# BAB 1. DASAR-DASAR GRAF

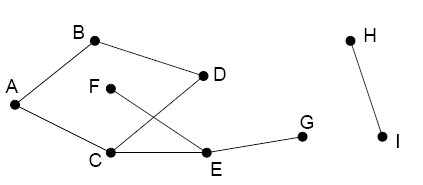
## 1.1. Pengertian Graf

Secara sederhana graf didefinisikan sebagai kumpulan titik yang dihubungkan oleh garis. Secara matematis, graf adalah pasangan himpunan (*V*,*E*) dimana *V* adalah himpunan tak kosong yang memiliki elemen disebut *vertices* dan *E* adalah kumpulan dari dua elemen subsets *V* yang disebut *edges*.

*Vertices* direpresentasikan dengan titik dan *edges* direpresentasikan dengan garis. Gambar 1.1 adalah contoh graph (*V*,*E*) dimana:

*V* = {*A*,*B*,*C*,*D*,*E*, *F*, *G*,*H*, *I*}, dan

*E* = {{*A*,*B*} , {*A*,*C*} , {*B*,*D*} , {*C*,*D*} , {*C*,*E*} , {*E*, *F*} , {*E*,*G*} , {*H*, *I*}} .



**Gambar 1.1.** Contoh graf

Sebuah *edge* selalu memiliki dua *endpoint*, misalnya *edge* {*H*,*I*} memiliki *endpoint* *H* dan *I*. Graf biasanya digunakan untuk memodelkan objek-objek diskrit dan hubungan antar objek-objek tersebut.

## 1.2. Graf Sederhana (*Simple Graph*)

Graf sederhanaadalah graf yang tidak mengandung *loops* atau *multiple edges*. *Loops* adalah *edge* yang memiliki *endpoint* sama, sedangkan *multiple edges* adalah *edge* yang memiiki pasangan *endpoint* sama.

Contoh graf sederhana dan bukan graf sederhana dapat dilihat pada Gambar 1.2. Gambar 1.2.a bukan graf sederhana karena memiliki *loop*. Gambar 1.2.b bukan graf sederhana karena memiliki *multiple edges*. Gambar 1.2.c merupakan graf sederhana karena tidak memiliki *loops* atau *multiple edges*.

|  |  |  |
| --- | --- | --- |
| a | b | c |

**Gambar 1.2.** a. Bukan graf sederhana karena memiliki *loop*. b. Bukan graf sederhana karena memiliki *multiple edges*. c. Graf sederhana

Beberapa jenisgraf khusus yang termasuk graf sederhana diantaranya :

**1. Graf Komplit (Complete Graph *K*n)**

Adalah graf dimana setiap pasang *vertices* selalu memiliki sebuah *edge*. Graf komplit dapat diamati pada Gambar 1.3. Simbol dari graf komplit adalah *K*n, dimana *n* menyatakan jumlah *vertices* dari graf komplit.



**Gambar 1.3.** Graf komplit *Kn* untuk *n*=1,2,3,4

**2. Cycle (*Cn*)**

Graf *cycle* adalah graf *C* = (*V*,*E*) dengan bentuk

*V* = {*v*1, *v*2, *v*3, *v*4, … , *v*n} dan

*E* = {{ *v*1, *v*2} , { *v*2, *v*3} , { *v*3, *v*4}, … , { *v*n, *v*1} } ,

dimana *n*≥3 dan *v*1, *v*2, *v*3, *v*4, … , *v*n adalah *vertices* yang berbeda. Graf *cycle* disimbolkan dengan *C*n dimana *n* adalah banyaknya *vertices*. Gambar 1.4 menunjukkan graf *cycle* *C*n untuk n=3, 4, dan 5.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| *C*3 | *C*4 | *C*5 |

**Gambar 1.4.** Graf cycle *Cn* untuk *n*=3,4,5

**3. Wheel** **(*Wn*)**

Graf *wheel* adalah graf *cyle* yang ditambahi sebuah *vertex* baru (*vm*) dimana *vm* terhubung ke seluruh *vertices* yang ada. Sehingga graf W = (V,E) dengan bentuk

*V* = {*v*1, *v*2, *v*3, *v*4, … , *v*n, *v*m } dan

*E* = {{ *v*1, *v*2} , { *v*2, *v*3} , { *v*3, *v*4}, … , { *v*n, *v*1}, { *v*1, *v*m} , { *v*2, *v*m} , { *v*3, *v*m}, … , { *v*n, *v*m } } ,

dimana *n*≥3 dan *v*1, *v*2, *v*3, *v*4, … , *v*n , *v*m adalah *vertices* yang berbeda. Graf *wheel* disimbolkan dengan *W*n dimana *n*+1 adalah banyanya *vertices*. Gambar 1.5 menunjukkan graf *wheel* *W*n untuk n=3, 4, dan 5.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| *W*3 | *W*4 | *W*5 |

**Gambar 1.5.** Graf wheel *Wn* untuk *n*=3,4,5

## 1.3. Graf dan Sub-graf

Jika *V*(*G*) dan *E*(*G*) adalah himpunan *vertices* dan *edge* pada graf *G*, serta V(*H*) dan *E*(*H*) adalah himpunan *vertices* dan *edge* pada graf *H*. Maka graf *H* disebut sub-graf dari graf *G* jika dan hanya jika V(H) ⊆ V(G) dan E(H) ⊆ E(G).

Jadi, jika *e* adalah *edge* pada graph *H* yang menghubungkan vertex *v* dan *u* maka *e* juga merupakan *edge* pada graph *G* yang menghubungkan vertex *v* dan *u* di graf *G*. Gambar 1.6 menunjukkan graf *H* yang merupakan sub-graf dari graf *G*.

|  |  |
| --- | --- |
|  |  |
| Graf G | Graf H |

**Gambar 1.6.** Graf *H* adalah sub-graf dari graf *G*

## 1.4. Graf Tak-berarah dan Graf Berarah

Graph tak-berarah (*undirected graph*) adalah graf yang tidak memiliki orientasi arah pada setiap *edge* yang dimiliki. Penulisan *edge* tidak memperhatikan urutan. Penulisan *edge* *e* = (*u*,*v*), dimana *edge* *e* adalah *edge* yang menghubungakan vertex *u* dan *v* sama saja dengan penulisan *e* = (*v*,*u*). Semua graf yang ditampilkan pada Gambar 1.1 sampai dengan Gambar 1.6 merupakan graf tak-berarah.

Graf berarah (*directed graph*/ *digraph*) adalah graf yang memiliki orientasi arah pada setiap *edge* yang dimiliki. Sehingga, penulisan *edge* *e* = (*u*,*v*) untuk *edge* *e* yang mengubungkan vertex *u* dan *v* berbeda maknanya dengan penulisan *edge* *e* = (*v*,*u*) yang menghubungkan vertex *v* dan *u*. Setiap *edge* pada digraph biasa juga disebut dengan *arc*.

|  |  |
| --- | --- |
|  |  |
| Graf G | Graf H |

**Gambar 1.6.** Graf *H* adalah sub-graf dari graf *G*

## 1.5. Representasi Graf

Graf memililiki kemudahan pemahaman ketika direpresentasikan dalam bentuk visual. Akan tetapi, ketika domain permasalahan graf dibawa pada pemrograman maka graf harus dapat direpresentasikan dalam struktur data. Secara umum terdapat dua cara yang dapat merepresentasikan graf, yaitu representasi matriks dan representasi list.

**1. Representasi matriks**

**a. *Adjacency matrix* (Matriks Ketetanggan)**





**Gambar 1.7.** Graf  *G* yang memiliki *adjacency matrix A*

*Adjacency matrix* dari sebuah graf *G* dengan *n* *vertices* adalah berupa matriks *A* = {*aij*} dengan ukuran *n*x*n*. *aij* bernilai 1 apabila terdapat sebuah *edge* yang menghubungkan *vertex* *i* dan *vertex* *j* dan *aij* bernilai 0 apabila tidak terdapat sebuah *edge* yang menghubungkan *vertex* *i* dan *vertex* *j*.

*A* adalah *adjacency matrix* dari graf *G* pada Gambar 1.7. Graf *G* memiliki 5 *vertices* sehingga *A* adalah matriks yang berukuran 5x5. Elemen *a*1,2 bernilai 1 karena terdapat sebuah *edge* yang menghubungkan *vertex* 1 dan *vertex* 2. Sedangkan elemen *a*1,3 bernilai 0 karena tidak terdapat sebuah *edge* yang menghubungkan *vertex* 1 dan *vertex* 3.

*B* adalah *adjacency matrix* dari graf *H* pada Gambar 1.8. Jika dibandingkan antara matriks A dan matriks B, dapat dilihat bahwa *adjacency matrix* dari graf tak berarah (A) bersifat simetris dan graf berarah (B) belum tentu memiliki *adjacency matrix* yang simetris.





**Gambar 1.8.** Graf  *H* yang memiliki *adjacency matrix B*

**b. *Incidence matrix* (Matriks Insiden)**

*Incidence**matrix* dari sebuah graf *G* dengan *m* *vertices* dan *n edges* adalah berupa matriks *A* = {*aij*} dengan ukuran *m*x*n*. *aij* bernilai 1 apabila vertex *i* merupakan salah satu *end point* dari edge *j* dan sebaliknya, *aij* bernilai 0 apabila vertex *i* bukan merupakan *end point* dari edge *j.*

*A* adalah *incidence**matrix* dari graf *G* pada Gambar 1.9. Graf *G* memiliki 5 *vertices* dan 8 *edges*, sehingga *A* adalah matriks yang berukuran 5x8. Elemen *a*1,1  dan *a*2,1 bernilai 1 karena vertex 1 dan vertex 2 adalah *end point* edge e1. *Incidence**matrix* hanya bisa untuk merepresentasikan graf tak berarah, sedangkan graf berarah tidak bisa menggunakan *incidence**matrix* karena definisi *end point* hanya terdapat pada graf tak berarah.





**Gambar 1.8.** Graf  *G* yang memiliki *adjacency matrix A*

**2. Representasi List**





**Gambar 1.9.** Graf  *G* yang memiliki *adjacency matrix A*

*Adjacency matrix* dari sebuah graf *G* dengan *n* *vertices* adalah berupa matriks *A* = {*aij*} dengan ukuran *n*x*n*. *aij* bernilai 1 apabila terdapat sebuah *edge* yang menghubungkan *vertex* *i* dan *vertex* *j* dan *aij* bernilai 0 apabila tidak terdapat sebuah *edge* yang menghubungkan *vertex* *i* dan *vertex* *j*.

*A* adalah *adjacency matrix* dari graf *G* pada Gambar 1.7. Graf *G* memiliki 5 *vertices* sehingga *A* adalah matriks yang berukuran 5x5. Elemen *a*1,2 bernilai 1 karena terdapat sebuah *edge* yang menghubungkan *vertex* 1 dan *vertex* 2. Sedangkan elemen *a*1,3 bernilai 0 karena tidak terdapat sebuah *edge* yang menghubungkan *vertex* 1 dan *vertex* 3.

*B* adalah *adjacency matrix* dari graf *H* pada Gambar 1.8. Jika dibandingkan antara matriks A dan matriks B, dapat dilihat bahwa *adjacency matrix* dari graf tak berarah (A) bersifat simetris dan graf berarah (B) belum tentu memiliki *adjacency matrix* yang simetris.





**Gambar 1.8.** Graf  *H* yang memiliki *adjacency matrix B*