



CP411107 Data Structures and Algorithms

## พื้นฐานทั่วไปของกราฟ

### ตัวอย่าง

เวอร์ชันที่เก็บ  $V(G)$  ได้แก่ A B C D E  
เอนซ์  $E(G)$  ได้แก่  
เส้นที่เชื่อมโยงจาก A ไป B  
เส้นที่เชื่อมโยงจาก A ไป C  
เส้นที่เชื่อมโยงจาก A ไป D  
เส้นที่เชื่อมโยงจาก A ไป E  
เส้นที่เชื่อมโยงจาก B ไป E

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

$$E(G) = \{(A,B), (A,C), (A,D), (A,E), (B,E)\}$$

CP411107 Data Structures and Algorithms

## พื้นฐานทั่วไปของกราฟ

### ชนิดของกราฟ

#### 1) กราฟแบบไม่มีทิศทาง (Undirected Graph)

จะเป็นกราฟที่มีเส้นเชื่อมโยงระหว่างเวอร์เทกซ์ทั้ง 2 จุดไม่มีทิศทางว่าจะจากเวอร์เทกซ์ใดไปยังเวอร์เทกซ์ใด การเรียนเชื่อมของเส้นเชื่อมโยงจะเขียนอยู่ในเครื่องหมายว่างเส้น

$$V(G) = \{a, b, c, d, e\}$$

$$E(G) = \{(a,b), (b,e), (a,c), (a,d), (a,e)\}$$

หรือ

$$E(G) = \{(b,a), (e,b), (c,a), (d,a), (e,a)\}$$

CP411107 Data Structures and Algorithms

## พื้นฐานทั่วไปของกราฟ

### ชนิดของกราฟ

#### 2) กราฟที่มีทิศทาง (Digraph)

กราฟแบบมีทิศทาง = กราฟที่มีเอดจ์เป็นหัวลูกศร ซึ่งแสดงทิศทางจากเวอร์เทกซ์หนึ่งไปยังอีกเวอร์เทกซ์หนึ่ง (Directed Graph)

**อาร์ค (Arcs)**

กรุงเทพ ขอนแก่น

จากภาพ มีเส้นทางจากกรุงเทพไปขอนแก่น  
ต.ไม่มีเส้นทางจากขอนแก่นไปกรุงเทพ

CP411107 Data Structures and Algorithms

## พื้นฐานทั่วไปของกราฟ

### ชนิดของกราฟ

#### 2) กราฟที่มีทิศทาง (Digraph)

$$V(G) = \{a, b, c, d\}$$

$$E(G) = \{<a,b>, <a,c>, <b,d>, <d,c>, <c,b>\}$$

CP411107 Data Structures and Algorithms

## พื้นฐานทั่วไปของกราฟ

### ชนิดของกราฟ

#### 2) กราฟที่มีทิศทาง (Digraph)

ภาพตัวอย่างกราฟแบบมีทิศทาง

$$V(G) = \{A, B, C, D, E, F, G\}$$

$$E(G) = \{<A,B>, <A,C>, <A,D>, <C,E>, <C,F>, <D,G>\}$$

CP411107 Data Structures and Algorithms

## พื้นฐานทั่วไปของกราฟ

### ชนิดของกราฟ

#### 2) กราฟที่มีทิศทาง (Digraph)

ภาพตัวอย่างกราฟแบบมีทิศทาง

$$V(G) = \{1, 2, 3, 4, 5\}$$

$$E(G) = \{<1,1>, <1,5>, <2,1>, <2,4>, <4,2>, <4,5>, <5,3>, <2,3>, <3,2>\}$$

**พื้นฐานทั่วไปของกราฟ**

ระดับขั้นเข้าและระดับขั้นออก

ระดับขั้นเข้า (In-degree) คือ จำนวนของเส้นที่เข้าไปยังเวอร์เทกซ์นั้น ๆ

ระดับขั้นออก(Out-degree) คือ จำนวนของเส้นที่ออกจากเวอร์เทกซ์นั้น ๆ

สามารถรับทราบการสอนพิเศษ วิทยาลัยการสอนพิเศษ มหาวิทยาลัยธรรมศาสตร์ มหาวิทยาลัยเชียงใหม่

14/38

**พื้นฐานทั่วไปของกราฟ**

ก) กราฟแบบไม่มีศักดิ์

เวอร์เทกซ์	ระดับขั้นเข้า	ระดับขั้นออก
a	2	2
b	2	2
c	2	2

ข) กราฟแบบมีศักดิ์

เวอร์เทกซ์	ระดับขั้นเข้า	ระดับขั้นออก
a	0	2
b	1	1
c	2	0

สามารถรับทราบการสอนพิเศษ วิทยาลัยการสอนพิเศษ มหาวิทยาลัยการสอนพิเศษ มหาวิทยาลัยเชียงใหม่

15/38

**พื้นฐานทั่วไปของกราฟ**

กราฟสมบูรณ์

กราฟที่ทุกเวอร์เทกซ์มีเส้นที่เขื่อน莫โยงไปยังเวอร์เทกซ์ที่เหลือทั้งหมด

ในการฟาร์มบูรณ์สามารถคำนวณจำนวนเส้นได้จาก  $N*(N-1)/2$

$3*2/2=3$

สามารถรับทราบการสอนพิเศษ วิทยาลัยการสอนพิเศษ มหาวิทยาลัยการสอนพิเศษ มหาวิทยาลัยเชียงใหม่

16/38

**พื้นฐานทั่วไปของกราฟ**

ก) กราฟสมบูรณ์แบบไม่มีศักดิ์

เวอร์เทกซ์	ระดับขั้นเข้า	ระดับขั้นออก
a	3	3
b	3	3
c	3	3
d	3	3

ข) กราฟสมบูรณ์แบบมีศักดิ์

เวอร์เทกซ์	ระดับขั้นเข้า	ระดับขั้นออก
a	3	3
b	3	3
c	3	3
d	3	3

สามารถรับทราบการสอนพิเศษ วิทยาลัยการสอนพิเศษ มหาวิทยาลัยการสอนพิเศษ มหาวิทยาลัยเชียงใหม่

17/38

**พื้นฐานทั่วไปของกราฟ**

สูตรหาจำนวนเส้นของกราฟมีศักดิ์  $= N * (N - 1)$

จากภาพที่ (ข) ซึ่งเป็นกราฟแบบมีศักดิ์ และจำนวนเวอร์เทกซ์ที่มีทั้งหมดเท่ากับ 4 เวอร์เทกซ์ จึงคำนวณหาจำนวนเส้นได้ดังนี้

สูตรหาจำนวนเส้นของกราฟมีศักดิ์  $= N * (N - 1)$

$$= 4 * (4 - 1)$$

$$= 4 * 3$$

$$= 12 \text{ เส้น}$$

สามารถรับทราบการสอนพิเศษ วิทยาลัยการสอนพิเศษ มหาวิทยาลัยการสอนพิเศษ มหาวิทยาลัยเชียงใหม่

19/38

**พื้นฐานทั่วไปของกราฟ**

เส้นทาง (Path) ของกราฟ

เส้นทาง (Path) ของกราฟ คือ ลำดับของเวอร์เทกซ์ที่ถูกเชื่อมต่อทั้งหมด โดยเริ่มตั้งแต่เวอร์เทกซ์แรกไปจนถึงเวอร์เทกซ์สุดท้ายหรือเวอร์เทกซ์ที่ต้องการ

เส้นทางจาก A ไป E  
P1 = (A, B, C, D, E) ความยาวของเส้นทางเท่ากับ 4

เส้นทางจาก A ไป H  
P2 = (A, B, C, F, G, H) ความยาวของเส้นทางเท่ากับ 5

สามารถรับทราบการสอนพิเศษ วิทยาลัยการสอนพิเศษ มหาวิทยาลัยการสอนพิเศษ มหาวิทยาลัยเชียงใหม่

20/38

CP411107 Data Structures and Algorithms

Graph

## พื้นฐานทั่วไปของกราฟ

กราฟที่มีน้ำหนัก (Weighted Graphs)

กราฟที่แต่ละเดคจ์จะมีค่าบวกกันเรื่องความหมายอย่างใดอย่างหนึ่ง เช่น ระยะทาง ความเร็ว เวลาเดินทาง ค่าโดยสาร เป็นต้น

```

graph LR
    A((ปทุมธานี)) --- B((กรุงเทพ)) [46]
    A --- C((ฉะเชิงเทรา)) [82]
    A --- D((นครนายก))
    B --- C [107]
    B --- D [100]
    C --- D [76]
    C --- E((สระบุรี)) [58]
    D --- E [29]
    D --- F((ปราจีนบุรี)) [29]
  
```

อาจารย์วิวัฒนาการสอนวิชาและ/orientation ให้เป็นการสอนที่มีโครงสร้าง ไม่ว่าในรายละเอียดแบบกัน

21/38

CP411107 Data Structures and Algorithms

Graph

## การดำเนินการของกราฟ

### การเพิ่มเวอร์ทีกซ์

Add Vertex

สามารถรับรายการของเพื่อนที่มาดู รับรายการของเพื่อนที่มาดู ไม่ว่าเพื่อนจะมีเพื่อนคนไหน

23/38

The diagram shows a graph with four nodes labeled a, b, c, and d. Nodes a and b are at the top, while c and d are at the bottom. They are connected by red lines forming a rectangle: a to b, b to c, c to d, and d to a. A small red circle is placed on the edge between nodes a and b. To the right of the graph is a rectangular box containing the text "Delete Vertex". Two arrows point from this box to the graph: one arrow points to the edge between nodes a and b (the edge with the red circle), and another arrow points to node d.

Graph

การดำเนินการของกราฟ

การลบเวอร์ทексทึ่งๆ

Delete Vertex

CP411107 Data Structures and Algorithms

Graph

## การดำเนินการของกราฟ

### การลบเอดจ์

The diagram shows a graph with four nodes labeled a, b, c, and d. Node a is connected to b and d. Node b is connected to a, c, and d. Node c is connected to b and d. Node d is connected to a, b, and c. A box labeled "Delete Edge" with the label  $\{b, c\}$  below it indicates the removal of the edge between nodes b and c. The resulting graph on the right has the same structure as the original except for the missing edge between b and c.

การดำเนินการของกราฟที่ไม่稠密 ไม่มีเพียงการลบกราฟ

26/38

CP411107 Data Structures and Algorithms Graph

### การดำเนินการของกราฟ

#### การค้นหาเวอร์เทกซ์

เป็นการเดินทางเข้าไปในกราฟ เพื่อทำการค้นหาเวอร์เทกซ์ที่ซึ่งหรือโนดที่ต้องการ หากพบ แจ้งคำแนะนำลับมา หากไม่พบแจ้งข้อผิดพลาด

27/38

CP411107 Data Structures and Algorithms Graph

### การสร้างกราฟด้วยเมทริกซ์

#### การแทนกราฟด้วยอาร์เรย์สองมิติ | | A | B | C | D | |---|---|---|---|---| | A | 0 | 1 | 1 | 1 | | B | 1 | 0 | 0 | 1 | | C | 1 | 0 | 0 | 1 | | D | 1 | 1 | 1 | 0 | 28/38

42:20 %

CP411107 Data Structures and Algorithms Graph

### การสร้างกราฟด้วยเมทริกซ์

#### การแทนกราฟ

	A	B	C	D	E
A	1	1	0	0	0
B	0	0	1	1	1
C	0	0	0	1	0
D	0	0	0	0	1
E	0	0	0	0	0

30/38

CP411107 Data Structures and Algorithms Graph

### การสร้างกราฟด้วยเมทริกซ์

#### การแทนกราฟ

	เชิงจราญ	อุบลราชธานี	กรุงเทพ
เชิงจราญ	0	1060	800
อุบลราชธานี	1060	0	630
กรุงเทพ	800	630	0

31/38

CP411107 Data Structures and Algorithms Graph

### การสร้างกราฟด้วยเมทริกซ์

#### การแทนกราฟด้วยลิงค์ลิสต์

เก็บ Adjacency List

	0	1	2	3
a	[a]	[1]	[2]	[3]
b	[b]	[0]	[2]	
c	[c]	[0]	[1]	[3]
d	[d]	[0]	[2]	

32/38

CP411107 Data Structures and Algorithms Graph

### การเดินทางเข้าไปในกราฟ

#### การท่องไปในกราฟ (Graph traversal)

การท่องไปในกราฟ (graph traversal) คือ กระบวนการเดินทางเข้าไปเยือนโนดในกราฟ โดยมีหลักในการทำงานคือ แคลลิโภนจะถูกเก็บไว้ในหนึ่งเดียว

- สำหรับการเดินทางในกราฟเพื่อยืนยันผลลัพธ์หนึ่งจะมีลักษณะเดียวกัน
- และในกราฟจะมีเส้นทางที่ซ้ำกัน แต่ละเส้นทางจะมีเส้นทางเดียว

เดินทางที่ซ้ำเดิมจะเป็นปัจจัยที่รบกวนการทำงานของกราฟ

และเมื่อเดินทางเสร็จแล้ว ให้รีเซ็ตค่าให้เป็น 0 ให้ลูปเดินทางใหม่

33/38

CP411107 Data Structures and Algorithms Graph

### การเดินทางเข้าไปในกราฟ

สำหรับเทคโนโลยีการเดินทางในกราฟมี 2 แบบดังนี้

การท่องแบบกราฟ (Breadth first traversal) วิธีนี้ทำโดยเลือกโหนดที่เป็นจุดเริ่มต้น คือมาให้ย้อนโหนดอื่นที่ใกล้กันที่สุดที่ไม่เคยถูกสำรวจ จนกว่าจะไม่เหลือโหนดที่ไม่ได้สำรวจ ตัวอย่างแสดงเรื่องของการท่องแบบกราฟที่ลงในหน้านี้ดังนี้

การท่องไปในแนวว้าว A B C D E F

ผลลัพธ์ที่ได้ สามารถใช้ในการทดสอบ ในการเขียนโปรแกรมที่ต้องการเดินทางในกราฟ มากำหนดเส้นทาง 34/38

CP411107 Data Structures and Algorithms Graph

### การเดินทางเข้าไปในกราฟ

การท่องไปในกราฟแนวว้าว

การท่องไปในแนวว้าว = A G B H E C F D

ผลลัพธ์ที่ได้ สามารถใช้ในการทดสอบ ในการเขียนโปรแกรมที่ต้องการเดินทางในกราฟ มากำหนดเส้นทาง 35/38

CP411107 Data Structures and Algorithms Graph

### การเดินทางเข้าไปในกราฟ

การท่องแบบลึก (Depth first traversal)

การท่องแบบลึกนักบุญก่อกราฟที่ไม่ระดับของกราฟ โดยก้าวนะเริ่มที่โหนดที่ไม่เคยถูกสำรวจแล้วเดินทางไปยังโหนดเดียวที่มีเส้นเชื่อมต่อจากโหนดปัจจุบัน จนกว่าจะไม่พบโหนดที่ไม่เคยถูกสำรวจ 再次进入此节点时，将回溯到最近的尚未访问过的节点。 从该节点继续搜索直到所有可能的路径都被探索或遇到一个已经访问过的节点。 一旦遇到一个已经访问过的节点，就停止搜索并返回到上一个尚未访问过的节点。 重复此过程直到所有可能的路径都被探索或遇到一个已经访问过的节点。

A B E C F D

ผลลัพธ์ที่ได้ สามารถใช้ในการทดสอบ ในการเขียนโปรแกรมที่ต้องการเดินทางในกราฟ มากำหนดเส้นทาง 36/38

CP411107 Data Structures and Algorithms Graph

### การเดินทางเข้าไปในกราฟ

Spanning Tree คือต้นไม้ที่ประกอบด้วยโหนดทุกโหนดของกราฟ โดยแต่ละรากของ โหนดจะต้องมีลักษณะเดียวกัน นั่นคือไม่มี loop หรือ cycle

- สมมติสถานการณ์ให้กราฟแสดงเส้นทางบินระหว่าง 7 เมือง แต่ด้วยเหตุผลทางธุรกิจทำให้ต้องปรับเส้นทางการบินไปใหม่ที่สุดและแข็งแกร่งสามารถเดินทางได้ทุกแห่ง

ผลลัพธ์ที่ได้ สามารถใช้ในการทดสอบ ในการเขียนโปรแกรมที่ต้องการเดินทางในกราฟ มากำหนดเส้นทาง 37/38

CP411107 Data Structures and Algorithms Graph

### สรุป

- พื้นฐานทั่วไปของกราฟ
- การดำเนินการของกราฟ
- การสร้างกราฟด้วยเมทริกซ์
- การสร้างกราฟด้วยจิงค์ลิส
- กราฟระบุที่หนึ้น
- การเดินทางเข้าไปในกราฟ

ผลลัพธ์ที่ได้ สามารถใช้ในการทดสอบ ในการเขียนโปรแกรมที่ต้องการเดินทางในกราฟ มากำหนดเส้นทาง 38/38

CP411107 Data Structures and Algorithms

### การบ้าน

1. จงเขียนโปรแกรมสร้างกราฟได้ตามการใช้อาร์เรย์ขนาด  $4 \times 4$  โดยเรียบฟังก์ชันเพิ่ม ลบ เออดจ์ และแสดงผลลัพธ์ของเออดจ์