BAB II

MATERI PENUNJANG

2.1 HIMPUNAN

2.1.1 PENGERTIAN

Himpunan adalah sekelompok obyek-obyek yang berada dalam satu kesatuan atau batasan, dan mempunyai sifat keterikatan diantara anggota-anggotanya.

Jika A adalah suatu himpunan dan x anggota dari himpunan A, maka dapat ditulis :

$$x \in A$$

Jika y bukan anggota himpunan A, dapat ditulis :

$$y \notin A$$

Definisi 2.1.1

Suatu himpunan dikatakan himpunan kosong jika himpunan tersebut tidak mempunyai anggota, dinotasikan $\emptyset$.

2.1.2 HUBUNGAN ANTAR HIMPUNAN

Definisi 2.1.2

Dua himpunan A dan B dikatakan sama ($A = B$) jika dan hanya jika untuk setiap anggota dari A adalah anggota B, dan untuk setiap anggota dari B adalah anggota A, sehingga dapat ditulis :

$$A = B \text{ j.h.j untuk setiap } x,\ x \in A \text{ j.h.j } x \in B$$

Definisi 2.1.3

Jika terdapat dua himpunan A dan B, di katakan A adalah himpunan bagian (Subset) dari B jika dan hanya jika untuk setiap anggota A adalah anggota B, tetapi untuk setiap anggota B belum tentu anggota A , ditulis $A \subset B$.

2.1.3 OPERASI ANTAR HIMPUNAN

Definisi 2.1.4

Irisan (Interseksi) dari dua himpunan A dan B, dengan tanda $A \cap B$ adalah himpunan yang anggota-anggotanya dimiliki oleh A dan juga dimiliki oleh B secara bersamaan.

$$A \cap B = \{ x \mid x \in A \text{ dan } x \in B \}$$

Definisi 2.1.5

Gabungan (union) dari dua himpunan A dan B, dengan tanda $A \cup B$ adalah himpunan yang anggota-anggotanya adalah anggota yang berada di A atau berada di B atau berada di kedua-duanya.

$$A \cup B = \{ x \mid x \in A \text{ atau } x \in B \}$$

Definisi 2.1.6

Selisih dari dua himpunan A dan B, dengan tanda A - B adalah himpunan yang anggota-anggotanya termasuk di A tetapi tidak termasuk di B.

$$A - B = \{ x \mid x \in A \text{ dan } x \notin B \}$$

2.2 GRAPH TAK BERARAH

2.2.1 PENGERTIAN

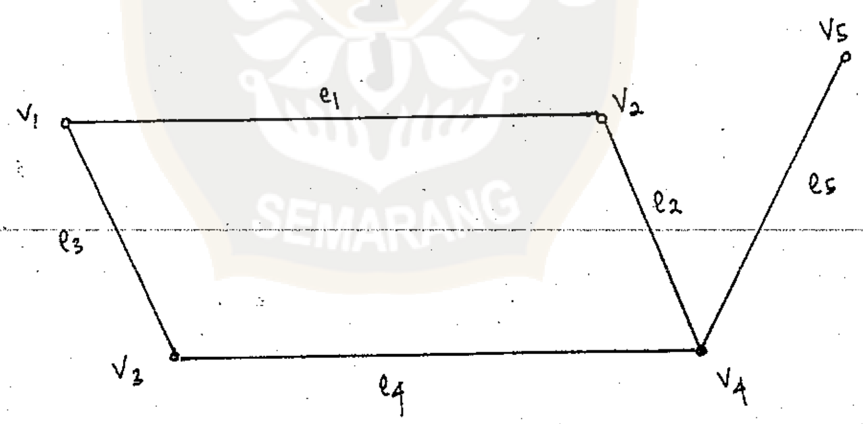
Definisi 2.2.1

Suatu graph G dinotasikan G=(V,E) adalah himpunan vertex (V) dimana $V=\{v_1, v_2, v_3, \ldots, v_n\}$ yang berhingga dan tidak kosong, dan himpunan edge (E) dimana $E = \{ e_1, e_2, \ldots, e_m\}$ berhingga.

Definisi 2.2.2

Graph tak berarah adalah graph yang semua edgenya tidak mempunyai arah dan biasanya disebut dengan graph pada definisi 2.2.1

Contoh :



gambar 2.2.1

Gambar 2.2.1 adalah suatu graph dengan :

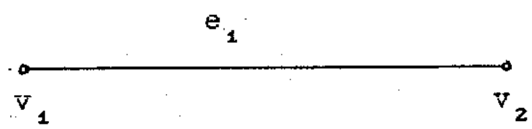
$V = \{ v_1 , v_2 , v_3 , v_4 , v_5 \}$

$E = \{ e_1 , e_2 , e_3 , e_4 , e_5 \}$

Definisi 2.2.3

Vertex v_i dan v_j disebut endvertex dari e_r jika e_r menghubungkan vertex v_i dan v_j.

Contoh :



Gambar 2.2.2

Pada gambar 2.2.2, vertex v_1 dan v_2 endvertex dari e_1.

Definisi 2.2.4

Suatu edge e_i dikatakan incident dengan vertex v_j, jika v_j endvertex dari e_i.

Contoh :

Pada gambar 2.2.1, edge e_1 dan e_2 incident dengan vertex v_2.

Definisi 2.2.5

Dua vertex dikatakan adjacent jika dihubungkan oleh sebuah edge.

Contoh :

Pada gambar 2.2.1, vertex v_1 adjacent dengan v_2.

Definisi 2.2.6

Dua edge dikatakan adjacent jika mereka incident pada vertex yang sama.

Contoh :

Pada gambar 2.2.1, e_1 dan e_2 adjacent.

Definisi 2.2.7

Derajat (degree) dari vertex v_i dinotasikan $d(v_i)$ adalah banyaknya edge yang incident dengan vertex v_i.

Contoh :

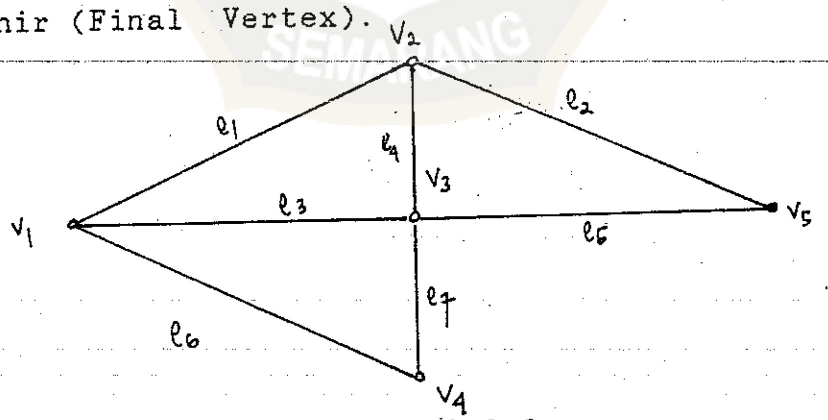
Pada gambar 2.2.1,

$d(v_1) = d(v_2) = d(v_3) = 2$

$d(v_4) = 3$

$d(v_5) = 1$

Diberikan suatu graph G seperti pada gambar 2.2.3, dimana vertex-vertexnya menunjukkan kota dan edge-edgenya menunjukkan jalan. Dari kota v_1 ke kota v_5 terdapat barisan edge-edge yang merupakan rute dari v_1 ke v_5. Suatu vertex berkorespondensi ke asal dinamakan vertex awal (Initial Vertex) dan vertex yang berkorespodensi ke tujuan dinamakan vertex akhir (Final Vertex).



Gambar 2.2.3

Contoh :

Pada gambar 2.2.3, dengan Intial vertex v_1 dan final vertex v_5 terdapat beberapa barisan edge misalnya (e_3, e_5), (e_3, e_4, e_2), $(e_1, e_4, e_3, e_6, e_7, e_5)$.

Yang harus dicatat bahwa setiap edge dalam suatu barisan yang dibicarakan selain edge pertama dan terakhir mempunyai satu endvertex yang bersamaan dengan edge sebelum nya dan endvertex lain dengan edge sesudahnya.

Contoh :

Dalam barisan (e_3, e_4, e_2) pada contoh diatas, vertex v_3 merupakan endvertex dari e_4 juga merupakan endvertex dari edge sebelumnya yaitu e_3, dan vertex v_2 endvertex dari e_4 juga merupakan endvertex dari edge sesudahnya yaitu e_2.

Jika setiap edge yang demikian dalam suatu barisan muncul hanya sekali, barisan ini dinamakan suatu train edge. Sehingga didapat definisi train edge sebagai berikut :

Definisi 2.2.8

Train Edge adalah barisan edge-edge dengan sifat sebagai berikut :

(i). Untuk edge e_u selain pertama dan terakhir edge-edge dalam barisan, satu endvertex dari e_u adalah endvertex dari edge sebelumnya dan endvertex lain dari e_u adalah endvertex dari edge sesudahnya.

(ii). Satu endvertex dari edge pertama adalah endvertex dari edge sesudahnya dan endvertex lain dari edge pertama adalah initial vertex.

(iii). Satu endvertex dari edge terakhir adalah endvertex dari edge sebelumnya dan endvertex lain dari edge terakhir adalah final vertex.

(iv). Setiap edge muncul tepat satu kali.

Definisi 2.2.9

Suatu train edge disebut train edge open jika initial vertex dan final vertexnya berbeda, sebaliknya jika initial vertex dan final vertexnya sama disebut train edge closed.

Contoh :

Pada gambar 2.2.3, barisan $(e_3, e_4, e_2, e_5, e_7)$ adalah train edge open yang initial vertexnya v_1 dan final vertexnya v_4. Barisan (e_1, e_4, e_3) adalah train edge closed.

Jika train edge open mempunyai sifat bahwa derajat setiap vertex kecuali initial vertex dan final vertex adalah 2 dan derajat dari initial vertex dan vinal vertex adalah 1, maka himpunan semua edge dalam train edge disebut path antara initial vertex dan final vertex.

Definisi 2.2.10

Path antara vertex i dan j adalah himpunan semua edge dalam train edge open yang memenuhi sifat sebagai berikut:

(i). Initial vertex dan final vertex adalah i dan j

(ii). Setiap vertex selain i dan j derajatnya 2 dan vertex i dan j derajatnya 1.

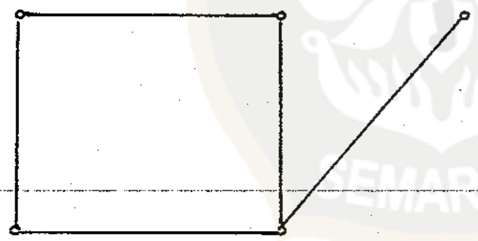
Contoh :

Himpunan semua edge dalam train edge (e_1, e_4, e_5) pada gambar 2.2.3 merupakan path antara vertex v_1 dan v_5.

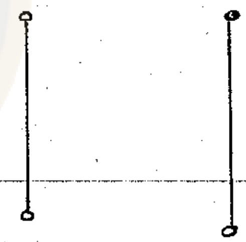
Definisi 2.2.11

Graph terhubung (Connected Graph) adalah jika setiap pasang vertex-vertexnya dihubungkan oleh path. Dan sebaliknya disebut graph tak terhubung (Disconnected Graph).

Contoh :



Gambar 2.2.4a



gambar 2.2.4b

Gambar 2.2.4a merupakan graph terhubung sedangkan gambar 2.2.4b merupakan graph tak terhubung.

Definisi 2.2.12

Sirkuit adalah graph terhubung yang setiap vertexnya berderajat 2.

Dari definisi ini, dapat dilihat bahwa suatu himpunan semua edge dalam train edge closed akan menjadi sirkuit jika derajat setiap vertexnya 2.

Contoh :

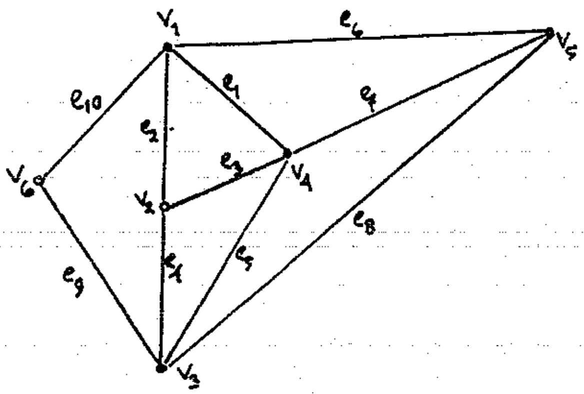
Himpunan edge dalam train edge closed (e_1, e_3, e_4) pada gambar 2.2.3 merupakan sirkuit tetapi himpunan edge dalam train edge closed $(e_6, e_7, e_5, e_2, e_4, e_3)$ pada gambar yang sama bukan merupakan sirkuit.

Definisi 2.2.13

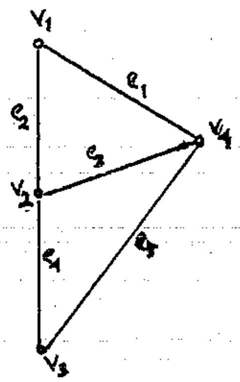
Suatu graph g dikatakan subgraph dari graph G jika seluruh vertex dan edgenya berada dalam G.

Konsep dari subgraph sama dengan konsep subset pada teori himpunan. Sebuah subgraph bisa dikatakan sebagai graph dimuat (bagian dari) graph yang lain. Simbol teori himpunan $g \subset G$, digunakan dengan kalimat " g adalah sebuah subgraph dari G ".

Contoh :



Gambar 2.2.5a



Gambar 2.2.5b

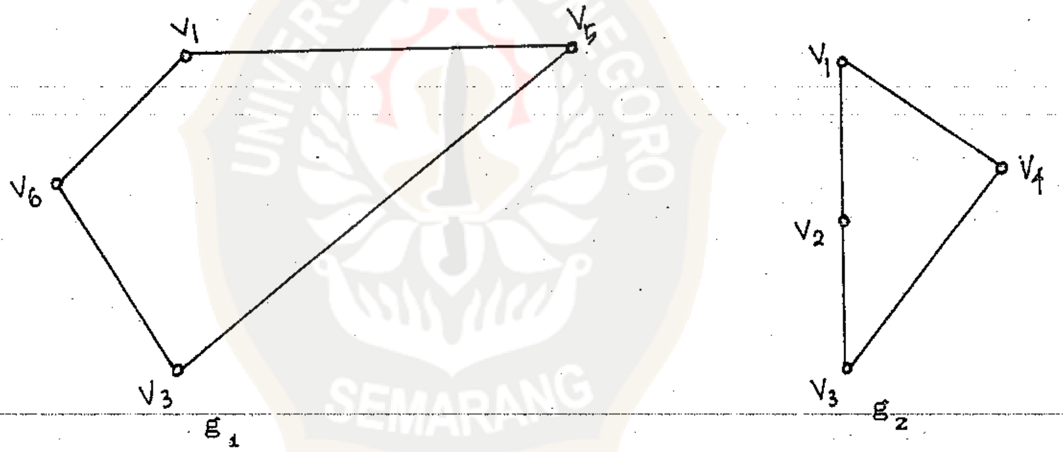
Gambar 2.2.5b merupakan salah satu subgraph dari graph pada gambar 2.2.5a.

Definisi 2.2.14

Dua subgraph g_1 dan g_2 dari graph G disebut edge terpisah (Edge Disjoint) jika g_1 dan g_2 tidak mempunyai edge bersama-sama.

Contoh :

g_2 dan g_2 pada gambar 2.2.6 merupakan edge disjoint dari graph G pada gambar 2.2.5a

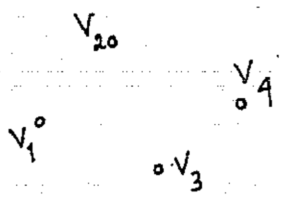


Gambar 2.2.6

Definisi 2.2.15

Suatu graph dikatakan null graph jika edge E kosong atau graph tersebut tidak mempunyai edge.

Contoh :



Gambar 2.2.7

Gambar 2.2.7 merupakan suatu null graph dengan $V = \{v_1, v_2, v_3, v_4\}$

2.2.2 OPERASI DALAM GRAPH

2.2.2.1 DEFINISI-DEFINISI

Definisi 2.2.16

Union dari dua graph $G_1 = (V_1, E_1)$ dan $G_2 = (V_2, E_2)$ adalah graph lain G_3 yang ditulis $G_3 = G_1 \cup G_2$, dimana himpunan vertexnya $V_3 = V_1 \cup V_2$ dan himpunan edgenya $E_3 = E_1 \cup E_2$.

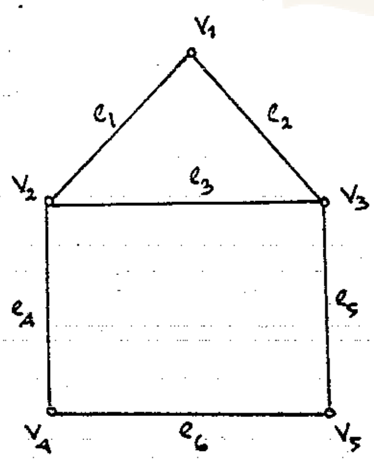
Definisi 2.2.17

Irisan (Intersection) dari dua graph G_1 dan G_2 adalah graph lain G_4 yang ditulis $G_4 = G_1 \cap G_2$, dimana himpunan vertexnya $V_4 = V_1 \cap V_2$ dan himpunan edgenya $E_4 = E_1 \cap E_2$.

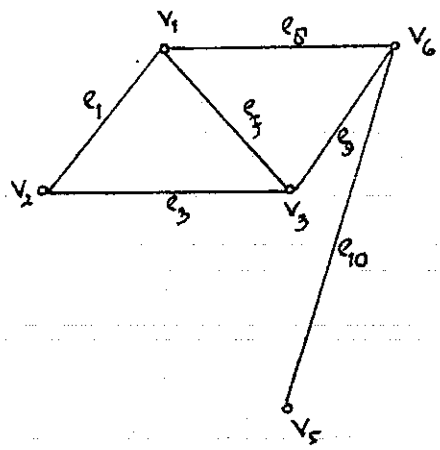
Definisi 2.2.18

Ring Sum dari dua graph G_1 dan G_2 adalah graph lain G_5 yang ditulis $G_5 = G_1 \oplus G_2$, dimana himpunan verteknya $V_5 = V_1 \cup V_2$ dan himpunan edgenya $E_5 = (E_1 \cup E_2) - (E_1 \cap E_2)$.

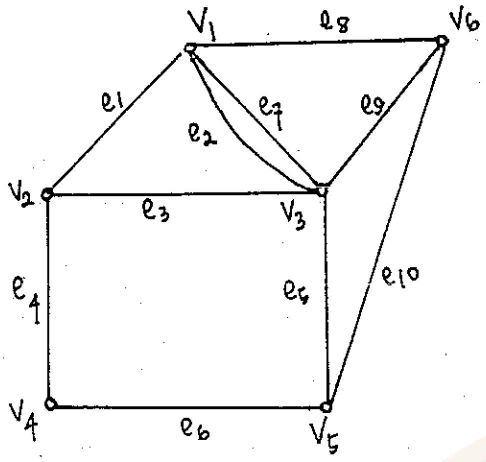
Contoh :



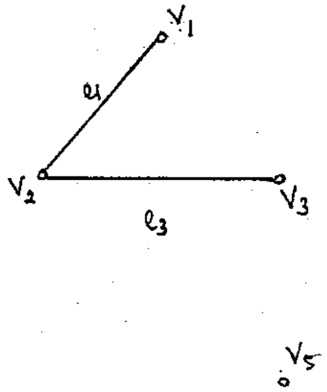
G_1



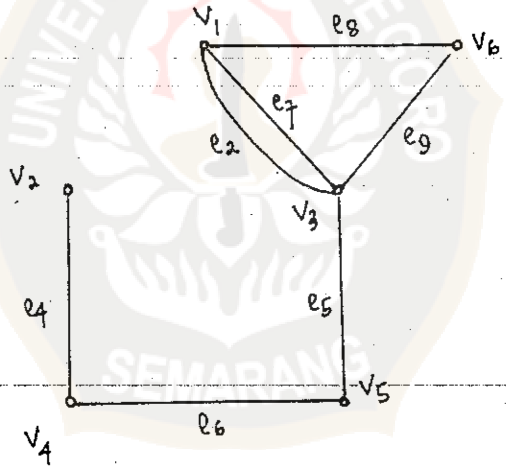
G_2



$G_3 = G_1 \cup G_2$



$G_4 = G_1 \cap G_2$



$G_5 = G_1 \oplus G_2$

Gambar 2.2.8

2.2.2.2 SIFAT-SIFAT OPERASI DALAM GRAPH

1. $G_1 \cup G_2 = G_2 \cup G_1$
2. $G_1 \cap G_2 = G_2 \cap G_1$
3. $G_1 \oplus G_2 = G_2 \oplus G_1$
4. $G \cup G = G \cap G = G$
5. $G \oplus G$ = Suatu null graph.

2.3 HIMPUNAN POTONG

Untuk menentukan himpunan potong dari suatu graph harus diketahui lebih dulu pohon bentangan (Spanning Tree)nya, kemudian ditentukan himpunan potong dasar (fundamental)nya sebelum himpunan potong keseluruhan dari suatu graph tersebut.

2.3.1 POHON BENTANGAN

Definisi 2.3.1

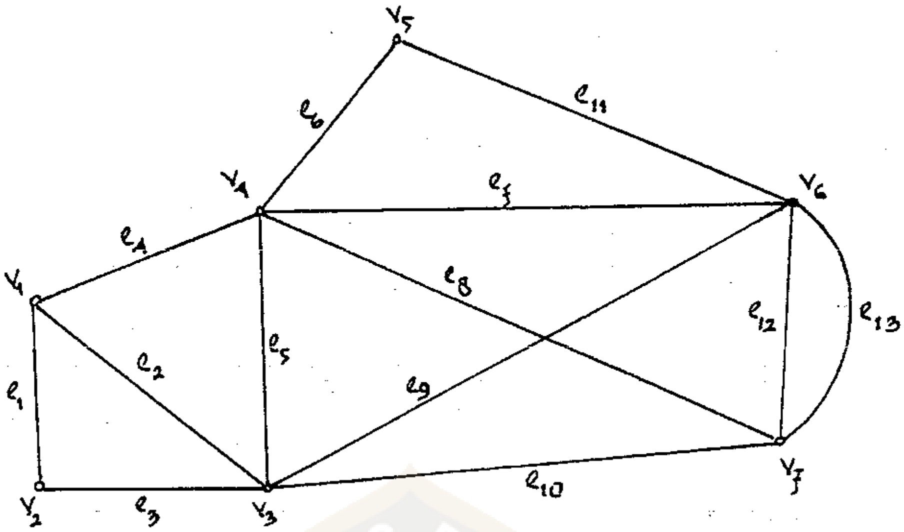
Suatu pohon (Tree) T adalah graph terhubung tanpa sirkuit.

Definisi 2.3.2

Suatu pohon T dikatakan sebagai pohon bentangan (Spanning Tree) dari graph terhubung G jika T adalah subgraph dari G dan T memuat seluruh vertex dari G.

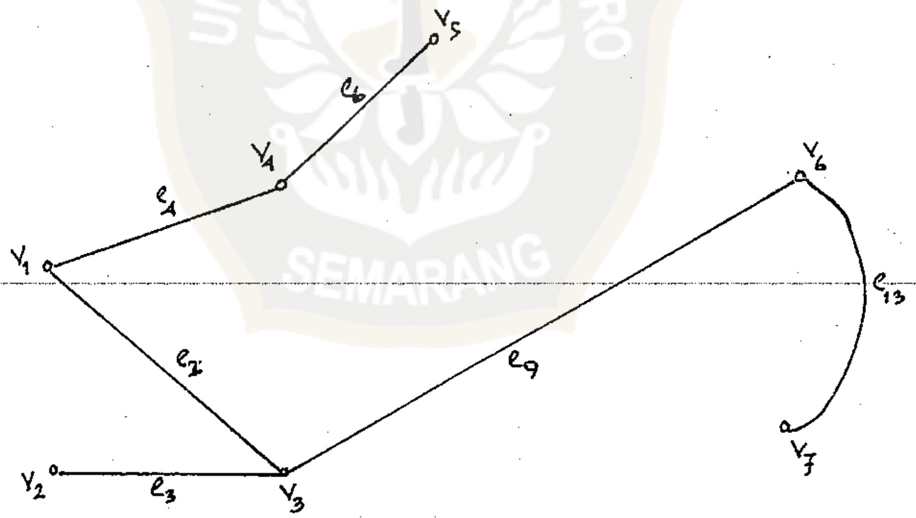
Untuk menentukan pohon bentangan T dari suatu graph terhubung G sangat sederhana. Jika graph G tidak mempunyai sirkuit maka pohon bentangan adalah dirinya sendiri. Dan jika graph G mempunyai sirkuit, menghapus sebarang edge dari sirkuit tersebut dan penghapusan ini harus tetap meninggalkan graph G terhubung. Jika masih ada sirkuit diulangi lagi penghapusan seperti semula, penghapusan diulang-ulang sampai edge pada sirkuit terakhir terhapus yang tetap meninggalkan graph terhubung, yang tidak mempunyai sirkuit dan yang memuat seluruh vertex dari G.

Contoh :



Gambar 2.3.1

Dari suatu graph terhubung pada gambar 2.3.1 dapat dutentukan pohon bentangan seperti pada gambar 2.3.2 di bawah ini



Gambar 2.3.2

Definisi 2.3.3

Cabang (Branch) dari T adalah edge dalam pohon bentangan T, Chord dari T adalah edge diluar T.

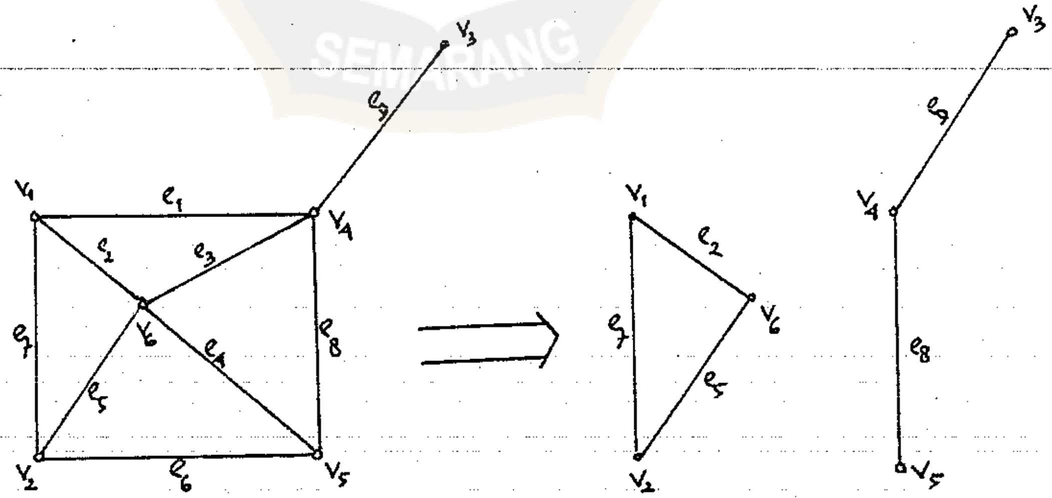
2.3.2 HIMPUNAN POTONG DAN HIMPUNAN POTONG FUNDAMENTAL

2.3.2.1 PENGERTIAN HIMPUNAN POTONG

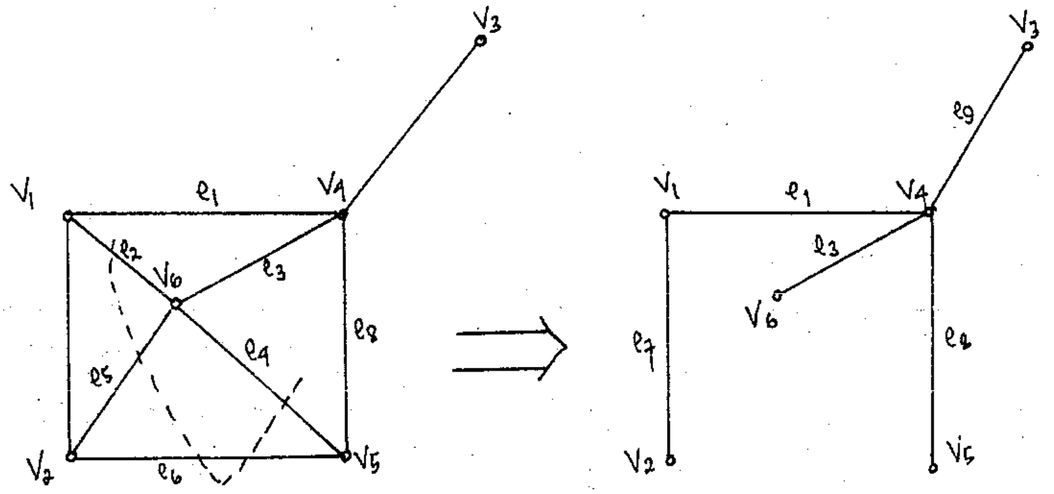
Pada suatu graph terhubung G, himpunan potong S adalah himpunan yang anggotanya edge-edge dalam G, yang apabila edge-edge tersebut dihilangkan dari G mengakibatkan graph G tak terhubung (Disconected).

Contoh :

Himpunan edge $\{e_1, e_3, e_4, e_6\}$ dalam graph terhubung G pada gambar 2.3.3a adalah himpunan potong sebab jika edge-edge tersebut dihilangkan dari G maka G akan menjadi graph tak terhubung. Sedangkan himpunan edge $\{e_2, e_4, e_5, e_6\}$ pada gambar 2.3.3b bukan merupakan himpunan potong sebab jika edge-edge tersebut dihilangkan dari G, G tetap terhubung.



a. $\{e_1, e_3, e_4, e_6\}$ adalah himpunan potong



b. $\{e_2, e_4, e_5, e_6\}$ bukan himpunan potong

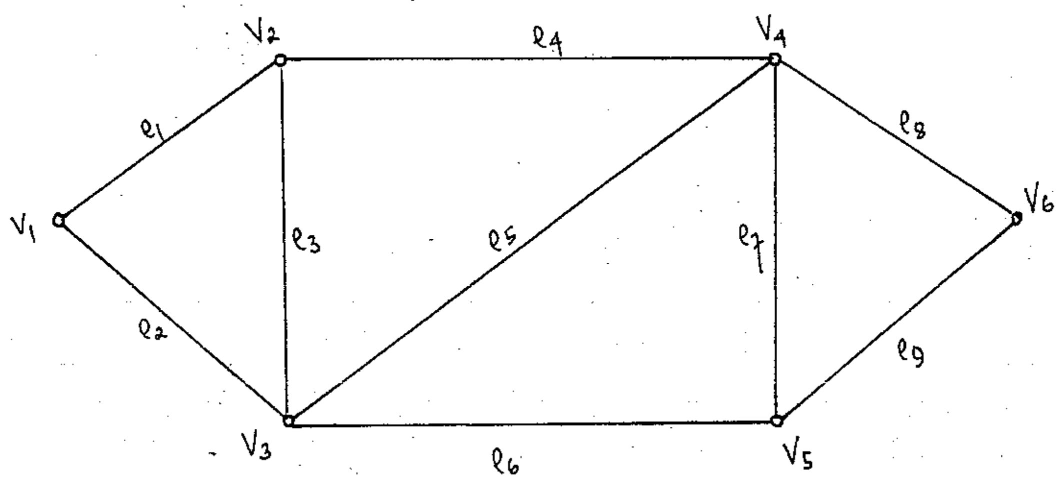
Gambar 2.3.3

2.3.2.2 PENGERTIAN HIMPUNAN POTONG FUNDAMENTAL

Sekarang akan dibicarakan pengertian himpunan potong fundamental yang akan digunakan untuk menentukan semua himpunan potong dalam suatu graph terhubung G. Dipandang suatu pohon bentangan T dari G. Dihapus sebarang cabang b dari T, {b} suatu himpunan potong dalam T maka {b} memotong atau memisah T menjadi dua pohon (salah satu atau dua-duanya mungkin memuat vertex tunggal yang disebut pohon vertex). Dipandang suatu pemisahan yang sama dari vertex dalam G dan himpunan potong S dalam G. Himpunan potong S akan memuat hanya satu cabang dari T dan edge yang lain merupakan chord dari T. Sehingga S memuat tepat satu cabang dari T dan disebut himpunan potong fundamental.

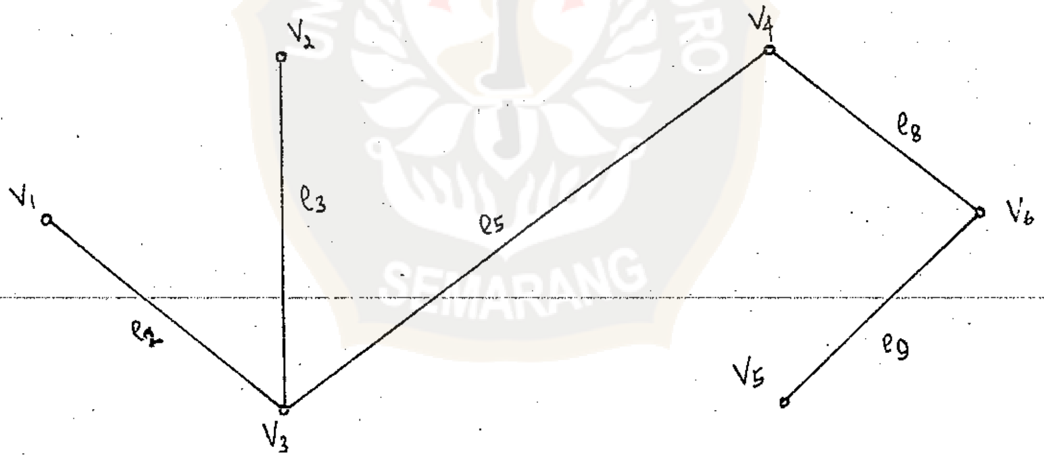
Contoh :

Suatu graph terhubung G pada gambar 2.3.4 akan ditentukan himpunan potong fundamentalnya.



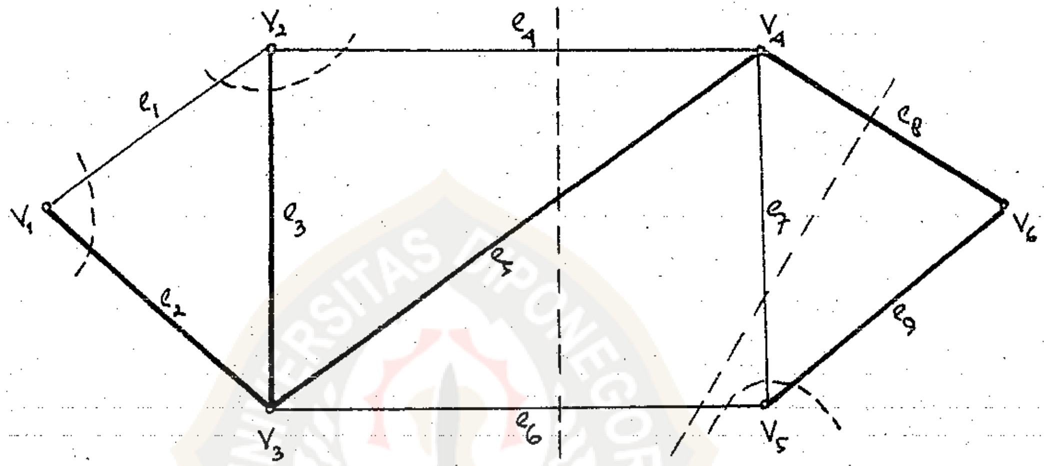
Gambar 2.3.4

Sebelumnya harus ditentukan dulu pohon bentangan T, misalnya seperti gambar 2.3.5



Gambar 2.3.5

Cabang dari pohon bentangan gambar 2.3.5 adalah e_2, e_3, e_5, e_8 dan e_9. Jika menghapus cabang e_2 maka akan didapatkan sebuah pohon vertex dan sebuah pohon yang tidak memuat edge e_2. Demikian juga jika dilakukan penghapusan cabang-cabang yang lain juga akan didapatkan dua pohon. Pemisahan ini kemudian dilihat pada G yang membentuk himpunan potong. Setiap himpunan potong memuat satu cabang, untuk lebih jelasnya dapat dilihat pada gambar 2.3.6 di bawah ini



Gambar 2.3.6

Pada gambar 2.3.6, suatu pohon bentangan ditunjukkan dengan garis tebal dan himpunan potong fundamentalnya ditunjukkan dengan garis putus-putus. Pada gambar tersebut terdapat lima himpunan potong fundamental, yaitu : $\{e_1, e_2\}$, $\{e_1, e_3, e_4\}$, $\{e_4, e_5, e_6\}$, $\{e_6, e_7, e_8\}$ $\{e_6, e_7, e_9\}$.

2.3.2.3 MENENTUKAN HIMPUNAN POTONG

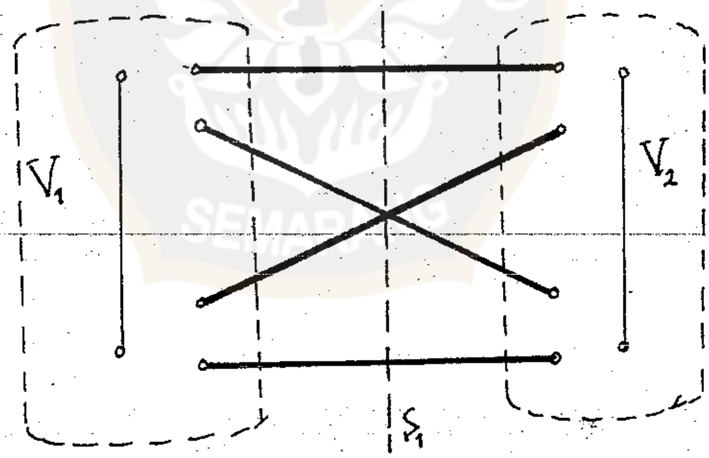
Setelah didapatkan himpunan potong fundamental dari graph terhubung G, maka dapat ditentukan semua himpunan potongnya. Sehubungan dengan hal itu diberikan theorema ring sum dari dua himpunan potong, sebagai berikut :

Theorema 2.3.1

Ring Sum dari dua himpunan potong dari suatu graph adalah himpunan potong yang lain atau suatu union edge terpisah dari dua himpunan potong.

Bukti :

Dimisalkan S_1 dan S_2 merupakan dua himpunan potong pada suatu graph terhubung G. Dimisalkan V_1 dan V_2 merupakan partisi (unik dan terpisah) dari himpunan vertex V dalam G berkorespondensi terhadap S_1. Dimisalkan V_3 dan V_4 merupakan partisi berkorespondensi terhadap S_2. Untuk jelasnya dapat dilihat pada gambar 2.3.7a dan 2.3.7b.

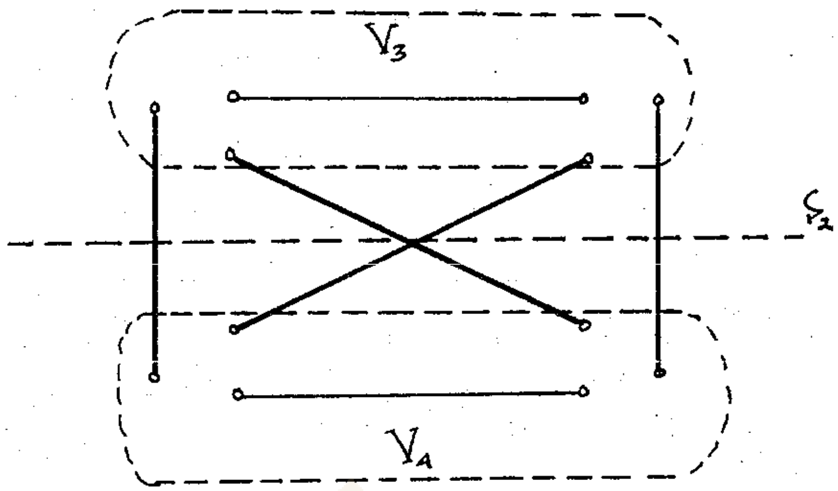


Gambar 2.3.7a

Dari gambar 2.3.7a terlihat bahwa

$V_1 \cup V_2 = V$

$V_1 \cap V_2 = \phi$



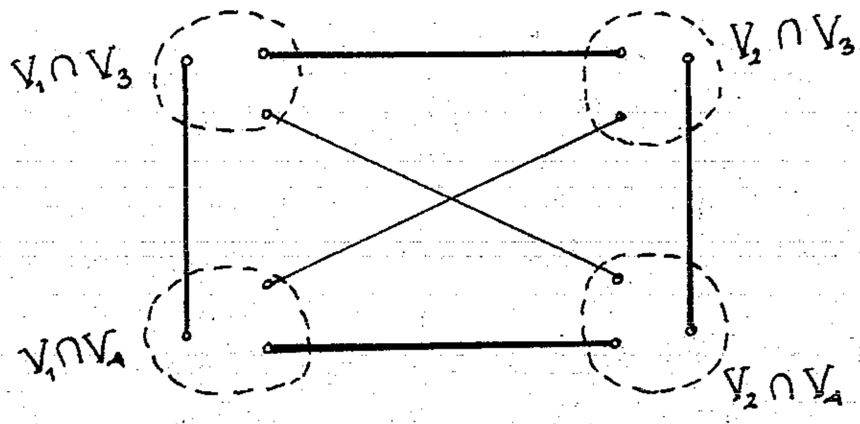
Gambar 2.3.7b

Dari gambar 2.3.7b terlihat bahwa

$V_3 \cup V_4 = V$

$V_3 \cap V_4 = \phi$

Sekarang dimisalkan himpunan bagian $(V_1 \cap V_4) \cup (V_2 \cap V_3)$ disebut V_5 dan didefinisikan sama dengan ring sum $V_1 \oplus V_3$. Demikian juga dimisalkan himpunan bagian $(V_1 \cap V_3) \cup (V_2 \cap V_4)$ disebut V_6 yang didefinisikan sama dengan ring sum $V_2 \oplus V_3$ dapat dilihat pada gambar 2.3.7c.



Gambar 2.3.7c

ring sum dari dua himpunan potong $S_1 \oplus S_2$ dapat dilihat hanya memuat edge-edge yang menghubungkan vertex-vertex di V_5 dengan vertex-vertex di V_6. Juga tidak ada edge-edge di luar $S_1 \oplus S_2$ yang menghubungkan vertex-vertex di V_5 dengan vertex-vertex di V_6. Jadi himpunan edge $S_1 \oplus S_2$ menghasilkan partisi dari V menjadi V_5 dan V_6 sedemikian sehingga :

$$V_5 \cup V_6 = V$$

$$V_5 \cap V_6 = \phi$$

Sehingga $S_1 \oplus S_2$ adalah himpunan potong jika subgraph-subgraph yang memuat V_5 dan V_6 masing-masing merupakan subgraph terhubung sesudah $S_1 \oplus S_2$ dihilangkan dari G. Sebaliknya kalau tidak demikian $S_1 \oplus S_2$ adalah union edge terpisah dari himpunan potong-himpunan potong.

Contoh :

Pada gambar 2.3.6 dipandang ring sum dari tiga pasang himpunan potong sebagai berikut

$\{e_4, e_5, e_6\} \oplus \{e_6, e_7, e_8\} = \{e_4, e_5, e_7, e_8\}$

$\{e_1, e_3, e_4\} \oplus \{e_4, e_5, e_6\} = \{e_1, e_3, e_5, e_6\}$

$\{e_4, e_5, e_7, e_8\} \oplus \{e_6, e_7, e_9\} = \{e_4, e_5, e_6, e_8, e_9\}$

$= \{e_4, e_5, e_6\} \cup \{e_8, e_9\}$

$\{e_4, e_5, e_7, e_8\}$ dan $\{e_1, e_3, e_5, e_6\}$ merupakan himpunan potong yang lain sedangkan $\{e_4, e_5, e_6, e_8, e_9\}$ merupakan union edge terpisah.

Untuk selanjutnya dengan menggunakan theorema 2.3.1 dapat ditentukan semua himpunan potong dari suatu graph terhubung G. Untuk contohnya, suatu graph terhubung G pada gambar 2.3.4 didapatkan himpunan potong-himpunan potongnya, yaitu :himpunan potong-himpunan potong fundamental dan hasil dari ring sum tiap pasang himpunan potong. Ring Sum yang lain untuk mendapatkan semua himpunan potong dari graph G pada gambar 2.3.4 adalah sebagai berikut:

$\{e_1, e_2\} \oplus \{e_1, e_3, e_4\} = \{e_2, e_3, e_4\}$

$\{e_4, e_5, e_6\} \oplus \{e_6, e_7, e_9\} = \{e_4, e_5, e_7, e_9\}$

$\{e_1, e_2\} \oplus \{e_1, e_3, e_5, e_6\} = \{e_2, e_3, e_5, e_6\}$

$\{e_1, e_3, e_4\} \oplus \{e_4, e_5, e_7, e_8\} = \{e_1, e_3, e_5, e_7, e_8\}$

$\{e_1, e_3, e_4\} \oplus \{e_4, e_5, e_7, e_9\} = \{e_1, e_3, e_5, e_7, e_9\}$

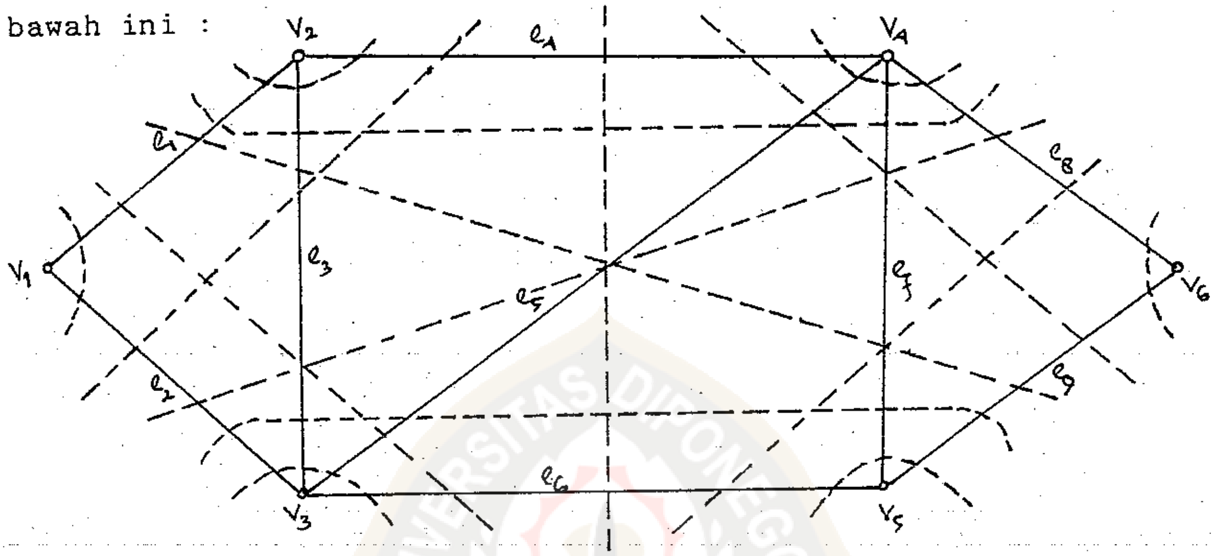
$\{e_2, e_3, e_4\} \oplus \{e_4, e_5, e_7, e_8\} = \{e_2, e_3, e_5, e_7, e_8\}$

$\{e_2, e_3, e_4\} \oplus \{e_4, e_5, e_7, e_9\} = \{e_2, e_3, e_5, e_7, e_9\}$

Jadi himpunan potong seluruhnya dari graph terhubung G pada gambar 2.3.4 di atas adalah :

$\{e_1, e_2\}, \{e_1, e_3, e_4\}, \{e_1, e_3, e_5, e_6\}, \{e_1, e_3, e_5, e_7, e_8\}$,
$\{e_1, e_3, e_5, e_7, e_9\}, \{e_2, e_3, e_4\}, \{e_2, e_3, e_5, e_6\}$,
$\{e_2, e_3, e_5, e_7, e_8\}, \{e_2, e_3, e_5, e_7, e_9\}, \{e_4, e_5, e_6\}$,
$\{e_4, e_5, e_7, e_8\}, \{e_4, e_5, e_7, e_9\}, \{e_6, e_7, e_8\}, \{e_6, e_7, e_9\}$, dan
$\{e_8, e_9\}$.

Himpunan potong-himpunan potong tersebut dapat dilihat pada gambar 2.3.8 yang ditunjukkan dengan garis putus-putus di bawah ini :



Gambar 2.3.8

Dengan demikian ditarik kesimpulan bahwa cara menentukan himpunan potang yang mempunyai n (finite) vertex adalah sebagai berikut :

1. Dari suatu graph G dengan n (finite) vertex yang diberikan, dibuat suatu pohon bentangan.
2. Kemudian dengan pohon bentangan tersebut ditentukan himpunan potong fundamentalnya.
3. Dari himpunan potong fundamental yang telah diperoleh dapat ditentukan semua himpunan potong dengan menggunakan operasi ring sum.