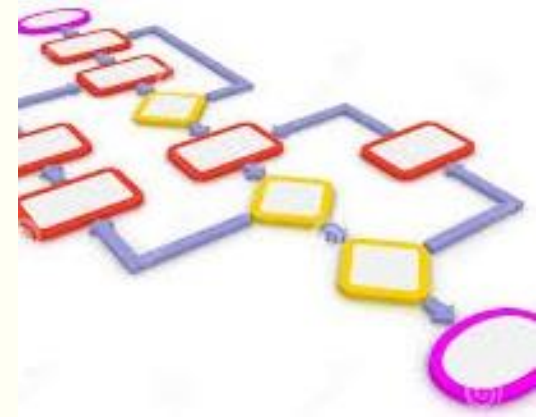
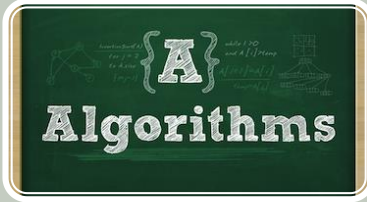


การออกแบบและวิเคราะห์ขั้นตอนวิธี
DESIGN AND ANALYSIS OF ALGORITHMS
02-212-212

อ.ธิดาวรรณ คล้ายศรี





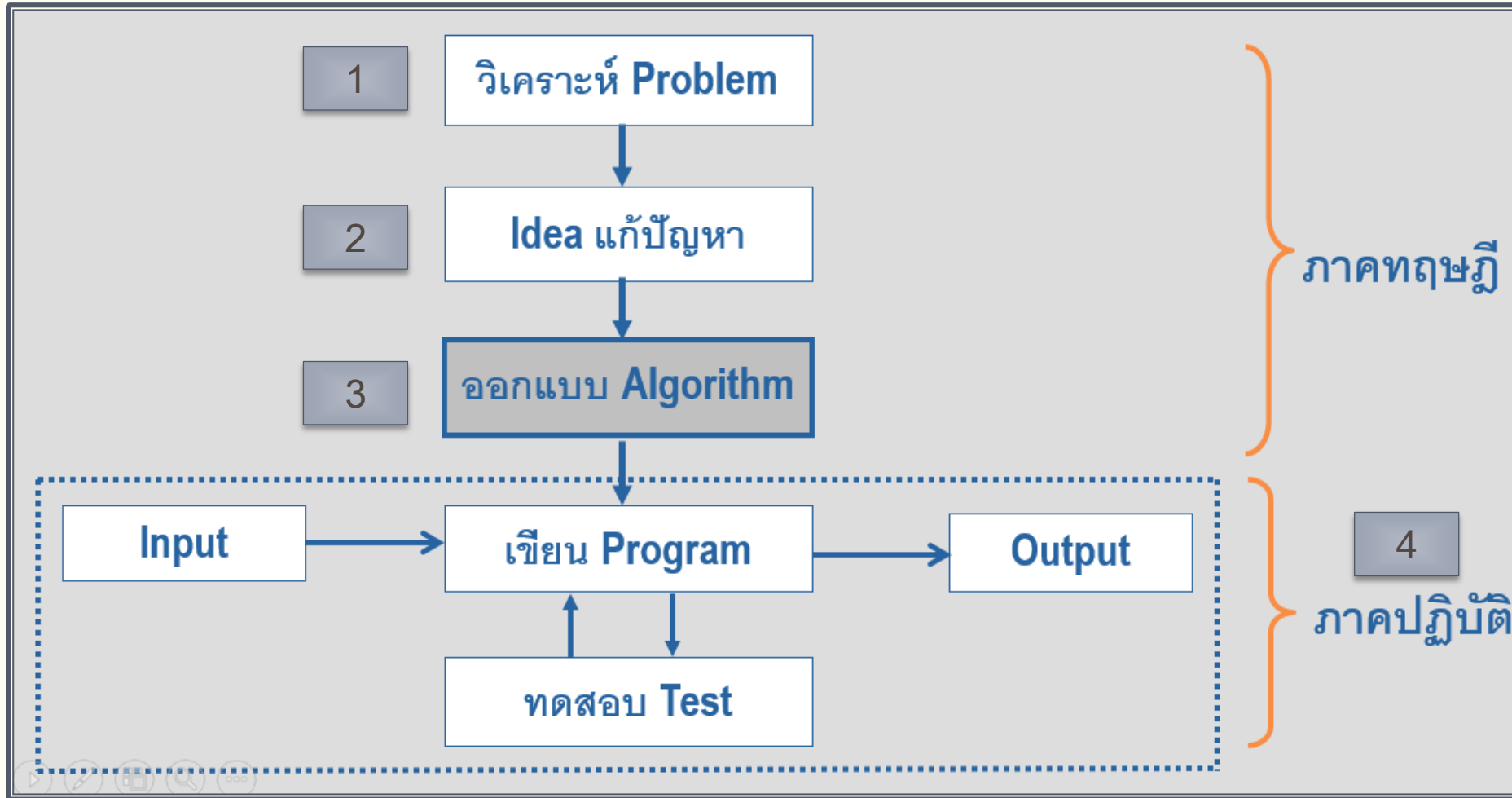
- 1.1 การแก้ปัญหาและขั้นตอนวิธี (Problems Solving and Algorithms)
- 1.2 An Introduction to Algorithm Design (บทนำการออกแบบขั้นตอนวิธี)
- 1.3 คณิตศาสตร์พื้นฐานเพื่อการวิเคราะห์ (Mathematic Preliminaries for Analysis)
- 1.4 Analysis of Algorithms (การวิเคราะห์อัลกอริธึม)
- 1.5 Growth of Functions (แนวโน้มการเพิ่มขึ้นของฟังก์ชัน)

ทบทวนก่อนเรียน

อ.ธิดาวรรณ คล้ายศรี



1.1 การแก้ปัญหาและขั้นตอนวิธี (Problems Solving and Algorithms)



1.1 การแก้ปัญหาและขั้นตอนวิธี (Problems Solving and Algorithms)

กระบวนการแก้ปัญหาด้วยคอมพิวเตอร์ ประกอบไปด้วย 4 ขั้นตอน

1. การวิเคราะห์ปัญหา (Problem Analysis) → เข้าใจปัญหา/ชัดเจน
2. หาแนวคิดในการแก้ปัญหา (Solution Idea) → มีวิธีใดบ้าง
3. การออกแบบและวิเคราะห์ขั้นตอนวิธีทางคอมพิวเตอร์ (Design and Analysis Computer Algorithm) → แบบวนซ้ำ (iterate)/เรียกตัวเอง (recursive)/ตรรกะ (logic)/การเปรียบเทียบ (comparison)
4. การพัฒนาและทดสอบโปรแกรม (Program Development and Testing)

1.1 การแก้ปัญหาและขั้นตอนวิธี (Problems Solving and Algorithms)

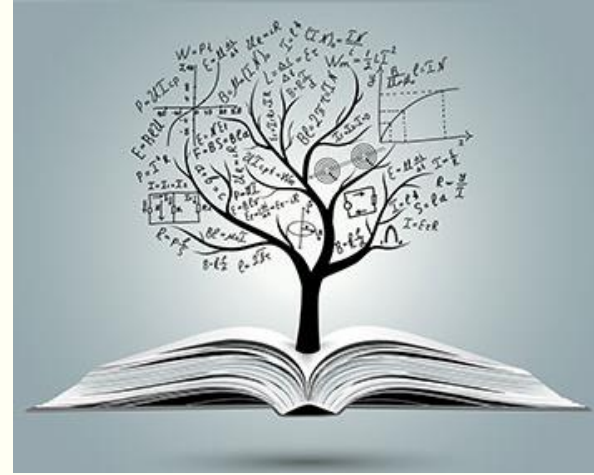
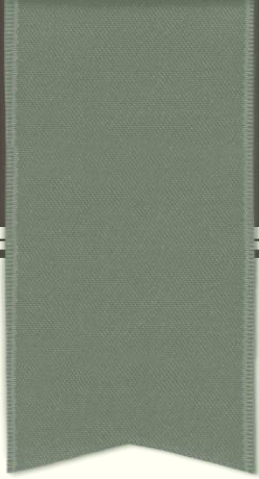
กระบวนการทำงาน (procedure) ใน algorithm เพื่อใช้แก้ปัญหาให้บรรลุเป้าหมายนั้นจะต้อง

- มีขั้นตอนการดำเนินงานที่ชัดเจน
- กระบวนการทำงานควรไม่ซับซ้อน เพื่อที่จะได้สามารถนำไปปฏิบัติการโดยเครื่องคอมพิวเตอร์
- กระบวนการทำงานมีจุดจบ (finiteness)
- นิยมเขียนในรูปแบบของคำสั่งจำลอง (pseudo code)

1.2 บทนำการออกแบบขั้นตอนวิธี (An Introduction to Algorithm Design)

ต้องการอัลกอริธึมที่ดี → สามารถหาผลลัพธ์ได้ถูกต้อง (ในเวลาจำกัด)

- ถูกต้อง
- มีประสิทธิภาพ
- ใช้หน่วยความจำให้น้อยที่สุด
- มีขั้นตอนวิธีที่ชัดเจน เข้าใจง่าย



1.3 คณิตศาสตร์พื้นฐานเพื่อการวิเคราะห์ (MATHEMATIC PRELIMINARIES FOR ANALYSIS)

1.3 คณิตศาสตร์พื้นฐานเพื่อการวิเคราะห์ (Mathematic Preliminaries for Analysis)

1.3.1 ออนุกรม (Series) นำมาใช้งานบ่อย:

- **อนุกรมเลขคณิต** $1+2+3+ \dots + n$
 $= \sum_{i=1}^n i = n(n+1) / 2 \rightarrow \sim n^2$
- **อนุกรมเรขาคณิต** $1+x+x^2+\dots+x^n$
 $= \sum_{i=0}^n x^i = (x^{n+1}-1) / (x-1) \rightarrow \sim x^n$

[ภาพโดย อ. จีระพร วีระพันธุ์]

1.3 คณิตศาสตร์พื้นฐานเพื่อการวิเคราะห์ (Mathematic Preliminaries for Analysis)

1.3.2 ฟังก์ชันความซับซ้อนด้านเวลาของ algorithm:

Discrete Mathematic ที่มีการใช้งานบ่อย:

- ฟังก์ชัน Exponential
- ฟังก์ชัน Logarithm
- ฟังก์ชัน Factorial

ฟังก์ชัน Exponential

ฟังก์ชัน Exponential อาศัยพื้นฐานทฤษฎีที่เกี่ยวข้องกับเลขยกกำลัง

ถ้า a, b เป็นจำนวนจริง โดยที่ $a \neq 0, b \neq 0$ และ m, n เป็นจำนวนเต็ม

$$1. a^m a^n = a^{m+n}$$

$$2. (a^m)^n = a^{mn}$$

$$3. (ab)^m = a^m b^m$$

$$4. (a/b)^m = a^m / b^m$$

$$5. a^m / a^n = a^{m-n}$$

ทฤษฎีที่เกี่ยวข้องกับฟังก์ชัน Exponential

ถ้า a, b เป็นจำนวนจริงบวก โดยที่ $a \neq 1, b \neq 1$ และ x, y เป็นจำนวนจริง, ตัวแปร หรือนิพจน์ทางคณิตศาสตร์

1. กฎของเลขยกกำลัง (Exponent Laws)

$$1.1 \ a^x a^y = a^{x+y}$$

$$1.2 \ (a^x)^y = a^{x \cdot y}$$

$$1.3 \ (ab)^x = a^x b^x$$

$$1.4 \ (a/b)^x = a^x / b^x$$

$$1.5 \ a^m / a^n = a^{m-n}$$

$$1.6 \ a^{-x} = 1/a^x$$

$$1.7 \ a^0 = 1$$

2. $ax = ay$ ก็ต่อเมื่อ $x = y$

3. ถ้า $x \neq 0$ แล้ว $a^x = b^x$ ก็ต่อเมื่อ $a = b$

ฟังก์ชัน Logarithms/ทฤษฎีที่เกี่ยวข้องกับฟังก์ชัน Logarithms

→ ฟังก์ชัน Logarithm เป็นส่วนกลับกันของ ฟังก์ชัน Exponential

กำหนด M, N, a และ b เป็นจำนวนจริงบวก โดยที่ $a \neq 1$, $b \neq 1$ และมี n เป็นจำนวนจริง แล้ว สามารถสรุปได้ ดังนี้

$$1. \log_a MN = \log_a M + \log_a N$$

$$2. \log_a M/N = \log_a M - \log_a N$$

$$3. \log_a 1 = 0$$

$$4. \log_a M^n = n \log_a M$$

$$5. \log_a a = 1$$

$$6. \log_a M = \log_a N \quad \text{ก็ต่อเมื่อ} \quad M=N$$

$$7. a^{\log_a M} = M$$

$$8. \log_{a^n} M = 1/n \log_a M$$

$$9. \log_{1/a} M = -\log_a M$$

$$10. \log_a M = 1/\log_M a \quad \text{เมื่อ} \quad M \neq 1$$

ฟังก์ชัน Factorial

ค่า n Factorial \rightarrow n!

หมายถึง ผลคูณของจำนวนเต็มบวกตั้งแต่ 1 ถึง n เมื่อ n เป็นจำนวนเต็มบวกใดๆ

เช่น factorial 8 (8!) = $8 \times 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1$

หรือ ผลคูณของจำนวนเต็มบวก n กับจำนวนที่ลดลงจาก n ทีละ 1 จนกระทั่งถึง 1

$$n! = n \times (n-1) \times (n-2) \times (n-3) \times \dots \times 2 \times 1$$

$$\text{หรือ } n! = 1 \times 2 \times 3 \times \dots \times (n-1) \times n$$

แบบฝึกหัด

- จงเขียนอัลกอริธึม ทำการหาค่า n factorial