

STRUKTUR DATA & ALGORITMA

IN OUR CLASSROOM



AN INTRODUCTION TO **GRAPH**



CAPAIAN PEMBELAJARAN

- Mahasiswa mendapatkan pemahaman mengenai cara kerja dan penyajian graph

Agenda.

- Konsep **Graph**
- Representasi **Graph**
- Penelusuran **Graph (Graph Traversal)**
- Lintasan terpendek (**Shortest Graph**)

KONSEP GRAPH

Definisi Graph.

“A **Graph (G)** consists two sets of V and E . V (**vertice**) is finite non-empty set of vertices and E (**edge**) is a set of pairs vertices”

Ellist Horowitz, Satraj Sahni, “*Fundamentals of Data Structures*”

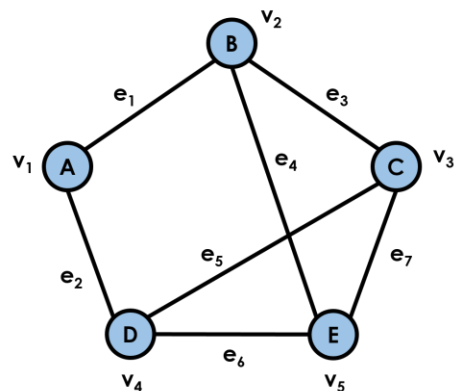
“A **Graph** consists of a set of **nodes** (or **vertices**) and a set of **arc** (or **edge**). Each arc in a graph is specified by a **pair of nodes**”

Aarn M.Tenebaum, “*Data Structures Using C and C++*”

5

KONSEP GRAPH

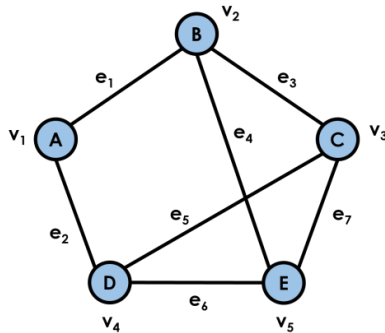
- Definisi **graph** sederhana sekumpulan dari simpul dan busur
- Bentuk busur (**edge/arc**) dapat dibedakan menjadi 2 macam:
 1. Graph Tak Berarah (**undirected graph** atau **non-direct graph**), adalah graph yang simpulnya hanya menghubungkan 2 vertex tanpa menunjukkan arah.
 2. Graph Berarah (**directed graph/digraph**), adalah graph yang simpulnya tidak hanya menghubungkan 2 vertex tetapi juga menunjukkan arah.



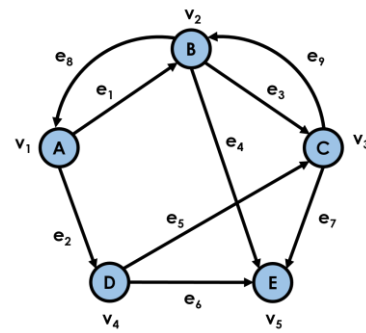
6

KONSEP GRAPH

Undirected Graph



Directed Graph



7

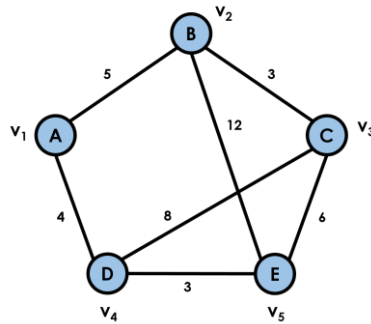
GRAPH | GRAPH BERBOBOT (WEIGHTED GRAPH)

- Bobot sebuah graph ditentukan dari nilai yang menyatakan hubungan antara 2 buah simpul.
- Bobot sebuah busur (**edge/arc**) dapat menyatakan:
 - panjang antara 2 buah titik (**node/vertex**),
 - jarak antara 2 kota yang diwakilkan dengan simpul, atau
 - selisih antara 2 buah simpul.

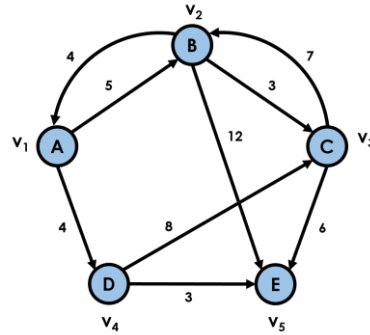
8

GRAPH | GRAPH BERBOBOT (WEIGHTED GRAPH)

Weighted Undirected Graph



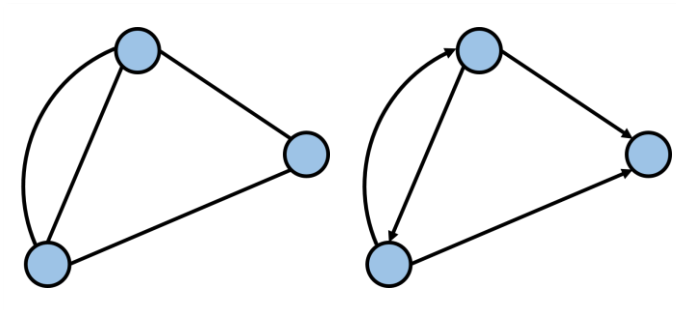
Weighted Directed Graph



9

GRAPH | GRAPH SEDERHANA (SIMPLE GRAPH)

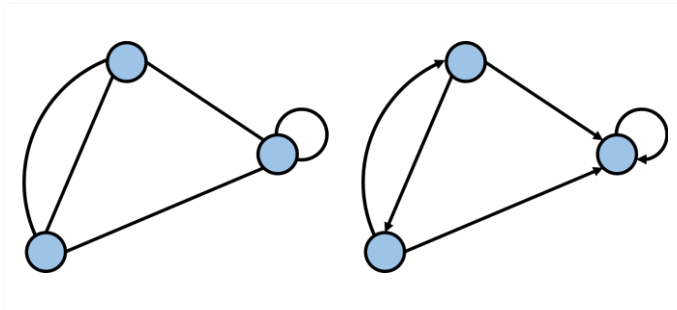
- Graph sederhana (**simple graph**), adalah graph yang mempunyai busur paling banyak hanya satu macam.



10

GRAPH | GRAPH TAK SEDERHANA (COMPLEX GRAPH)

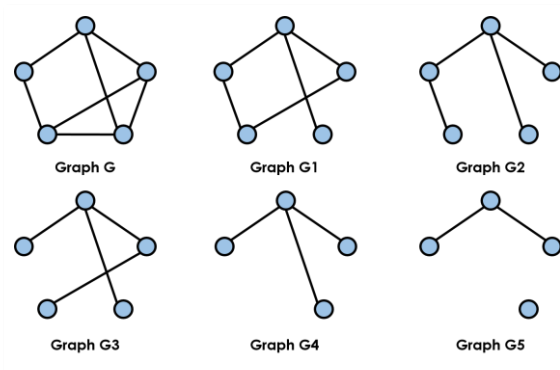
- Graph tak sederhana (**simple graph**), adalah graph yang mempunyai busur yang memiliki bentuk busur lebih dari satu macam.



11

GRAPH | SUBGRAPH

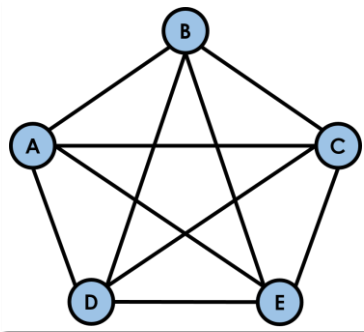
- Subgraph**, adalah graph yang merupakan bagian dari sebuah graph utama.



12

GRAPH | FULL CONNECTED GRAPH

- **Full connected graph**, adalah graph yang semua simpulnya saling terhubung satu dengan lainnya secara penuh.



Untuk menyatakan jumlah total busur dalam **full connected graph** berlaku rumus: $m = n(n-1)/2$, dimana **m**: jumlah busur, **n**: jumlah node

13

ISTILAH-ISTILAH GRAPH

Istilah dalam Graph.

1. Incident
2. Degree
3. Indegree
4. Outdegree
5. Adjacent
6. Successor
7. Predecessor
8. Path
9. Cycle

14

REPRESENTASI GRAPH

- Penyajian (representasi) graph dalam sistem komputer harus dinyatakan dalam sebuah struktur data yang menggambarkan graph tersebut.
- Penyajian (representasi) graph dapat dinyatakan dalam bentuk:
 1. **Matrix** dan
 2. **Linked-list**.

17

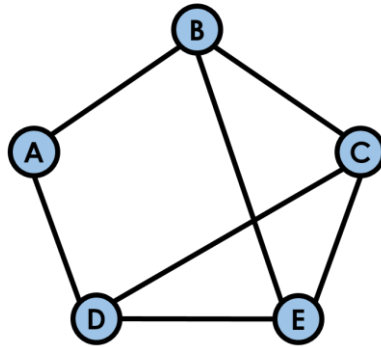
REPRESENTASI GRAPH

- Penyajian (representasi) graph dalam bentuk **matrix** terdiri dari:
 1. *Adjacency Matrix Graph*
 2. *Directed Invers Adjacency Matrix Graph*
 3. *Insidence Matrix Graph*
 4. *Vector Matrix Graph*
- Penyajian (representasi) graph dalam bentuk **linked-list** terdiri dari:
 1. *Adjacency List Graph*
 2. *Directed Invers Adjacency List Graph*

18

REPRESENTASI GRAPH | MATRIX

1. A. Adjacency Matrix (Undirected Graph)



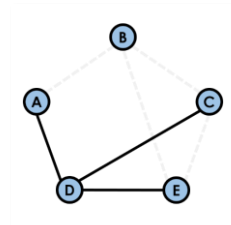
	A	B	C	D	E
A 0	0	1	0	1	0
B 1	1	0	1	0	1
C 2	0	1	0	1	1
D 3	1	0	1	0	1
E 4	0	1	1	1	0

19

REPRESENTASI GRAPH | MATRIX

- Graph disajikan dalam matrix $n \times n$, dimana n = jumlah simpul
- Matrix yang terbentuk adalah **matrix simetris** dengan sumbu simetris berbentuk diagonal
- Data yang terdapat dalam baris dan kolom dapat menyatakan derajat (**degree**) sebuah simpul.

	A	B	C	D	E
A 0	0	1	0	1	0
B 1	1	0	1	0	1
C 2	0	1	0	1	1
D 3	1	0	1	0	1
E 4	0	1	1	1	0

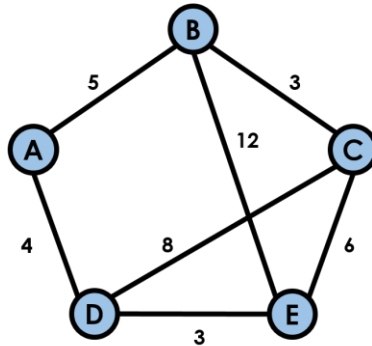


	A	B	C	D	E
A 0	0	1	0	1	0
B 1	1	0	1	0	1
C 2	0	1	0	1	1
D 3	1	0	1	0	1
E 4	0	1	1	1	0

20

REPRESENTASI GRAPH | MATRIX

Weight Adjacency Matrix (Undirected Graph)

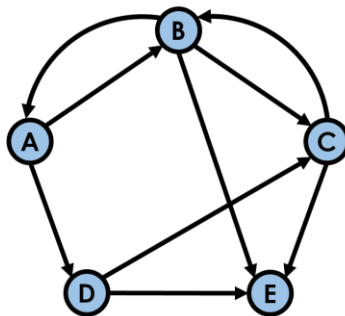


	A	B	C	D	E
	0	1	2	3	4
A 0	0	5	999	4	999
B 1	5	0	3	999	12
C 2	999	3	0	8	6
D 3	4	999	8	0	3
E 4	999	12	6	2	0

21

REPRESENTASI GRAPH | MATRIX

1. B. Adjacency Matrix (Directed Graph)

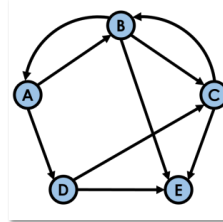


	dari	ke	A	B	C	D	E
			0	1	2	3	4
A 0	0	1	0	1	0	0	0
B 1	1	0	1	0	1	0	1
C 2	0	1	0	0	0	1	0
D 3	0	0	1	0	0	1	0
E 4	0	0	0	0	0	0	1

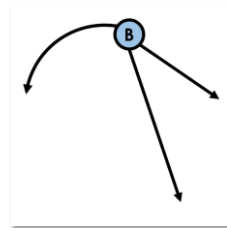
22

REPRESENTASI GRAPH | MATRIX

- Graph disajikan dalam matrix $n \times n$, dimana n = jumlah simpul
- Matrix yang terbentuk dapat berupa **matrix simetris** maupun **asimetris**
- Data yang terdapat dalam satu baris dapat menyatakan **outdegree** sebuah simpul yang bersangkutan



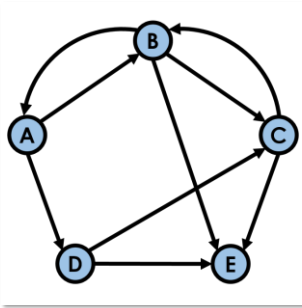
	A	B	C	D	E
A 0	0	1	0	1	0
B 1	1	0	1	0	1
C 2	0	1	0	0	1
D 3	0	0	1	0	1
E 4	0	0	0	0	0



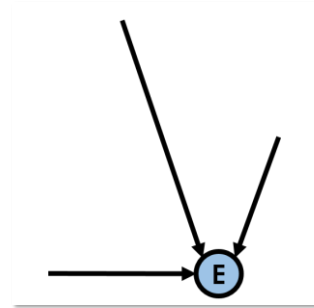
23

REPRESENTASI GRAPH | MATRIX

- Data yang terdapat dalam satu kolom dapat menyatakan **indegree** sebuah simpul yang bersangkutan



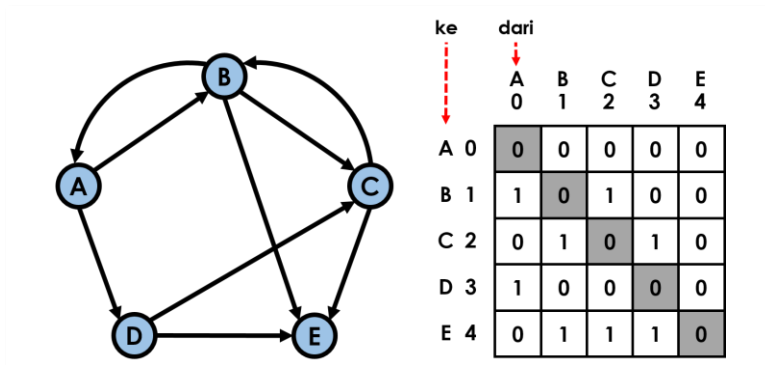
	A	B	C	D	E
A 0	0	1	0	1	0
B 1	1	0	1	0	1
C 2	0	1	0	0	1
D 3	0	0	1	0	1
E 4	0	0	0	0	0



24

REPRESENTASI GRAPH | MATRIX

2. Invers Adjacency Matrix (Directed Graph)



25

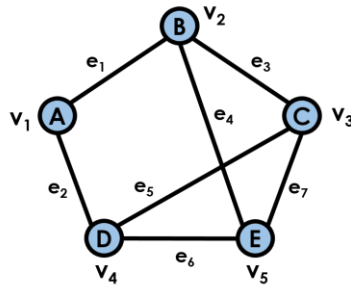
REPRESENTASI GRAPH

- Data yang terdapat dalam satu baris dapat menyatakan **indegree** sebuah simpul yang bersangkutan
- Data yang terdapat dalam satu kolom dapat menyatakan **outdegree** sebuah simpul yang bersangkutan

26

REPRESENTASI GRAPH | MATRIX

3. Incidence Matrix (Undirected Graph)

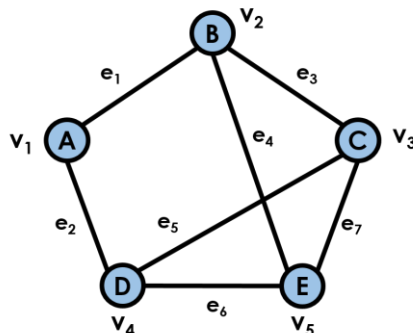


	e ₁	e ₂	e ₃	e ₄	e ₅	e ₆	e ₇
A 0	1	1	0	0	0	0	0
B 1	1	0	1	1	0	0	0
C 2	0	0	1	1	1	0	1
D 3	0	1	0	0	1	1	0
E 4	0	0	0	1	0	1	1

27

REPRESENTASI GRAPH | MATRIX

4. Vector Matrix (Undirected Graph)

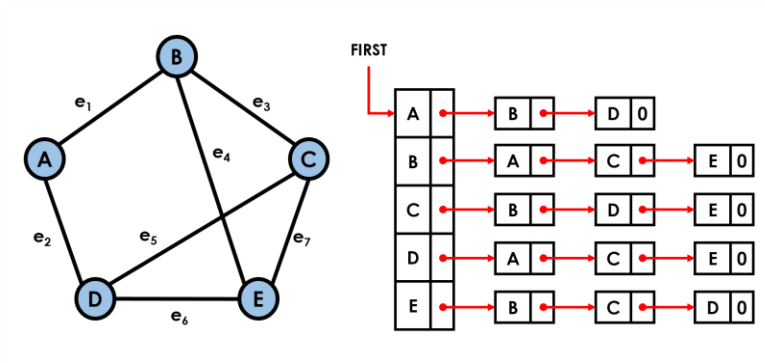


	e	e	e	e
A 0	1	2	0	0
B 1	1	0	3	4
C 2	3	4	7	1
D 3	2	5	6	0
E 4	4	6	7	1

28

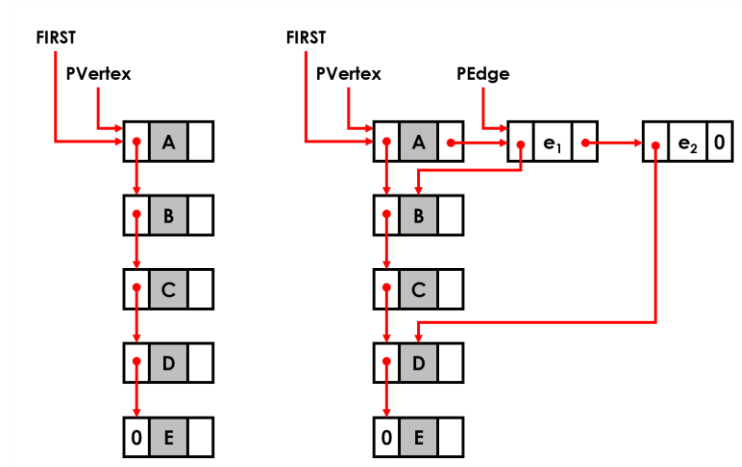
REPRESENTASI GRAPH | LINKED LIST

1. Adjacency List (Undirected Graph)



29

REPRESENTASI GRAPH | LINKED LIST



30

REPRESENTASI GRAPH | LINKED LIST

- Terdapat 2 buah jenis simpul penyajian graph dalam **linked-list**:
 1. **Simpul-Vertex**: **pointer left** digunakan untuk menunjuk simpul berikutnya dan NULL jika tidak ada simpul yang perlu ditunjuk, sedangkan **pointer right** digunakan untuk menunjuk **simpul-edge** yang pertama.
 2. **Simpul-Edge**: **pointer left** digunakan untuk menunjuk **simpul vertex tujuan**, sedangkan **pointer right** digunakan untuk menunjuk **simpul-edge** berikutnya jika masih ada dan NULL jika tidak **simpul-edge** yang ditunjuk.

31

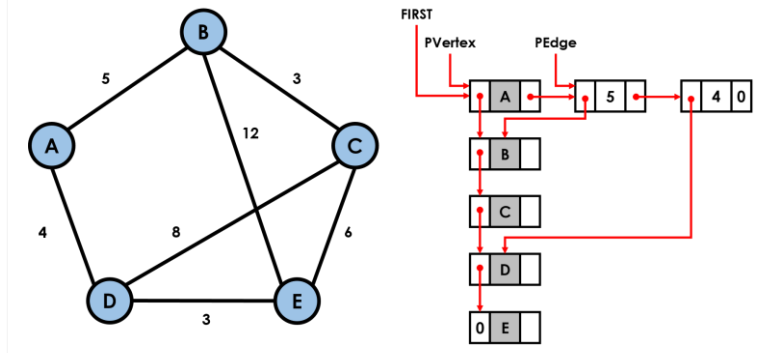
REPRESENTASI GRAPH | LINKED LIST

- Terdapat 2 buah jenis simpul penyajian graph dalam **linked-list**.
 1. **Simpul-Vertex**: **pointer left** digunakan untuk menunjuk simpul berikutnya dan NULL jika tidak ada simpul yang perlu ditunjuk, sedangkan **pointer right** digunakan untuk menunjuk **simpul-edge** yang pertama.
 2. **Simpul-Edge**: **pointer left** digunakan untuk menunjuk **simpul vertex tujuan**, sedangkan **pointer right** digunakan untuk menunjuk **simpul-edge** berikutnya jika masih ada dan NULL jika tidak **simpul-edge** yang ditunjuk.

32

REPRESENTASI GRAPH | LINKED LIST

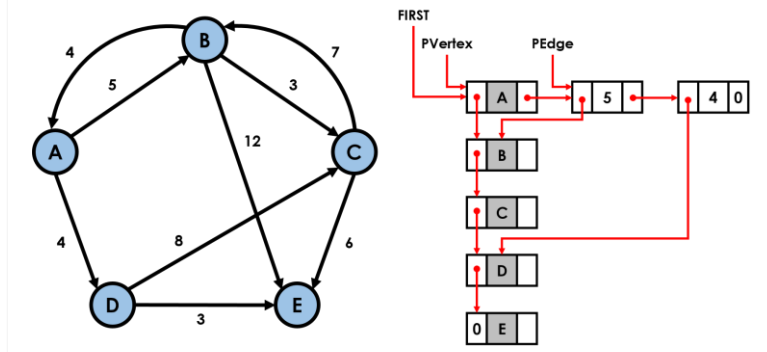
Weight Adjacency List (Undirected Graph)



33

REPRESENTASI GRAPH | LINKED LIST

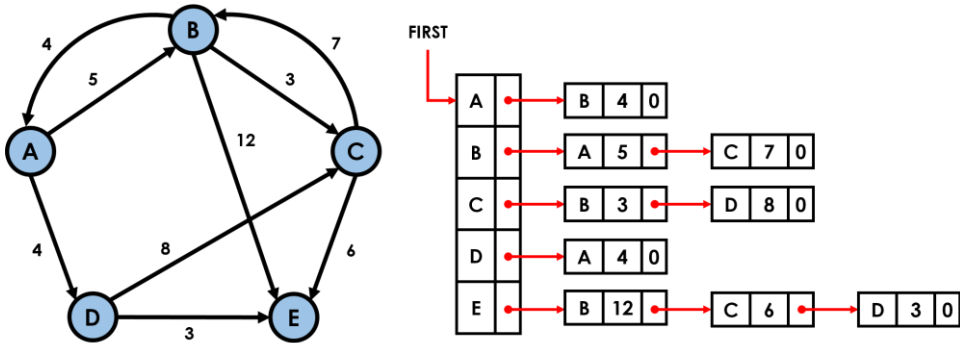
Weight Adjacency List (Directed Graph)



34

REPRESENTASI GRAPH | LINKED LIST

2. Inverse Weight Adjacency List (Directed Graph)



35

AN INTRODUCTION TO GRAPH

