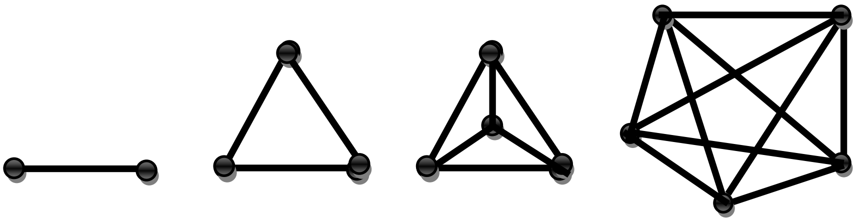
II. TINJAUAN PUSTAKA

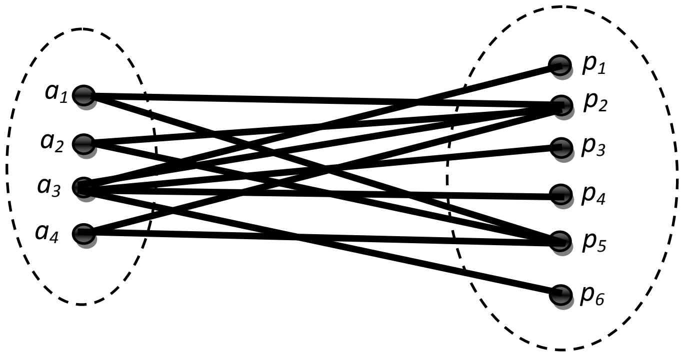
Dalam bab ini akan dijelaskan beberapa pengertian tentang graf, isomorfis graf, *Cyclic-Cubes*, *Wrapped Butterfly Networks* (WB) (*n*,*k*) dan beberapa istilah yang berkaitan dengan bahasan dalam penelitian ini. Hal mendasar yang harus diketahui dan difahami untuk menunjang penelitian ini adalah pengertian graf. Dalam suatu referensi dijelaskan bahwa graf didefinisikan sebagai berikut:

Definisi 2.1 Graf $G = (V,E)$ terdiri dari objek $V = \{v_1, v_2 \ldots.\}$ yang disebut *vertex* (titik) yang tidak kosong, dan objek $E = \{e_1, e_2 \ldots..\}$ yang unsur-unsurnya disebut *edge* (garis) yang boleh kosong, sehingga setiap *edge* e_{ij} diidentifikasi dengan pasangan (v_i, v_j) dari *vertex*. *Vertex* v_i, v_j berhubungan dengan edge e_{ij} disebut *vertex* akhir dari e_{ij}. Representasi paling umum dari graf adalah dengan cara diagram, dimana *vertex* direpresentasikan sebagai titik dan setiap *edge* sebagai garis yang menghubungkan *vertex* (Deo,1989).

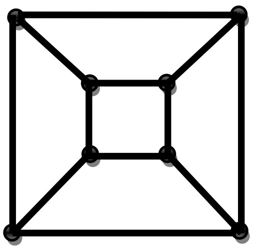
Graf mempunyai banyak jenis dan bentuk yang berbeda-beda seperti graf Lengkap, graf Bipartite, graf Kubus, graf Euler, dan masih banyak lagi jenis graf yang lain dengan bentuk sebagai berikut:



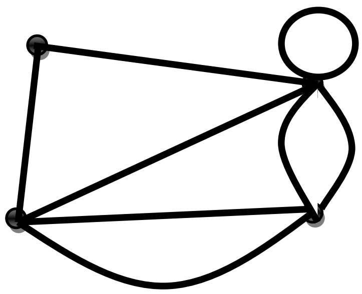
Gambar 1. Graf Lengkap



Gambar 2. Graf Bipartite

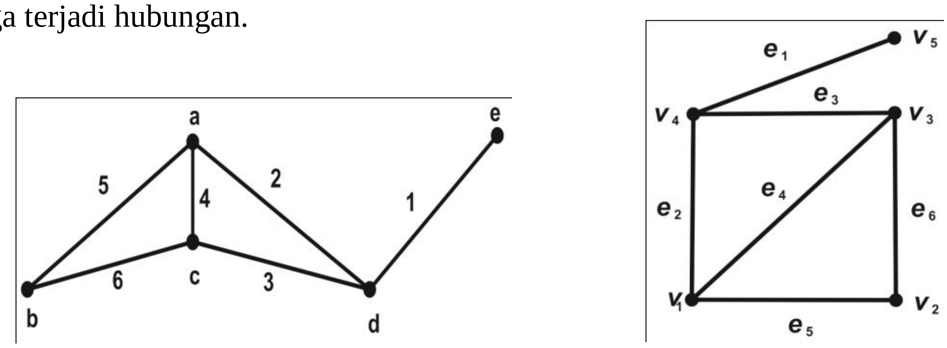


Gambar 3. Graf Kubus



Gambar 4. Graf Euler

Dari semua graf tersebut ada yang memiliki bentuk yang kongruen atau isomorfis antara satu dengan yang lainya. Dalam geometri, dua bangun atau bidang dikatakan ekuivalen (dan disebut dengan kongruen) jika keduanya memiliki ciri-ciri atau sifat yang sama. Begitupun, dua graf dikatakan ekuivalen (dan disebut isomorfis) jika keduanya memiliki ciri-ciri yang sama pada istilah dalam teori graf. Dua graf G dan G^* dikatakan isomorfis jika ada korespondensi 1-1 antara *vertex* dari kedua graf tersebut dan antara *edge* dari kedua graf tersebut sehingga terjadi hubungan.



(a) **(b)**

Gambar 5. Contoh dua graf yang isomorfis

Sebagai contoh, dua graf pada Gambar 1 adalah isomorfis. Korespondensi antara dua graf adalah: *vertexs a,b,c,d,* dan *e* berkoresponden dengan v_1, v_2, v_3, v_4, dan v_5. *Edges* 1,2,3,4,5, dan 6 berkoresponden dengan e_1, e_2, e_3, e_4, e_5, dan e_6.

Hal tersebut di atas dapat dibuktikan dengan definisi dari isomorfis dari dua graf harus memiliki:

1. Jumlah *vertex* yang sama.
2. Jumlah *edge* yang sama.
3. Mempertahankan *adjacency* dari setiap *vertex*.

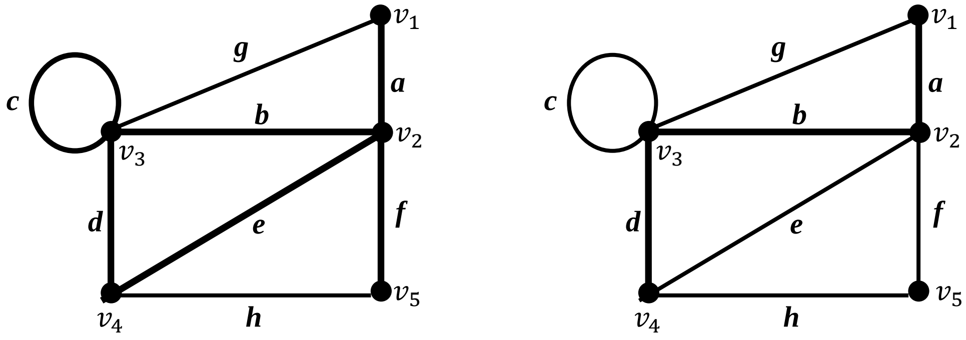
(Deo,1989).

Dalam buku yang lain disebutkan bahwa isomorfisme dari G ke H adalah fungsi bijeksi $f : V(G) \rightarrow V(H)$ dimana $(u, v) \epsilon E(G)$ jika dan hanya jika $f(u), f(v) \epsilon E(H)$. G isomorfis dengan H dilambangkan dengan $G \cong H$. Jika G isomorfis dengan H dan H isomorfis dengan G, maka G dan H dikatakan saling isomorfis (Hsu dan Lin, 2009).

Graf adalah kumpulan *vertex* yang dihubungkan dengan *edge* dan setiap *vertex* memiliki jumlah *edge* yang berbeda untuk graf yang berbeda pula. Misal G adalah graf tanpa *loop* dan *v* adalah *vertex* dari graf G. *Degree* dari *v* adalah jumlah *edge* yang bertemu (menempel) pada *v* dan dinotasikan dengan deg *v* (Wilson dan Watkins,1990).

Dalam teori graf ada istilah graf tertutup atau yang biasa disebut dengan *circuit.* Untuk memahami tentang *circuit* ada istilah *walk* dan *path* yang harus dipahami. *Walk* didefinisikan sebagai urutan berhingga dari *vertex* dan *edge*, dimulai dan diakhiri dengan *vertex*, sehingga setiap *edge* adalah terjadi dengan *vertex* sebelumnya dan mengikutinya. Tidak ada *edge* muncul lebih dari satu kali dalam *walk*. Hal yang mungkin terjadi dalam *walk* adalah dimulai dan diakhiri dengan *vertex* yang sama, *walk* yang demikian disebut dengan *walk* tertutup. Suatu *walk* yang tidak tertutup disebut dengan *walk* terbuka.

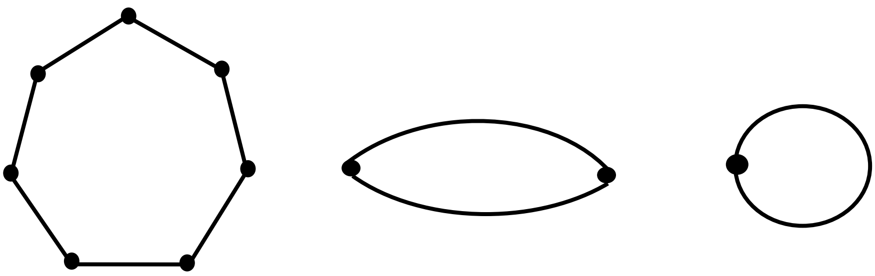
Path adalah *walk* terbuka yang tidak ada *vertex* yang dilalui lebih dari satu kali. *Circuit* adalah suatu *walk* tertutup yang tidak mempunyai pengulangan *vertex* kecuali *vertex* awal dan akhir (Deo, 1989).



(a) *Walk* Terbuka (b) *Path*

Gambar 6. Contoh *Walk* dan *Path*

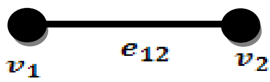
Berikut ini akan diberikan contoh graf yang merupakan *circuit.*



Gambar 7. Tiga *circuit* yang berbeda

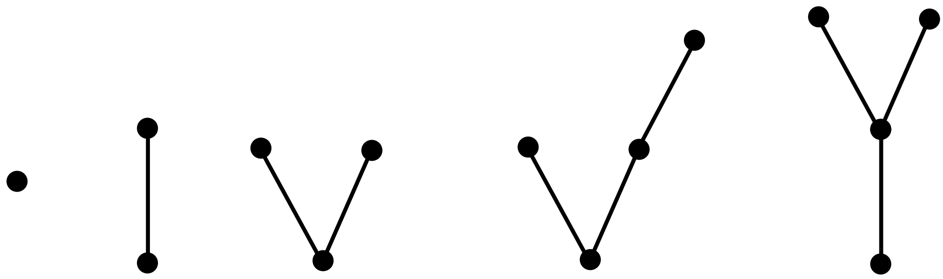
Dua *vertex* yang saling berhubungan disebut bertetangga atau *adjacent*.

Jika $e = \{u,v\}$ adalah suatu garis yang menghubungkan titik u dan v pada graf G, maka titik u disebut bertetangga (*adjacent*) dengan titik v dan garis e menempel (*incidence*) pada u dan v (Lipschutz and Lipson,2002).



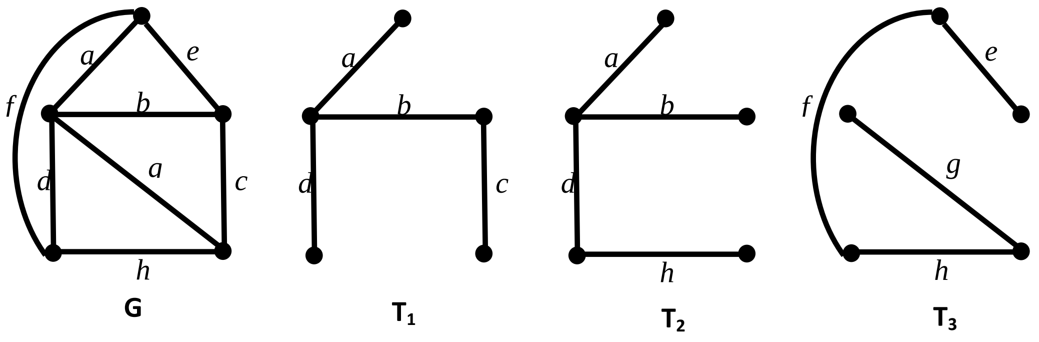
Gambar 8. Graf dengan garis e_{12} *incidence* pada titik v_1 dan v_2 , serta titik v_1 dan v_2 *adjacent*.

Graf memiliki keunikan dan keragaman bentuk dan jenisnya. Contohnya, dalam penjabaran sebelumnya telah dijelaskan graf *circuit*. Namun, ada juga graf yang tidak mengandung *circuit* yaitu pohon atau *tree*. Pohon (*tree*) adalah graf terhubung yang tidak mengandung *circuit*. Dilihat dari definisi pohon (*tree*) maka pohon (*tree*) tidak memiliki *loop* atau garis paralel (karena keduanya element *circuit*). *Tree* dengan satu *vertex* disebut *null tree* (*tree* tanpa simpul) (Deo, 1989).



Gambar 9. Contoh *tree* dengan *vertex* 1,2,3, dan 4

Suatu *tree* T disebut sebagai *spanning tree* dari graf terhubung G jika T adalah subgraf dari G dan T mengandung semua *vertex* dari graf G.



Gambar 10. Graf dan 3 *Spanning Tree.*

Setiap graf terhubung memiliki paling sedikit satu *spanning tree*. *Edge* pada suatu *spanning tree* T disebut dengan *branch*. *Edge* dari graf G yang tidak terdapat pada *spanning tree* disebut *chord*. Pada gambar 10, untuk graf G dan *spanning tree* T_1 maka *edge a,b,c* dan *d* adalah *branch* dan *edge e,f,g* dan *h* adalah *chord.* Untuk graf G dan *spanning tree* T_2, *edge a,b,d* dan *h* adalah *branch* dan *edge c,e,f,* dan *g* adalah *chord.* Untuk graf G dengan spanning tree T_3, *e,f,g* dan *h* adalah *branch* dan *edge a,b,c* dan *d* adalah *chord.* Jumlah *branch* adalah *rank* dan jumlah *chord* adalah *nullity* (Deo, 1989).

Selain dari istilah-istilah yang telah diuraikan sebelumnya, ada jenis graf yang unik dalam segi bentuk dan nama yang dimiliki seperti graf *Cyclic-Cubes*, *Wrapped Butterfly Networks* (WB) (*n*,*k*) seperti yang telah disebutkan di sebelumnya.

Definisi 2.2 Graf *Cyclic-Cubes* didefinisikan sebagai graf G_n^k yang memiliki nk^n *vertex* dan setiap *vertex* direpresentasikan dengan *n*-bit vektor, yang merupakan permutasi siklik dari $t_1^{i_1}t_2^{i_2}\ldots t_n^{i_n}$ untuk $1 \leq i_1, i_2, \ldots, i_n \leq k$. Dengan kata lain, *vertex* di G_n^k dapat dituliskan sebagai berikut:

$$V\,(G_n^k) = \left\{ t_j^{i_j} t_{j+1}^{i_{j+1}} \ldots t_n^{i_n} t_1^{i_1} \ldots t_{j-1}^{i_{j-1}} \middle| \, 1 \leq j \leq n \; dan \; 1 \leq i_1, i_2, ., i_n \leq k \right\}$$

Contoh pada graf G_3^2, $t_2^1t_1^2t_3^1$, $t_3^1t_2^1t_1^1$ bukan merupakan *vertex* dan $t_1^2t_2^1t_3^1$, $t_2^2t_3^1t_1^2$ merupakan *vertex*.

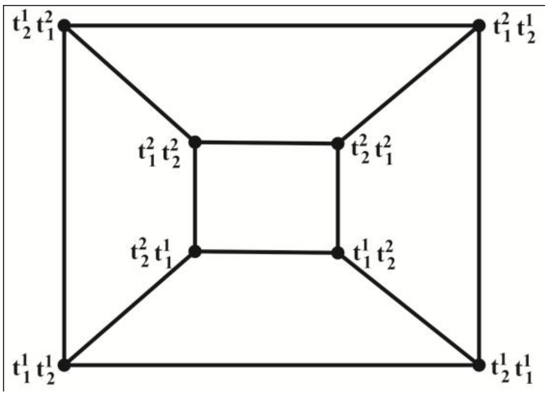
Untuk mendefinisikan *edge* pada graf G_n^k, pertama akan didefinisikan fungsi f_s untuk setiap $1 \le s \le k$, pemetaan $V(G_n^k)$ onto kepada dirinya sendiri, sesuai dengan definisi berikut ini:

$$f_s\left(t_j^{i_j} t_{j+1}^{i_{j+1}} \dots t_n^{i_n} t_1^{i_1} \dots t_{j-1}^{i_{j-1}}\right) = t_{j+1}^{i_{j+1}} \dots t_n^{i_n} t_1^{i_1} \dots t_{j-1}^{i_{j-1}} t_j^s \; untuk \; 1 \le s \le k$$

Setiap f_j adalah fungsi bijektif karena dipetakan pada dirinya sendiri. Setiap vertex $x \in V\,(G_n^k)$ mempunyai pasangan $2k$ vertex $f_s(x)$ dan $f_j^{-1}(x)$ untuk semua $1 \le j \le k$. Contohnya pada graf G_4^2, *vertex* $t_1^1 t_2^2 t_3^1 t_4^1$ akan memiliki pasangan *vertex* $t_2^2 t_3^1 t_4^1 t_1^1$, $t_2^2 t_3^1 t_4^1 t_1^2$, $t_4^1 t_1^1 t_2^2 t_3^1$, dan $t_4^2 t_1^1 t_2^2 t_3^1$ (Hsing dan Kuan Lin, 2009).

Definisi 2.2 yang telah dijabarkan sebelumnya dapat digunakan untuk menggambar graf *Cyclic-cubes.*

Berikut ini adalah salah satu contoh bentuk graf *Cyclic-cubes*,



Gambar 11. Graf G_2^2

Definisi 2.3 Graf *Wrapped Butterfly Networks* (WB) (*n*,*k*) didefinisikan sebagai graf yang mempunyai *vertex* $n.k^n$ dan setiap *vertex*nya direpresentasikan dengan (n-1)-bit vektor $a_0a_1 \ldots\ldots a_{n-1}i$ dimana $0 \leq i \leq n-1$ dan $0 \leq a_j \leq k$ untuk semua $0 \leq j \leq n-1$.

Dua *vertex* $a_0a_1 \ldots\ldots a_{n-1}i$ dan $b_0ab_1 \ldots\ldots ba_{n-1}j$ dikatakan bertetangga (*adjacent*) pada WB (*n*,*k*) jika dan hanya jika *j-i=1* (mod n) dan $a_t = b_t$ untuk semua $0 \leq t \neq j \leq n-1$ (Hsing dan Kuan Lin, 2009).