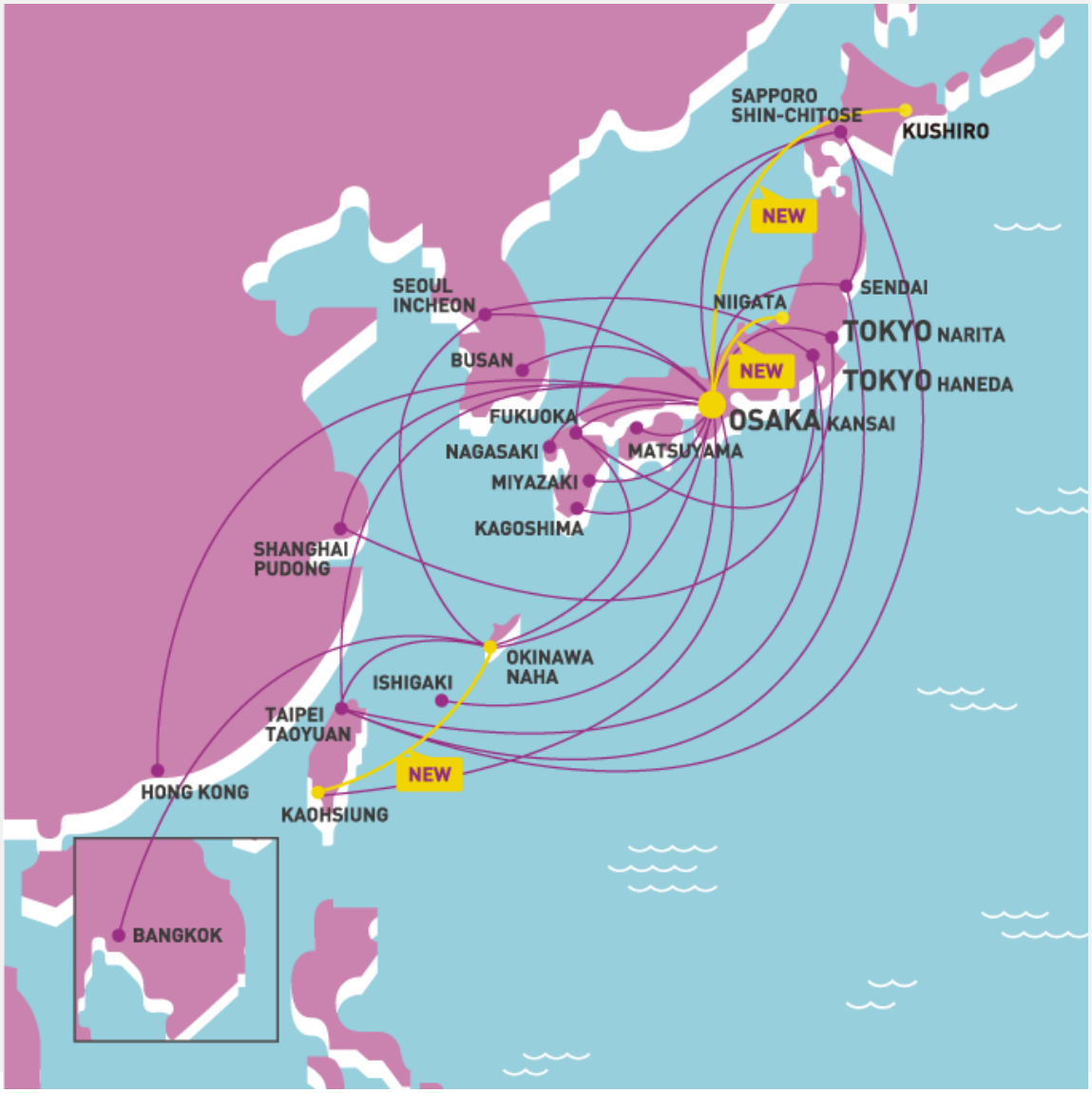
Module4—Graph

1. กราฟคืออะไร ทำไมต้องใช้โครงสร้างข้อมูลแบบกราฟ
2. นิยามของกราฟ digraph, undigraph, vertice, edge, adjacent, incident, degree, path
3. การเก็บกราฟด้วย adjacency list
4. การท่องกราฟด้วย BFS

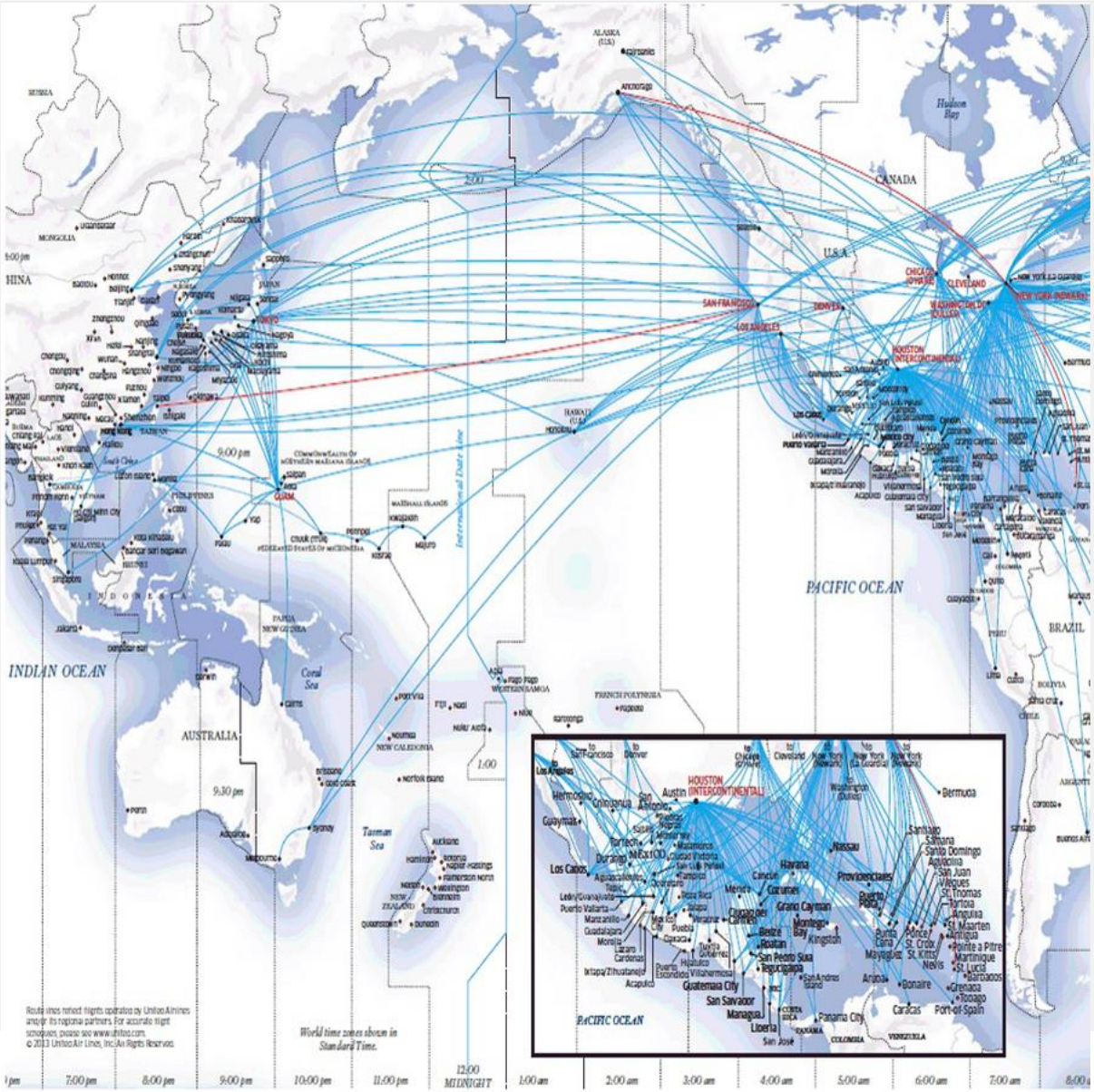


เก็บข้อมูล

- มีทิศทาง
- มีระยะทาง

ตัวอย่างงาน

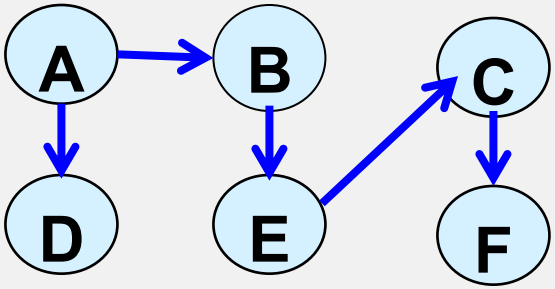
- การวางข่าย งานคอมพิวเตอร์
- การวิเคราะห์เส้นทางวิกฤติ
- ปัญหาเส้นทาง ที่สั้นที่สุด

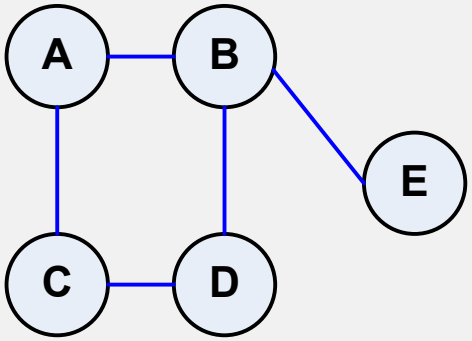

CANADA
U.S.A.
PACIFIC OCEAN
INDIAN OCEAN
AUSTRALIA
BRAZIL
HOUSTON (INTERCONTINENTAL)
SAN FRANCISCO
LOS ANGELES
DENVER
CHICAGO (O'HARE)
CLEVELAND
NEW YORK (NEWARK)
WASHINGTON (DULLES)
Hudson Bay
Coral Sea
Tasman Sea
International Date Line
World time zones shown in Standard Time.
MIDNIGHT
Route lines reflect flights operated by United Airlines and/or its regional partners. For accurate flight schedules, please see www.united.com.
© 2013 United Air Lines, Inc. All Rights Reserved.

4.1 Definition

เกี่ยวอย่างมีทิศทาง

1. directed graph(digraph)
2. Undirected graph(Undigraph)



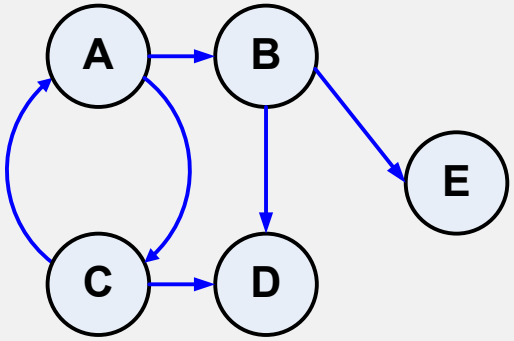


4.1 Definition

1) A directed graph(or digraph) G : is a pair(V,E), where V is a finite set and E is binary relation on V.

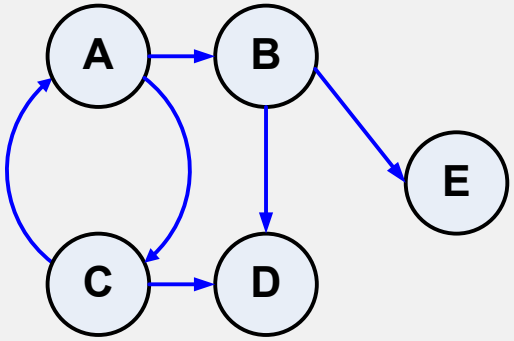
- The ***set V*** is called the **vertex set** of G, and its elements are called vertices.

V={A,B,C,D,E}



(A,B) (A,C) (B,D) (B,E)
(C,A) (C,D)

- The ***set E*** is called the **edge set** of G, and its elements are called edges.



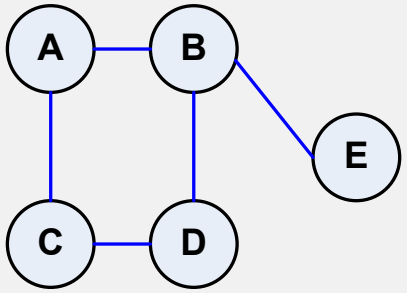
The edge set E consists of ordered pairs of vertices

มีลำดับ

V={A,B,C,D,E}
E={ (A,B), (A,C), (B,D), (B,E), (C,A),(C,D) }

2) An undirected graph G :

G = (V,E) , the edge set E consists of unordered pairs of vertices, rather than ordered pairs.



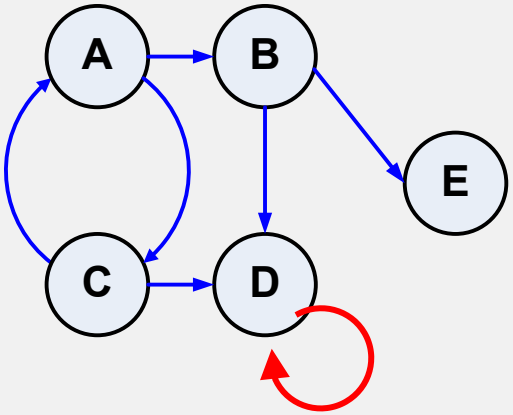
V={A,B,C,D,E}

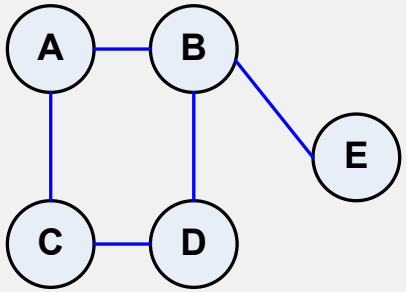
E={ (A,B), (B,A), (A,C), (C,A), (B,D),(D,B), (B,E), (E,B), (C,D),(D,C) }

3) self-loops : edges (เส้นเชื่อม) from a vertex to itself.

Undigraph self loop are forbidden, and so every edge consists of exactly two distinct vertices. $u \neq v$

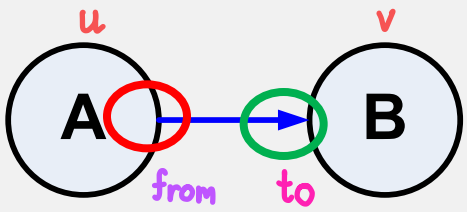
Cycle : a cycle in a graph is a non-empty trail in which only the first and last vertices are equal





4) Incident :

digraph : if (u,v) is an edge in a directed graph G=(V,E), we say that(u,v) is



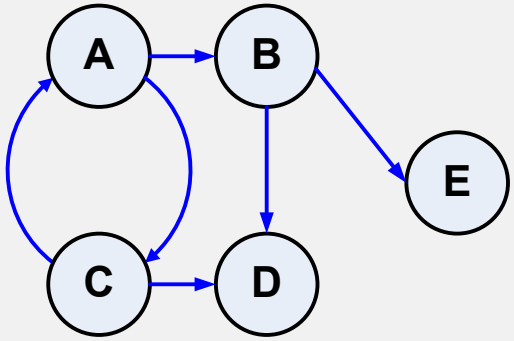
incident from or leaves vertex u

incident to or enters vertex v.

Question edge (B,E)

Incident from ..B..

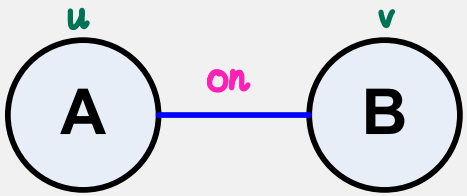
Incident to ..E..



undigraph : if (u,v) is an edge in a directed graph G=(V,E), we say that(u,v) is ***incident on*** vertices u and v

ติดต่อ

is ***incident on*** vertices u and v



Question edge (A,B)

Incident from ....

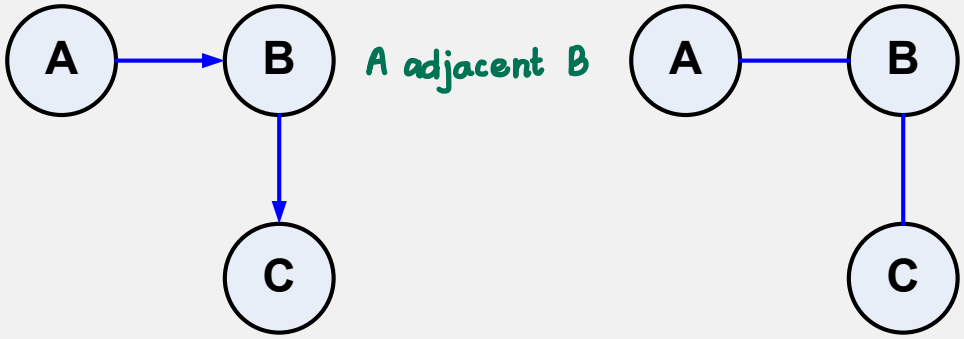
Incident to ....

Edge (A,B) is incident on vertices A and B

5) adjacent ประชิด

Digraph : If (u,v) is an edge in a graph G=(V,E), we say that v is adjacent to vertex u. If v is adjacent to u denote by u -> v

Undigraph : Adjacent relation is symmetric.

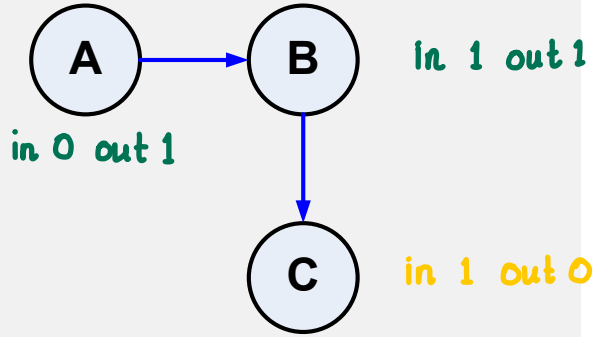


B ประชิด A A->B

B ประชิด A และ A ประชิด B

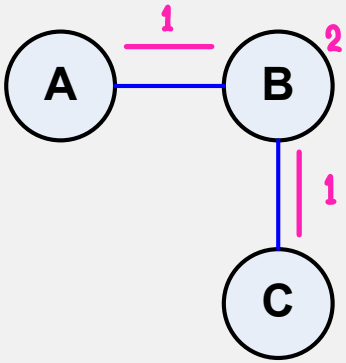
6) Degree

Digraph : in degree (number of edges entering it.), out degree (Opposite in degree.)



A in degree = 0
A out degree = 1

Undigraph : is the number of edges incident on it.

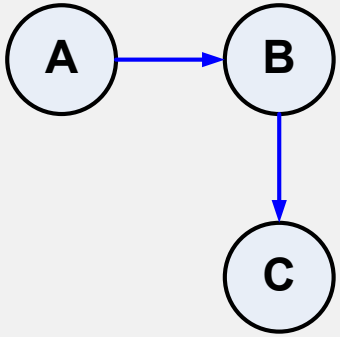


A degree = 1

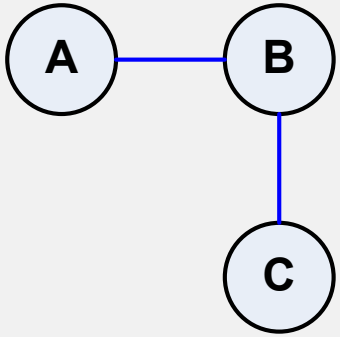
B degree = 2

C degree = 1

7) path of length : Path k from a vertex u to a vertex u' in a graph G=(V,E) is a sequence $(v_o,v_1,..,v_k)$ of vertices such that $u = v_0$, $u' = v_k$, and $(v_{i-1}, v_i) \in E$ for i = 1, 2, .., k



A,C (A,B,C)



A,C

(A,B,A,B,C)

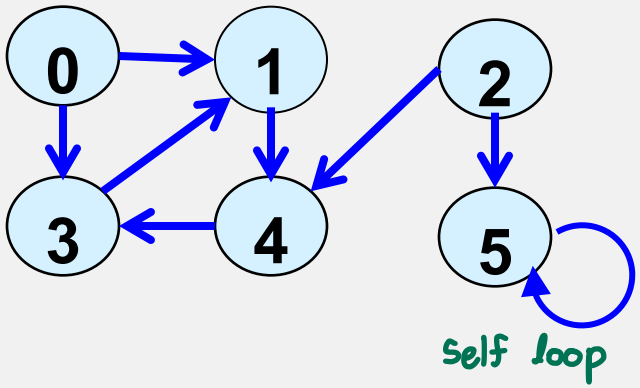
(A,B,C)

4.2 โครงสร้างข้อมูลที่ใช้เก็บข้อมูลในรูปแบบกราฟคือ

1. Adjacency list
2. Adjacency Matrix

โครงสร้างข้อมูลทั้ง 2 แบบนี้จะสามารถเก็บกราฟได้ทั้ง digraph และ undigraph

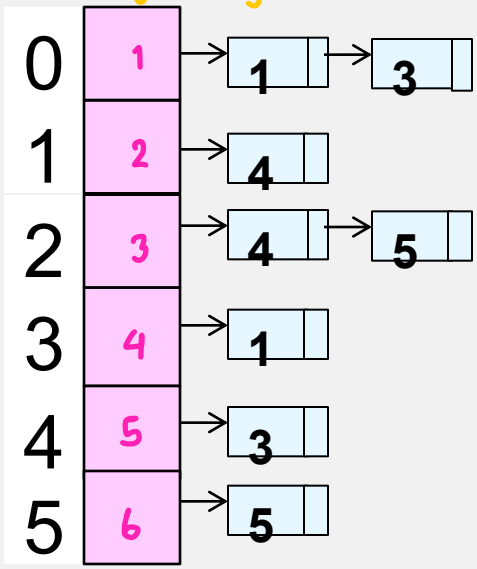
Digraph



1. Adjacency list ✓
2. Adjacentcy matix

Com

adjacency list



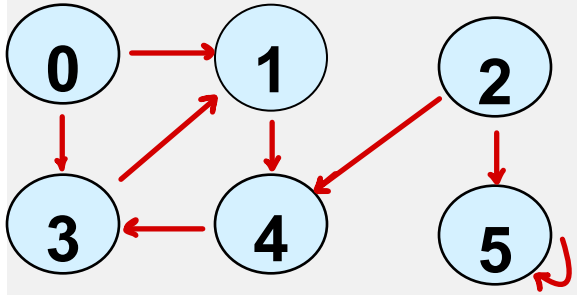
โครงสร้างในการเก็บกราฟ

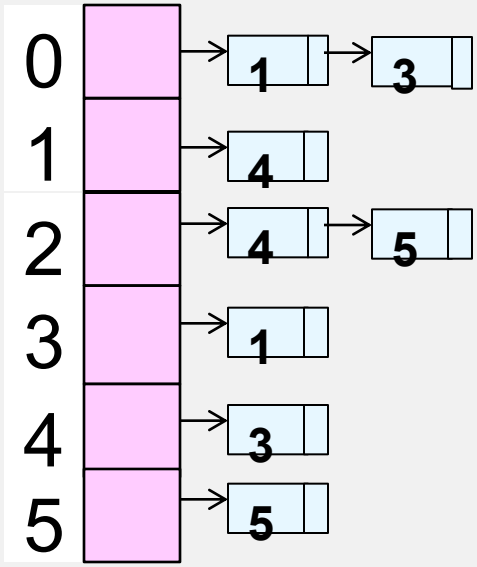
	0	1	2	3	4	5
0	0	1	0	1	0	0
1	0	0	0	0	1	0
2	0	0	0	0	1	1
3	0	1	0	0	0	0
4	0	0	0	1	0	0
5	0	0	0	0	0	1

```
struct record * adj [6];
```

```
struct record {
    int value;
    struct record *next;
};
```

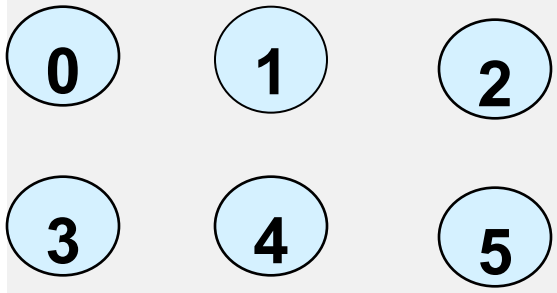
Digraph





1. Adjacency list
2. Adjacentcy matix

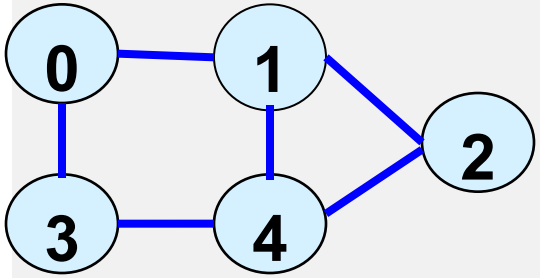
Digraph



	0	1	2	3	4	5
0	0	1	0	1	0	0
1	0	0	0	0	1	0
2	0	0	0	0	1	1
3	0	1	0	0	0	0
4	0	0	0	1	0	0
5	0	0	0	0	0	1

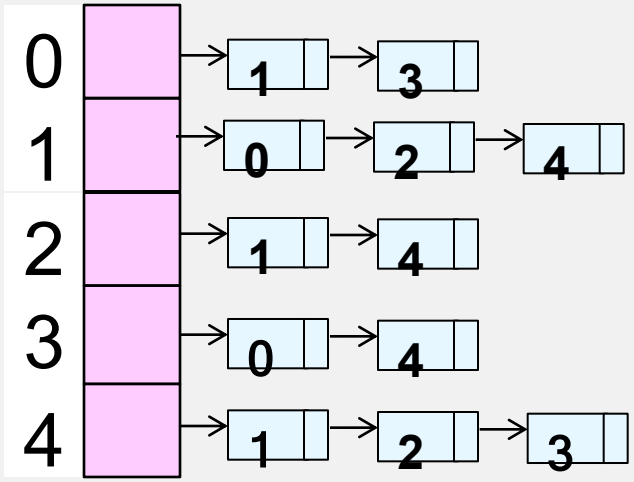
1. Adjacency list
2. Adjacentcy matix

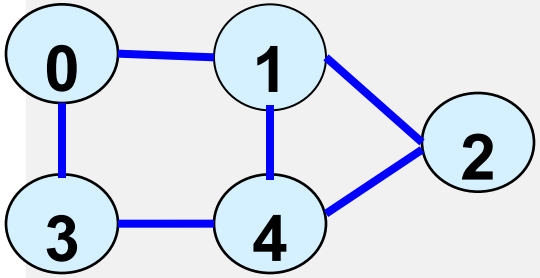
Undigraph



Adjacency list

12 loops forward - Back



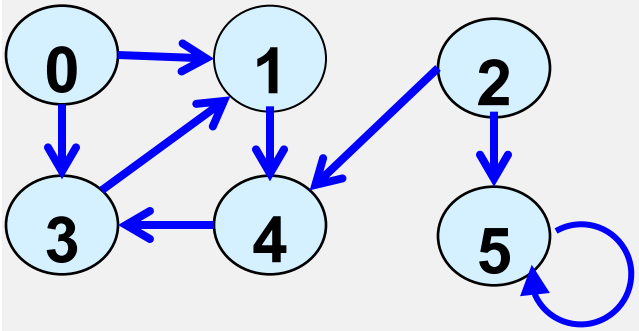


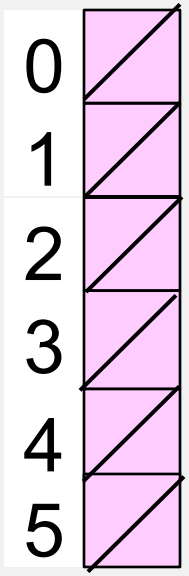
Adjacency Matrix

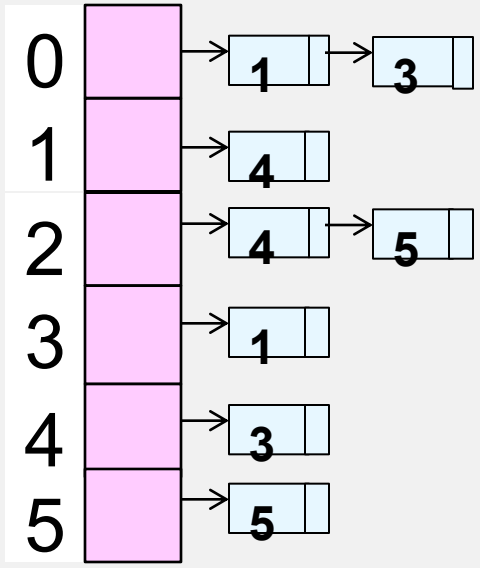
	0	1	2	3	4
0	0	1	0	1	0
1	1	0	1	0	1
2	0	1	0	0	1
3	1	0	0	0	1
4	0	1	1	1	0

mirror

การบ้าน ข้อ 1







Enter
#0 : 1 3 -1
#1 : 4 -1
...

```
for ( int i = 0 ;  i < 6 ; i++){
    adj [i] = NULL ;
```