = U \cup \{u_{i+1}\}$

$= U \cup \{u_3\}$

$= \{u_0, u_1, u_2, u_3\}$

dan $\bar{U} = \bar{U} - \{c\}$

$= \{x, a, b, c, v\} - \{c\}$

$= \{x, a, b, v\}$

L. 5 $i = i + 1$

$= 2 + 1$
$= 3$

Jika $i \neq p - 1$ maka kembali ke L.2

L.2 $\forall\ V \in \bar{U} \ni (u_3 c) \in E(G)$ diperoleh v

Apakah $d(v) \leq d(u_3) + \omega(u_3 v)$?

$\infty \leq 5 + 3$

$= 8$ (tidak)

PARENT $(v) = u_3$

L.3 $m = \min\{d(V) \mid V \in U\}$

jika $v \in \bar{U} \ni d(v) = m = 8$

m output IV

yang merupakan jarak u_3 *dan* v *dan* $\boxed{u_{3+1} = u_4 = v}$

L. 4 $U = U \cup \{u_{i+1}\}$

$= U \cup \{u_4\}$

$= \{u_0, u_1, u_2, u_3, u_4\}$

dan $\bar{U} = \bar{U} - \{v\}$

$= \{x, a, b, v\} - \{v\}$

$= \{x, a, b\}$

Jadi $U = \{u_0, u_1, u_2, u_3, u_4\} = \{u, z, y, c, v\}$ yang merupakan lintasan terpendek pada rute perjalanan antar kota u dengan kota v

Bisa juga digunakan cara seperti berikut :

Langkah 1 :

Kita buat tabel jarak sebagai berikut

Tabel 3 . Jarak antar 2 kota:

u	x	y	z	a	b	c	v
ux=4 uy=6 uz=2	xy=3 xa=4	yb=2 yc=1	zy=2 zc=5	ab=2 av=3	bv=3	cv=3	

Langkah 2.

Kita mulai dengan simpul u sebagai simpul awal. Beri harga = 0. Ambil simpul dengan jarak terdekat dari u, dalam hal ini z(=2) kemudian lingkari uz.Semua rusuk lain yang berakhir di z kita hapus (tidak ada rusuk lain yang berakhir di z). Beri nilai = 2 di belakangnya. simpul yang telah diberi harga, ditandai dengan (*).

Tabel 4. Tabel Jarak antar 2 kota yang sudah dihapus rusuk lain yang berakhir di z

u*(0)	x	y	z*(2)	a	b	c	v
ux=4 uy=6 uz=2	xy=3 xa=4	yb=2 yc=1	zy=2 zc=5	ab=2 av=3	bv=3	cv=3	

Langkah 3 :

Dari simpul u dan z (yang telah ditandai (*)), dicari simpul lain yang jaraknya terdekat dihitung dari u. Jadi harus diperhitungkan nilai yang telah tertulis di simpul (0 untuk u dan 2 untuk z). Di sini ux = 4 dan uzy = 2 + 2 =4 merupakan nilai minimum. Apabila dipilih misalnya ux Beri nilai = 4 pada x, lingkari zx dan hapus rusuk lain yang menuju x. Di sini tidak diperoleh rusuk lain yang menuju x jadi tak ada rusuk yang dihapus.

Tabel 5. Jarak antar kota *u* dan v yang sudah dihapus rusuk lain yang berakhir di *x*

$u^*(0)$	x^*	y	$z^*(2)$	a	b	c	v
ux=4 uy=6 uz=2	xy=3 xa=4	yb=2 yc=1	zy=2 zc=5	ab=2 av=3	bv=3	cv=3	

Langkah 4 :

Dari simpul u dan z (yang telah ditandai (*)), dicari simpul lain yang jaraknya terdekat dihitung dari u. Jadi harus diperhitungkan nilai yang telah tertulis di simpul (0 untuk u dan 2 untuk z). Di sini ux = 4 dan uzy = 2 + 2 =4 merupakan nilai minimum. Boleh dipilih salah atau, misalnya uzy. Beri nilai=4 pada y, lingkari zy dan hapus rusuk lain yang menuju y, yaitu uy dan xy.

Tabel 6 . Jarak antar kota *u* dan v yang telah dihapus rusuk lain yang berakhir di y

u*(0)	x*(4)	y*(4)	z*(2)	a	b	c	v
ux=4 uz=2	xa = 4	yb=2 yc=1	zy=2 zc=5	ab=2 av=3	bv=3	cv=3	

Langkah 5 :

Dari simpul u, z dan y (yang telah diberi tanda (*)), dicari simpul lain yang jaraknya terdekat dihitung dari u. Dengan tetap memperhitungkan nilai yang telah tertulis di simpul 0 untuk u, 2 untuk z dan 4 untuk y).Disini uzyc=5 yang merupakan nilai minimum. Kemudian beri nilai 5 pada c, kemudian lingkari yc. Dan hapus rusuk yang lain yang menuju c, yaitu zc.

Tabel 7 . Jarak antar kota *u* dan v yang telah dihapus rusuk lain yang berakhir di *c*

u*(0)	x*(4)	y*(4)	z*(2)	a	b	c*(5)	v
ux=4 uz=2	xa =4	yb=2 yc=1	zy=2 zc=5	ab=2 av=3	bv=3	cv=3	

Langkah 6 :

Dari simpul u, z dan y (yang telah diberi tanda (*)), dicari simpul lain yang jaraknya terdekat dihitung dari u. Dengan tetap memperhitungkan nilai yang telah tertulis di simpul 0 untuk u, 2 untuk z dan 4 untuk y).Disini uzyb=6 . Kemudian beri nilai pada b=6. Kemudian hapus rusuk lain yang menuju b yaitu ab.

Tabel 8 . Jarak antar kota *u* dan v yang telah dihapus rusuk lain yang berakhir di *b*

u*(0)	x*(4)	y*(4)	z*(2)	a	b*(6)	c*(5)	v
ux=4 uz=2	xa = 4	yb=2 yc=1	zy=2	av=3	bv=3	cv=3	

Langkah 7 :

Dari simpul u, x (yang telah diberi tanda (*)), dicari simpul lain yang jaraknya terdekat dihitung dari u. Dengan tetap memperhitungkan nilai yang telah tertulis di simpul 0 untuk u, 4 untuk x).Disini uxa=8 .Kemudian beri nilai pada a= 8

Tabel 9 . Jarak terpendek antar kota *u* dengan kota *v*

u*(0)	x*(4)	y*(4)	z*(2)	a*(11)	b*(6)	c*(5)	v
ux=4 uz=2	xa = 4	yb=2 yc=1	zy=2	av=3	bv=3	cv=3	

Dari tabel 9 diatas diperoleh 3 lintasan dari u ke v yaitu (u, x, a, v), (u, z, y, c, v) dan (u, z, y, b, v) dan diperoleh lintasan minimum dari simpul u ke simpul v jarak/panjangnya = 8, dengan urutan sebagai berikut: u $\rightarrow z \rightarrow y \rightarrow c \rightarrow v$

BAB IV

KESIMPULAN DAN SARAN

4 . 1 Kesimpulan

Berdasarkan uraian dan pembahasan dalam skripsi ini dapat ditarik kesimpulan sebagai berikut :

1. Matriks ketetanggaannya dapat digunakan untuk mempresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut.
2. Representasi visual dari Graf adalah menyatakan objek sebagai simpul, sedangkan hubungan antar objek dinyatakan dengan rusuk(edge). Bila rusuknya diberi arah (tanda panah) maka representasi graf disebut digraf (Graf berarah)
3. Untuk keperluan pemerosesan Graf di program komputer, suatu graf dapat direpresentasikan dalam bentuk matriks ketetanggaan atau matriks bersisian.
4. Matriks ketetanggaan untuk graf sederhana dan tidak berarah selalu simetriks, sedangkan untuk digraf belum tentu simetriks dan akan simetri jika grafnya adalah digraf lengkap.
5. Matriks ketetanggaan nol-satu (zero-one) tidak dapat digunakan untuk mempresentasikan graf yang mempunyai sisi ganda .
6. Keuntungan Representasi Graf dengan matrik ketetanggaan adalah elemen matriksnya dapat diakses langsung melalui indeks, selain itu dapat ditentukan langsung apakah simpul i dan j bertetangga.

4. 2 Saran-saran

1. Penulisan ini masih jauh dari kesempurnaan karena pemprosesan graf melalui program komputer belum digunakan. Karena itu penulis menyarankan agar penggunaan metode ini dikembangkan melaui program komputer.
2. Studi mengenai teori Graf perlu ditingkatkan karena memiliki segi terapan di banyak bidang ilmu.

DAFTAR PUSTAKA

1. Anthony Ralston dan Stephen B. Maurer, "*Discrete Algoritmic Mathematics* " , Addison Wesley Publishing Company New York, Amerika, 1991.

2. Anton, H. dan Rorres, C., "*Penerapan Aljabar Linear*" , Pantur Silaban, Erlangga, Jakarta, 1988.

3. H. S ., Suryadi , "*Teori Graf Dasar Seri Diktat Kuliah*", Gunadarma. Jakarta, 1994

4. Munir rinaldi, "*Matematika Diskrit*". Informatika, Bandung, 2001

5. Robin J.Wilson, "*Graph Theory*", Longman, England, 1979.

6. Slamet Sumantri dan Makalive hendrik, " *Matematika Kombinatorik*", PT Elex Media Komputindo , Jakarta, 1992.