MODUL

MATEMATKA DISKRIT

MATERI GRAPH

Oleh:

Yemi Kuswardi

PROGRAM STUDI PENDIDIKAN MATEMATIKA
FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN
UNIVERSITAS SEBELAS MARET SURAKARTA

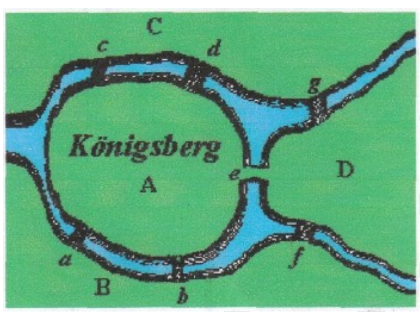
2019

YEMI KUSWARDI

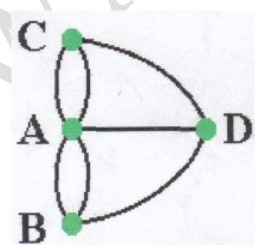
TEORI GRAPH

Dalam mata kuliah Matematika Diskrit terdapat topik atau bahasan mengenai Teori Graph. Topik ini pertama kali dikemukakan pada tahun 1937 oleh seorang matematikawan bernama Leonhard Euler. Masalah ini muncul dilatarbelakangi adanya permasalahan yang timbul di daerah asalnya yang dikenal dengan "Tujuh Jembatan Konigsberg". Masalah "Tujuh Jembatan Konigsberg" itu adalah sebagai berikut.

Di kota Konigsberg terdapat tujuh jembatan yang melintasi sungai Pregel seperti pada Gambar 1. Diceritakan bahwa para penduduk menghabiskan hari Minggu mereka berjalan untuk menemukan cara agar dari titik tertentu mereka dapat berjalan di antara kota dengan menyeberangi jembatan tepat sekali sehingga kembali ke titik semula. Untuk membuktikan apakah penduduk dapat melakukan hal tersebut Euler merepresentasikan masalah tersebut ke dalam bentuk seperti pada Gambar 2.



Gambar 1



Gambar 2

Pada Gambar 2, setiap daerah dilambangkan dengan titik dan jembatan dilambangkan dengan sisi. Gambar 2 merupakan sebuah graph.

Selain contoh di atas ada banyak sekali permasalahan sehari-hari yang dapat direpresentasikan dalam bentuk graph. Peta yang sering kita jumpai dapat juga direpresentasikan sebagai graph di mana kota sebagai titik dan sisi sebagai jalannya.

Definisi:

Sebuah graph G(V(G),E(G)) berisikan dua himpunan yaitu

- Himpunan hingga tak kosong V(G)

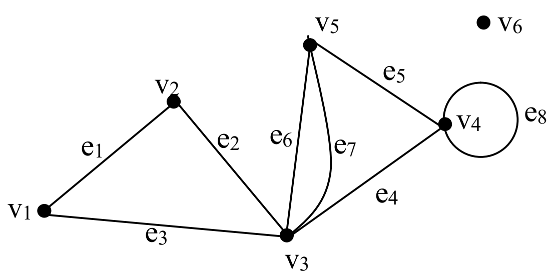
  Elemen-elemen V(G) disebut titik sehingga V(G) merupakan himpunan titik-titik di graph G

- Himpunan hingga (mungkin kosong) E(G)

  Elemen-elemen E(G) disebut sisi sehingga E(G) merupakan himpunan sisi-sisi di graph G

Setiap elemen e dalam E(G) merupakan sebuah pasangan tak berurutan dari titik-titik di V(G).

Perhatikan contoh graph G pada Gambar 3 berikut ini



Gambar 3

Graph G pada Gambar 3 terdiri dari himpunan titik V(G) = {v_1, v_2, v_3, v_4, v_5, v_6} dan himpunan sisi E(G) = {e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8}. Pada graph G tersebut terdapat sebuah sisi yang berawal dan berakhir pada titik yang sama yaitu sisi e_8 = (v_4, v_4) yang berawal dan berakhir pada titik v_4. Sisi seperti ini disebut **loop.** Selain itu pada graph G juga terdapat dua sisi yang mempunyai ujung yang sama yaitu e_6 = (v_3, v_5) dan e_7 = (v_3, v_5). Dua sisi yang mempunyai dua titik ujung yang sama seperti ini disebut sebagai **sisi-sisi rangkap.** Suatu titik yang bukan merupakan titik ujung dari sisi manapun seperti titik v_6 pada graph G disebut **titik terisolasi.**

Dua buah titik pada sebuah graph dikatakan berhubungan langsung (*adjacent*) jika kedua titik tersebut dihubungkan oleh sebuah sisi. Pada graph G di atas

v_1 berhubungan langsung dengan v_2 dan v_3

v_2 berhubungan langsung dengan v_1 dan v_3

v_3 berhubungan langsung dengan v_1, v_2, v_4, dan v_5

v_4 berhubungan langsung dengan v_3 dan v_5

v_5 berhubungan langsung dengan v_3 dan v_4

Sisi e dikatakan terkait (*incident*) pada titik u dan titik v jika titik u dan titik v berhubungan langsung, sehingga u dan v merupakan titik ujung/titik akhir dari sisi e. Pada graph G di atas

e_1 terkait pada v_1 dan v_2

e_2 terkait pada v_2 dan v_3

e_3 terkait pada v_1 dan v_3

e_4 terkait pada v_3 dan v_4

e_5 terkait pada v_4 dan v_5

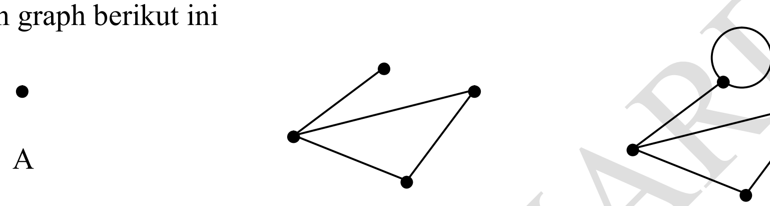
e_6 dan e_7 terkait pada v_3 dan v_5

e_8 terkait pada v_4

Graph Sederhana

Graph G(V,E) disebut graph sederhana jika graph G tersebut tidak memiliki loop atau sisi rangkap.

Perhatikan graph berikut ini



A B C

Gambar 4

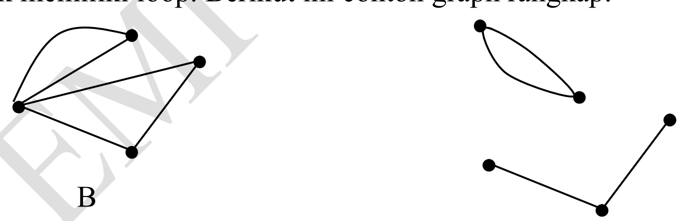
Keterangan : Graph A graph sederhana

Graph B graph sederhana

Graph C bukan merupakan graph sederhana

Graph rangkap (*multi graph*)

Graph G(V,E) disebut graph rangkap jika graph tersebut memiliki sisi rangkap tetapi tidak memiliki loop. Berikut ini contoh graph rangkap.



B C

Gambar 5

Graph Kosong

Graph G(V,E) disebut graph kosong jika graph tersebut tidak memiliki sisi. Berikut ini contoh graph kosong.

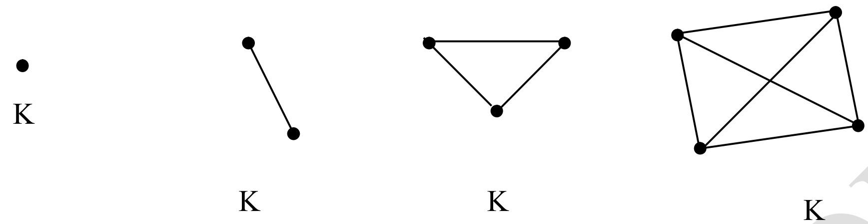
D P

Gambar 6

YEMI KUSWARDI

Graph Komplit

Graph G(V,E) disebut graph komplit jika graph G tersebut graph sederhana dan setiap dua titik pada graph G tersebut dihubungkan oleh sebuah titik. Graph komplit dengan n titik dilambangkan dengan K_n. Berikut ini contoh graph komplit.

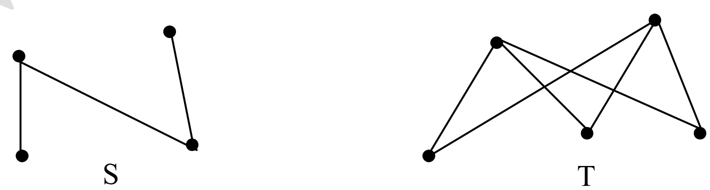


Gambar 7

Jika G adalah graph lengkap dengan n titik, maka dapat dengan mudah ditunjukkan bahwa jumlah sisinya adalah $^1/2$ n (n-1).

Graph Bipartisi

Graph G(V.E) adalah graph bipartisi. Jika V(G) dapat dipartisi menjadi dua subhimpunan X dan Y yang saling asing ($X \cup Y = V(G)$ dan $X \cap Y = \phi$) sedemikian rupa sehingga setiap sisi pada G mempunyai satu titik ujung di X dan satu titik ujung di Y. Pada graph bipartisi apabila setiap titik di X terhubung dengan setiap titik di Y begitu pula sebaliknya maka graph tersebut disebut graph bipartisi komplit. Graph bipartisi komplit yang titik-titiknya terpartisi dalam subhimpunan X beranggotakan m titik dan Y beranggotakan n titik dilambangkan dengan $K_{m,n}$ atau $K_{n,m}$. Perhatikan graph berikut ini.



Gambar 8

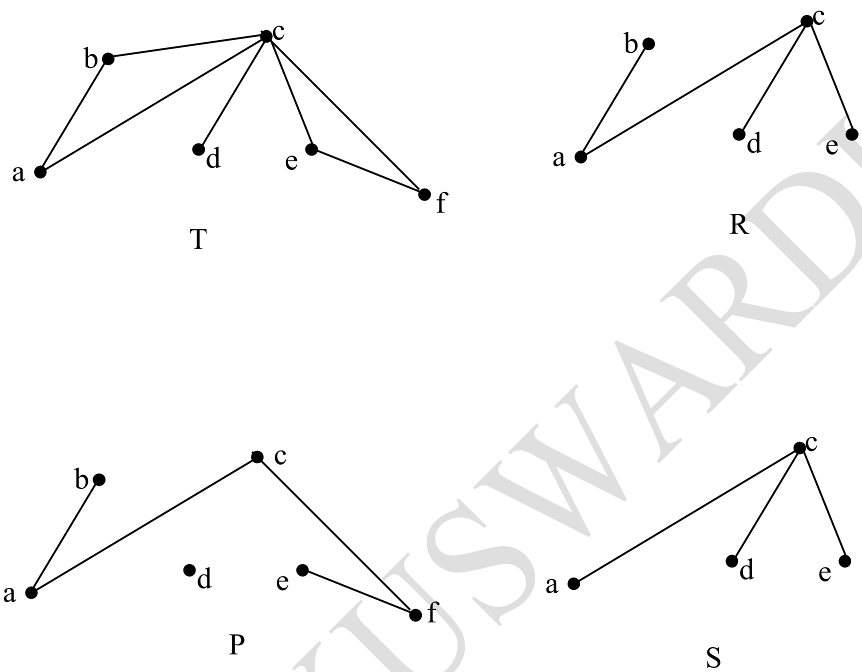
Keterangan Graph S merupakan graph bipartisi

Graph T merupakan graph bipartisi komplit (= $K_{2,3}$)

Graph Bagian (*subgraph*)

Sebuah graph H disebut graph bagian dari graph G ($H \subseteq G$) jika $V(H) \subseteq V(G)$ dan $E(H) \subseteq E(G)$. Jika $H \subseteq G$ dan $V(H) = V(G)$ maka H disebut graph bagian rentang (*spanning subgraph*) dari graph G. Misalkan $V_1 \subseteq V(G)$. Graph bagian rentang dari G yang dibangun oleh V_1 (= $G[V_1]$) adalah sebuah graph bagian dari G yang himpunan titik-titiknya adalah V_1 dan himpunan sisinya beranggotakan semua ssi G yang mempunyai titik akhir di V_1.

Perhatikan graph berikut ini.



Gambar.9

Keterangan: Graph P, R, S merupakan graph bagian dari graph T

Graph S graph bagian dari graph R

Graph P graph bagian rentang dari graph T

Graph S adalah graph bagian rentang dari graph T yang dibangun oleh $V_1 = \{a, c, d, e\}$

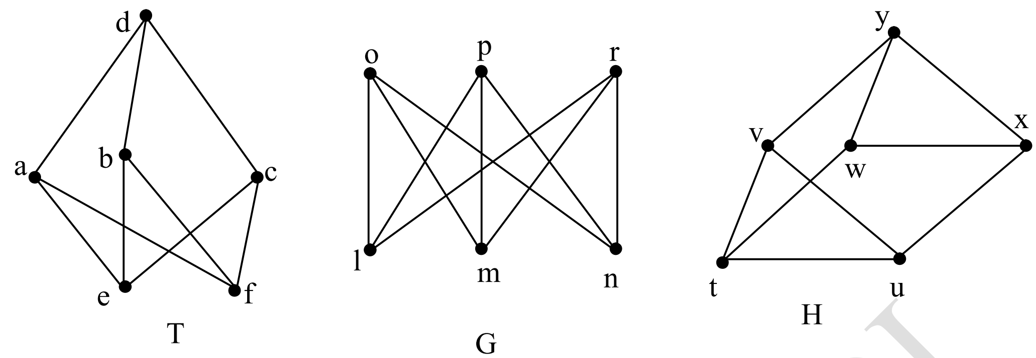
Isomorfik

Graph G dan graph H disebut isomorfik $(= G \infty H)$ jika

- terdapat korespondensi satu-satu antara V(G) dan V(H)
- banyak sisi yang menghubungkan titik u dan v di V(G) sama dengan banyaknya sisi yang menghubungkan dua titik di V(H) yang berkorespondensi satu-satu dengan titik-titik u dan v

YEMI KUSWARDI

Sebagai akibat: jika graph G dan H isomorfik maka V(G) = V(H) dan E(G) = E(H) (tidak berlaku sebaliknya). Perhatikan graph berikut ini.



Gambar 10

Keterangan: Graph T isomorfik dengan graph G

Graph H tidah isomorfik dengan graph T maupun graph G

Graph T isomorfik dengan graph G karena

- terdapat korespondensi satu-satu antara V(G) = {l, m, n, o, p, q} dan V(T) = {a, b, c, d, e, f}, yaitu a$\leftrightarrow$o, b$\leftrightarrow$p, c$\leftrightarrow$r, d$\leftrightarrow$l, e$\leftrightarrow$m, f$\leftrightarrow$n
- Pada graph T: a berhubungan langsung dengan d, e, f

  b berhubungan langsung dengan d, e, f

  c berhubungan langsung dengan d, e, f

  Pada graph G:

  o yang berkorespondensi dengan a berhubungan langsung dengan l, m, n yang masing masing berkorespondensi dengan d, e, f.

  p yang berkorespondensi dengan b berhubungan langsung dengan l, m, n yang masing masing berkorespondensi dengan d, e, f.

  r yang berkorespondensi dengan c berhubungan langsung dengan l, m, n yang masing masing berkorespondensi dengan d, e, f.

Jalan (*Walk*)

Sebuah jalan di graph G adalah sebuah barisan berhingga dan tak kosong yang suku-sukunya bergantian titik dan sisi sedemikian sehingga v_{i-1} dan v_i adalah titik-titik akhir sisi e_i.

Misalkan $W = v_0e_1v_1e_2v_2e_3v_3 \ \ldots \ e_kv_k$ untuk $1 \leq i \leq k$

Maka W disebut jalan dari v_0 ke v_k atau jalan-(v_0,v_k)

v_0 disebut titik awal dari W

v_k disebut titik akhir dari W

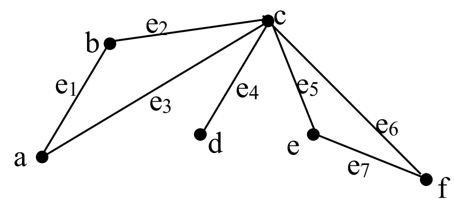
YEMI KUSWARDI

$v_1, v_2, v_3, \ldots, v_{k-1}$ disebut titik –titik internal dari W

k disebut panjang dari W

Perhatikan bahwa panjang dari jalan W adalah banyaknya sisi dalam W. Sebuah jalan W dengan panjang positif disebut tertutup jika titik awal dan titik akhir dari W adalah sama.

Perhatikan graph G berikut ini.



G

Gambar 11

Pada graph G di atas ada beberapa jalan dari a ke f diantaranya:

$ae_1b\ e_2c\ e_6f$ jalan-(a,f) dengan panjang 3

$ae_1b\ e_2c\ e_5ee_7f$ jalan-(a,f) dengan panjang 4

$ae_3c\ e_6f$ jalan-(a,f) dengan panjang 2

$ae_3ce_5ee_7f$ jalan-(a,f) dengan panjang 3

$ae_1be_2ce_3ae_1be_2ce_6f$ jalan-(a,f) dengan panjang 6

Pada graph G di atas ada beberapa jalan tertutup dari a ke a diantaranya

$ae_1b\ e_2c\ e_3a$ jalan-(a,a) dengan panjang 3

$ae_1be_2ce_6fe_7ee_5ce_3a$ jalan-(a,a) dengan panjang 6

$ae_1be_2ce_6fe_7ee_5ce_2be_1a$ jalan-(a,a) dengan panjang 7

Jejak (*Trail*)

Sebuah jalan apabila semua sisinya berbeda disebut jejak. Pada graph G Gambar 11, $ae_1be_2ce_6f$ adalah jalan-(a,f) yang semua sisinya berbeda, maka $ae_1be_2ce_6f$ merupakan salah satu contoh jejak-(a,f). sedangkan $ae_1be_2ce_3ae_1be_2ce_6f$ adalah jalan-(a,f) tetapi bukan jejak-(a,f) karena ada sisi yaitu e_1 yang dilalui lebih dari satu kali.

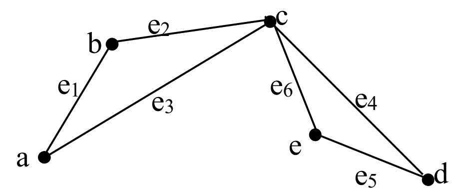
Jejak tertutup atau sering disebut sirkit adalah sebuah jalan tertutup yang semua sisinya berbeda. Pada graph G Gambar 11, $ae_1be_2ce_3a$ adalah jalan-(a,a) yang semua sisinya berbeda, maka $ae_1be_2ce_3a$ disebut sirkit, sedangkan

YEMI KUSWARDI

$ae_1be_2ce_6fe_7ee_5ce_2be_1a$ jalan-(a,a) yang bukan merupakan sirkit karena pada jalan tertutup tersebut ada sisi yang dilalui lebih dari satu kali yaitu e_1.

Sebuah sirkit pada sebuah graph yang memuat semua sisi pada graph tersebut disebut sirkit Euler. Sebuah Graph yang memuat sirkit Euler disebut graph Euler.

Perhatikan graph H berikut ini.



H

Gambar 12

Pada graph H di atas memuat sebuah sirkit $ae_1be_2ce_4de_5ee_6ce_3a$. Sirkit tersebut memuat semua sisi pada graph H, maka sirkit $ae_1be_2ce_4de_5ee_6ce_3a$ ini disebut sirkit Euler. Karena graph H memuat sirkit Euler maka graph H tersebut merupakan graph Euler.

Perhatikan kembali graph G pada Gambar 11. Graph G tersebut tidak memuat sebuah sirkit yang memuat semua sisi pada graph G tersebut, maka graph G pada Gambar 11 bukan merupakan graph Euler.

Lintasan (*Path*)

Sebuah lintasan di graph G adalah sebuah jalan apabila semua sisi dan semua titik berbeda.

Perhatikan kembali graph G pada Gambar 11,

$ae_1b\ e_2c\ e_6f$ adalah sebuah lintasan-(a,f) dengan panjang 3.

$ae_1b\ e_2c\ e_5ee_7f$ adalah sebuah lintasan-(a,f) dengan panjang 4.

$ae_3c\ e_6f$ adalah sebuah lintasan-(a,f) dengan panjang 2,

$ae_3ce_5ee_7f$ adalah sebuah lintasan-(a,f) dengan panjang 3,

$ae_1be_2ce_3ae_1be_2ce_6f$ bukan merupakan lintasan karena ada titik yang dilewati lebih dari satu kali, yaitu titik b.

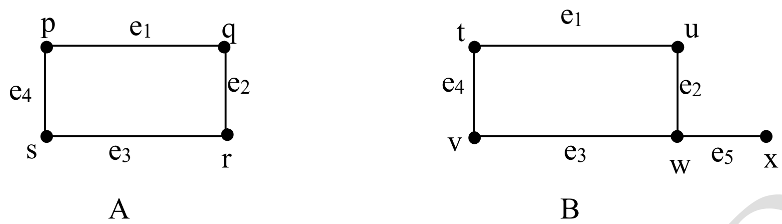
Sebuah lintasan yang tertutup atau sebuah jejak tertutup/sirkit yang titik awal dan semua titik internalnya berbeda disebut sikel (*cycle*). Pada graph H di atas $ae_1be_2ce_3a$ adalah sirkit yang titik awal dan semua titik internalnya berbeda, maka $ae_1be_2ce_3a$ merupakan sebuah sikel. Pada graph H Gambar 12 $ae_1be_2ce_4de_5ee_6ce_3a$.

YEMI KUSWARDI

adalah sebuah sirkit, namun bukan sebuah sikel karena pada sirkit tersebut ada titik internal yang sama yaitu c dilewati lebih dari satu kali.

Sebuah sikel yang memuat semua titik pada sebuah graph disebut sikel Hamilton. Graph yang memuat sikel Hamilton disebut Graph Hamilton.

Perhatikan graph berikut ini



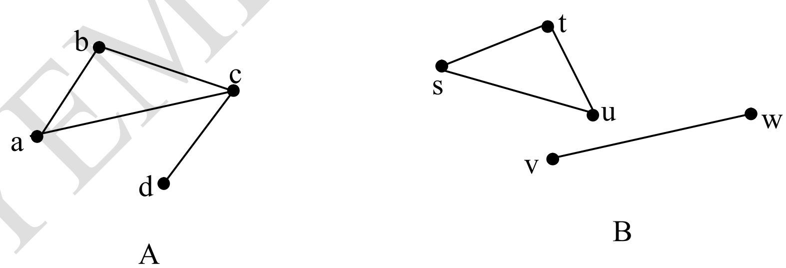
Gambar 13

Pada graph A di atas memuat sikel yaitu $pe_1qe_2re_3se_4p$. sikel pada graph A tersebut memuat semua titik pada graph a tersebut, maka sikel tersebut disebut sikel Hamilton. Karena memuat sikel Hamilton maka graph A merupakan graph Hamilton. Padagraph B dia tas tidak memuat sikel Hamilton maka graph B tersebut bukan merupakan graph Hamilton.

Graph Terhubung

Sebuah graf G dikatakan terhubung jika untuk setiap dua titik u dan v di G terdapat lintasan di G yang menghubungkan kedua titik tersebut.

Perhatikan graph berikut ini.



Gambar 14

Graph A di atas adalah graph terhubung karena setiap dua titik yang berbeda di A terdapat lintasan yang menghubungkan dua titik tersebut. Graph B bukan merupakan graph terhubung karena ada dua titik di G yang tidak dihubungkan oleh lintasan, contoh titik $u,v \in V(G)$, tetapi tidak ada lintasan di G yang menghubungkan titik u dan v tersebut.

Catatan : graph disebut graph terhubung.

YEMI KUSWARDI

Komponen Graph

Syarat sebuah graf dikatakan komponen dari graf G adalah

- Sebuah graf bagian
- Terhubung maksimal (titik dan sisi) dari graf G

  Terhubung maksimal: tidak ada lagi graf bagian lain yang terhubung dan memuat dia

Perhatikan kembali graph A dan B pada Gambar 14.

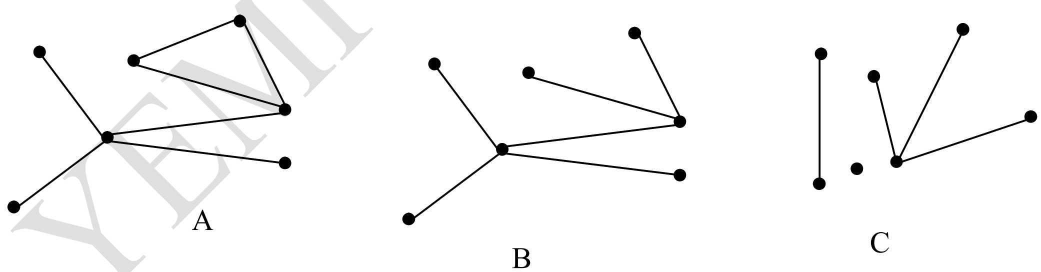
Graph A merupakan graph terhubung, maka komponen graph A adalah graph A itu sendiri. Graph B adalah graph tidak terhubung dengan dua komponen, yaitu

Gambar 15

Pohon (*Tree*) dan Hutan (*Forest*)

Sebuah graph dikatakan pohon apabila graph tersebut terhubung dan tidak memiliki sikel. Sedangkan sebuah graf yang setiap komponennya berupa pohon disebut Hutan.

Perhatikan graph berikut ini.



Gambar 16

Keterangan   Graph A bukan pohon

Graph B pohon

Graph C hutan

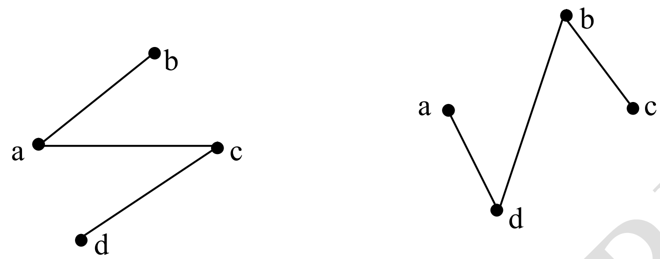
YEMI KUSWARDI

Komplemen Graph

Jika G graf sederhana maka komplemen graf G (= $\overline{G}$) mempunyai ciri

- Himpunan titik $\overline{G}$ sama dengan himpunan titik di G
- Dua titik u dan v di $\overline{G}$ berhubungan langsung jika dan hanya jika dua titik u dan v tersebut tidak berhubungan langsung di G

Perhatikan Graph G dan komplemen dari graph G berikut ini



G $\qquad\qquad$ $\overline{G}$

Gambar 17

Derajat Titik

Derajat titik v di graph G adalah banyaknya sisi G yang terkait di titik v. Derajat titik v di graph G dilambangkan dengan $d_G(v)$ atau d(v). Derajat minimum dari G dilambangkan dengan $\delta(G)$ didefinisikan sebagai berikut :

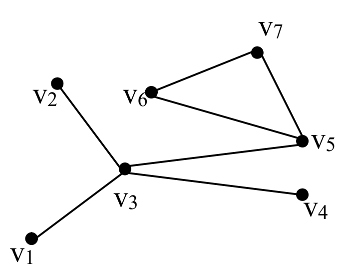
$$\delta(G) = \text{minimum } \{d(v) | \; v \in V(G)\}$$

Sedangkan derajat maksimum dari G dilambangkan dengan $\Delta(G)$ didefinisikan sebagai berikut:

$$\Delta(G) = \text{maksimum } \{d(v) | \; v \in V(G)\}$$

Graf G disebut beraturan-k jika setiap titik di G berderajat k

Perhatikan graph A berikut ini.



A

Gambar 18

YEMI KUSWARDI

Pada graph A tersebut $d(v_1) = 1$, $d(v_2) = 1$, $d(v_3) = 4$, $d(v_4) = 1$, $d(v_5) = 3$, $d(v_6) = 2$, $d(v_7) = 2$

$$\delta(A) = \text{minimum } \{d(v) \mid v \in V(A)\} = 1$$

$$\Delta(A) = \text{maksimum } \{d(v) \mid v \in V(A)\} = 4$$

Teorema (Jabat Tangan)

Untuk setiap graph G berlaku

$$\sum_{v \in V(G)} d(v) = 2\,|E(G)|$$

Akibat dari teorema tersebut maka banyak titik yang berderajat ganjil dalam suatu graph adalah genap.

Berikut ini bukti bahwa banyak titik erderajat ganjil pada suatu graph adalah genap.

Bukti : misalkan G(V(G),E(G)) adalah graph. V(G) dapat dipartisi menjadi dua himpunan A dan B dengan

A = himpunan titik-titik di graph G yang berderajat Ganjil

B = himpunan titik-titik di graph G yang berderajat Genap

Menurut teorema jabat tangan untuk setiap graph G berlaku

$$\sum_{v \in V(G)} d(v) = 2\,|E(G)|$$

$$\leftrightarrow \sum_{v \in A \cup B} d(v) = 2\,|E(G)|$$

$$\leftrightarrow \sum_{v \in A} d(v) + \sum_{v \in B} d(v) = 2\,|E(G)|$$

$$\leftrightarrow \sum_{v \in A} d(v) + \mathit{GENAP} = \mathit{GENAP}$$ (semua vertek di B berderajat genap maka jumlah semua derajatvertek di B genap)

$$\leftrightarrow \sum_{v \in A} d(v) = \mathit{GENAP}$$

Karena semua vertek di A berderajat genap maka jumlah semua derajat vertek di A akan genap jika banyaknya vertek di A genap. Jadi, terbukti bahwa banyaknya vertek berderajat ganjil pada graph G genap

YEMI KUSWARDI

LATIHAN:

1. Mungkinkah dalam suatu pesta yang dihadiri 7 orang, setiap yang hadir mengetahui tepat 3 orang yang lain?
2. Jika diketahui G graph beraturan dengan 10 sisi, maka berapakah jumlah vertek pada graph tersebut?
3. Jika G graph sederhana dengan n vertek m sisi dan k komponen, maka buktikan bahwa $m \geq n - k$
4. Jika G graph dengan derajat setiap titiknya paling sedikit dua, maka G memuat sikel
5. Jika G graph bipartisi sederhana dengan n vertek dan m sisi, maka buktikan bahwa $m \leq \frac{n^2}{4}$
6. Jika G graph bipartisi sederhana, maka tunjukkan panjang sikel di graph G adalah genap!

YEMI KUSWARDI

Penyajian Graph Dalam Bentuk Matriks

Sebuah graf dapat disajikan dalam bentuk matriks terhubung langsung (*adjacency matrix*) dan matriks keterkaitan (*incidence matrix*)

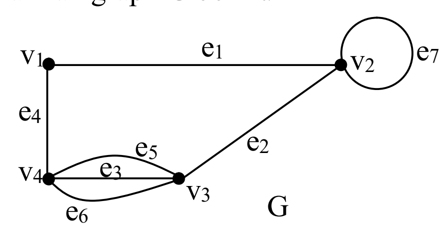
Matriks Terhubung Langsung

Misal G adalah sebuah graph dengan $V(G) = \{ v_1, v_2, v_3, \ldots, v_n \}$, matriks terhubung langsung dari G adalah matrik matriks bujur sangkar berordo n,

$$A(G) = (a_{ij})$$

Dengan a_{ij} : banyaknya sisi yang menghubungkan titik v_i dan titik v_j

Perhatikan graph G berikut ini



$$A(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 3 \\ 1 & 0 & 3 & 0 \end{bmatrix} \end{array}$$

Gambar 19

Perhatikan bahwa A(G) adalah matriks simetris. Jika graph G tidak punya loop maka setiap unsur A(G) pada diagonal utama adalah nol. Jika G graph sederhana maka graph tersebut tidak memiliki sisi rangkap dan loop sehingga unsur-unsur dari A(G) adalah nol atau satu. Jika G graph komplit dengan n titik maka A(G) merupakan matriks berordo n dan unsur-unsur pada diagonal utama matriks tersebut 0 serta unsur-unsur selain diagonal utama adalah 1.

Dari uraian di atas dapat disimpulkan bahwa syarat sebuah matriks agar matriks tersebut merupakan matriks terhubung langsung sebuah graph adalah:

- Matriks bujur sangkar
- Simitris terhadap diagonal utama
- Elemen-elemen matriks bulat non negatif

Perhatikan kembali matrik A(G) di atas. Unsur-unsur pada $A(G)=a_{ij}$ yang terletak pada baris ke-i dan kolom ke-j menunjukkan banyaknya jalan-(v_i,v_j) dengan panjang satu. Sekarang, kalikan matriks A(G) dengan dirinya sendiri, maka akan diperoleh

YEMI KUSWARDI

$$A^2(G) = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 3 \\ 1 & 0 & 3 & 0 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 3 \\ 1 & 0 & 3 & 0 \end{bmatrix} = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 2 & 1 & 4 & 0 \\ 1 & 3 & 1 & 4 \\ 4 & 1 & 10 & 0 \\ 0 & 4 & 0 & 10 \end{bmatrix} \end{array}$$

Perhatikan bahwa unsur $A^2(G)$ yang terletak pada baris ke-i dan kolom ke-j menyatakan banyaknya jalan-(v_i,v_j) dengan panjang dua.

Contoh :

unsur $A^2(G)$ yang terletak pada baris ke-1 dan kolom ke-3 adalah 4. Jadi, ada 4 jalan-(v_1,v_3) di graph G dengan panjang 2, yaitu $v_1e_1v_2e_2v_3$, $v_1e_4v_4e_5v_3$, $v_1e_4v_4e_3v_3$, dan $v_1e_4v_4e_6v_3$,

Berikut ini diberikan generalisasi dari observasi di atas.

Teorema

Misalkan G graph dengan $V(G) = \{v_1, v_2, . . . ,v_n\}$. Jika A matriks berhubungan langsung dari G, maka unsur A^k untuk suatu bilangan bulat positif k yang terletak di baris ke-i dan kolom ke-j menyatakan banyaknya jalan di G dengan panjang k yang menghubungkan titik v_1 dan titik v_j.

Matriks Keterkaitan

Misal G adalah sebuah graph dengan $V(G) = \{ v_1, v_2, v_3, \ldots,v_n\}$ dan $E(G) = \{e_1, e_2, e_3, \ldots e_s\}$, matriks keterkaitan dari G adalah matriks segiempat berordo n x s

$$M(G) = (m_{ij})$$

Dengan n : banyaknya titik di G

s : banyaknya sisi di G

$$m_{ij} = \begin{cases} 0, & e_j \text{ tidak terkait } v_i \\ 1, & e_j \text{ terkait } v_i \text{ dan } e_j \text{ bukan loop} \\ 2, & e_j \text{ terkait } v_i \text{ dan } e_j \text{ adalah loop} \end{cases}$$

Perhatikan kembali graph G pada Gambar 18. Berikut ini matriks keterkaitan dari graph tersebut.

$$M(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{ccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 2 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

YEMI KUSWARDI

Perhatikan bahwa jumlah semua unsur M(G) yang terletak pada baris ke-i menyatakan derajat dari titik v_i di graph G, sedangkan jumlah semua unsur M(G) yang terletak pada kolom selalu 2.

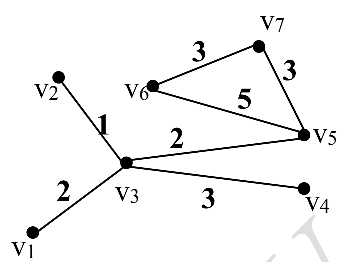
YEMI KUSWARDI

PENERAPAN GRAPH

Graph Bobot

Graph bobot adalah sebuah graph yang setiap sisinya dikaitkan dengan bilangan real. Sebuah bilangan real w(e) yang dikaitkan pada setiap sisi pada sebuah graph sering disebut bobot dari e. bobot dari sebuah grap G dilambangkan dengan w(G) adalah jumlah bobot semua sisi G.

Perhatikan graph bobot B berikut ini.



B

Gambar 20

Bobot dari graph B tersebut adalah w(B) = 2 + 1 + 3 + 2 + 5 + 3 + 3 = 19

Lintasan Terpendek

Pada sebuah graph bobot G, misalkan u dan v sembarang titik pada graph G maka yang dimaksud lintasan terpendek antara u dan v di G adalah sebuah lintasan (u,v) di G yang panjangnya minimal. Panjang lintasan pada sebuah graph bobot adalah jumlah bobot dari semua sisi dalam lintasan tersebut. Sedangkan jarak dari u ke v di G (disimbolkan dengan $d_G(u,v)$) didefinisikan sebagai panjang lintasan terpendek antara titik u dan titik v di G.

Perhatikan kembali Gambar 20. Ada beberapa lintasan (v_3,v_6), yaitu lintasan (v_3, v_5, v_7, v_6) dengan panjang lintasan 2 + 3 + 3 = 8 dan lintasan (v_3,v_5,v_6) dengan panjang lintasan 2 + 5 = 7. Dengan demikian lintasan terpendek dari v_3 ke v_6 adalah (v_3,v_5,v_6), sehingga $d_G(v_3,v_6) = 7$.

Andaikan pada Gambar 20, titik dalam graph tersebut mewakili kota, sisi mewakili jalan antara dua kota dan bobot masing-masing sisi menyatakan panjang jalan yang diwakili sisi tersebut. Misalkan kita berada di kota v_1 akan pergi ke kota v_7, maka ada beberapa lintasan yang dapat dilalui, yaitu lintasan (v_1, v_3, v_5, v_7) atau $(v_1, v_3, v_5, v_6, v_7)$ yang secara berturut-turut panjangnya 7 dan 12. tentu saja dari segi

YEMI KUSWARDI

aplikasi diantara dua lintasan tersebut lintasan (v_1, v_3, v_5, v_7) yang paling menguntungkan. Persoalannya adalah bagaimana menentukan lintasan terpendek antara dua titik dalam sebuah graph bobot tertentu.

Untuk mencari panjang lintasan terpendek dari sebuah titik s ke sebuah titik t di graph bobot G, dimana bobot setiap sisi G adalah bilangan non-negatif, dapat digunakan algoritma yang dikembangkan oleh Dijkstra pada tahun 1959. Berikut ini algoritma Dijkstra.

Algoritma Dijkstra

- Input : Graph bobot G dengan s, t $\in$ V(G)
- Step 1: Label titik s dengan $\lambda(s) = 0$

  Label titik-titik di G selain s dengan $\lambda(v) = \infty$

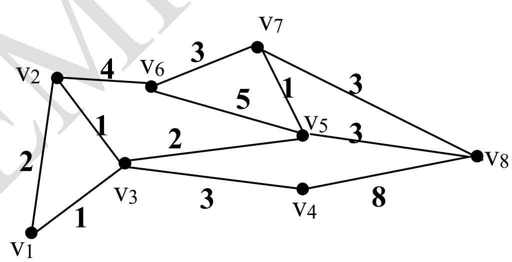
  Tulis T=V(G) atau T=himpunan titik di G yang berlabel sementara
- Step 2: Misal u $\in$ T dengan $\lambda(u) = \text{minimum}$
- Step 3: Jika u = t maka **STOP!!!!!!**

  Jika u $\neq$ t, lanjut ke step 4
- Step 4: Untuk setiap sisi e = uv , v $\in$ T ganti label v dengan

  $\lambda(v) = \text{minimum}\ \{\lambda(v), \lambda(u) + w(e)\}$
- Step 5: Tulis T = T – {u}

  Lanjut ke Step 2

Perhatikan graph bobot G berikut ini.



Berikut ini akan dicari lintasan terpendek dari v_1 ke v_8 di graph bobot G dengan menerapkan algoritma Dijkstra.

Input : graph bobot G

<u>Iterasi 1</u>

Step 1: Label v_1 dengan $\lambda(v_1) = 0$

Label v_i dengan $\lambda(v_i) = \infty$ untuk setiap $2 \leq i \leq 8$

Tulis T = {v_1,v_2,v_3,v_4, . . . ,v_8}

Step 1 ini dapat kita tulis dalam bentuk tabel seperti berikut ini.

Titik v_i	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
$\lambda(v_i)$	**0**	∞	∞	∞	∞	∞	∞	∞
T	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8

Step 2: $u = v_1 \in T$ dengan $\lambda(v_1)$ minimal

Step 3: Karena $u \neq v_8$ maka lanjut ke Step 4

Step 4: Terdapat 2 sisi di G yang terkait dengan v_1, yaitu v_1v_2, v_1v_3 dengan $v_2, v_3 \in T$ sehingga label v_2, v_3 berubah menjadi:

$$\lambda(v_2) = \text{minimum}\,\{\lambda(v_2), \lambda(v_1) + w(v_1v_2)\} = \min\{\infty, 0+2\} = 2$$

$$\lambda(v_3) = \text{minimum}\,\{\lambda(v_3), \lambda(v_1) + w(v_1v_3)\} = \min\{\infty, 0+1\} = 1$$

step 5: Tulis T= T – $\{v_1]$ = $\{v_2, v_3, v_4, \ldots, v_8\}$

Pada tahap ini dikatakan bahwa titik v_1 telah dilabel permanen dengan label $\lambda(v_1) = 0$

Iterasi 2

Step 1:

Titik v_i	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
$\lambda(v_i)$	**0**	**2**	**1**	∞	∞	∞	∞	∞
T	-	v_2	v_3	v_4	v_5	v_6	v_7	v_8

Step 2: $u = v_3 \in T$ dengan $\lambda(v_3)$ minimal

Step 3: Karena $u \neq v_8$ maka lanjut ke Step 4

Step 4: Terdapat 3 sisi di G yang terkait dengan v_3, yaitu v_3v_2, v_3v_4, v_3v_5 dengan $v_2, v_4, v_5 \in T$, sehingga label v_2, v_4,$v_5$berubah menjadi:

$$\lambda(v_2) = \text{minimum}\,\{\lambda(v_2), \lambda(v_3) + w(v_3v_2)\} = \min\{2, 1+1\} = 2$$

$$\lambda(v_4) = \text{minimum}\,\{\lambda(v_4), \lambda(v_3) + w(v_3v_4)\} = \min\{\infty, 1+3\} = 4$$

$$\lambda(v_5) = \text{minimum}\,\{\lambda(v_5), \lambda(v_3) + w(v_3v_5)\} = \min\{\infty, 1+2\} = 3$$

step 5: Tulis T= T – $\{v_3]$ = $\{v_2, v_4, \ldots, v_8\}$

Pada tahap ini dikatakan bahwa titik v_3 telah dilabel permanen dengan label $\lambda(v_3) = 1$

Iterasi 3

Step 1:

Titik v_i	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
$\lambda(v_i)$	**0**	**2**	**1**	**4**	**3**	∞	∞	∞
T	-	v_2	-	v_4	v_5	v_6	v_7	v_8

Step 2: $u = v_2 \in T$ dengan $\lambda(v_2)$ minimal

Step 3: Karena $u \neq v_8$ maka lanjut ke Step 4

Step 4: Terdapat 1 sisi di G yang terkait dengan v_2, yaitu v_2v_6 dengan $V_6 \in T$,

YEMI KUSWARDI

sehingga label v_6 berubah menjadi:

$$\lambda(v_6) = \text{minimum}\{\lambda(v_6), \lambda(v_2) + w(v_2 v_6)\} = \min\{\infty, 2+4\} = 6$$

step 5: Tulis T= T – {v_2] = {v_4, . . . ,v_8}

Pada tahap ini dikatakan bahwa titik v_2 telah dilabel permanen dengan label $\lambda(v_2) = 2$

Iterasi 4

Step 1:

Titik v_i	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
λ (v_i)	0	2	1	4	3	6	∞	∞
T	-	-	-	v_4	v_5	v_6	v_7	v_8

Step 2: u = $v_5 \in$ T dengan λ (v_5) minimal

Step 3: Karena u $\neq v_8$ maka lanjut ke Step 4

Step 4: Terdapat 3 sisi di G yang terkait dengan v_5, yaitu v_5v_6, v_5v_7, v_5v_8 dengan v_6, v_7,$v_8 \in$ T , sehingga label v_6, v_7,v_8 berubah menjadi:

$$\lambda(v_6) = \text{minimum}\{\lambda(v_6), \lambda(v_5) + w(v_5 v_6)\} = \min\{6, 3+5\} = 6$$

$$\lambda(v_7) = \text{minimum}\{\lambda(v_7), \lambda(v_5) + w(v_5 v_7)\} = \min\{\infty, 3+1\} = 4$$

$$\lambda(v_8) = \text{minimum}\{\lambda(v_8), \lambda(v_5) + w(v_5 v_8)\} = \min\{\infty, 3+3\} = 6$$

step 5: Tulis T= T – {v_5] = {v_4,v_6, v_7 ,v_8}

Pada tahap ini dikatakan bahwa titik v_5 telah dilabel permanen dengan label $\lambda(v_5) = 3$

Iterasi 5

Step 1:

Titik v_i	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
λ (v_i)	0	2	1	4	3	6	4	6
T	-	-	-	v_4	-	v_6	v_7	v_8

Step 2: u = $v_4 \in$ T dengan λ (v_4) minimal

Step 3: Karena u $\neq v_8$ maka lanjut ke Step 4

Step 4: Terdapat 1 sisi di G yang terkait dengan v_4 yaitu $v_4v_8$dengan $v_8 \in$ T , sehingga label v_8 berubah menjadi:

$$\lambda(v_8) = \text{minimum}\{\lambda(v_8), \lambda(v_4) + w(v_4 v_8)\} = \min\{6, 4+8\} = 6$$

step 5: Tulis T= T – {v_4] = {v_6, v_7 ,v_8}

Pada tahap ini dikatakan bahwa titik v_4 telah dilabel permanen dengan label $\lambda(v_4) = 4$

YEMI KUSWARDI

Iterasi 6

Step 1:

Titik v_i	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
λ (v_i)	**0**	**2**	**1**	**4**	**3**	**5**	**4**	**6**
T	-	-	-	-	-	v_6	v_7	v_8

Step 2: u = $v_7 \in$ T dengan λ (v_7) minimal

Step 3: Karena u $\neq v_8$ maka lanjut ke Step 4

Step 4: Terdapat 2 sisi di G yang terkait dengan v_7 yaitu v_7v_6, v_7v_8 dengan v_6, $v_8 \in$ T , sehingga label v_6, v_8 berubah menjadi:

$$\lambda(v_6) = \text{minimum}\{\lambda(v_6), \lambda(v_7) + w(v_7v_6)\} = \min\{6, 4+3\} = 6$$

$$\lambda(v_8) = \text{minimum}\{\lambda(v_8), \lambda(v_4) + w(v_4v_8)\} = \min\{6, 4+3\} = 6$$

step 5: Tulis T= T – {v_7] = {v_6,v_8}

Pada tahap ini dikatakan bahwa titik v_7 telah dilabel permanen dengan label $\lambda(v_7) = 4$

Iterasi 7

Step 1:

Titik v_i	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
λ (v_i)	**0**	**2**	**1**	**4**	**3**	**5**	**4**	**6**
T	-	-	-	-	-	v_6	-	v_8

Step 2: u = $v_8 \in$ T dengan λ (v_8) minimal

Step 3: Karena u $= v_8$ maka Stop!!!

Lintasan terpendek dari v_1 ke v_8 telah diperoleh dengan panjang lintasan terpendek sebesar $\lambda(v_8) = 6$.

Untuk mengetahui lintasan terpendek tersebut, untuk selanjutnya digunakan metode telusur balik dari v_8 ke v_1. Perhatikan penerapan metode telusur balik berikut ini.

$$\lambda(v_8) = 6 = 3 + 3 = \lambda(v_5) + w(v_5v_8)$$

$$\lambda(v_5) = 3 = 1 + 2 = \lambda(v_3) + w(v_3v_5)$$

$$\lambda(v_3) = 1 = 0 + 1 = \lambda(v_1) + w(v_1v_3)$$

Jadi, $\lambda(v_8) = 6 = w(v_1v_3) + w(v_3v_5) + w(v_5v_8)$

Dengan demikian lintasan terpendek dengan panjang 6 dari v_1 ke v_8 di graph bobot G adalah lintasan (v_1,v_3, v_5, v_8).

YEMI KUSWARDI

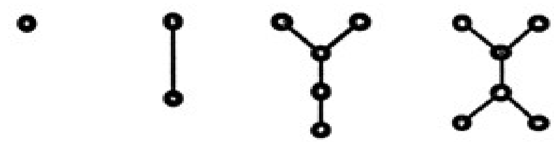
Pohon

Pohon (*tree*) adalah graf terhubung yang tidak memiliki sikel.

<u>Sifat-sifat Pohon :</u>

Misalkan $G = (V, E)$ adalah graf sederhana dan banyak titiknya n buah. Pernyataan-pernyataan di bawah ini adalah ekivalen.

1. G adalah pohon.
2. Setiap pasang titik di G terdapat tepat satu lintasan.
3. G terhubung dan memiliki $n - 1$ buah sisi.
4. G tidak mengandung sikel dan memiliki $n - 1$ buah sisi.
5. G terhubung dan semua sisinya adalah jembatan.



T_1 T_2 T_3 T_4

Pohon Rentang

Misalkan G adalah sebuah graph, sebuah pohon di G yang memuat semua titik G disebut pohon rentang (*spanning tree*) dari G. Setiap graph terhubung pasti memuat pohon rentang, berlaku pula sebaliknya sebuah graph apabila memuat pohon rentang maka pasti graph tersebut merupakan graph terhubung.

Pada sebuah graph bobot terhubung G, pohon rentang di G yang memiliki bobot minimum disebut pohon rentang minimum dari graph G. Untuk mencari pohon rentang minimum pada sebuah graph bobot terhubung dapat digunakan algoritma Prim. Algoritma ini sangat praktis dan efisien untuk mencari pohon rentang minimum pada sebuah graph bobot terhubung. Berikut ini langkah-langkah dari algoritma Prim.

Input : Graph bobot terhubung G dengan n titik

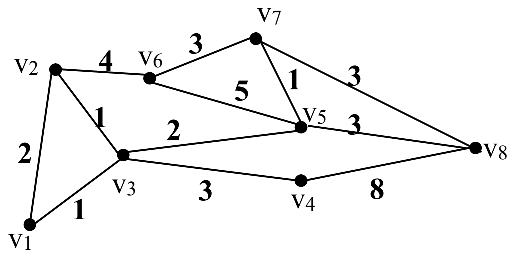
Step 1: pilih sebuah titik dari G dan tulis T_1=v

Step 2: pilih sebuah sisi e_k dengan bobot minimum yang menghubungkan sebuah titik T_k dengan sebuah titik G yang bukan di T_k. jika terdapat lebih dari satu sisi yang demikian, pilih salah satu sisi sembarang.
Tulis $T_{k+1} = T_k \cup e_k$

Srep 3: Jika n – 1 sisi telah terpilih (k = n – 1) Stop!!!
Beri pesan T_{k+1} adallah pohon rentang minimum di G
Jika k < n – 1 maka kembali ke Step 2

YEMI KUSWARDI

Berikut ini contoh penerapan Algoritma Prim untuk mencari pohon rentang minimum pada graph G berikut ini

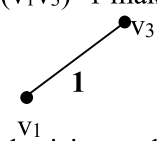


Input : Graph bobot terhubung G dengan 8 titik

Step 1: pilih sebuah titik dari G misal titik v_1 dan tulis T_1=v_1

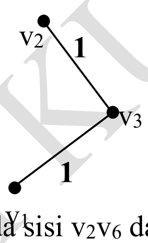
Step 2: Karena v_1 terkait pada sisi v_1v_2, v_1v_3, dengan v_2,v_3 tidak di T_1 dan $W(v_1v_2)=2$, $W(v_1v_3)=1$ maka pilih v_1v_3 karena bobot minimum.

$T_2 = T_1 \cup v_1v_3$



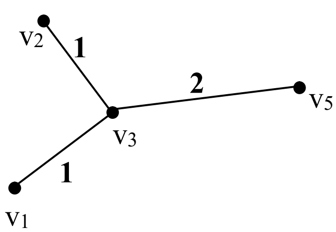
Step 2: Karena v_1 terkait pada sisi v_1v_2 dan v_3 terkait pada sisi v_3v_2 , v_3v_5 , v_3v_4 dengan v_2,v_4,v_5 tidak di T_2 sedangkan $W(v_1v_2)=2$, $W(v_3v_2)=1$, $W(v_3v_5)=2$, $W(v_3v_4)=3$ maka pilih v_3v_2 karena bobot minimum.

$T_3 = T_2 \cup v_3v_2$



Step 2: Karena v_2 terkait pada sisi v_2v_6 dan v_3 terkait pada sisi v_3v_5 , v_3v_4 dengan v_4,v_5 ,v_6 tidak di T_3 sedangkan $W(v_2v_6)=4$, $W(v_3v_5)=2$, $W(v_3v_4)=3$ maka pilih v_3v_5 karena bobot minimum.

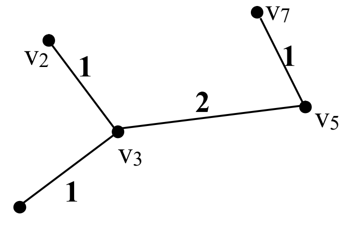
$T_4 = T_3 \cup v_3v_5$



YEMI KUSWARDI

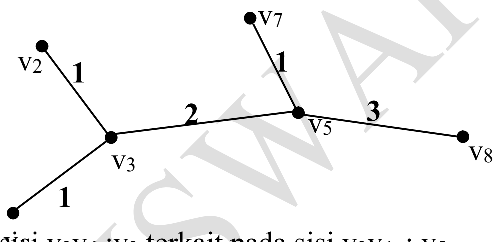
Step 2: Karena v_2 terkait pada sisi v_2v_6 dan v_3 terkait pada sisi v_3v_4 serta v_5 terkait pada v_5v_6, v_5v_7,v_5v_8 dengan v_4,v_6 ,v_7 ,v_8 tidak di T_4 sedangkan $W(v_2v_6)=4$, $W(v_3v_4)=3$, $W(v_5v_6)=5$, $W(v_5v_7)=1$, $W(v_5v_8)=3$ maka pilih v_5v_7 karena bobot minimum.

$T_5 = T_4 \cup v_5v_7$



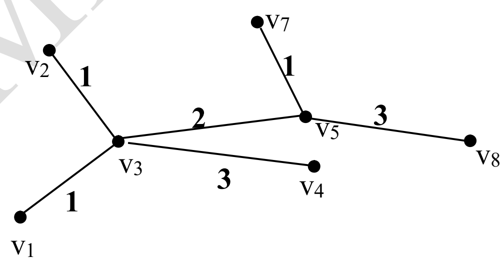
Step 2: Karena v_2 terkait pada sisi v_2v_6 ;v_3 terkait pada sisi v_3v_4 ; v_5 terkait pada v_5v_6, v_5v_8 dan v_7 terkait pada v_7v_6,v_7v_8 dengan v_4, v_6,v_8 tidak di T_5 sedangkan $W(v_2v_6)=4$, $W(v_3v_4)=3$, $W(v_5v_6)=5$, $W(v_5v_8)=3$ $W(v_7v_6)=3$, $W(v_7v_8)=3$, maka pilih v_3v_4 , v_5v_8 , v_7v_6, v_7v_8 karena bobot minimum.

$T_6 = T_5 \cup v_5v_8$



Step 2: Karena v_2 terkait pada sisi v_2v_6 ;v_3 terkait pada sisi v_3v_4 ; v_5 terkait pada v_5v_6 dan v_8 terkait pada v_8v_4 dengan v_4,v_6 tidak di T_6 sedangkan $W(v_2v_6)=4$, $W(v_3v_4)=3$, $W(v_5v_6)=5$, $W(v_7v_6)=3$ $W(v_4v_8)=8$, maka pilih v_3v_4 , v_7v_6 karena bobot minimum.

$T_7 = T_6 \cup v_3v_4$



Karena sebanyak $8 – 1 = 7$ sisi telah terpilih maka STOP T_7 tersebut merupakan pohon renyang minimum di G dengan bobot sebesar $1 + 1 + 2 + 3 + 1 + 3 = 11$

YEMI KUSWARDI

DAFTAR PUSTAKA

Adiwijaya. 2016. *Matematika Diskrit dan Aplikasinya*. Bandung: Penerbit Alfabet

Clark and Holton. 1991. *Graph Theory*. New Zealand: World Scientific

https://archive.org/details/AppliedCombinatorics6thEditionByAlanTucker2012PDF/page/n335

I Ketut Budayasa. 2001. *Matematika Diskrit*. Surabaya: Unesa University Press

Jok Jek Siang. 2009. *Matematika Diskrit dan Aplikasinya pada Komputer*. Yogyakarta: Andi Offset

Rinaldi Munir. 2005. *Matematika Diskrit*. Edisi ketiga. Bandung: Informatika

Townsend. 1987. *Discrete Mathematics: Applied Combinatorics and Graph Theory*. California:The Benjamin/Cummings Publishing Company. Inc

Wono Setya Budi. 2014. *Kombinatorik*. Jakarta: Penerbit Erlangga.

YEMI KUSWARDI

YEMI KUSWARDI