

คณิตศาสตร์คอมพิวเตอร์

กุญแจปลดล็อกโลกเทคโนโลยีดิจิทัลมีเดีย

สาขาวิชาเทคโนโลยีดิจิทัลมีเดีย (ปวช. 1)
วิทยาลัยเทคโนโลยีอาเซี่ยนบริหารธุรกิจ



เรียนคณิตศาสตร์ ไปทำไม?

ตรรกะและอัลกอริทึม

สร้างระบบความคิดที่เป็นขั้นตอน
เขียนโค้ดได้ไม่บก

โครงสร้างข้อมูล

จัดระเบียบข้อมูลหลากหลายให้ค้นหาและใช้งานได้ไว

กราฟิกและเกม 3D

คำนวณพิกัดและเมทริกซ์เพื่อสร้างโลกเสมือนจริง

ความปลอดภัย

ไคคณิตศาสตร์เข้ารหัส (RSA)
ป้องกันแฮกเกอร์ขโมยข้อมูล

บิต (Bit) จุดเริ่มต้นของทุกสิ่ง

สภาวะปิด (OFF) = 0

สภาวะเปิด (ON) = 1



Bit (Binary Digit) คือหน่วยข้อมูลที่เล็กที่สุดในคอมพิวเตอร์
ข้อมูลทุกอย่าง (ภาพ, เสียง, ข้อความ) เกิดจากการต่อสวิตช์ 0 และ 1 เหล่านี้นับล้านตัว!

ระบบเลขฐานในคอมพิวเตอร์

เลขฐาน 2 (Binary)	สัญลักษณ์: 0, 1	การใช้งาน: ภาษาเครื่องแท้ๆ
เลขฐาน 8 (Octal)	สัญลักษณ์: 0 ถึง 7	การใช้งาน: ย่อเลขฐานสองให้สั้นลง
เลขฐาน 10 (Decimal)	สัญลักษณ์: 0 ถึง 9	การใช้งาน: ภาษามนุษย์ที่ใช้ในชีวิตประจำวัน
เลขฐาน 16 (Hexadecimal)	สัญลักษณ์: 0 ถึง 9 และ A, B, C, D, E, F	การใช้งาน: แสดงสีในเว็บ และที่อยู่หน่วยความจำ



คอมพิวเตอร์รู้จักตัวอักษรได้อย่างไร?

มนุษย์พิมพ์ตัวอักษร 'A'

A

ระบบจะแปลงอักษร
เป็นชุดตัวเลขผ่านตาราง
มาตรฐาน ASCII

'A' กลายเป็นข้อมูล 8 บิต:
01000001 (เก็บลงหน่วยความจำ)

01000001

ระบบอื่นๆ ที่ควรรู้:
BCD (ใช้ 6 บิต)
และ EBCDIC (ใช้ 8 บิต)

การจัดการเลขติดลบในโลกของ 0 กับ 1

คอมพิวเตอร์ไม่มีเครื่องหมายลบ (-)
สมองของมันจึงต้องใช้วิธีที่เรียกว่า
Complements

0101 → 1010

1's Complement: สลับขั้วข้อมูลทุกบิต
(เปลี่ยน 0 เป็น 1 และ 1 เป็น 0)

1010 + 1 → 1011

2's Complement: เอาผลลัพธ์จาก 1's มาบวกเพิ่มอีก 1
(นี่คือระบบที่ CPU ส่วนใหญ่ใช้จริง!)

ตรรกศาสตร์เบื้องต้น (Introduction to Logic)

ประพจน์ (Proposition)
คือ ประโยคที่บอกได้ชัดเจนว่า
จริง (1) หรือ เท็จ (0) เท่านั้น ห้ามกำกวม!

เท็จ (False)
สาคดิส: 0

จริง (True)
สาคดิส: 1

$1 + 1 = 2 \rightarrow$ **จริง (1)**

กรุงเทพฯ หิมะตก $\rightarrow$ **เท็จ (0)**

คอมพิวเตอร์จะทำงานได้
ต้องมีตรรกะที่ชัดเจนแบบนี้เสมอ

พีชคณิตบูลีน (Boolean Algebra)



คิดค้นโดยนักคณิตศาสตร์
จอร์จ บูล (George Boole)

คณิตศาสตร์ที่ไม่ได้คำนวณตัวเลข
แต่คำนวณ **ความจริง** กับ **ความเท็จ**

AND
(Conjunction):
ต้องจริงทั้งคู่
ถึงจะผ่าน!

OR
(Disjunction):
แค่จริงอย่างใดอย่าง
หนึ่ง ก็ผ่านแล้ว!

NOT
(Negation):
พลิกกลับหน้ามือ
เป็นหลังมือ!

พื้นฐานลอจิกเกต (Logic Gates) อีฐัก่อนแรกของสมองกล

AND Gate
(รูปครึ่งวงกลม)

OR Gate
(รูปหัวลูกศรโค้ง)

NOT Gate
(รูปสามเหลี่ยมมีวงกลม)

วงจรลอจิกแปลง **ตรรกะ** ให้กลายเป็น **กระแสไฟฟ้า**

กลไกการตัดสินใจ

Input A: ลูกบอลมืด (0)

Input B: ลูกบอลเรืองแสง (1)

ตรรกะ AND (และ):

$1 \text{ AND } 1 = 1$ (เรืองแสง!)
กรณีอื่น = 0 ทั้งหมด

ตรรกะ OR (หรือ):

$0 \text{ OR } 0 = 0$ (มืดมืด)
มี 1 โผล่มาแต่ตัวเดียว = 1 (เรืองแสง!)

Output Z:
ลูกบอลมืด (0) หล่นออกมา!

จากลอจิกเกต สู่หน่วยประมวลผล (CPU)

ลอจิกเกตแค่ 1 ตัวอาจดูไม่มีประโยชน์
แต่เมื่อนำเกตมาเรียงต่อกันเป็นวงจร
หลายล้านตัว มันจะกลายเป็น
สมอง (Processor)

ALU (Arithmetic Logic Unit):
โซนคำนวณคณิตศาสตร์และตรรกะ

Main Memory:
โซนเก็บคำสั่งและข้อมูลหลัก

Register File:
โซนพักข้อมูลชั่วคราว
ขณะคำนวณ

คณิตศาสตร์ไม่ต่อเนื่อง (Discrete Math) หัวใจของซอฟต์แวร์

โลกความจริงเป็นแบบ
ต่อเนื่อง (Continuum)
เช่น คลื่นเสียง, อุณหภูมิ
(แบ่งย่อยได้ไม่สิ้นสุด)

โลกคอมพิวเตอร์เป็นแบบ
ไม่ต่อเนื่อง (Discrete)
ข้อมูลต้องเป็นชั้นๆ นับได้ชัดเจน
เช่น ตัวอักษรบนคีย์บอร์ด, ไฟล์ภาพ

เพราะข้อมูลคอมพิวเตอร์มีลักษณะเป็น ชั้นๆ
เราจึงต้องมีวิธี จัดระเบียบ มันให้มีประสิทธิภาพ!

โครงสร้างข้อมูล (Data Structures)

การจัดวางข้อมูลในรูปแบบที่เหมาะสม เพื่อง่ายต่อการค้นหาและประมวลผล

Array (แถวลำดับ):

ข้อมูลเรียงติดกันเป็นระเบียบ
เข้าถึงง่าย

Linked List (รายการเชื่อมโยง):

ข้อมูลอยู่กระจัดกระจาย
แต่มีสายโซ่ชี้ไปยังตัวถัดไป

Tree (โครงสร้างต้นไม้):

ข้อมูลแบบลำดับชั้น
เหมาะกับการค้นหาแบบแยกซ้าย-ขวา

โครงสร้างข้อมูล Stacks (แถวลำดับ):

ข้อมูลเรียงติดกันกรีนอวคม
ในค้อยขอสมควาร

Tree (โครงสร้างต้นไม้):

ข้อมูลแบบลำดับชั้น
เหมาะกับการค้นหาแบบแยกซ้าย-ขวา

ทฤษฎีกราฟ (Graph Theory) สู่การใช้งานจริง



ภาพรวมสู่การเป็นนักพัฒนาดิจิทัลมืออาชีพ



ทุกแพลตฟอร์ม เกม 3D หรือแอปพลิเคชันที่คุณใช้งาน ล้วนขับเคลื่อนด้วย คณิตศาสตร์คอมพิวเตอร์ พื้นฐานที่แข็งแกร่งจะทำให้คุณต่อยอดเทคโนโลยีได้ไม่มีที่สิ้นสุด!



Q&A | พร้อมลุยหรือยัง?

โดย อ.เก่ง สาขาเทคโนโลยีธุรกิจดิจิทัล ปวช.1