



Basic python CODING

เรียนง่ายเป็นเร็ว

เปลี่ยนการเขียนโปรแกรมให้กลายเป็นเรื่องง่าย ออกแบบให้ทุกคนเขียนได้อย่างรวดเร็ว

ผู้แต่ง ผู้ช่วยศาสตราจารย์ ดร.สุชาติ คุ้มมะณี บรรณาธิการ ภิรพล ภูเขาเจริญ



มีเพียง “ความรู้” เท่านั้นที่มนุษย์ใช้พลิก “โลก” และเปลี่ยน “ชีวิต”
เราจึงสร้างสรรค์ และส่งมอบ “ความรู้” ในรูปแบบที่ดีกว่า
เพื่อให้คนไทย “เรียนรู้” ได้ตลอดชีวิต



basic Python coding

เรียนง่ายเป็นเร็ว

Writer	ผู้ช่วยศาสตราจารย์ ดร.สุชาติ คุ้มมะณี
Editor	ภีรพล คชาเจริญ
Graphic Designers	วสันต์ พึ่งพูลผล, ถมทวานิตย์ มนตรีมนโรรมย์
Page Layout	สิริลักษณ์ วาระเลิศ
Proofreader	สุนทรี บรรลือศักดิ์
Publishing Coordinators	วรพล ณธิกุล, สุพัตรา อจาปุรุ, ปฐมพล ธรรมศรีสกุล

Microsoft Windows เป็นเครื่องหมายการค้าของบริษัท Microsoft Corporation, Python IDLE เป็นเครื่องหมายการค้าของบริษัท Python Software Foundation, PyCharm เป็นเครื่องหมายการค้าของบริษัท JetBrains, Komodo IDE เป็นเครื่องหมายการค้าของบริษัท ActiveState, PyDev เป็นเครื่องหมายการค้าของบริษัท Eclipse และเครื่องหมายการค้าอื่นๆ ที่อ้างถึงเป็นของบริษัทนั้นๆ บริษัท ไอดีซี พรีเมียร์ จำกัด จัดตั้งขึ้นเพื่อเผยแพร่ความรู้ที่มีคุณภาพสู่ผู้อ่านชาวไทย เรายินดีรับงานเขียนของนักวิชาการและนักเขียนทุกท่าน ท่านผู้สนใจกรุณาติดต่อผ่านทางอีเมลที่ infopress@idcpremier.com หรือทางโทรศัพท์หมายเลข 0-2962-1081 (อัตรานาที 10 คู่สาย) โทรสาร 0-2962-1084

สร้างสสคโดย



พิมพ์ครั้งที่ 1 สิงหาคม 2562

ข้อมูลทางบรรณานุกรม

ผู้ช่วยศาสตราจารย์ ดร.สุชาติ คุ้มมะณี
basic Python coding เรียนง่ายเป็นเร็ว

นนทบุรี : ไอดีซีฯ, 2562

336 หน้า

1. ไพรอน

2. ภาษาที่ใช้ในการเขียนโปรแกรม

1 ชื่อเรื่อง

005.13

ISBN 885-916-100-568-1

จัดพิมพ์และจัดจำหน่ายโดย



บริษัท ไอดีซี พรีเมียร์ จำกัด

200 หมู่ 4 ชั้น 19 หอ 1901

อาคารจัสตินอินเตอร์เนชั่นแนลทาวเวอร์

ถ.แจ้งวัฒนะ อ.ปากเกร็ด จ.นนทบุรี 11120

โทรศัพท์ 0-2962-1081 (อัตรานาที 10 คู่สาย)

โทรสาร 0-2962-1084

สมาชิกสัมพันธ์

โทรศัพท์ 0-2962-1081-3 ต่อ 121

โทรสาร 0-2962-1084

ร้านค้าและตัวแทนจำหน่าย

โทรศัพท์ 0-2962-1081-3 ต่อ 112-114

โทรสาร 0-2962-1084

ราคา 295 บาท

EDITOR'S

NOTE

ในปี พ.ศ. 2561 เกิดความเปลี่ยนแปลงขึ้นในระบบการศึกษาขั้นพื้นฐานในบ้านเรา เมื่อมีการบรรจุวิชาวิทยาการคำนวณ หรือ Computing Science เข้ามาในหลักสูตรระดับประถมและมัธยม โดยปูพื้นฐานการเรียนรู้ไว้อย่างเป็นลำดับขั้นตอน เริ่มเรียนกันตั้งแต่ชั้นประถมศึกษาไปจนถึงชั้นมัธยมศึกษา และได้เรียนเขียนโปรแกรมในระดับที่สูงขึ้นในสายอาชีพ หรือสายอุดมศึกษาในสาขาที่เกี่ยวข้อง เช่น สาขาวิทยาการคอมพิวเตอร์ สาขาวิศวกรรมคอมพิวเตอร์ สาขาวิศวกรรมซอฟต์แวร์ สาขาวิศวกรรมไอซีที เป็นต้น

หนังสือเล่มนี้ ทางสำนักพิมพ์อินโฟเเพรสและอาจารย์ผู้เขียนตั้งใจออกแบบเนื้อหา เพื่อที่จะให้เป็นหนังสือสอนเขียนโปรแกรมสำหรับผู้เริ่มต้น สามารถเรียนรู้การเขียนโปรแกรมได้ด้วยตนเองแม้ไม่มีพื้นฐานมาก่อน กำหนดขอบเขตเนื้อหาให้สอดคล้องกับหลักสูตรการเรียนในระดับมัธยม อย่างไรก็ตาม หนังสือเรียนตามหลักสูตรอาจจะมีข้อจำกัดในด้านการนำเสนอเนื้อหา เพราะต้องคำนึงถึงผู้เรียนและผู้สอน จำนวนชั่วโมงการเรียนการสอน ความพร้อมทางด้านอุปกรณ์การสอน (ต้องเรียนในห้องคอมพิวเตอร์)

ด้วยเหตุนี้ ทางสำนักพิมพ์จึงจัดทำหนังสือเล่มนี้ขึ้นมา เพื่อให้เป็นคู่มือการเรียนเขียนโปรแกรมด้วยภาษา Python ที่สมบูรณ์ทั้งเนื้อหาและภาพประกอบที่ออกแบบมาอย่างดี ช่วยให้จดจำเนื้อหาได้ง่าย นอกจากนี้หนังสือเล่มนี้จะเหมาะกับนักเรียนชั้นมัธยมแล้ว ยังเหมาะกับผู้ที่สนใจศึกษาภาษาไพธอนไม่ว่าจะเป็นครู นักศึกษา หรือบุคคลทั่วไปที่อยากเรียนเขียนโปรแกรมด้วยตนเอง

กวีพล ochaเจริญ

Senior Editor

Digital Technology

PREFACE

วิทยาการคำนวณ หรือ Computing Science คือ วิชาที่จะสอนให้ผู้เรียนสามารถสร้างกระบวนการคิดเชิงวิเคราะห์อย่างเป็นระบบ และสามารถนำไปปรับใช้เพื่อแก้ไขปัญหาต่างๆ ได้อย่างสร้างสรรค์ โดยอาศัยศักยภาพการประมวลผลของระบบคอมพิวเตอร์ได้อย่างมีประสิทธิภาพ

วิชาวิทยาการคำนวณ ได้รับการบรรจุเข้าเป็นส่วนหนึ่งของหลักสูตรแกนกลางการศึกษาขั้นพื้นฐาน ในกลุ่มสาระการเรียนรู้วิทยาศาสตร์ ซึ่งเริ่มใช้เรียนกันตั้งแต่ปี พ.ศ. 2561 เป็นต้นไป จากการปรับปรุงหลักสูตรในครั้งนี้ พบว่าภาษาไพธอน (Python Language) ถูกบรรจุอยู่ในหนังสือเรียน “วิชาวิทยาการคำนวณ” ด้วย ดังนั้น หนังสือเล่มนี้จึงถูกออกแบบให้มีเนื้อหาครอบคลุมวิชาวิทยาการคำนวณ โดยเฉพาะพื้นฐานการคิดเชิงคำนวณ (Computational Thinking) ซึ่งมีองค์ประกอบที่สำคัญ 5 ส่วน คือ การแบ่งปัญหาใหญ่ให้เป็นปัญหาย่อย (Decomposition), การจำแนกรูปแบบของปัญหา (Pattern Recognition), การคิดเชิงนามธรรม (Abstraction), การออกแบบขั้นตอนวิธีการแก้ปัญหา (Algorithm Design) และโครงสร้างข้อมูล (Data Structure)

หนังสือเล่มนี้จึงเหมาะสำหรับนักเรียน นักศึกษา ครูผู้สอน อาจารย์ และผู้ที่มีความสนใจต้องการเรียนรู้พื้นฐานเกี่ยวกับวิทยาการคำนวณ เพื่อนำไปใช้แก้ปัญหาต่างๆ ด้วยระบบคอมพิวเตอร์ได้อย่างมีประสิทธิภาพ

ผู้เขียนหวังเป็นอย่างยิ่งว่า หนังสือเล่มนี้จะช่วยให้ผู้อ่านสามารถเขียนโปรแกรมด้วยไพธอน เพื่อช่วยในการแก้ไขปัญหาต่างๆ ได้ไม่มากนักน้อย หากหนังสือเล่มนี้มีข้อบกพร่องประการใด ผู้เขียนขออภัยขอรับไว้ทั้งหมด

ด้วยความนับถืออย่างสูง

ผู้ช่วยศาสตราจารย์ ดร.สุชาติ คุ่มมะณี

Email : suchart.k@msu.ac.th

www.facebook.com/suchart.khummanee

Chapter 01 วิทยาการคำนวณ วิชาใหม่ที่ทุกคนต้องได้เรียน

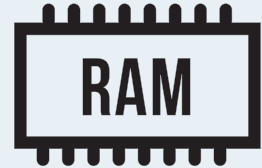
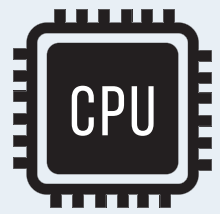
สร้างทักษะใหม่ให้เด็กยุคดิจิทัล เพื่อรับมือกับโลกอนาคต.....	2
▶ แรกรู้จักวิทยาการคำนวณ.....	2
▶ แรกรู้จัก Programming Language และ Python.....	5
สรุปท้ายบท.....	8

Chapter 02 คอมพิวเตอร์เบื้องต้นก่อนเขียนโปรแกรม

วิวัฒนาการคอมพิวเตอร์ (Computers in the Revolution)	10
ส่วนประกอบของเครื่องคอมพิวเตอร์	14
▶ คอมพิวเตอร์ฮาร์ดแวร์	14
▶ คอมพิวเตอร์ซอฟต์แวร์.....	18
การทำงานของระบบคอมพิวเตอร์.....	20
▶ หน่วยประมวลผลกลาง	20
▶ หน่วยความจำ.....	20
▶ อุปกรณ์รับข้อมูล/แสดงผล.....	23
▶ หลักการทำงานของระบบคอมพิวเตอร์	24
สรุปท้ายบท.....	25
แบบฝึกหัดท้ายบท.....	25

Chapter 03 การเตรียมซอฟต์แวร์สำหรับระบบคอมพิวเตอร์

ขั้นตอนการประมวลผลคำสั่งของคอมพิวเตอร์.....	28
▶ คอมไพเลอร์และอินเตอร์พรีเตอร์ต่างกันอย่างไร.....	30
สรุปท้ายบท.....	34
แบบฝึกหัดท้ายบท.....	34

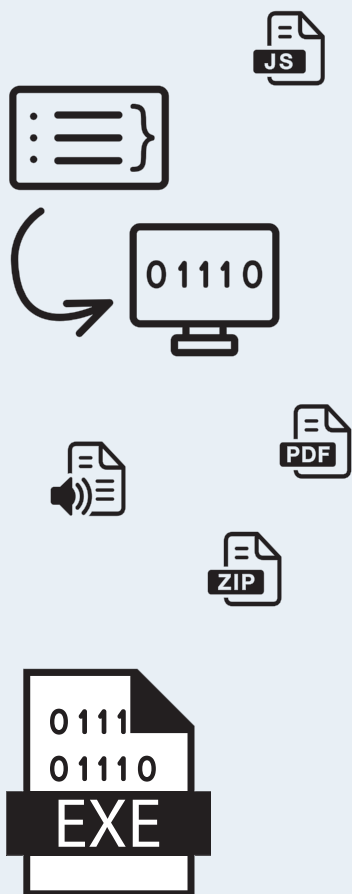


Chapter 04 การเขียนโปรแกรมในแบบที่คอมพิวเตอร์เข้าใจได้

การตีโจทย์ปัญหาไปเป็นภาษาคอมพิวเตอร์ (Problem Solving)	36
▶ อัลกอริทึมและโฟลว์ชาร์ต (Algorithm and Flowchart).....	38
▶ การเขียนโฟลว์ชาร์ต (Flowchart).....	45
▶ การเขียนซูดโค้ด (Pseudo Code)	51
การแปลงโฟลว์ชาร์ตและซูดโค้ด (Converting flowchart and pseudo)...	54
สรุปท้ายบท	56
แบบฝึกหัดท้ายบท	56

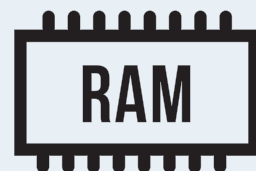
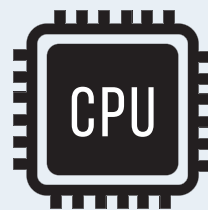
Chapter 05 รู้จักไพธอนและเครื่องมือพัฒนาโปรแกรม

การกำเนิดของภาษาไพธอน	58
ฟีเจอร์เด่นของภาษาไพธอน (Features of Python)	59
เครื่องมือในการเขียนโปรแกรมไพธอน (IDEs and Code Editors for Python)	62
▶ ไพธอนไอดีล (Python IDLE)	62
▶ อีคลิปส์ (Eclipse PyDev)	63
▶ โคโมโด (ActiveState Komodo IDE)	64
▶ ไพชาร์ม (PyCharm)	65
การติดตั้งและทดสอบโปรแกรมรันไพธอน (Installing and Testing Python Tools)	66
▶ การติดตั้งโปรแกรม Python IDLE	66
▶ การทดสอบโปรแกรม Python IDLE	68
▶ การติดตั้งโปรแกรม PyCharm	70
▶ การทดสอบโปรแกรม PyCharm	74
สรุปท้ายบท	80
แบบฝึกหัดท้ายบท	80



Chapter 06 การเขียนโปรแกรมไพธอนเบื้องต้น

ส่วนประกอบของโปรแกรมไพธอน (The Anatomy of a Python Program).....	82
คำสงวนในภาษาไพธอน (Reserved Words).....	85
การตั้งชื่อตัวแปรในไพธอน (Naming).....	86
การใช้งานตัวแปรในไพธอน (Variables).....	87
เขียนโปรแกรมครั้งแรกกับไพธอน (The first programming with Python)...	89
▶ แนะนำหน้าต่างการทำงานของ IDLE	89
▶ ความแตกต่างของ Shell Window และ Edit Window	92
▶ Python IDLE กับโค้ดหลากสี	93
▶ มือใหม่หัดใช้งาน Python IDLE	93
▶ มือใหม่หัดเขียนโปรแกรมครั้งแรก.....	96
▶ ความผิดพลาดจากการเขียนโปรแกรม (Bugs).....	101
การแสดงผลสัฟท์ทางจอภาพ	107
การรับอินพุตทางคีย์บอร์ด.....	108
สรุปท้ายบท	109
แบบฝึกหัดท้ายบท	109



Chapter 07 ชนิดข้อมูล นิพจน์ และตัวดำเนินการ

ชนิดของข้อมูล (Data Type).....112

- ▶ ข้อมูลชนิดตัวเลข (Numbers).....113

นิพจน์ (Expression).....124

- ▶ ตัวดำเนินการทางคณิตศาสตร์ (Arithmetic Operators).....126
- ▶ ตัวดำเนินการเปรียบเทียบ (Comparison Operators).....127
- ▶ ตัวดำเนินการตรรกศาสตร์ (Logical Operators).....129
- ▶ ตัวดำเนินการระดับบิต (Bitwise Operators).....131
- ▶ ตัวดำเนินการกำหนดค่า (Assignment Operators).....135
- ▶ ตัวดำเนินการชนิดพิเศษ (Special Operators).....138

ลำดับความสำคัญของตัวดำเนินการ (Precedence Order).....140

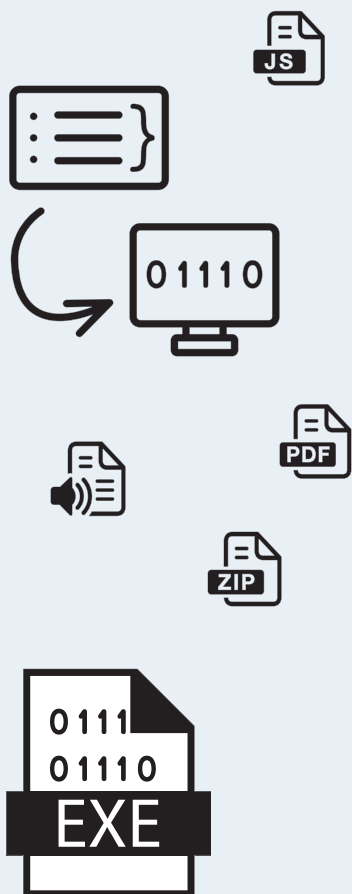
กฎการเชื่อมโยง (Associativity Rule).....141

การแปลงชนิดข้อมูล (Data Type Conversion).....142

- ▶ การแปลงชนิดข้อมูลโดยอัตโนมัติด้วยตัวแปลภาษา.....142

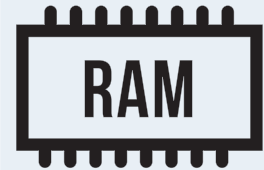
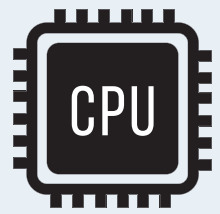
ทบทวนความเข้าใจกับการเขียนโปรแกรม (Comprehension check with programming).....149

แบบฝึกหัดท้ายบท.....157



Chapter 08 การดำเนินการกับข้อมูลในไผรอน

การดำเนินการกับข้อมูลชนิดสตริง (String)	160
▶ การประกาศตัวแปรสตริงและตัวอักษร (Declare).....	161
▶ การเข้าถึงข้อมูลแต่ละตำแหน่งในสตริง (Access)	161
▶ ตัวดำเนินการกับสตริง (Operators).....	163
▶ Built-in ฟังก์ชันกับสตริง.....	164
▶ เมธอดกับสตริง (Methods).....	164
การดำเนินการกับข้อมูลชนิดลิสต์ (List).....	165
▶ การประกาศตัวแปรลิสต์ (Declare).....	166
▶ การเข้าถึงสมาชิกในลิสต์ (Access)	167
▶ การปรับปรุงและลบสมาชิกในลิสต์ (Update & Delete)	169
▶ ตัวดำเนินการกับลิสต์ (Operators).....	170
▶ Built-in ฟังก์ชันและเมธอดกับลิสต์ (Built-in Function & Methods).....	171
การดำเนินการกับข้อมูลชนิดทูเพิล (Tuple).....	171
▶ การประกาศตัวแปรทูเพิล (Declare).....	172
▶ การเข้าถึงสมาชิกในทูเพิล (Access)	173
▶ การปรับปรุงและลบสมาชิกในทูเพิล (Update & Delete).....	174
▶ ตัวดำเนินการกับทูเพิล (Operators).....	174
การดำเนินการกับข้อมูลชนิดดิกชันนารี (Dictionary).....	175
▶ การประกาศตัวแปรดิกชันนารี (Declare)	176
▶ การเข้าถึงสมาชิกในดิกชันนารี (Access).....	177
▶ การแก้ไข เพิ่ม และลบสมาชิกในดิกชันนารี (Update, Insert and Delete)	178
▶ ฟังก์ชันและเมธอดกับดิกชันนารี (Operators).....	180



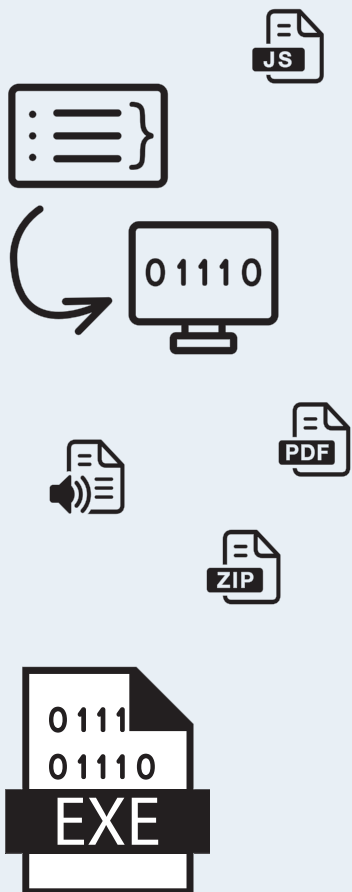
การดำเนินการกับข้อมูลชนิดเซต (Set).....	181
▶ การสร้างตัวแปรเซต (Create).....	182
▶ การเข้าถึงสมาชิกในเซต (Access).....	184
▶ การเพิ่มและลบสมาชิกในเซต (Insert and Remove).....	184
▶ ตัวดำเนินการกับเซต (Operators).....	185
สรุปท้ายบท.....	186
แบบฝึกหัดท้ายบท.....	186

Chapter 09 โครงสร้างการควบคุมแบบมีทางเลือก

if (ถ้า).....	188
if...else (ถ้า...มิฉะนั้น).....	192
if...elif (ถ้า...มิฉะนั้น ถ้า...).....	194
nested if (คำสั่ง if แบบซ้อนซ้อน).....	196
switch...case (เลือกเหตุการณ์อย่างใดอย่างหนึ่ง).....	199
สรุปท้ายบท.....	200
แบบฝึกหัดท้ายบท.....	201

Chapter 10 โครงสร้างการควบคุมแบบวนรอบทำซ้ำ

คำสั่ง while.....	204
คำสั่ง for.....	210
คำสั่ง break, continue และ else.....	217
▶ คำสั่งออกจากลูปก่อนกำหนด (break).....	218
▶ คำสั่งกระโดดข้ามลูป (continue).....	221
▶ คำสั่งกำหนดให้ทำงานหลังจากออกจากลูป (else).....	223
คำสั่ง pass (ยกเว้นการทำงาน).....	225
ลูปซ้อน (nested loop).....	226
สรุปท้ายบท.....	230
แบบฝึกหัดท้ายบท.....	230

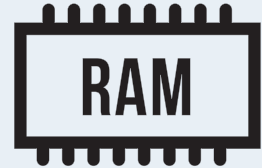
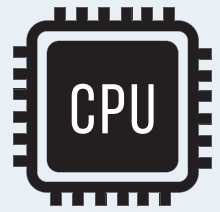


Chapter 11 ฟังก์ชัน

การสร้างและใช้งานฟังก์ชัน (Creating and using functions).....	232
การส่งค่าให้กับฟังก์ชัน (Passing values to functions).....	236
การคืนค่าจากฟังก์ชัน (Returning values from functions).....	240
ขอบเขตของตัวแปร (Scope of Variables)	248
▶ ตัวแปรแบบโลคอล (Local Variable)	249
▶ ตัวแปรแบบโกลบอล (Global Variable).....	252
สรุปท้ายบท	253
แบบฝึกหัดท้ายบท	253

Chapter 12 การจัดการแฟ้มข้อมูล

การทำงานกับไฟล์มีอะไรบ้าง	256
การเปิดไฟล์ (Open file).....	257
การอ่านข้อมูลจากเท็กซ์ไฟล์ (Reading data from a text file).....	259
▶ การอ่านไฟล์ทั้งหมดในครั้งเดียว.....	259
▶ การอ่านไฟล์ทีละบรรทัด.....	263
การเขียนข้อมูลลงเท็กซ์ไฟล์ (Writing data to a text file)	265
▶ การเขียนข้อมูลแบบต่อเนื่อง (ต่อท้ายไฟล์เดิม)	267
สรุปท้ายบท	280
แบบฝึกหัดท้ายบท	280



Chapter 13 เรียนรู้การสร้างกราฟิกง่ายๆ กับโมดูลเต๋าไพธอน

เรียนรู้การสร้างกราฟิกง่ายๆ กับโมดูลเต๋าไพธอน.....	282
เมธอดควบคุมการเคลื่อนที่ (Moving)	284
เมธอดควบคุมการวาด (Drawing).....	294
เมธอดควบคุมจอภาพ (Screen)	299
ตัวอย่างการวาดรูปต่างๆ	304
สรุปท้ายบท	311

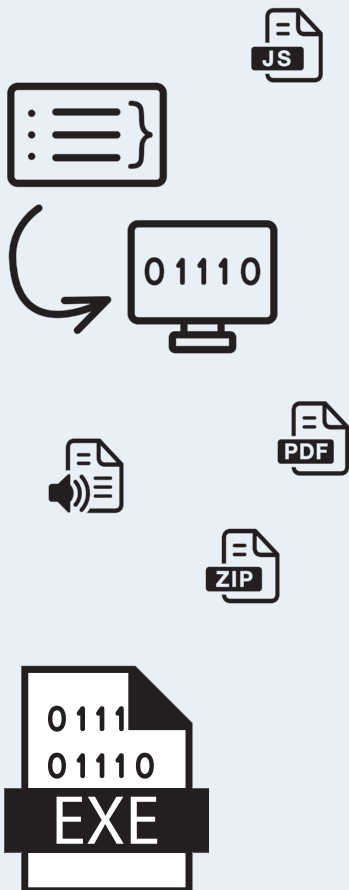
เฉลยแบบฝึกหัดท้ายบท 312

ภาคผนวก ฟังก์ชันทางคณิตศาสตร์พื้นฐานที่จำเป็น 319

บรรณานุกรม 321

ดัชนี 322

ดาวน์โหลด Source Code



วิทยาการคำนวณ วิชาใหม่ที่ทุกคนต้องได้เรียน (Computer Science for All)

ทำไมต้องเรียนวิชาวิทยาการคำนวณ ไม่อยากเป็นโปรแกรมเมอร์ทำไมต้องเรียนเขียนโปรแกรมด้วย แล้วทำไมต้องเป็นภาษาไพธอน ในเมื่อมีภาษาโปรแกรมให้เลือกมากมาย และการเขียนโปรแกรมมันยากไปไหมสำหรับเด็ก มาหาคำตอบให้หายสงสัยกันในบทนี้



สร้างทักษะใหม่ให้เด็กยุคดิจิทัล เพื่อรับมือกับโลกอนาคต

คนระดับมันสมองจากทั่วโลกต่างเห็นพ้องต้องกันว่า ทักษะทางด้านไอทีและการเขียนโปรแกรมควรจะเป็นเรื่องพื้นฐานของคนทุกอาชีพในโลกแห่งอนาคต และหากประเทศใดยกระดับความสามารถทางด้านนี้แก่ประชากรในประเทศได้ ก็จะช่วยให้ประเทศมีความมั่นคงทางเศรษฐกิจ แข่งขันกับนานาประเทศได้

แตรู้จักวิชาวิทยาการคำนวณ

ประเทศชั้นนำของโลกแทบทุกประเทศ ต่างให้ความสำคัญกับวิชาวิทยาการคำนวณ (Computing Science : CS)

ใครบ้างต้องเรียนวิชา วิทยาการคำนวณ

ตั้งแต่เดือนพฤษภาคม
พ.ศ. 2561 นักเรียน
ประถม-มัธยม ต้องเริ่ม
เรียนวิทยาการคำนวณ
โดยจะค่อยๆ ขยายไปจน
ครอบคลุมทุกระดับชั้น
ปีระหว่าง ป.1-ม.6 ในปี
พ.ศ. 2563-2564

อย่างในประเทศสหรัฐอเมริกา ก็มีการประกาศโครงการ **Computing Science For All** ที่ส่งเสริมให้นักเรียนชาวอเมริกันทุกคนมีโอกาสเรียนวิชาวิทยาการคำนวณ เพราะเชื่อมั่นว่ามันเป็นทักษะพื้นฐานที่ทุกคนต้องมีเพื่อรับมือกับโลกอนาคต ทักษะที่ว่าคือ แทนที่เด็กๆ จะใช้งานคอมพิวเตอร์ในฐานะ ‘ผู้ใช้’ เพียงอย่างเดียว แต่เปลี่ยนเป็น ‘ผู้สร้าง’ หรือ ‘นักตร’ สูการสร้างสรรค์นวัตกรรมทางเศรษฐกิจในอนาคต

ในโลกแห่งอนาคต ทุกประเทศต้องก้าวไปสู่โลกดิจิทัล อุปกรณ์เกือบทุกอย่างสามารถสั่งการได้ด้วยระบบคอมพิวเตอร์ หรือเป็น **Internet of Things (IoT)** เริ่มเข้าสู่ยุคปัญญาประดิษฐ์ (**Artificial Intelligence : AI**), **Machine Learning**, **Blockchain** และ **Big Data** เป็นเรื่องที่คนในยุคต่อไปต้องเข้าใจ และเรื่องที่สำคัญอย่างยิ่งที่ต้องพัฒนาทุกคนให้มีทักษะด้านไอที เพื่อที่จะเอาตัวรอดได้นั่นเอง และทั้งหมดนี้คือเหตุผลว่า ทำไมถึงต้องเรียนวิชาวิทยาการคำนวณ

“Computing Science for Next-Gen”

‘วิทยาการคำนวณ’ ต่างจากวิชาเดิมอย่างไร

วิชาคอมพิวเตอร์ แต่เดิมเน้นการนำเทคโนโลยีสารสนเทศและการสื่อสาร (ICT) ไปใช้ในชีวิตประจำวัน และเรียนการเขียนโปรแกรมตอน ม.1 ขึ้นไป แต่วิชาวิทยาการคำนวณมีการปรับปรุงเนื้อหาใหม่ โดยมีองค์ประกอบหลัก 3 ส่วนคือ

1 วิทยาการคอมพิวเตอร์ (Computer Science : CS)

1

การแก้ปัญหาอย่างเป็นขั้นตอนและเป็นระบบ

2 เทคโนโลยีสารสนเทศและการสื่อสาร (Information Communication Technology : ICT)

2

รวบรวมข้อมูล การประมวลผล การประเมินผล
การนำเสนอข้อมูลหรือสารสนเทศ เพื่อแก้ปัญหาในชีวิตจริง

3 การรู้เรื่องดิจิทัล (Digital Literacy : DL)

3

การใช้เทคโนโลยีสารสนเทศและการสื่อสารอย่างปลอดภัย
และการรู้เท่าทันสื่อ

วิทยาการคอมพิวเตอร์ เป็นเนื้อหาส่วนที่เกี่ยวข้องกับการคิดแก้ปัญหา และคิดสร้างสรรค์แบบนักคอมพิวเตอร์ โดยใช้ทักษะการคิดเชิงคำนวณประกอบกับการเขียนโปรแกรมคอมพิวเตอร์ นับเป็น 1 ใน 3 ส่วนของวิชาวิทยาการคำนวณ

ส่วนเทคโนโลยีสารสนเทศและการสื่อสาร และการรู้เรื่องดิจิทัลก็ปรับให้ทันสมัย ครอบคลุมการรักษาข้อมูลส่วนตัว และการรู้เท่าทันสื่อหรือข่าวลวง สร้างภูมิคุ้มกันการใช้ชีวิตในโลกยุคที่เปลี่ยนแปลงอย่างรวดเร็วและต้องเผชิญการแข่งขันสูง

วิทยาการคำนวณกับการเขียนโปรแกรม

เป็นความเข้าใจผิดที่ว่า การบรรจุวิชาวิทยาการคำนวณเข้าไปในหลักสูตรการศึกษาขั้นพื้นฐานในระดับประถมศึกษา และชั้นมัธยมศึกษาเป็นการบังคับให้เด็กทุกคนต้องเรียนเขียนโปรแกรม และก้าวสู่อาชีพโปรแกรมเมอร์ในที่สุดจริงๆ แล้วเรากำลังสร้างเด็กให้คิดอย่างเป็นระบบ มีกระบวนการคิดเชิงวิเคราะห์ รับมือกับปัญหาที่มีความซับซ้อนได้ และสามารถแก้ไขปัญหาได้อย่างสร้างสรรค์

โดยในระดับประถมนั้น เด็กยังไม่ต้องเรียนเขียนโปรแกรมอะไรเลย ยังไม่ต้องเรียนเรื่องนี้กับคอมพิวเตอร์ สำหรับเด็กประถมจึงเน้นไปที่กิจกรรม Unplugged ที่ให้เด็กคิดแก้ปัญหา เช่น ถ้าเด็กจะวางแผนเดินทางจากโรงเรียนไป

บ้านเพื่อนให้ไปถึงก่อนค่ำ แต่มีเงินอยู่ไม่มากนัก เด็กจะต้องวางแผนว่าจะขึ้นรถเมล์สายใดจึงจะประหยัดทั้งเงินและเวลา หรืออาจใช้บัตรค่า เช่น ให้เด็กใช้บัตรภาพลูกศรกับแผนที่เพื่อวางแผนเดินทางไปบ้านเพื่อน หรือใช้เกมเป็นสื่อการเรียนรู้แบบ Code.org หรือ CodingThailand.org เป็นต้น

ส่วนการเขียนโปรแกรมคอมพิวเตอร์ที่ภาษาอังกฤษเรียกว่า **Coding** หรือ **Programming** นั้น จะเริ่มต้นในระดับมัธยมเป็นต้นไป แต่ก็ปฏิเสธไม่ได้ว่า การเรียนวิชาวิทยาการคำนวณเป็นการสร้างเส้นทางสู่อาชีพโปรแกรมเมอร์ อาชีพที่มีอนาคตสดใส ค่าตอบแทนสูง สร้างโอกาสในการทำงานที่ดีของเด็กที่โตมาในยุคดิจิทัล

เป้าหมายของการเรียนวิชาวิทยาการคำนวณ

1. ต้องการพัฒนาผู้เรียนให้ใช้ทักษะการคิดเชิงคำนวณ สามารถคิด วิเคราะห์ แก้ปัญหาอย่างเป็นขั้นตอนและเป็นระบบ
2. เพื่อให้มีทักษะในการค้นหาข้อมูลหรือสารสนเทศ ประเมิน จัดการ วิเคราะห์ สังเคราะห์ และนำเสนอสารสนเทศไปใช้ในการแก้ปัญหา
3. เพื่อประยุกต์ใช้ความรู้ด้านวิทยาการคอมพิวเตอร์ สื่อดิจิทัล เทคโนโลยีสารสนเทศและการสื่อสารในการแก้ปัญหาในชีวิตจริง การทำงานร่วมกันอย่างสร้างสรรค์เพื่อประโยชน์ต่อตนเองหรือสังคม
4. ใช้เทคโนโลยีสารสนเทศและการสื่อสารอย่างปลอดภัย รู้เท่าทัน มีความรับผิดชอบ มีจริยธรรม

รู้จัก Programming Language ไละ: Python

ภาษาโปรแกรมมิ่งคืออะไร (Programming Language)

เนื่องมาจากว่า ในการทำงานของระบบคอมพิวเตอร์นั้น จำเป็นต้องมีซอฟต์แวร์เป็นสื่อกลางระหว่างเครื่องคอมพิวเตอร์ (Computer Machine) และยูสเซอร์ (User) ผู้ใช้งาน เพราะทั้งสองฝ่ายไม่สามารถสื่อสารระหว่างกันได้ คำว่า “ซอฟต์แวร์” ในที่นี้หมายถึงระบบปฏิบัติการ และโปรแกรมต่างๆ เกิดจากภาษาโปรแกรมที่ไม่ใช่ภาษาที่มนุษย์ใช้ แต่เป็นภาษาเฉพาะที่ถูกพัฒนาขึ้นมาโดยมนุษย์ ซอฟต์แวร์จึงไม่ต่างจากล่ามแปลภาษา ที่ทำให้ตัวเครื่องกับผู้ใช้งานสามารถสื่อสารระหว่างกันได้

กล่าวโดยสรุป ภาษาโปรแกรมคือ ภาษาที่ออกแบบมาเพื่อสื่อสารชุดคำสั่งไปควบคุมอุปกรณ์ฮาร์ดแวร์ เราใช้ภาษาเหล่านี้ในการสร้างโปรแกรมการใช้งานต่างๆ ขึ้นมา โดยผู้เชี่ยวชาญที่มีความรู้ความเข้าใจภาษาโปรแกรมเป็นอย่างดีที่เราเรียกว่า “โปรแกรมเมอร์ (Programmer)”

ซอฟต์แวร์เกิดขึ้นมาได้อย่างไร

ซอฟต์แวร์ถูกสร้างขึ้นมาจากภาษาโปรแกรม แต่ภาษาในที่นี้ไม่ได้หมายถึงภาษามนุษย์อย่างที่พวกเรารู้จักกัน แต่เป็นภาษาที่ถูกพัฒนาขึ้นมาโดยเฉพาะ ตัวซอฟต์แวร์เกิดจากการเขียนโปรแกรม (Programming) หรือการโค้ดดิ้ง (Coding) ด้วยภาษาโปรแกรมภาษาใดภาษาหนึ่ง โดยมีผู้เชี่ยวชาญด้านการโค้ดดิ้งคือ โปรแกรมเมอร์และนักพัฒนาซอฟต์แวร์ ที่ทำงานร่วมกันอย่างเป็นระบบจนได้ซอฟต์แวร์สำเร็จรูปเพื่อการใช้งาน

ภาษาสำหรับเขียนโปรแกรมมีอะไรบ้าง

ภาษาโปรแกรม มีอยู่ด้วยกันมากมายหลายภาษาในปัจจุบัน เช่นเดียวกับกับภาษาของมนุษย์ที่มีทั้งภาษาอังกฤษ ภาษาไทย ภาษาจีน ภาษาสเปน ฯลฯ ภาษาโปรแกรมก็มีไม่ต่ำกว่า 40 ภาษา ยกตัวอย่างเช่น Python, Java, C#, JavaScript, PHP, Swift เป็นต้น



แล้วทำไมต้องเป็นภาษาไพธอน?

ไพธอน (Python) เป็นภาษาโปรแกรมที่ใช้เขียนโปรแกรมที่ถือว่าเรียนรู้ง่ายที่สุดแล้วในยุคนี้ และองค์กรชั้นนำระดับโลกอย่าง Google ยังเลือกใช้ภาษานี้ จนกลายเป็นกระแสแพร่หลายได้รับความนิยมไปทั่วโลกอย่างรวดเร็ว และยังคงเติบโตอย่างต่อเนื่อง มีผลการสำรวจล่าสุดเป็นที่ประจักษ์ว่า ภาษาไพธอน ขึ้นแท่นเป็นอันดับหนึ่ง แข่งหน้าภาษาโปรแกรมทุกตัว ด้วยเพราะมีคุณสมบัติที่เหมาะสมสำหรับพัฒนาโปรแกรม แอปพลิเคชันในโลกยุคใหม่มากกว่านั่นเอง

ณ ปัจจุบันนี้ ภาษาที่กำลังมาแรงที่สุดในยุคคือ **ภาษาไพธอน** เนื่องจากการสำรวจของ Google Trends (พ.ศ. 2561) พบว่าภาษาไพธอนได้รับความนิยมมากกว่าภาษาจาวา ซึ่งเป็นภาษาที่มีผู้ใช้งานมากที่สุดไปแล้ว

สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี (สสวท.) แนะนำให้สอนเขียนโปรแกรมด้วยภาษาไพธอน ก็เพราะว่าภาษาไพธอนเป็นภาษาที่มีโครงสร้างและรูปแบบคำสั่งที่ไม่ซับซ้อน เข้าใจง่าย มีความใกล้เคียงกับภาษาอังกฤษที่มนุษย์เราใช้ในการสื่อสารกันมากที่สุด และสามารถนำมาใช้ในการเขียนโปรแกรมที่มีความซับซ้อนได้เช่นเดียวกับภาษาการเขียนโปรแกรมอื่นๆ ส่วนในระดับมหาวิทยาลัยทั้งในและต่างประเทศ ต่างแนะนำไพธอนเป็นภาษาแรกในการเรียนเขียนโปรแกรม

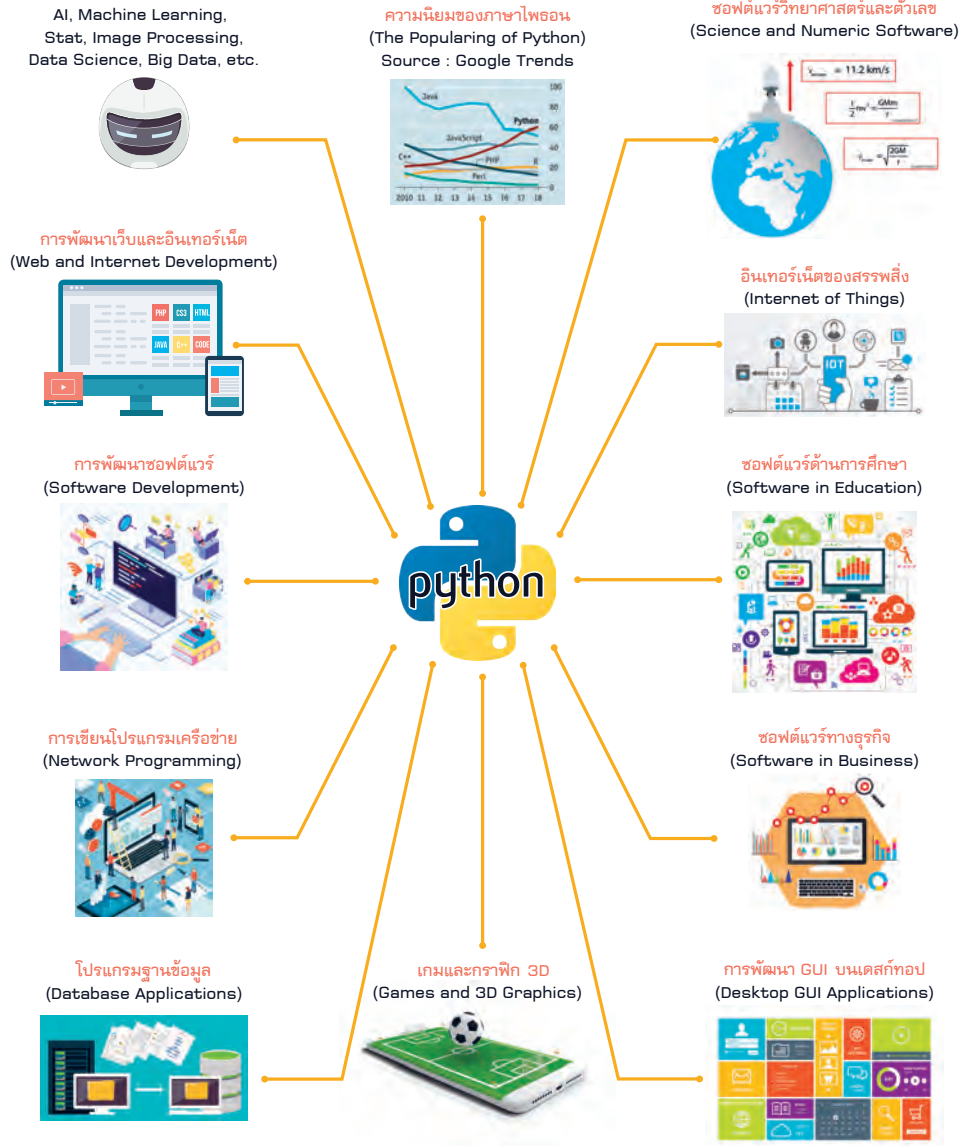


ประเทศไคเรียน Computing Science บ้าง

ประเทศที่บรรจุวิชาไว้หลักสูตรระดับชาติแล้ว ได้แก่ ฟินแลนด์ สหรัฐอเมริกา อังกฤษ เอสโตเนีย เกาหลีใต้ รวมถึงประเทศจีนที่ให้เด็กชั้นมัธยมศึกษาตอนปลายกว่า 40 โรงเรียน ได้เรียนวิชา AI แล้ว

ไพธอนทำอะไรได้บ้าง

เหตุที่ภาษาไพธอนได้รับความนิยมอย่างสูงสุด และเติบโตอย่างรวดเร็วที่สุด ก็เพราะมันสามารถนำไปประยุกต์ใช้ในงานได้อย่างกว้างขวาง และมีแนวโน้มที่จะเติบโตไปเรื่อยๆ ในอนาคต



▲ ไพธอนเป็นภาษาที่ครอบคลุมการทำงานเกือบทุกด้าน

“ไพทอน” กับ “ไพธอน”
เขียนแบบไหนถูก

คำว่า ‘Python’ ในภาษาไทย เราจะเขียนกันอยู่ 2 แบบ ได้แก่ ‘ไพทอน’ กับ ‘ไพธอน’ เมื่อไปตรวจสอบตามหลักภาษาแล้วพบว่า ‘th’ นั้น แทนด้วยอักษรได้ 3 ตัวคือ ท ธ ฐ แต่ ‘ฐ’ บ้านเราไม่นิยมใช้ จึงอนุโลมใช้ได้ทั้ง 2 แบบ

“

หากเปรียบ “เสาและคาน” เป็นรากฐานที่สำคัญของบ้านฉันใด “พื้นฐานการเขียนโปรแกรม” ก็เป็นหัวใจของการพัฒนาโปรแกรมฉันนั้นเช่นกัน เพราะฉะนั้นเป้าหมายหลักของหนังสือเล่มนี้คือ การปูพื้นฐานการเขียนโปรแกรมด้วยไพธอน เพื่อนำความรู้ที่ได้ไปต่อยอดสู่การเขียนโปรแกรมขั้นสูงต่อไป

”

สรุปท้ายบท

การเรียนวิชาวิทยาการคำนวณ ไม่ว่าผู้เรียนจะชอบหรือไม่ชอบก็ตาม แต่อย่างน้อยจะได้ประโยชน์ในเรื่องของการคิด คิดอย่างเป็นระบบ หรือคิดเป็นแก้ปัญหาเป็น นำไปประยุกต์ใช้กับชีวิตประจำวันได้ทันทีในทุกสาขาอาชีพ แต่จะได้ประโยชน์มากสำหรับคนที่นำความรู้นี้ไปต่อยอดเรียนในสาขาวิทยาการคอมพิวเตอร์ สาขาวิศวกรรมซอฟต์แวร์ หรือสาขาทางไอที ที่ต้องเรียนวิชาเขียนโปรแกรมในระดับที่สูงขึ้น และอาจใช้ความรู้นี้เพื่อทำงานในฐานะโปรแกรมเมอร์ หรือนักพัฒนาโปรแกรม ซึ่งเป็นอาชีพที่มีอนาคต และรายได้สูงเป็นอันดับต้นๆ เลยทีเดียว



คอมพิวเตอร์เบื้องต้น ก่อนเขียนโปรแกรม

(Basic understanding before writing
computer programs)

ก่อนจะเริ่มเรียนเขียนโปรแกรม ต้องเริ่มต้นเรียนรู้และเข้าใจในพื้นฐานของระบบคอมพิวเตอร์เสียก่อน เพื่อให้รู้จักองค์ประกอบของระบบ ส่วนประกอบของเครื่องคอมพิวเตอร์มีอะไรบ้าง และหน้าที่การทำงานของอุปกรณ์แต่ละชิ้น จำแนกได้ว่าจะอะไรคือสิ่งที่เรียกว่า “ฮาร์ดแวร์” และอะไรคือ “ซอฟต์แวร์” และเรียนรู้กระบวนการที่เกิดขึ้นเมื่อโปรแกรมต่างๆ ทำงานบนคอมพิวเตอร์ ทั้งหมดนี้คือพื้นฐานเบื้องต้นที่จำเป็นต้องเรียนรู้ให้เข้าใจก่อนเป็นลำดับแรก



วิวัฒนาการคอมพิวเตอร์

(Computers in the Revolution)

เทคโนโลยีทางด้านคอมพิวเตอร์เกิดขึ้นมากกว่า 70 ปี เราสามารถแบ่งการพัฒนาคอมพิวเตอร์จากอดีตถึงปัจจุบันได้ 4 ยุคด้วยกัน



พ.ศ. 2487
เครื่อง MARK I



พ.ศ. 2489
เครื่อง ENIAC



พ.ศ. 2492
เครื่อง EDVAC



พ.ศ. 2494
เครื่อง UNIVAC

ยุคที่ 1 หลอดสุญญากาศ (พ.ศ. 2487-2501) เป็นคอมพิวเตอร์ที่ใช้หลอดสุญญากาศจำนวนมาก เพื่อให้เครื่องสามารถทำงานได้อย่างรวดเร็ว แต่ต้องใช้พลังงานไฟฟ้าในการทำงานสูง ทำให้มีปัญหาเรื่องความร้อน

ปี พ.ศ. 2487 **เครื่องมาร์ควัน (MARK I)** พัฒนาโดย Howard Akien แห่งมหาวิทยาลัยฮาร์วาร์ด ร่วมกับวิศวกรของ IBM โดยเครื่องดังกล่าวทำงานแบบเครื่องจักรกลและไฟฟ้า

ปี พ.ศ. 2489 **เครื่องอีนิแอค (ENIAC : Electronic Numerical Integration And Calculator)** พัฒนาโดย John W. Mauchly และ Persper Eckert วิศวกรจากมหาวิทยาลัยเพนซิลวาเนีย ทั้งสองได้พัฒนาเครื่องคอมพิวเตอร์ที่สร้างจากอุปกรณ์อิเล็กทรอนิกส์เป็นเครื่องแรกของโลก

ปี พ.ศ. 2492 **เครื่องเอ็ดวาค (EDVAC : Electronic Discrete Variable Automatic Calculator)** พัฒนาโดย Dr. John Von Neumann ผู้เสนอแนวคิดในการสร้างคอมพิวเตอร์ที่มีหน่วยความจำที่ใช้เก็บข้อมูล และโปรแกรมการทำงานหรือชุดคำสั่ง

ปี พ.ศ. 2494 **เครื่องยูนิแอค (UNIVAC : Universal Automatic Computer)** เป็นคอมพิวเตอร์เครื่องแรกที่ผลิตในเชิงพาณิชย์เพื่อขายหรือให้เช่า และเริ่มมีการซื้อขายเพื่อใช้งานอย่างแพร่หลาย



พ.ศ. 2502

เครื่อง Mainframe Computer

ยุคที่ 2 ทรานซิสเตอร์ (พ.ศ. 2502-2506) เป็นคอมพิวเตอร์ที่ใช้ทรานซิสเตอร์ (Transistors) แทนหลอดสุญญากาศ เพราะทรานซิสเตอร์มีขนาดเล็ก ใช้พลังงานไฟฟ้าต่ำ ทำงานได้เร็วกว่า คงทนกว่า เชื่อถือได้ และมีราคาถูก

ปี พ.ศ. 2490 คอมพิวเตอร์ที่ใช้ทรานซิสเตอร์ถูกสร้างสำเร็จ โดยทีมนักวิทยาศาสตร์จากห้องปฏิบัติการเบล (Bell Lab) 3 คน ได้แก่ Dr. John Bardeen, Dr. Walter Brattain, Dr. William Shockley ส่วนทางด้านซอฟต์แวร์มีการพัฒนาภาษาคอมพิวเตอร์ขึ้นมาใหม่คือ ภาษาแอสเซมบลี (Assembly), ภาษาฟอร์แทรน (Fortran), ภาษาโคบอล (COBOL) เป็นต้น และในยุคนี้องค์ได้มีการสร้างเมนเฟรมคอมพิวเตอร์ (Mainframe Computer)



พ.ศ. 2507
เครื่อง Minicomputer

ยุคที่ 3 วงจรรวม (พ.ศ. 2507-2512) เป็นช่วงเวลาที่มีการสร้างทรานซิสเตอร์จำนวนมากลงบนแผ่นซิลิคอนขนาดเล็ก และเกิดวงจรรวมบนแผ่นซิลิคอนที่เรียกว่า **“ไอซี” (IC : Integrated Circuit)** โดยวงจรรวมแต่ละตัวจะมีทรานซิสเตอร์บรรจุอยู่ภายในจำนวนมาก

การใช้ไอซีเป็นส่วนประกอบทำให้เครื่องคอมพิวเตอร์มีความซับซ้อนขึ้น ประสิทธิภาพในการทำงานดีขึ้น และสามารถลดขนาดของตัวเครื่องให้เล็กลงได้อีกด้วย ต้นทุนในการผลิตก็ถูกลง คอมพิวเตอร์ในยุคไอซีถูกเรียกว่า **มินิคอมพิวเตอร์ (Minicomputer)**



พ.ศ. 2514
เครื่อง Microcomputer

ยุคที่ 4 VLSI (พ.ศ. 2514-ปัจจุบัน) เป็นยุคที่มีการสร้างวงจรมารวมในแผ่นซิลิคอน เรียกว่า **“วีแอลเอสไอ” (VLSI : Very Large Scale Integration)** เป็นวงจรรวมที่รวมเอาทรานซิสเตอร์นับแสนนับล้านตัวมาบรรจุอยู่ในแผ่นซิลิคอนขนาดเล็ก และผลิตเป็นหน่วยประมวลผลที่เรียกว่า **“ไมโครโปรเซสเซอร์” (Microprocessor)**

การใช้ VLSI เป็นวงจรรวมในเครื่องคอมพิวเตอร์ ทำให้เครื่องมีขนาดเล็กลง สามารถตั้งบนโต๊ะได้ เรียกว่า **“ไมโครคอมพิวเตอร์” (Microcomputer)** และส่งผลให้เกิดคอมพิวเตอร์ส่วนบุคคล (PC : Personal Computer) ที่มีการใช้งานกันอย่างแพร่หลาย และมีผู้ใช้งานกันทั่วโลก

ส่วนประกอบของเครื่องคอมพิวเตอร์

เราสามารถแบ่งองค์ประกอบของระบบคอมพิวเตอร์ออกได้เป็น 2 ส่วนหลักๆ คือ ส่วนของคอมพิวเตอร์ฮาร์ดแวร์ และส่วนของคอมพิวเตอร์ซอฟต์แวร์ โดยทั้งสองส่วนนี้จะต้องมีการทำงานร่วมกัน เพื่อให้เกิดการตอบสนองต่อการสั่งงานจากผู้ใช้ ซึ่งมีรายละเอียดดังนี้

คอมพิวเตอร์ฮาร์ดแวร์

คอมพิวเตอร์ฮาร์ดแวร์ (Computer Hardware) หมายถึง ส่วนประกอบของเครื่องคอมพิวเตอร์ ซึ่งก็คือ ชิ้นส่วนอุปกรณ์ต่างๆ (Devices) ที่ประกอบกันขึ้นเป็นเครื่องคอมพิวเตอร์นั่นเอง ทั้งส่วนประกอบภายในเครื่อง ส่วนประกอบภายนอกเครื่อง หรืออุปกรณ์ต่อพ่วง

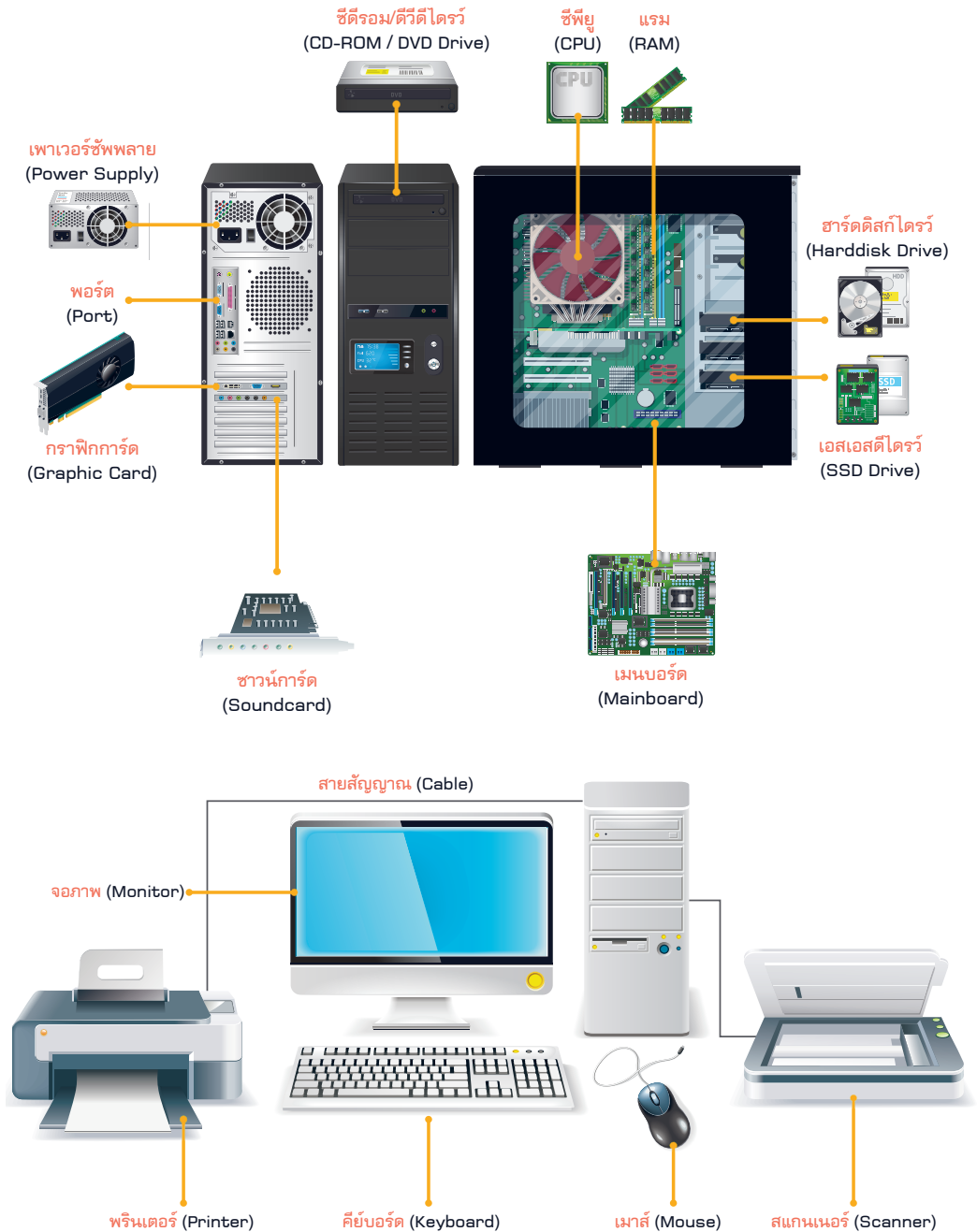
ส่วนประกอบภายใน

ส่วนประกอบภายใน (Internal Hardware) หรือ (Internal Devices) เป็นอุปกรณ์คอมพิวเตอร์ที่ประกอบกันอยู่ภายในตัวเครื่อง หากต้องการรู้ว่าอุปกรณ์เหล่านี้มีอะไรบ้าง ก็จำเป็นต้องเปิดฝาครอบเครื่องออก จะพบว่ามีอุปกรณ์ต่างๆ มากมาย ติดตั้งอยู่บนแผงวงจรขนาดใหญ่ที่เรียกว่า **“เมนบอร์ด (Mainboard)”** ต่อไปนี้มาทำความรู้จักกับชิ้นส่วนต่างๆ ภายในเครื่อง PC ว่าแต่ละส่วนเรียกว่าอะไร และทำหน้าที่อย่างไร

ส่วนประกอบภายนอก

อุปกรณ์ภายนอก (External Hardware) หรือ (External Devices) เป็นอุปกรณ์ที่เชื่อมต่อกันอยู่ภายนอกตัวเครื่องคอมพิวเตอร์ โดยมีสายสัญญาณต่อพ่วงระหว่างตัวเครื่องและอุปกรณ์ จึงนิยมเรียกอุปกรณ์เหล่านี้ว่า **“อุปกรณ์ต่อพ่วง (Peripherals)”** นั่นเอง ต่อไปนี้คือ อุปกรณ์ภายนอกมาตรฐาน





▲ ส่วนประกอบของเครื่องคอมพิวเตอร์

- **ซีพียู (CPU)** เป็นสมองของคอมพิวเตอร์ ทำหน้าที่ประมวลผลข้อมูล และควบคุมการทำงานของระบบ ประสิทธิภาพ และความเร็วในการทำงานของเครื่องจะขึ้นอยู่กับซีพียูเป็นหลัก
- **แรม (RAM)** เป็นที่พักของข้อมูลและคำสั่งที่จำเป็นต้องใช้งาน ซึ่งข้อมูลและโปรแกรมจะถูกเก็บอย่างถาวรอยู่ในฮาร์ดดิสก์ แต่จะถูกโหลดมาไว้ในที่แรมแบบชั่วคราวเฉพาะที่ใช้งาน เพื่อรอส่งต่อให้ซีพียูประมวลผล เหตุที่ซีพียูไม่โหลดจากฮาร์ดดิสก์โดยตรงก็เพราะว่า ฮาร์ดดิสก์ทำงานช้ากว่าซีพียูและแรมมากนั่นเอง เครื่อง PC ที่มีแรมมากจะช่วยให้ระบบทำงานได้อย่างรวดเร็ว
- **เมนบอร์ด (Mainboard) หรือมารเธอร์บอร์ด (Motherboard)** เป็นเหมือนศูนย์กลางเชื่อมต่ออุปกรณ์ทั้งหมดของเครื่อง PC มีลักษณะเป็นแผ่นวงจรขนาดใหญ่ที่ใช้ติดตั้งอุปกรณ์ต่างๆ เช่น ซีพียู แรม ชิปเซต และการ์ดต่างๆ รวมทั้งเชื่อมต่อกับอุปกรณ์ต่อพ่วงมากมาย เช่น เพาเวอร์ซัพพลาย ฮาร์ดดิสก์ ซีดีรอมไดรฟ์ คีย์บอร์ด เมาส์ เป็นต้น
- **ฮาร์ดดิสก์ไดรฟ์ (Harddisk Drive)** เป็นอุปกรณ์เก็บข้อมูลหลักของเครื่อง PC มีความจุสูง เก็บข้อมูลได้อย่างถาวร ข้อมูลที่ว่าหมายถึง ระบบปฏิบัติการ โปรแกรม ไฟล์ต่างๆ เช่น ไฟล์ข้อมูล ไฟล์ภาพ ไฟล์วิดีโอ เป็นต้น
- **เอสเอสดีไดรฟ์ (SSD Drive)** นั้นย่อมาจาก Solid State Drive เป็นฮาร์ดดิสก์ที่มีเทคโนโลยีการเก็บข้อมูลแบบ Flash Memory คือ อ่านเขียนลงในชิป ซึ่งทำงานเร็วกว่าฮาร์ดดิสก์จานหมุนแบบเดิม นอกจากเรื่องของความเร็วแล้ว ฮาร์ดดิสก์ SSD ยังมีความทนทานกว่า หากถูกกระแทก
- **ซีดีรอม/ดีวีดีไดรฟ์ (CD-ROM/DVD Drive)** เป็นอุปกรณ์ที่ใช้อ่านข้อมูลจากแผ่นซีดีหรือดีวีดีประเภทต่างๆ เช่น แผ่นติดตั้งโปรแกรม แผ่นเพลง หรือแผ่นภาพยนตร์ เป็นต้น มีให้เลือกทั้งแบบอ่านได้อย่างเดียว หรือทั้งอ่านและเขียนแผ่น (Read/Write : RW Drives)
- **ซาว์นการ์ด (Sound Card)** เป็นอุปกรณ์ที่ทำให้คอมพิวเตอร์สามารถฟังเพลง ดูหนัง หรือเล่นเกมได้ ซาว์นการ์ดที่มีคุณภาพดีก็จะให้เสียงที่มีคุณภาพสมจริง มีให้เลือกทั้งแบบบิวต์อิน (Built-in) และซาว์นการ์ด (Sound Card)

- **การ์ดจอ (Monitor Card) หรือกราฟิกการ์ด (Graphic Card)** เป็นการ์ดที่ทำหน้าที่โหลดข้อมูลมาแสดงผลบนจอภาพในรูปแบบกราฟิก แต่เนื่องจากข้อมูลกราฟิกสมัยใหม่นั้นมีสีที่เหมือนจริง คุณภาพสูง ซึ่งต้องใช้พลังในการคำนวณและประมวลผลสูง กราฟิกการ์ดจึงมีส่วนที่เรียกว่า **“GPU (Graphic Processing Unit)”** เพื่อเป็นสมองกลด้านกราฟิกโดยเฉพาะ จะดึงงานส่วนนี้แยกออกมาจากซีพียู
- **เพาเวอร์ซัพพลาย (Power Supply)** เป็นแหล่งพลังงานของอุปกรณ์ทั้งหมดในเครื่อง PC โดยจะทำหน้าที่เป็นอะแดปเตอร์ เพื่อแปลงพลังงานไฟฟ้ากระแสสลับให้เป็นพลังงานไฟฟ้ากระแสตรง จ่ายให้กับอุปกรณ์วงจรอิเล็กทรอนิกส์ต่างๆ
- **คีย์บอร์ด (Keyboard)** เป็นอุปกรณ์ป้อนข้อมูลที่เป็นตัวอักษร ทั้งตัวหนังสือ ตัวเลข เครื่องหมาย สัญลักษณ์ และปุ่มฟังก์ชันการทำงานต่างๆ เช่น ปุ่ม Esc ยกเลิก, ปุ่ม F1/F2/F3 ปิด-ลด-เพิ่มเสียง, ปุ่ม F11/F12 ลด-เพิ่มแสง เป็นต้น
- **เมาส์ (Mouse)** เป็นอุปกรณ์อำนวยความสะดวกในการกำหนดตำแหน่งการทำงานบนหน้าจอ เช่น เลือกตำแหน่งพิมพ์ เลือกข้อความ คลิกเลือกเมนู คลิกเลือกคำสั่ง ดับเบิลคลิก เพื่อเปิดโปรแกรม เป็นต้น เป็นการดึงหน้าที่บางอย่างออกมาจากคีย์บอร์ด ทั้งนี้เพื่อให้การทำงานสะดวกเร็วขึ้น
- **จอภาพ (Monitor)** เป็นอุปกรณ์แสดงผลข้อมูลเพื่อตอบโต้การทำงานกับยูสเซอร์
- **พริ้นเตอร์ (Printer)** เป็นอุปกรณ์พิมพ์ข้อความและภาพลงบนกระดาษ
- **สแกนเนอร์ (Scanner)** เป็นอุปกรณ์อ่านข้อมูลจากกระดาษและบันทึกเป็นไฟล์ภาพ มีความคล้ายกับเครื่องซีร็อกซ์ แต่ต่างกันตรงที่เครื่องซีร็อกซ์เป็นการก๊อปปี้จากกระดาษสู่กระดาษ แต่สแกนเนอร์จะก๊อปปี้จากกระดาษสู่ไฟล์ภาพ



ฟลอปปีดิสก์ไดรฟ์ (Floppy Disk Drive)

เป็นไดรฟ์สำหรับอ่านแผ่นบันทึกข้อมูลที่เรียกว่า “แผ่นฟลอปปีดิสก์” หรือเรียกสั้นๆ ว่า “แผ่นดิสก์” ซึ่งมีความจุในการเก็บข้อมูลน้อยมาก ไม่นิยมใช้งานแล้วในปัจจุบัน



คอมพิวเตอร์ซอฟต์แวร์

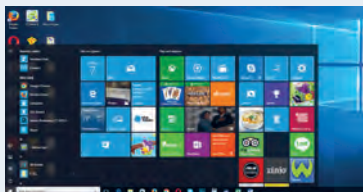
คอมพิวเตอร์ซอฟต์แวร์ (Computer Software) หมายถึง โปรแกรมหรือชุดคำสั่งที่ใช้สั่งงานให้คอมพิวเตอร์ทำงานตามที่เราต้องการ ซึ่งหมายถึงระบบปฏิบัติการ (Operating System : OS) และโปรแกรมต่างๆ ที่เราใช้งานกันอยู่ทุกวัน ไม่ว่าจะเป็นระบบปฏิบัติการ เช่น วินโดวส์ (Windows OS), ลินุกซ์ (Linux OS) หรือโปรแกรม เช่น ไมโครซอฟท์ เวิร์ด (Microsoft Word), ไมโครซอฟท์ เอ็กเซล (Microsoft Excel) เป็นต้น สำหรับซอฟต์แวร์ในระบบคอมพิวเตอร์แบ่งออกเป็น 2 ประเภทคือ ซอฟต์แวร์ระบบปฏิบัติการ และซอฟต์แวร์ประยุกต์

ซอฟต์แวร์ระบบปฏิบัติการ

ระบบปฏิบัติการ หรือโอเอส เป็นซอฟต์แวร์ที่ทำหน้าที่ควบคุมการทำงานของฮาร์ดแวร์ทั้งหมด โดยคอมพิวเตอร์ทุกเครื่องจะต้องติดตั้งระบบปฏิบัติการอย่างใดอย่างหนึ่งเสมอ ซึ่งปัจจุบันมีระบบปฏิบัติการหลายสายพันธุ์ที่ได้รับความนิยม เช่น ยูนิกซ์ (Unix), ลินุกซ์ (Linux), วินโดวส์ (Windows), ไอโอเอส (iOS), แอนดรอยด์ (Android) เป็นต้น



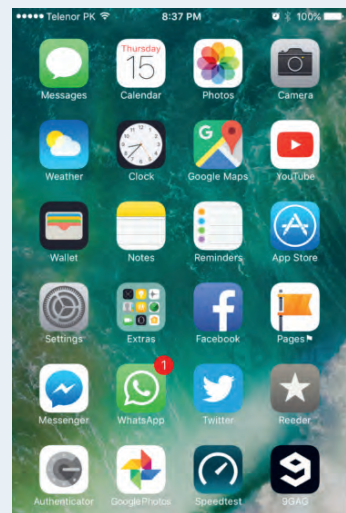
ยูนิกซ์/ลินุกซ์



วินโดวส์



ไอโอเอส



แอนดรอยด์

▲ ระบบปฏิบัติการสายพันธุ์ต่างๆ

ซอฟต์แวร์ประยุกต์ (Application Software)

ซอฟต์แวร์ประยุกต์คือ ชุดคำสั่งที่พัฒนาขึ้นเพื่อสั่งให้เครื่องคอมพิวเตอร์ทำงานเฉพาะด้านตามที่ผู้ต้องการ ซอฟต์แวร์ประยุกต์จำแนกได้เป็น 2 ประเภทคือ

ซอฟต์แวร์เฉพาะด้าน (Special Purpose)

ถูกพัฒนาให้เหมาะสมกับงานเฉพาะด้าน เช่น โปรแกรมบัญชีสำเร็จรูป โปรแกรมสำหรับการซื้อขายในร้านค้า โปรแกรมจัดการสต็อกสินค้า โปรแกรมการทำบัญชีจ่ายเงินเดือน และ โปรแกรมระบบเช่าซื้อ เป็นต้น ตัวอย่างโปรแกรมเฉพาะด้าน เช่น

- EASY-ACC
- Payroll and Salary
- Inventory Management
- Coffee Shop POS
- iSales
- iPurchases
- TaxBiz

ซอฟต์แวร์สำหรับงานทั่วไป (General Purpose)

หรือเรียกว่า ซอฟต์แวร์สำเร็จรูป เป็นซอฟต์แวร์นำมาประยุกต์ใช้กับงานส่วนบุคคลได้อย่างหลากหลาย ผู้ใช้ไม่จำเป็นต้องเขียนโปรแกรมเอง ไม่ต้องเสียเวลาในการเรียนรู้มาก ตัวอย่างโปรแกรมสำเร็จรูปที่นิยม เช่น

- Microsoft Word
- Microsoft Excel
- Microsoft PowerPoint
- Photoshop
- SPSS
- Google Chrome
- Games

การทำงานของระบบคอมพิวเตอร์

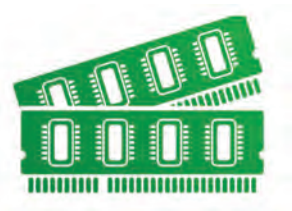
หลังจากได้รู้จักองค์ประกอบ และส่วนประกอบของเครื่องคอมพิวเตอร์ไปเป็นที่เรียบร้อยแล้ว ในหัวข้อนี้เรามาดูกันต่อว่า คอมพิวเตอร์ทำงานอย่างไร

เราสามารถแบ่งระบบคอมพิวเตอร์ออกได้เป็น 3 ส่วน ได้แก่ หน่วยประมวลผลกลาง (Central Processing Unit : CPU) หรือโปรเซสเซอร์ (Processor), หน่วยความจำ (Memory) และอุปกรณ์รับข้อมูล/แสดงผล (Input/Output) ซึ่งแต่ละส่วนมีหน้าที่แตกต่างกันดังนี้



หน่วยประมวลผลกลาง

หน่วยประมวลผลกลาง หรือซีพียู (Central Processing Unit : CPU) เป็นส่วนที่ทำหน้าที่ในการประมวลผลข้อมูล และควบคุมการทำงานของระบบ เปรียบเหมือนกับสมองของคนที่ทำหน้าที่ในการคิด คำนวณ และควบคุมการทำงานทุกชิ้นส่วนของระบบคอมพิวเตอร์



หน่วยความจำ

หน่วยความจำ (Memory) เป็นส่วนที่ใช้สำหรับเก็บข้อมูลและโปรแกรมต่างๆ มีทั้งแบบที่ใช้เก็บข้อมูลแบบถาวรและแบบชั่วคราว หน่วยความจำเป็นส่วนที่มีความจำเป็นต่อการทำงานของระบบคอมพิวเตอร์ ถ้าไม่มีหน่วยความจำคอมพิวเตอร์ก็เปรียบเหมือนกับคนที่ไม่มียังมีส่วนที่ใช้จดจำ เราแบ่งหน่วยความจำออกเป็น 2 ประเภท ได้แก่ หน่วยความจำหลักและหน่วยความจำสำรอง ซึ่งมีรายละเอียดดังต่อไปนี้