

Graph

โครงสร้างข้อมูลและอัลกอริทึมเบื้องต้น 305214 / 235012

1

Topic

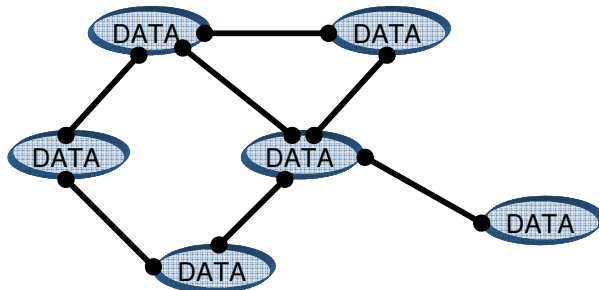
- ☐ Graph Definition
- ☐ Graph Type
- ☐ Graph Representation
- ☐ Graph Operation

2

Graph

กราฟ (graph) เป็นโครงสร้างข้อมูลที่มีคุณสมบัติ

- เป็นข้อมูลชนิดเดียวกัน
- ไม่เป็นข้อมูลลำดับเชิงเส้น (non-linear)
- มีสมาชิกตัวถัดไปและก่อนหน้าได้มากกว่าหนึ่งตัว

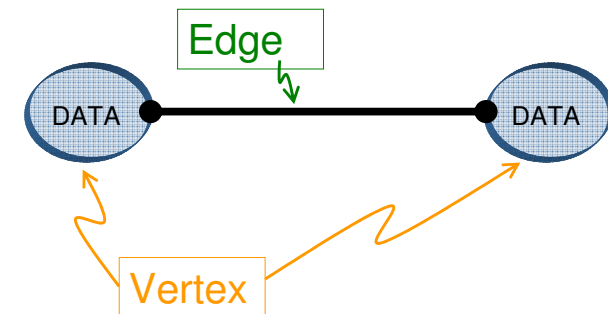


3

Graph

ส่วนประกอบของกราฟ

- Vertex - โหนดของกราฟ
- Edge - ส่วนที่เชื่อมโยงโหนดเข้าด้วยกัน



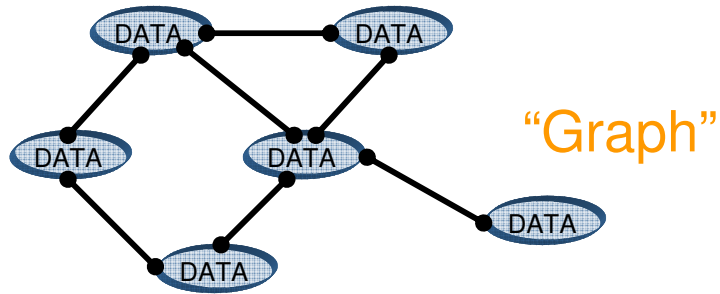
4

Graph

นิยามของกราฟ

“กลุ่มของโหนดที่เชื่อมโยงกันด้วยกลุ่มของเส้น”

- กลุ่มของโหนดเรียกว่า “vertex”
- กลุ่มของเส้นเรียกว่า “edge”



5

Graph

นิยามของกราฟ

กำหนด G เป็นกราฟ มี V เป็น vertex และมี E เป็น edge จะได้ว่า

$$G = (V, E)$$

โดยที่

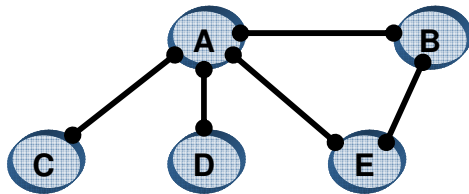
$V(G)$ คือเซตของ vertex ไม่เป็นเซตว่างและมีจำนวนจำกัด

$E(G)$ คือเซตของ Edge ซึ่งเขียนด้วยคู่อันดับของ vertex

6

Graph

ตัวอย่างกราฟ



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

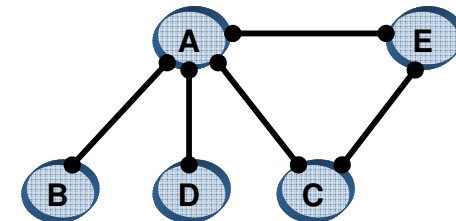
$$E(G) = \{(A,B), (A,C), (A,D), (A,E), (B,E)\}$$

7

Graph

ตัวอย่างกราฟ

- พิจารณา binary tree



Father มีลูกได้มากกว่า 2 \rightarrow General Tree

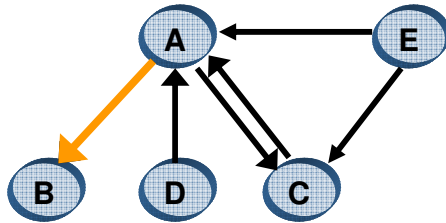
ลูกมี Father ได้มากกว่า 1 \rightarrow Graph

8

Graph Type

Directed Graph / Digraph

■ กราฟแบบมีทิศทาง



โหนด A กับ B

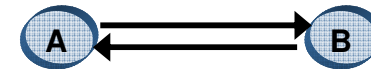
มี Edge จาก A ไปยัง B แต่ไม่มี Edge จาก B ไปยัง A

9

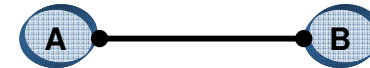
Graph Type

Directed Graph / Digraph

■ กราฟแบบมีทิศทาง



■ กราฟแบบไม่มีทิศทาง



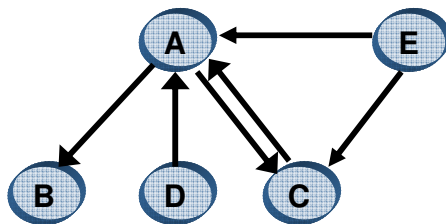
Edge ของ non-directed graph เทียบเท่ากับ
directed graph ที่มี edge ระหว่างโหนดทั้งไปและกลับ

10

Graph Type

Directed Graph / Digraph

■ การเขียนแฉง edge E(G)



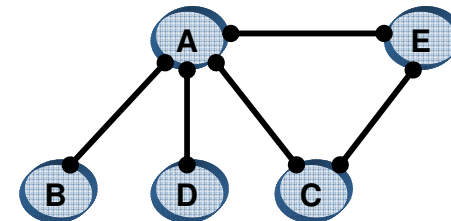
$$E(G) = \{(A,B), (A,C), (C,A), (D,A), (E,A), (E,C)\}$$

11

Graph Type

non-Directed Graph

■ การเขียนแฉง edge E(G)



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

or

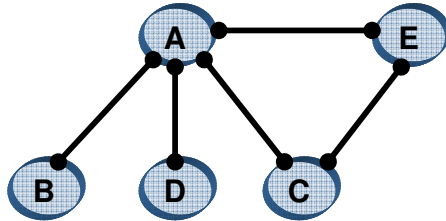
$$E(G) = \{(B,A), (C,A), (D,A), (E,A), (E,C)\}$$

12

Graph Type

Indegrees / Outdegrees

- Indegrees – จำนวน edge ที่เข้ามายัง vertex
- Outdegree – จำนวน edge ที่ออกไปจาก vertex



A → indegree = 4 , outdegree = 4

C → indegree = 2 , outdegree = 2

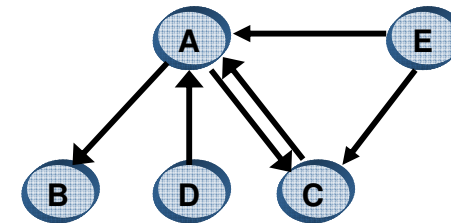
* Non-directed graph indegree = outdegree

13

Graph Type

Directed Graph / Digraph

- Indegree / Outdegree



A → indegree = 3 , outdegree = 2

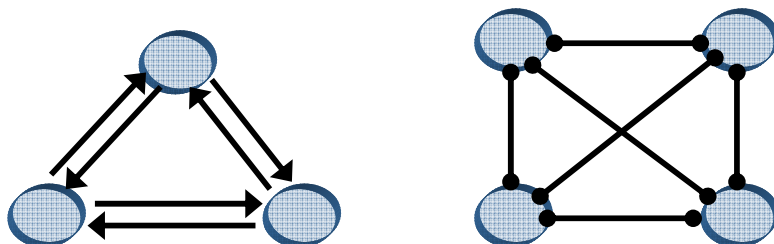
C → indegree = 2 , outdegree = 1

14

Graph Type

Complete Graph

- ทุก vertex มี edge เชื่อมไปยัง vertex ที่เหลือทั้งหมด



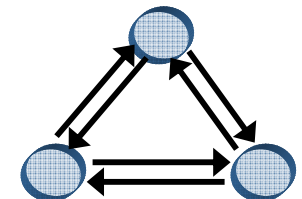
15

Graph Type

Complete Graph

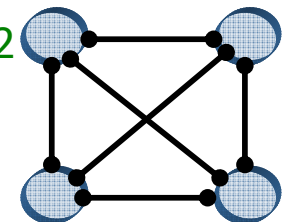
- Directed Graph

จำนวน Edge = $n * (n - 1)$
เมื่อ n เป็นจำนวน vertex



- Non-Directed Graph

จำนวน Edge = $(n * (n - 1)) / 2$
เมื่อ n เป็นจำนวน vertex

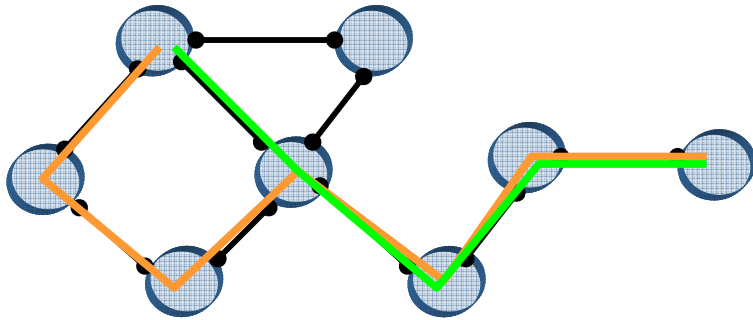


16

Graph Type

Path

- เส้นทาง – ลำดับของ vertex ซึ่งเรียกกันจาก vertex หนึ่งถึง vertex หนึ่งโดยเรียงกันตาม edge ที่เชื่อมต่อ

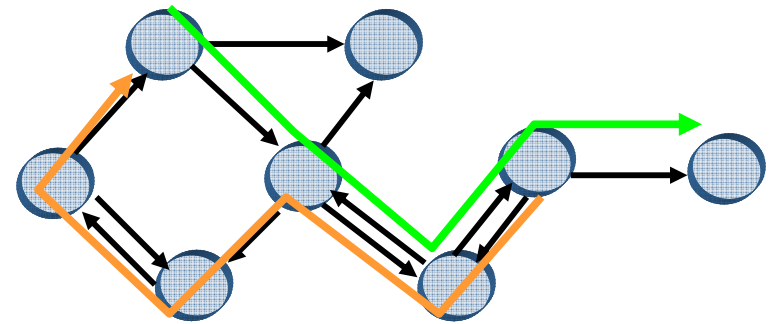


17

Graph Type

Directed Graph Path

- Path ของ directed graph ขึ้นอยู่กับทิศทางของ Edge ด้วย

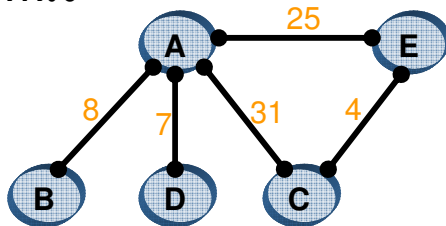


18

Graph Type

Weight Graph

- Graph ที่ edge มีค่ากำกับเป็นอย่างไรอย่างหนึ่ง



$A \rightarrow E \rightarrow C$ Weight = 29

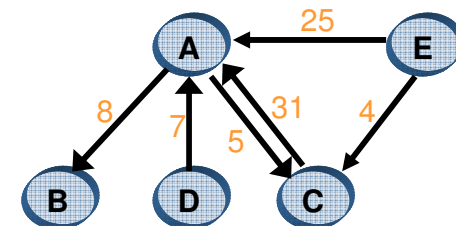
$A \rightarrow C$ Weight = 31

19

Graph Type

Weight Graph

- น้ำหนัก (weight) ของกราฟแบบมีทิศทาง



$A \rightarrow C$

Weight = 5

$C \rightarrow A$

Weight = 31

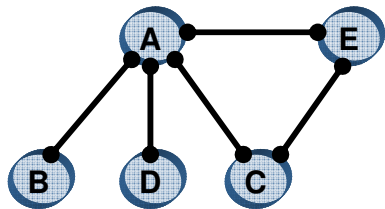
มีค่ากำกับของแต่ละทิศทาง ค่าไปและกลับไม่จำเป็นต้องเท่ากัน

20

Graph Representation

การแทนค่ากราฟ

- ใช้ array ในการเก็บค่า



มี edge → 1
ไม่มี edge → 0

การแทนค่ากราฟไม่มีทิศทาง
ด้วย array 2 มิติ

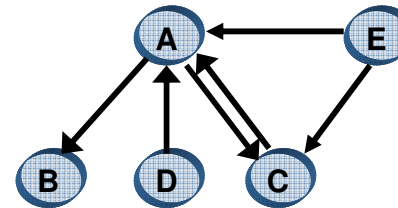
	A	B	C	D	E
A	0	1	1	1	1
B	1	0	0	0	0
C	1	0	0	0	1
D	1	0	0	0	0
E	1	0	1	0	0

21

Graph Representation

การแทนค่ากราฟ

- การใช้ array แทนค่ากราฟแบบมีทิศทาง



แทนจาก row ไป column

การแทนค่ากราฟมีทิศทาง
ด้วย array 2 มิติ

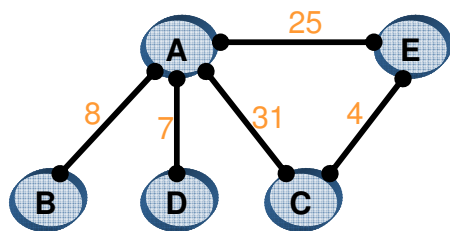
	A	B	C	D	E
A	0	1	1	0	0
B	0	0	0	0	0
C	1	0	0	0	0
D	1	0	0	0	0
E	1	0	1	0	0

22

Graph Representation

การแทนค่ากราฟ

- แทนค่ากราฟไม่มีทิศทางแบบมีน้ำหนัก



ใช้น้ำหนักแทนค่า 1

การแทนค่ากราฟไม่มีทิศทาง
ด้วย array 2 มิติ

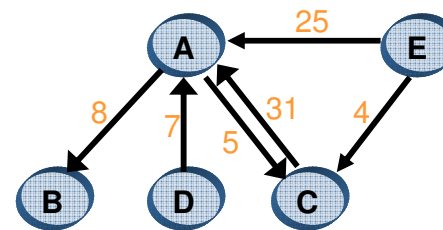
	A	B	C	D	E
A	0	8	31	7	25
B	8	0	0	0	0
C	31	0	0	0	4
D	7	0	0	0	0
E	25	0	4	0	0

23

Graph Representation

การแทนค่ากราฟ

- การใช้ array แทนค่ากราฟแบบมีน้ำหนักและทิศทาง



ใช้น้ำหนักแทนค่า 1

การแทนค่ากราฟมีทิศทาง
ด้วย array 2 มิติ

	A	B	C	D	E
A	0	8	5	0	0
B	0	0	0	0	0
C	31	0	0	0	0
D	7	0	0	0	0
E	25	0	4	0	0

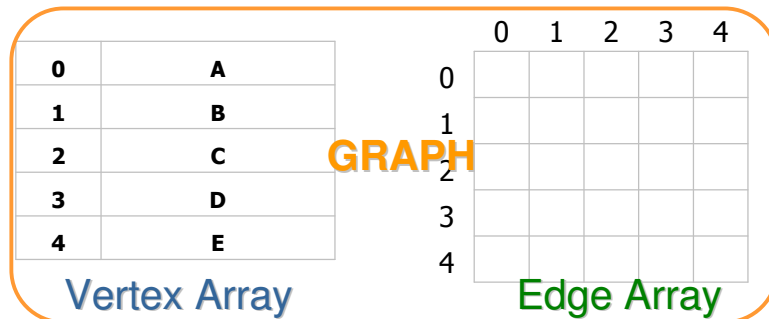
24

Graph Representation

การประกาศตัวแปรเพื่อใช้แทนค่ากราฟ

ประกอบไปด้วย array 2 ส่วน

- ส่วนที่ใช้เก็บค่า Vertex เป็น array มิติเดียว
- ส่วนที่ใช้เก็บค่า Edge เป็น array 2 มิติ



25

Graph Representation

การประกาศตัวแปรเพื่อใช้แทนค่ากราฟ

Vertex Array

- มีขนาดเท่ากับจำนวนโหนด (หรือจำนวนโหนดที่มากที่สุด-เผื่อไว้สำหรับการเพิ่ม)
- ชนิดข้อมูลเป็นชนิดของข้อมูลของโหนดกราฟ

0	A
1	B
2	C
3	D
4	E

26

Graph Representation

การประกาศตัวแปรเพื่อใช้แทนค่ากราฟ

Edge Array

- มีขนาดเท่ากับขนาดของ vertex array กำลังสอง
- ชนิดข้อมูลเป็นชนิดของข้อมูลของน้ำหนัก Edge

	0	1	2	3	4
0					
1					
2					
3					
4					

27

Graph Representation

การประกาศตัวแปรเพื่อใช้แทนค่ากราฟ

```
typedef struct{
    int VertexCount;
    char VertexList[max_vertices];
} VertexListType;
```

```
typedef struct{
    VertexListType vertices;
    int edge[max_vertices][max_vertices];
} GraphType;
```

28

Graph Representation

การประกาศตัวแปรเพื่อใช้แทนค่ากราฟ

```
GraphType Graph;
```

Vertex Array การอ้างอิงค่า vertex

```
Graph.vertices.VertexList[n]
```

Edge Array การอ้างอิงค่าน้ำหนัก edge

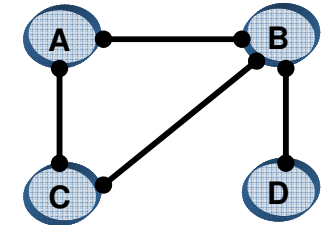
```
Graph.edge[n][m]
```

29

Graph Operation

การดำเนินการกับกราฟ

- การแทรก vertex
- การลบ vertex
- การแทรก edge
- การลบ edge
- การค้นหา vertex

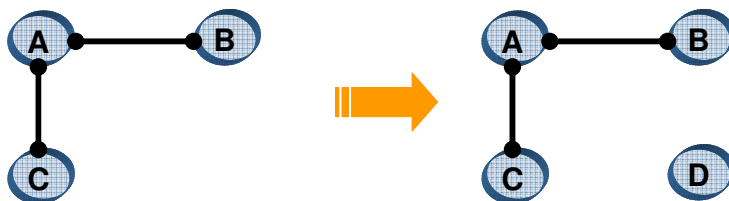


30

Graph Operation

การแทรก vertex

- เป็นการเพิ่ม vertex ลงไปในกราฟ
- กระบวนการยังไม่เสร็จสิ้นจนกว่าจะกระทำการแทรก edge ที่เชื่อมต่อกับ vertex ใหม่

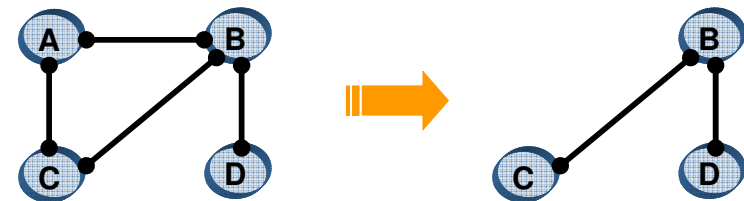


31

Graph Operation

การลบ vertex

- เป็นการลบ vertex ออกจากกราฟ
- ต้องลบ edge ที่เชื่อมต่อกับ vertex นั้นทั้งหมด

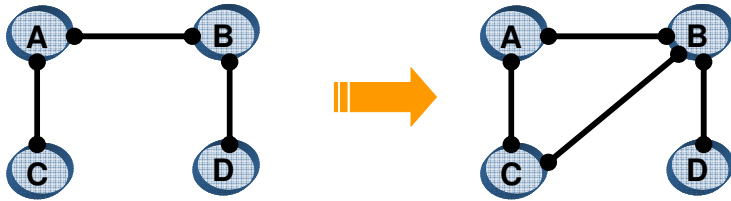


32

Graph Operation

การแทรก edge

- เป็นการเพิ่ม edge เข้าไปในกราฟ
- edge ต้องเชื่อมต่อเพิ่มเติมระหว่าง vertex สองอันใด ๆ

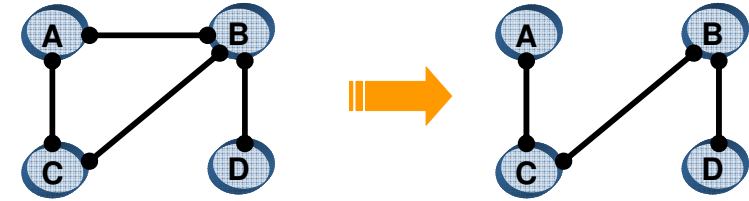


33

Graph Operation

การลบ edge

- เป็นการลบ edge ออกจากกราฟ
- อาจส่งผลให้ vertex ขาดออกจากกราฟได้

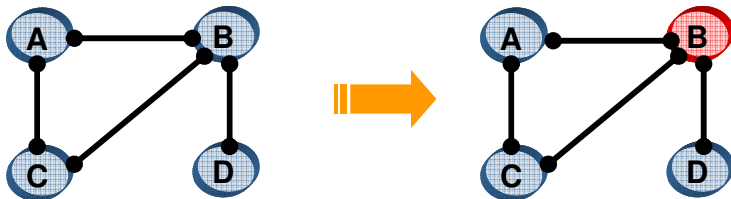


34

Graph Operation

การค้นหา vertex

- ใช้ค้นหา vertex ที่สอดคล้องกับค่าที่กำหนด
- ขึ้นอยู่กับวิธีการท่องกราฟ



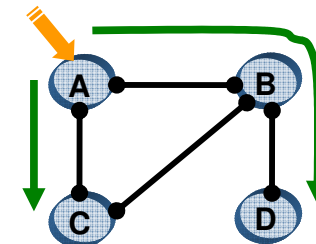
35

Graph Operation

Graph Traversal การท่องกราฟ

Depth-First Search การท่องไปแนวลึกก่อน

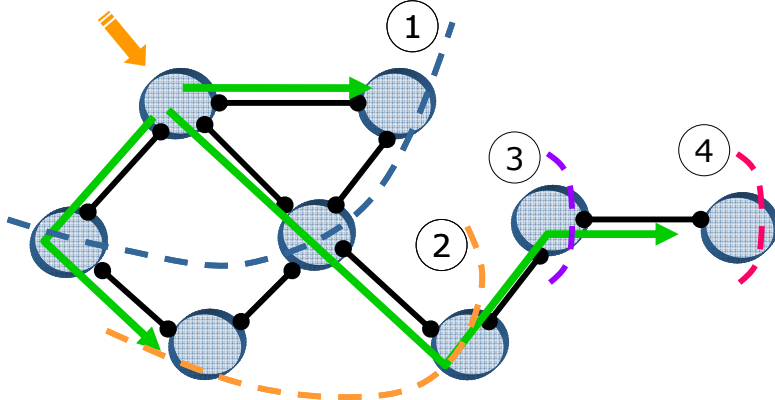
- ลงไปในแนวลึกสุดก่อนไปในแนวกว้าง (โหนดถัดไป)



36

Graph Operation

Depth-First Search



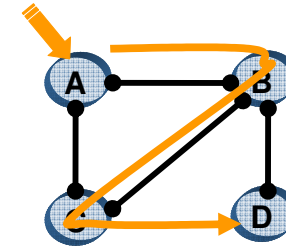
37

Graph Operation

Graph Traversal การท่องกราฟ

Breadth-First Search การท่องไปแนว
กว้างก่อน

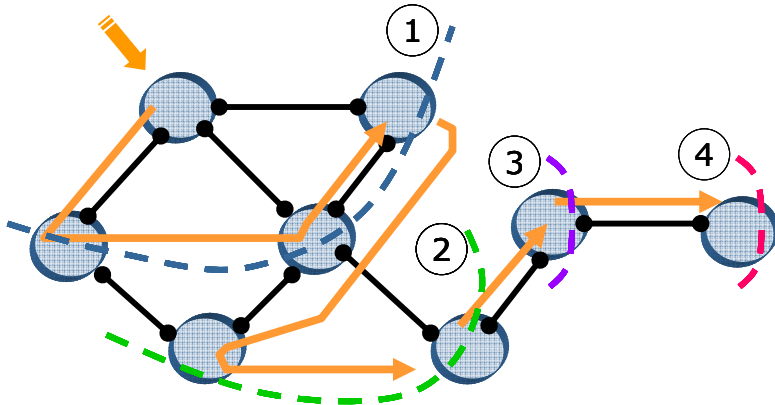
■ ลงไปในแนวกว้างให้หมดก่อนลงไป
ในแนวลึก



38

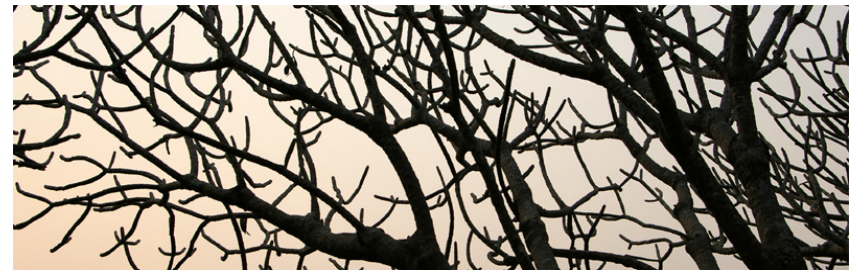
Graph Operation

Breadth-First Search



39

คำถามข้อสงสัย



40