254383 Algorithm Design and Analysis

อ.พิเศษพงศ์ สุธาพันธ์
ภาควิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ

fppt.com

สังเขปรายวิชา 254383 การออกแบบและวิเคราะห์อัลกอริทึม (Algorithm Design and Analysis)

จำนวนหน่วยกิต: 3 หน่วยกิต *(ภาคบรรยาย 2 ชั่วโมง/สัปดาห์ ภาคปฏิบัติ 2 ชั่วโมง/สัปดาห์ นอกเวลา 5 ชั่วโมง/สัปดาห์)*

คำอธิบายรายวิชา: แนวคิดเบื้องต้นเกี่ยวกับการออกแบบและวิเคราะห์ อัลกอริทึม สัญลักษณ์เชิงเส้นกำกับ การวิเคราะห์เชิงเส้นกำกับ อัลกอริทึมแบ่งแยกและเอาชนะ การโปรแกรมแบบพลวัต อัลกอริทึมเชิงละโมบ เอ็นพีบริบูรณ์

fppt.com

สังเขปรายวิชา 254383 การออกแบบและวิเคราะห์อัลกอริทึม (Algorithm Design and Analysis)

กิจกรรมการเรียนการสอน

1. บรรยาย ซักถาม อภิปราย
2. รายงาน
3. วิเคราะห์ตัวอย่างอัลกอริทึม
4. แบบฝึกหัด

การวัดผล

1. แบบฝึกหัดและรายงาน 50%
2. สอบกลางภาค (Mid-Term) 20%
3. สอบปลายภาค (Final) 20%
4. การเข้าชั้นเรียนและแสดงความคิดเห็น 10%

คะแนนรวม

%100

fppt.com

ทำไมยังต้องเรียน ในเมื่อมี AI?

3

เสาหลักสำคัญ

1. การตรวจสอบ (Verification): AI เขียนโค้ดได้ แต่ "มนุษย์" ต้องวิเคราะห์ว่ามันทำงานได้ถูกต้องและปลอดภัยจริงหรือไม่

2. Algorithm Efficiency **:** โค้ดที่ทำงานได้ ไม่เท่ากับโค้ดที่ "ประหยัดทรัพยากร" การวิเคราะห์ Big O ช่วยลดค่าใช้จ่ายระบบ Cloud

3. การแก้ไขปัญหาเชิงลึก (Deep Logic): ปัญหาใหม่ที่ AI ยังไม่เคยพบ ต้องการมนุษย์ที่มีตรรกะระดับสูงในการวางโครงสร้าง

fppt.com

Week 1

แนวคิดเบื้องต้นเกี่ยวกับการออกแบบและวิเคราะห์อัลกอริทึม

อ.พิเศษพงศ์ สุธาพันธ์
ภาควิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ

fppt.com

อัลกอริทึม(Algorithm)

Algorithm คือ อะไร?

ชุดคำสั่งที่ความชัดเจนและมีจัดเรียงลำดับคำสั่งในการทำงานแล้ว เพื่อใช้ในการแก้ไขปัญหาหรือหาคำตอบ

fppt.com

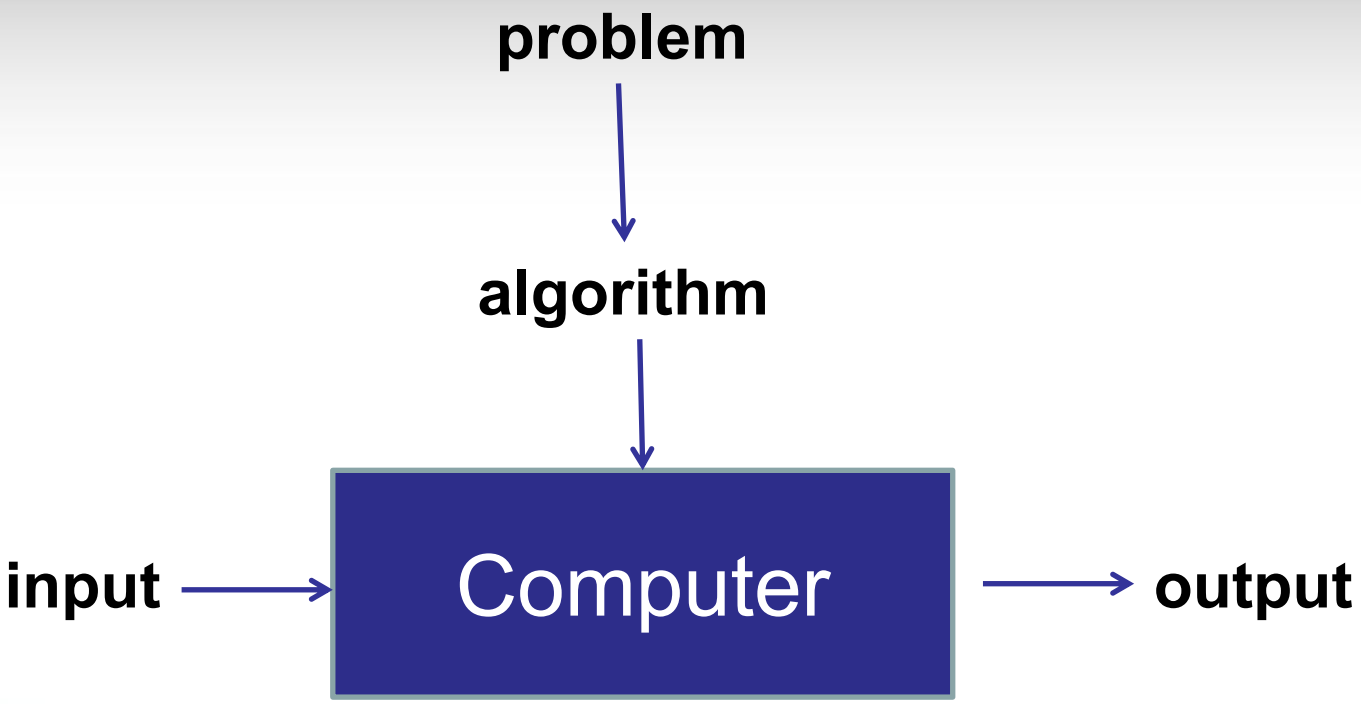
อัลกอริทึมคืออะไร?

Algorithm คือ ชุดคำสั่งที่ชัดเจนและเรียงลำดับขั้นตอนอย่างเป็นระบบ เพื่อแก้ไขปัญหาหรือหาคำตอบภายใต้เงื่อนไขที่กำหนด

"ในยุคนี้ อัลกอริทึมคือพิมพ์เขียวที่สั่งการให้ทั้งโปรแกรมเมอร์และ AI ทำงานได้อย่างแม่นยำ"

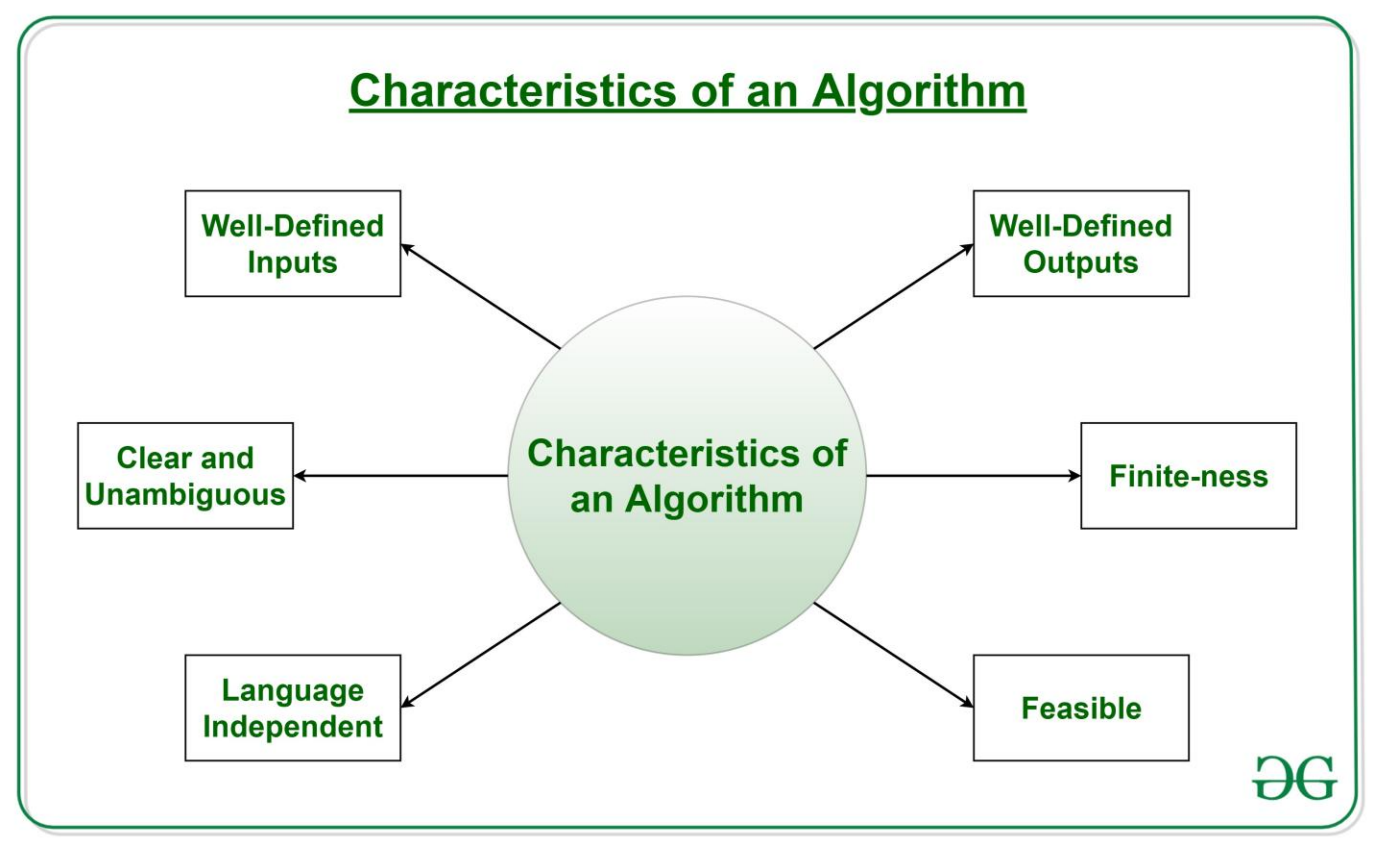
fppt.com

อัลกอริทึม(Algorithm)



อัลกอริทึม(Algorithm)

คุณสมบัติของอัลกอริทึม



https://www.geeksforgeeks.org/introduction-to-algorithms/

fppt.com

อัลกอริทึม(Algorithm)

- **Clear and Unambiguous**: ชัดเจนไม่คลุมเครือ
- **Well-Defined Inputs**: กำหนด input ชัดเจน
- **Well-Defined Outputs:** กำหนด output ชัดเจน
- **Finite-ness:** ต้องมีที่สิ้นสุด , ไม่ควรติด loop หรือ infinite คือ ทำงานไม่มีที่สิ้นสุด
- **Feasible:** เรียบง่ายไม่ยึดติดกับเทคโนโลยีใด
- **Language Independent:** ไม่ยึดติดกับภาษา

fppt.com

https://www.geeksforgeeks.org/introduction-to-algorithms/

อัลกอริทึม(Algorithm)

แนวคิดของอัลกอริทึม

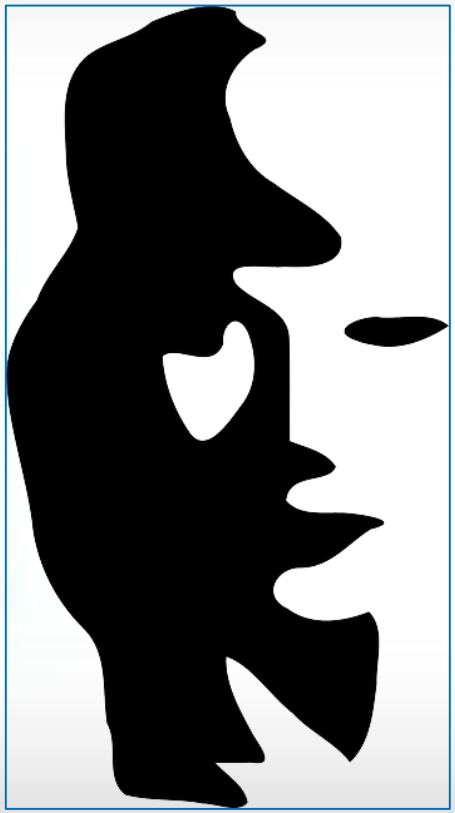
- มีการทำงานเป็นขั้นตอน
- ไม่มีความคลุมเครือ
- input ต้องเหมาะสมกับการทำงาน
- อัลกอริทึมหลายตัวที่สามารถแก้ปัญหาที่เหมือนกัน
  - แนวคิดที่ใช้ต่างกัน
  - ความเร็วต่างกัน
  - การใช้ทรัพยากรต่างกัน

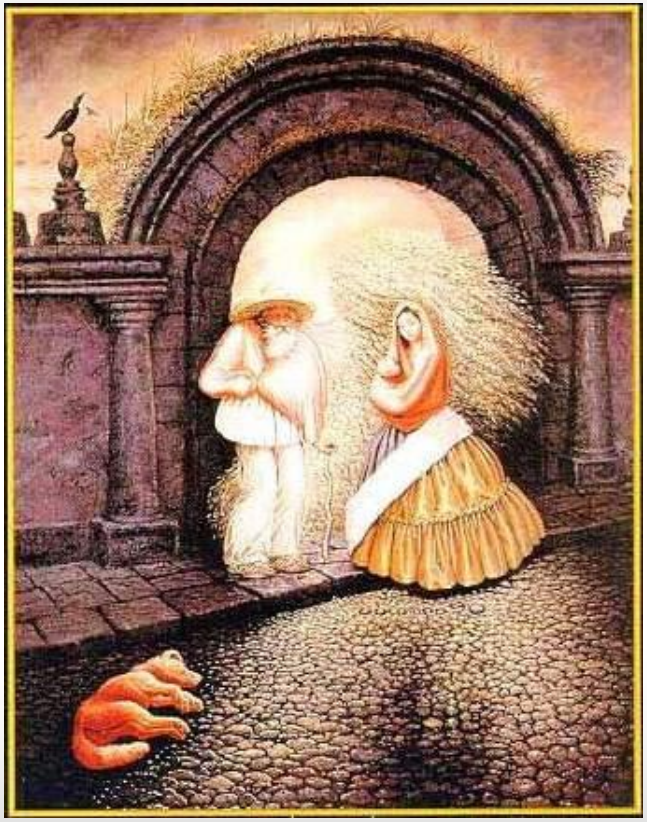
อัลกอริทึม(Algorithm)

- มีการทำงานเป็นขั้นตอน

อัลกอริทึม(Algorithm)

- ไม่มีความคลุมเครือ





อัลกอริทึม(Algorithm)

- **input ต้องเหมาะสมกับการทำงาน**

เรียงข้อมูล	5	10	120	15	3	45
แบบ 1	3	5	10	15	45	120
แบบ 2	10	120	15	3	45	5

อัลกอริทึม(Algorithm)

การหาตัวหารร่วมมาก (Greatest common divisor)

ตัวอย่าง หาตัวหารร่วมมากของ 32,18

คำตอบคือ 2

สามารถใช้อัลกอริทึมในการหา ได้หลายอัลกอริทึม

อัลกอริทึม(Algorithm)

ตย.1 Euclid's algorithm (แบบ Structured description)

$$\boxed{\gcd(m,n) = \gcd(n, m \bmod n)}$$

Step 1 if n = 0, return the value of m as the answer and stop; otherwise, proceed to Step 2.

Step 2 Divide m by n and assign the value of the remainder to r.

Step 3 Assign the value of n to m and the value of r to n. Go to Step 1.

อัลกอริทึม(Algorithm)

ตย.2 Euclid's algorithm (แบบ Pseudocode)

```
Euclid(m,n)
While n != 0 do
        r = m mod n
        m =  n
        n = r
return m
```

อัลกอริทึม(Algorithm)

ตย.3 Consecutive integer checking algorithm for gcd(m,n)

Step 1 Assign the value of min(m,n) to t.

Step 2 Divide m by t. If the remainder of this division is 0, go to Step 3; otherwise, go to Step 4.

Step 3 Divide n by t. If the remainder of this division is 0, return the value of t as the answer and stop; otherwise, process to Step 4.

Step 4 Decrease the value of t by 1. Go to Step 2.

อัลกอริทึม(Algorithm)

ตย.4 Middle-school procedure for gcd(m,n)

Step 1 Find the prime factors of m.

Step 2 Find the prime factors of n.

Step 3 Identify all the common factors in the two prime expansions found in Step 1 and Step 2.

Step 4 Compute the product of all the common factors and return it as the greatest common divisor of the number given.

อัลกอริทึม(Algorithm)

ตัวอย่าง **Algorithm Sieve(n)**

ใช้ในการหา prime number ที่มีค่าน้อยกว่าหรือเท่ากับ n

https://en.wikipedia.org/wiki/Sieve_of_Eratosthenes

อัลกอริทึม(Algorithm)

ตัวอย่าง Algorithm Sieve(n)

```
for p = 2 to n do a[p] = p
for p = 2 to floor(sqrt(n)) do
        if a[p] != 0
                j = p*p
                while j <= n do
                        a[j] = 0
                        j = j + p
i = 0
```

```
for p  = 2 to n do
        if a[p] != 0
                l[i] = a[p]
                i = i + 1
return l
```

วงจรการแก้ปัญหา (6 Steps)

Understand	Decide	Design	Prove	Analyze	Code
เข้าใจปัญหา	เลือกวิธีการ	ออกแบบ	พิสูจน์ผล	วิเคราะห์	เขียนโปรแกรม

fppt.com

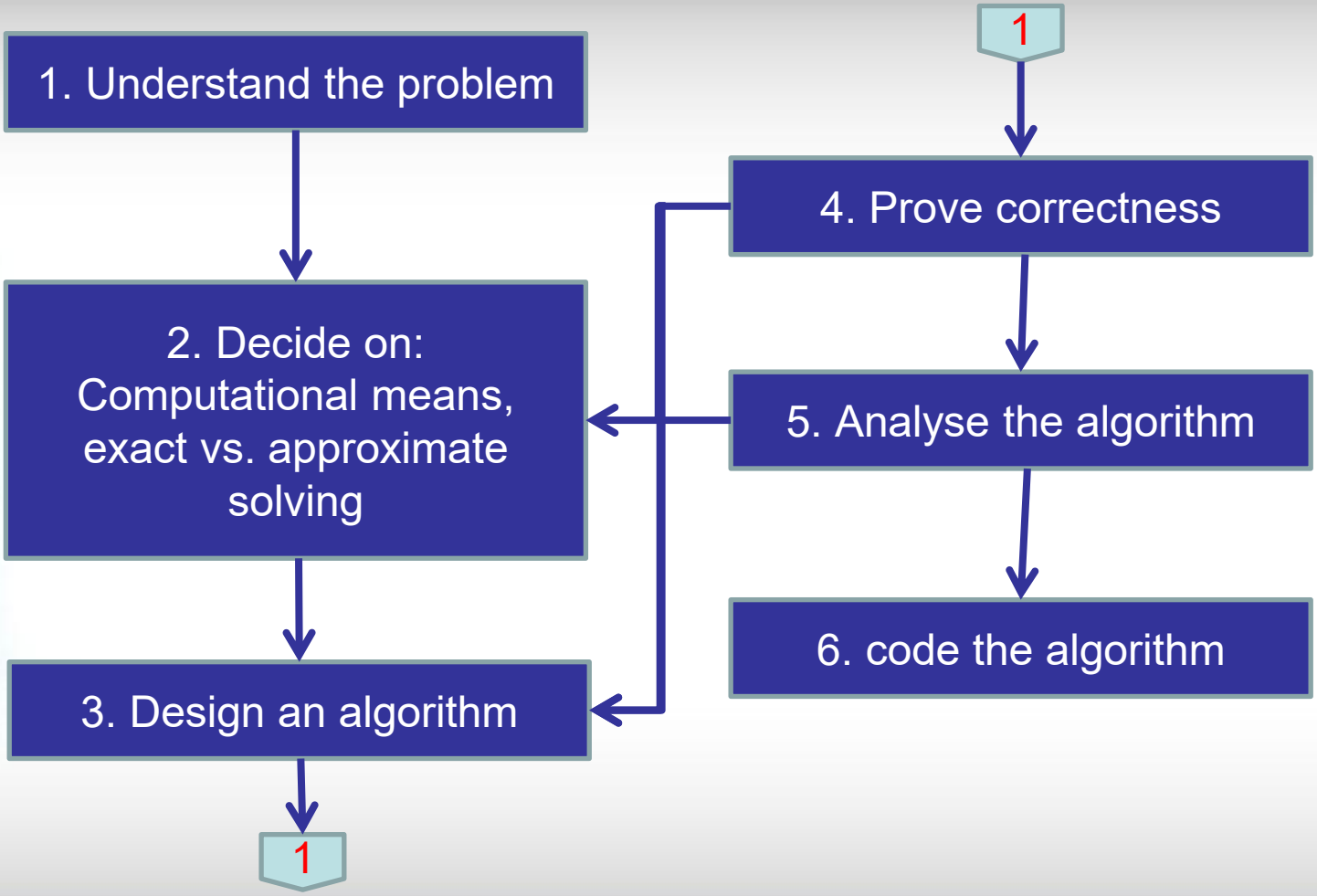
การออกแบบอัลกอริทึม

1. ต้องรู้ปัญหาที่จะแก้โดยอัลกอริทึมนี้
2. เงื่อนไขและข้อกำหนดต่าง ๆของปัญหาที่ต้องใช้ในขณะทำการแก้ไขปัญหา
3. อินพุต ที่จะใช้ในการแก้ปัญหา
4. เอาต์พุต ที่คาดว่าจะได้เมื่อปัญหาถูกแก้ไขแล้ว
5. โซลูชั่น ที่ใช้แก้ปัญหา ภายใต้เงื่อนไขที่กำหนด

fppt.com

https://www.geeksforgeeks.org/introduction-to-algorithms/

หลักการพื้นฐานของวิธีการแก้ไขปัญหา



fppt.com

หลักการพื้นฐานของวิธีการแก้ไขปัญหา

- ทำความเข้าใจปัญหา (1)
- สืบหาความสามารถของอุปกรณ์ที่ใช้คำนวณ (2)
- เลือกวิธีการแก้ปัญหาระหว่าง Exact และ Approximate (2)
- เทคนิคการออกแบบอัลกอริทึม (3)

fppt.com

หลักการพื้นฐานของวิธีการแก้ไขปัญหา (ต่อ)

- การออกแบบอัลกอริทึมและโครงสร้างข้อมูล (3)
- วิธีการระบุอัลกอริทึม (Pseudocode , Flowchart) (3)
- พิสูจน์ความถูกต้องของอัลกอริทึม (4)
- วิเคราะห์อัลกอริทึม (5)
- เขียนโค๊ตจากอัลกอริทึม (6)

fppt.com

ทำไมเราต้อง "วิเคราะห์" อัลกอริทึม?

เมื่อข้อมูล (n) เพิ่มขึ้น อัลกอริทึมที่ไม่มีประสิทธิภาพจะทำให้ระบบล่มหรือเปลืองต้นทุน Cloud อย่างมหาศาล

fppt.com

โครงสร้างข้อมูล: หัวใจของอัลกอริทึม

โครงสร้างข้อมูล	ตัวอย่างการใช้งานจริง	ความสำคัญ
Linear (Array/List)	Database Records, Undo Systems	เข้าถึงข้อมูลลำดับ
Trees (Binary/B-Tree)	File Systems, Search Indexes	ค้นหาข้อมูลความเร็วสูง
Graphs	GPS Navigation, Social Networks	วิเคราะห์ความสัมพันธ์
Dictionaries / Maps	Caches, JSON Key-Value	เรียกใช้ข้อมูลทันที

fppt.com

ชนิดปัญหาที่สำคัญ

- **Sorting**
- **Searching**
- **String Processing**
- **Graph problems** (TSP, graph-coloring)
- **Combinatorial problems**
- **Geometric problems** (closest-pair, convex-hull)
- **Numerical problems**

fppt.com

String Processing

ต.ย.

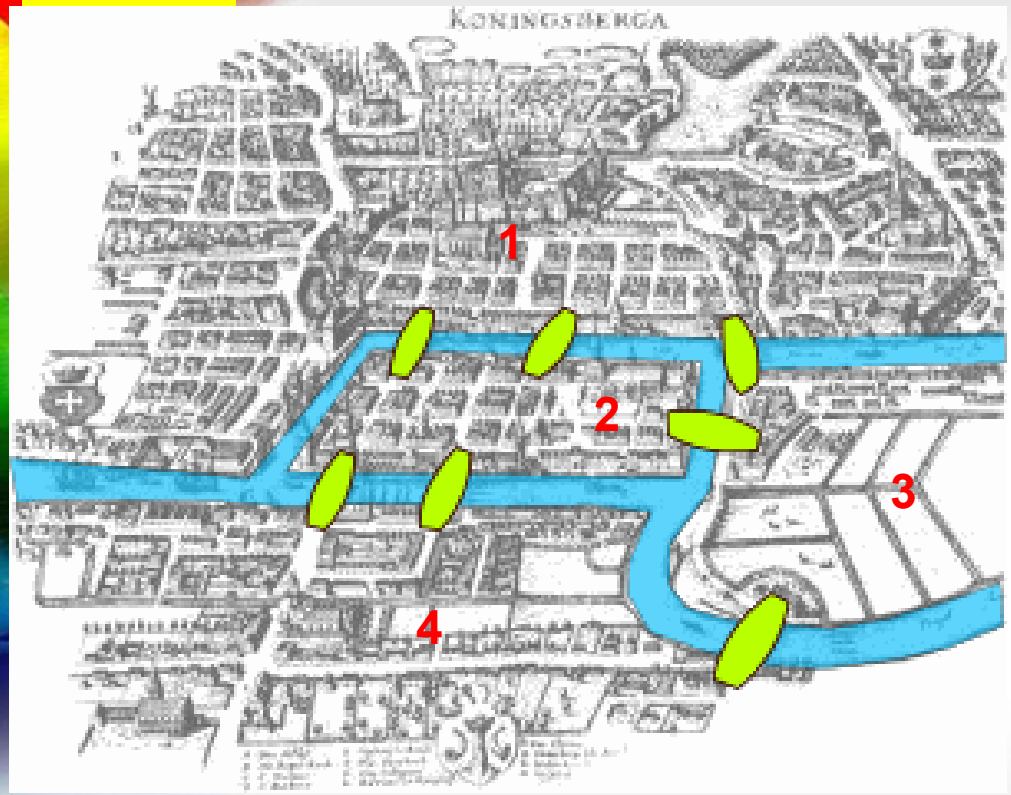
1. การค้นหาคำในประโยค
2. การค้นหาประโยคในเนื้อหา

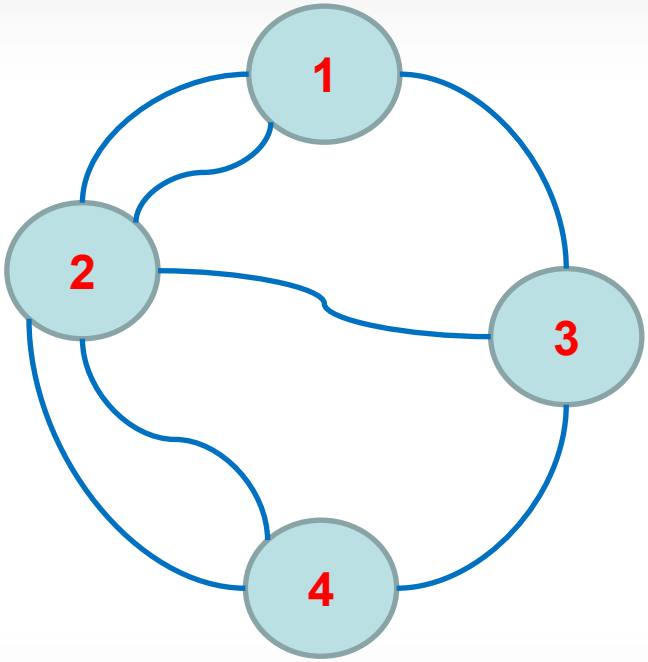
https://en.wikipedia.org/wiki/String_searching_algorithm

fppt.com

Graph problems

ต.ย. สะพานเคอนิกส์แบร์ก (Konigsberg bridges)





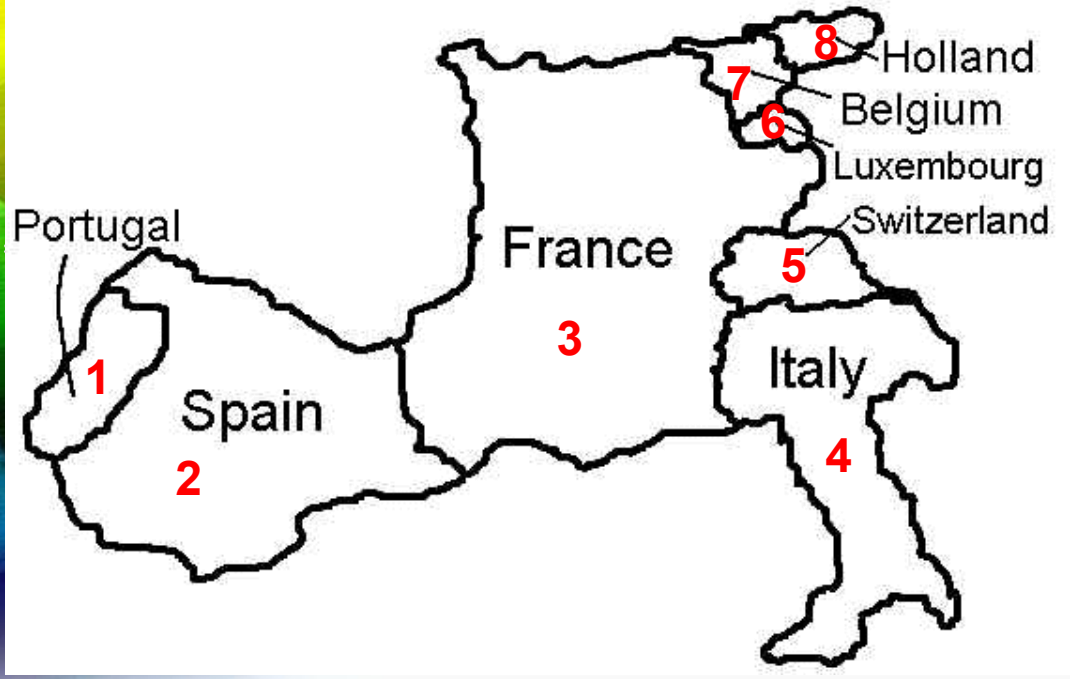
https://en.wikipedia.org/wiki/Graph_theory

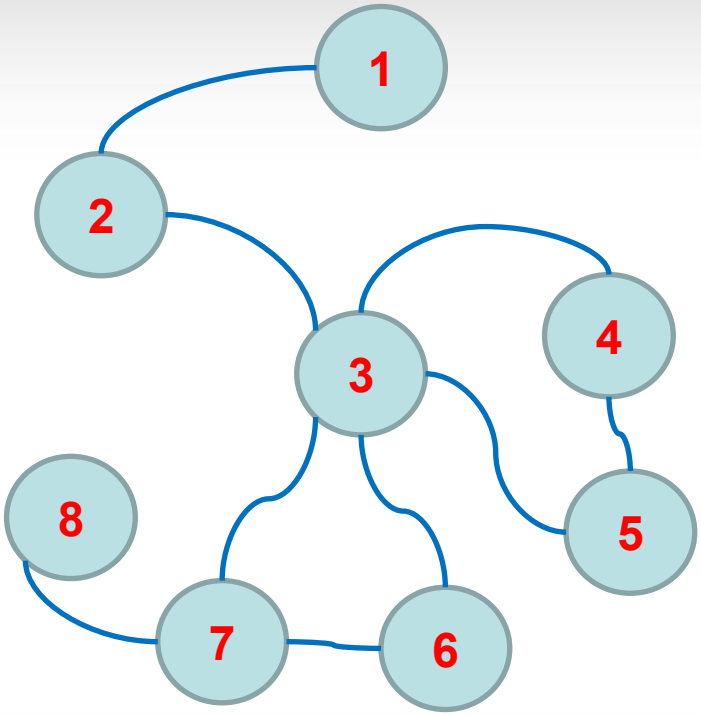
fppt.com

Graph problems

ต.ย.

การให้สี(coloring)





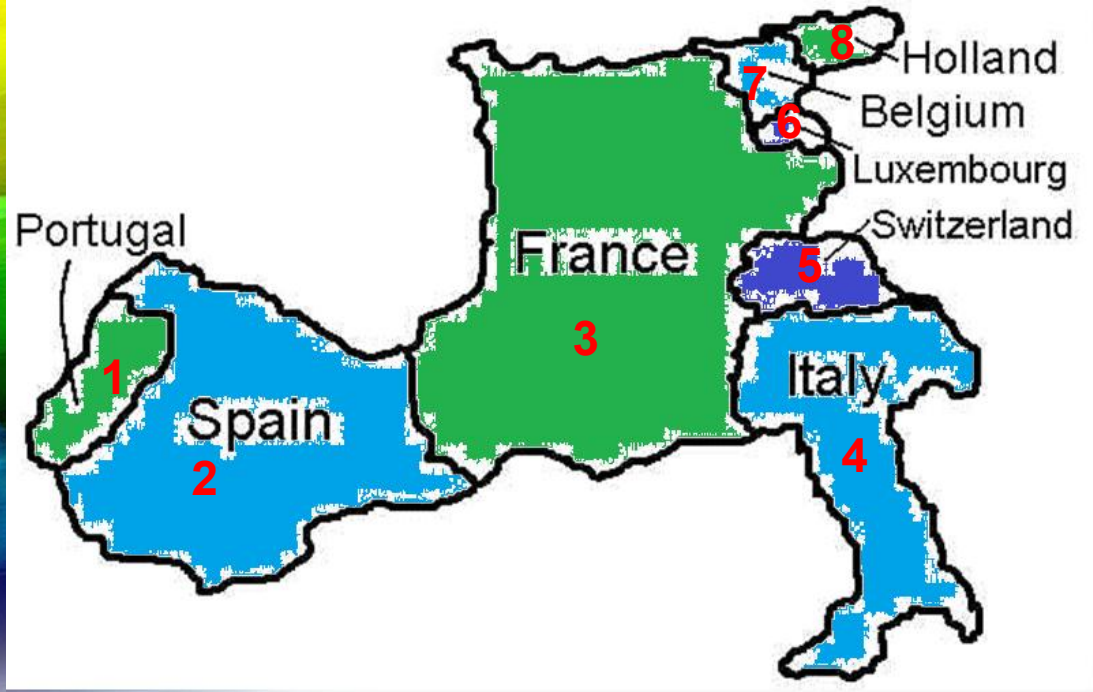
https://en.wikipedia.org/wiki/Graph_theory

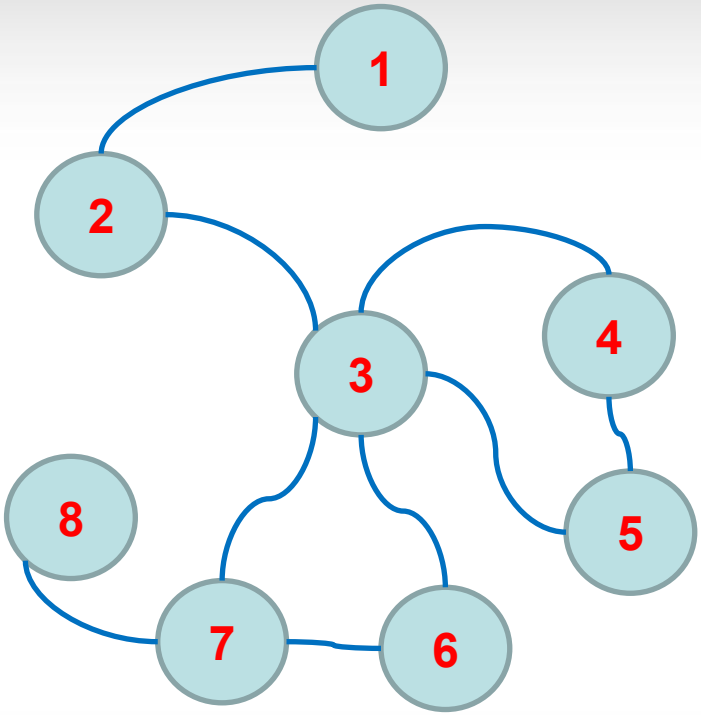
fppt.com

Graph problems

ต.ย.

การให้สี(coloring)





https://en.wikipedia.org/wiki/Graph_theory

fppt.com

หลักการพื้นฐานโครงสร้างข้อมูล

- Linear Data Structures
- Graphs
- Trees
- Sets and Dictionaries

fppt.com

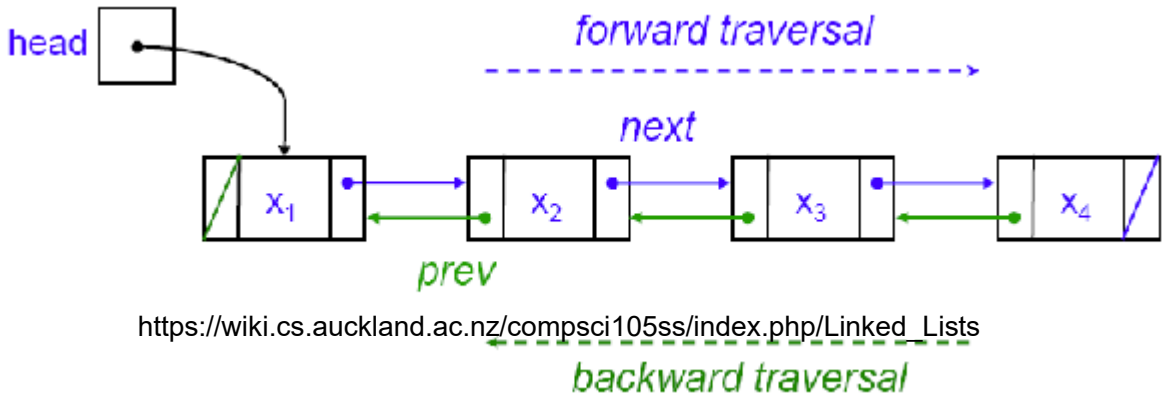
หลักการพื้นฐานโครงสร้างข้อมูล

Linear Data Structures

- Array

Item[0]	Item[1]	…	Item[n-1]

- Linked list



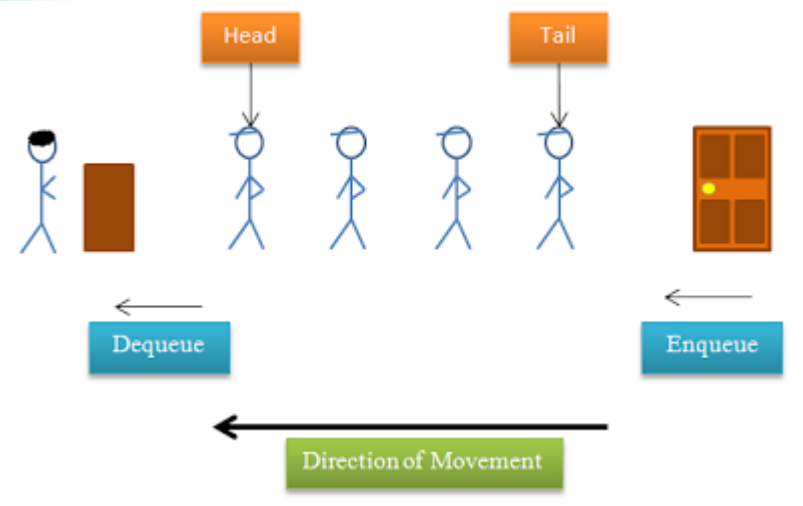
https://wiki.cs.auckland.ac.nz/compsci105ss/index.php/Linked_Lists

fppt.com

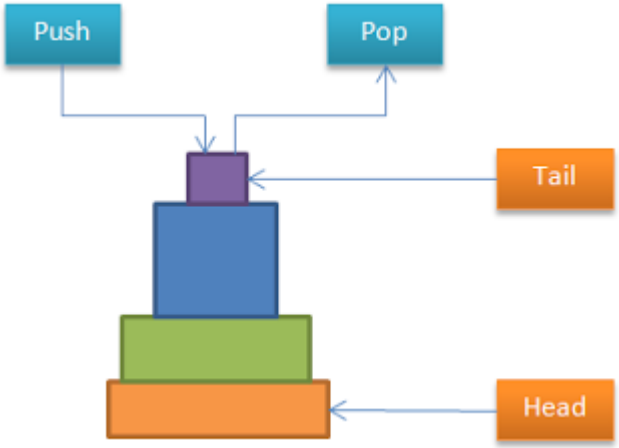
หลักการพื้นฐานโครงสร้างข้อมูล

รูปแบบของ list

- queue



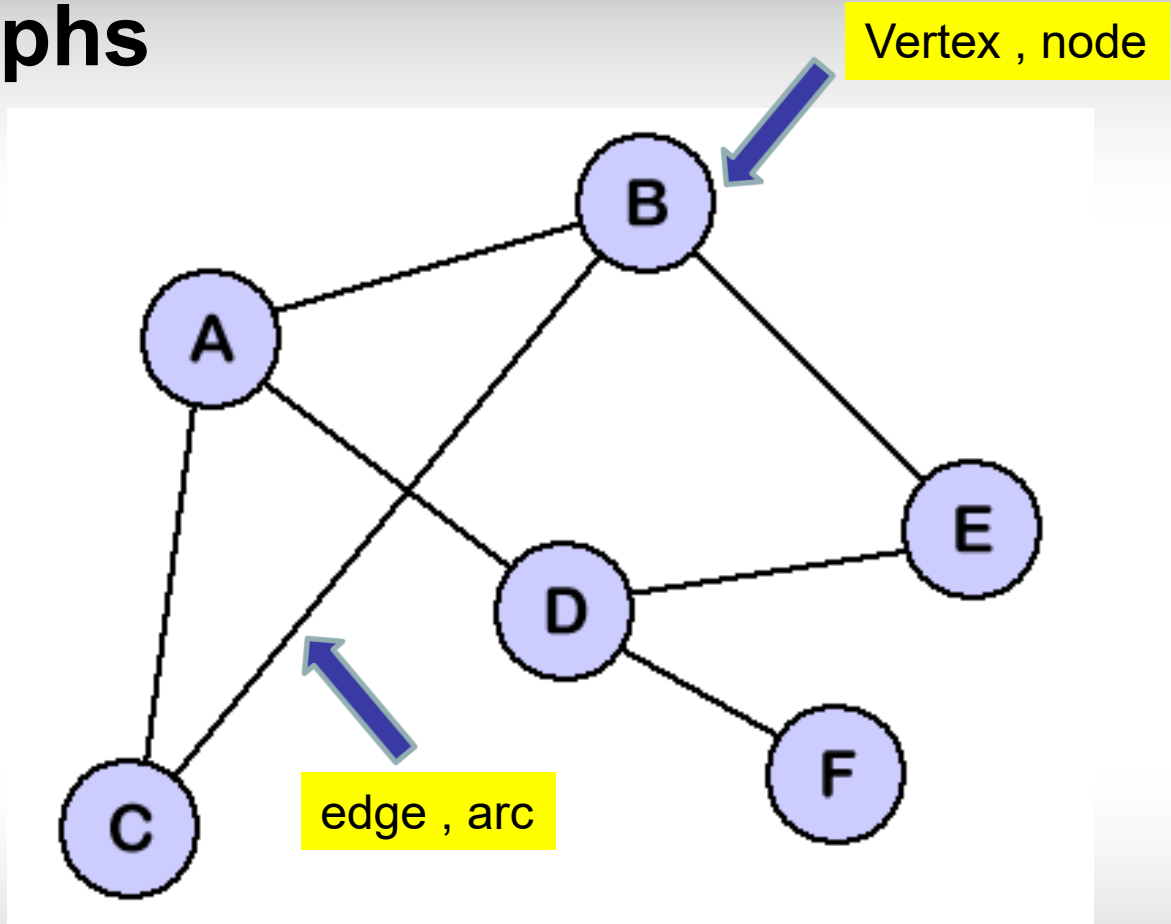
- stack



fppt.com

หลักการพื้นฐานโครงสร้างข้อมูล

Graphs

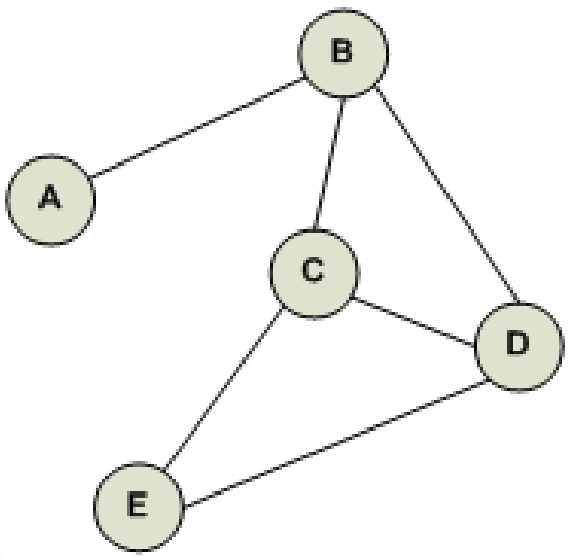


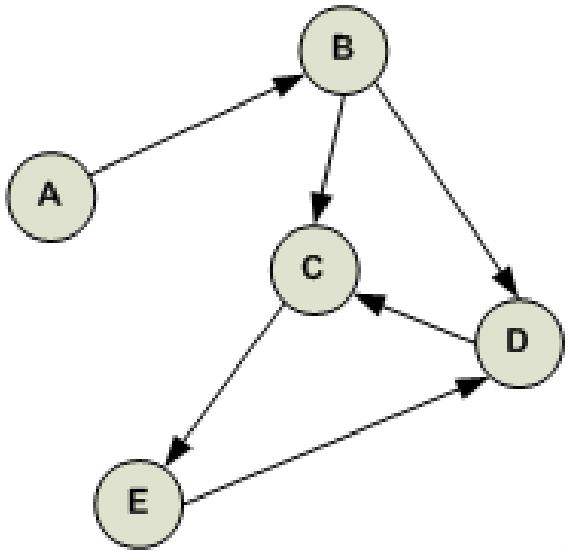
fppt.com

หลักการพื้นฐานโครงสร้างข้อมูล

Graphs

- Undirected graph
- Directed graph





http://www.codediesel.com/algorithms/building-a-graph-data-structure-in-php/

Fig 1. Undirected Graph

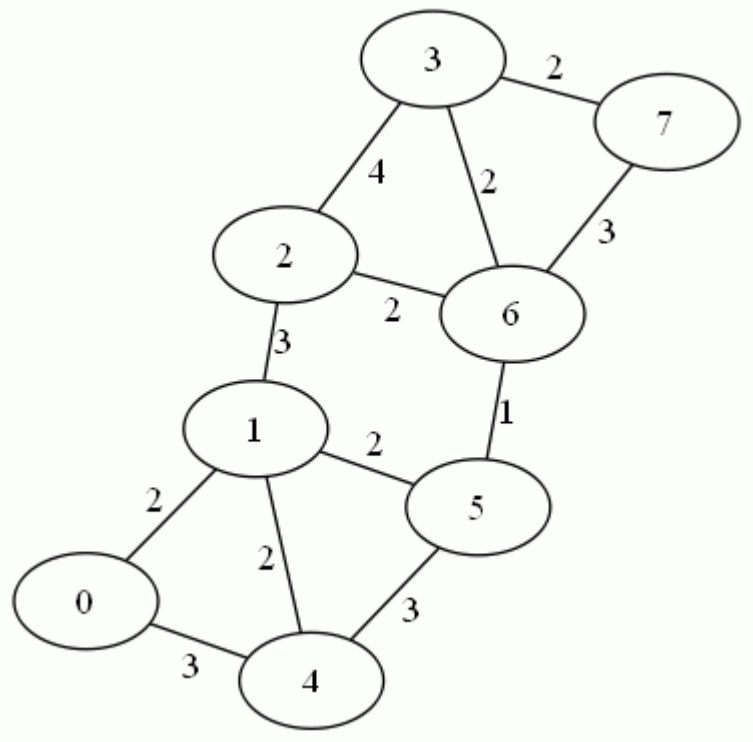
Fig 2. Directed Graph

fppt.com

หลักการพื้นฐานโครงสร้างข้อมูล

Graphs

- Weighted graph



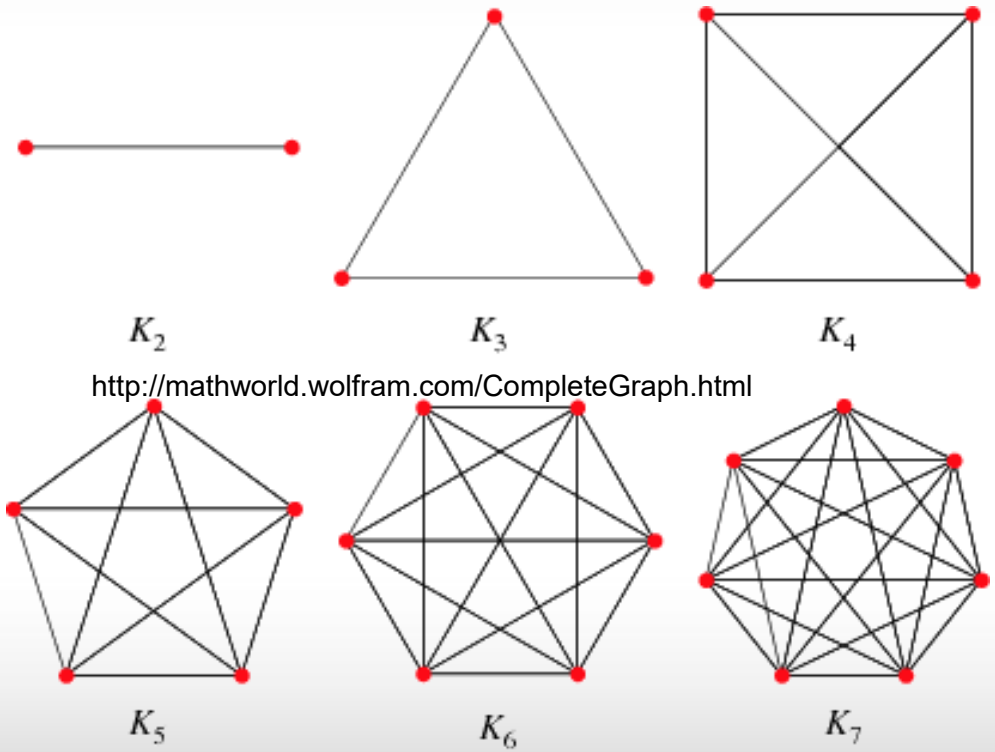
fppt.com

http://www.boost.org/doc/libs/1_46_1/libs/graph/doc/graph_theory_review.html/

หลักการพื้นฐานโครงสร้างข้อมูล

Graphs

- Complete graph



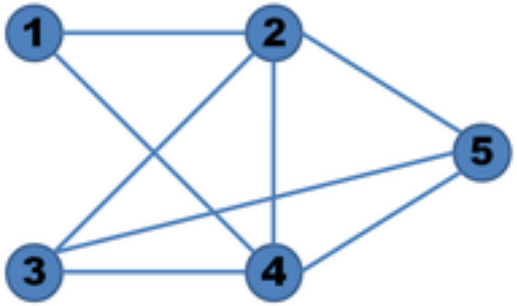
http://mathworld.wolfram.com/CompleteGraph.html

fppt.com

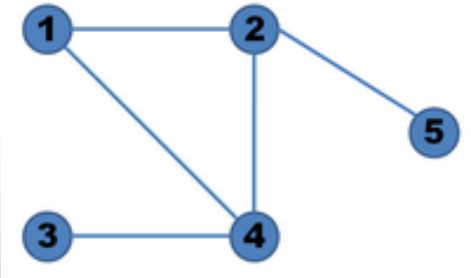
หลักการพื้นฐานโครงสร้างข้อมูล

Graphs

- Dense graph



- Sparse graph



fppt.com

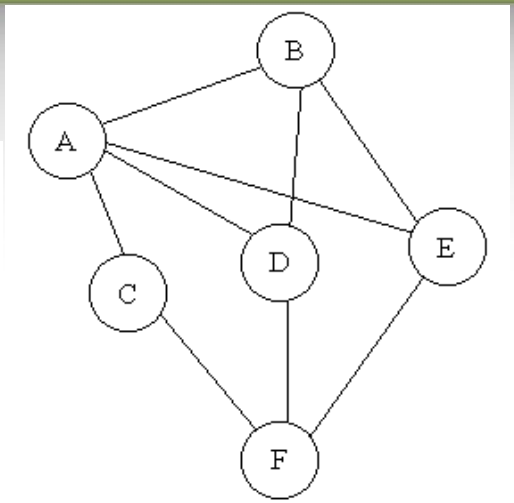
http://techportal.inviqa.com/2009/09/07/graphs-in-the-database-sql-meets-social-networks/

หลักการพื้นฐานโครงสร้างข้อมูล

Graph Representations

- Adjacency lists

A	→ B → C → D → E
B	→ A → D → E
C	→ A → F
D	→ A → B → F
E	→ A → B → F
F	→ C → D → E



- Adjacency matrix

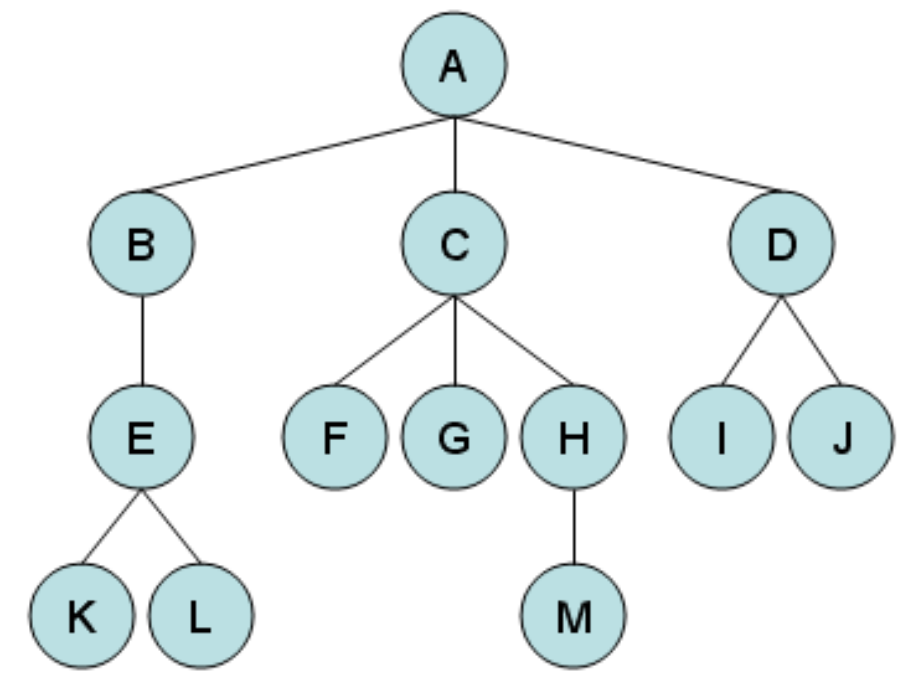
	A	B	C	D	E	F
A	-	1	1	1	1	
B	1	-		1	1	
C	1		-			1
D	1	1		-		1
E	1	1			-	1
F			1	1	1	-

fppt.com

http://campus.murraystate.edu/academic/faculty/bob.pilgrim/245/programming_22.html /

หลักการพื้นฐานโครงสร้างข้อมูล

Trees



http://www.csupomona.edu/~ftang/courses/CS241/notes/trees.htm

fppt.com

http://campus.murraystate.edu/academic/faculty/bob.pilgrim/245/programming_22.html /

หลักการพื้นฐานโครงสร้างข้อมูล

Sets and Dictionaries

ตย. sets

S = {2,3,5,7}
S = {n: n is a prime number smaller than 10}

ตย. Dictionaries

"Great Expectations": "John“
"Pride and Prejudice": "Alice“
"Wuthering Heights": "Alice"

fppt.com

http://en.wikipedia.org/wiki/Associative_array

อ้างอิง

[1] Introduction to the design & analysis of algorithm, 3rd, Anany Levitin

fppt.com

แบบฝึกหัด

1. อัลกอริทึม คือ อะไร
2. สามารถเขียนอัลกอริทึมได้ในรูปแบบใดบ้าง
3. อธิบายหลักการในการแก้ไขปัญหามีขั้นตอนอย่างไรบ้าง
4. ปัญหาพื้นฐานมีอะไรบ้าง และยกตัวอย่างมาปัญหาละ 2 ตัวอย่าง
5. โครงสร้างข้อมูลพื้นฐานมีอะไรบ้าง

fppt.com