

หนังสือเรียนรายวิชาเพิ่มเติม

กลุ่มสาระการเรียนรู้การงานอาชีพและเทคโนโลยี

ตามหลักสูตรแกนกลางการศึกษาขั้นพื้นฐาน พุทธศักราช ๒๕๕๑

พื้นฐานการเขียน โปรแกรมคอมพิวเตอร์ Visual Basic 2015

ผู้เรียบเรียง

พัฒพงษ์ อมรวงศ์

ผู้ตรวจ

๑. วิเชียร วิสูงเร

๒. วิวัฒน์ สุนันทวงศ์

๓. ภูซังค์ จันทร์เปล่ง

บรรณาธิการ

รุ่งทิวา ศิริনারรัตน์



บริษัท มีเดีย อินเทลลิเจนซ์ เทคโนโลยี จำกัด
Media Intelligence Technology Co.,Ltd.

หนังสือเรียนรายวิชาเพิ่มเติม
กลุ่มสาระการเรียนรู้การงานอาชีพและเทคโนโลยี
ตามหลักสูตรแกนกลางการศึกษาขั้นพื้นฐาน พุทธศักราช ๒๕๕๑

พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ Visual Basic 2015

ผู้เรียบเรียง

พัฒพงษ์ อมรวงศ์

ผู้ตรวจ

๑. วิเชียร วิสูงเร

๒. วิวัฒน์ สุนันทวงศ์

๓. ฤชงค์ จันทร์เปล่ง

บรรณาธิการ

รุ่งทิวา ศิริณารัตน์

ราคา ๙๙ บาท

สงวนลิขสิทธิ์ตามพระราชบัญญัติ

ข้อมูลทางบรรณานุกรมของหอสมุดแห่งชาติ

พัฒพงษ์ อมรวงศ์.

หนังสือเรียน พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ Visual Basic 2015.

- ปทุมธานี : มีเดีย อินเทลลิเจนซ์ เทคโนโลยี, ๒๕๕๙.

๒๒๔ หน้า.

๑. การเขียนโปรแกรมคอมพิวเตอร์ I. ชื่อเรื่อง

ISBN 978-616-7319-58-2

จัดพิมพ์และจัดจำหน่ายโดย



บริษัท มีเดีย อินเทลลิเจนซ์ เทคโนโลยี จำกัด

Media Intelligence Technology Co.,Ltd.

๔๙/๘๓ หมู่ ๗ ต.คลองสอง อ.คลองหลวง จ.ปทุมธานี ๑๒๑๒๐

โทรศัพท์ ๐-๒๕๑๖-๕๖๙๐-๑ โทรสาร ๐-๒๕๑๖-๕๖๙๒

www.mitmedia.com E-mail : mediamit@yahoo.com

ออกแบบ : KanomDesign Studio

พิมพ์ที่ : บริษัท เกลียวการพิมพ์ (1988) จำกัด

คำนำ

หนังสือเรียน พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ Visual Basic 2015 นี้จัดทำขึ้นเพื่อใช้ประกอบการเรียนการสอนวิชาการเขียนโปรแกรมคอมพิวเตอร์ วิชาเทคโนโลยีสารสนเทศหรือวิชาที่เกี่ยวข้อง สำหรับโรงเรียน หรือสถานศึกษาทุกระดับซึ่งจัดการเรียนการสอนที่ให้ความสำคัญกับวิชาการเขียนโปรแกรมคอมพิวเตอร์เป็นวิชาหลัก

ในการเขียนหนังสือเล่มนี้ ผู้เขียนได้เน้นเนื้อหาในเชิงการฝึกปฏิบัติเป็นหลัก โดยออกแบบขั้นตอนการให้เนื้อหาจากง่ายไปยากให้ผู้เรียนปฏิบัติตาม และมีภาพประกอบเพื่อเสริมการเรียนรู้ ผู้เรียนจึงสามารถใช้หนังสือเล่มนี้เป็นคู่มือการเรียนรู้ได้ทั้งการเรียนรู้ด้วยตนเอง และฝึกปฏิบัติในชั้นเรียน แต่ละบทมีกิจกรรมและการประเมินผลการเรียนรู้ให้ผู้เรียนได้ฝึกทักษะและทบทวนความรู้ อันจะทำให้ผู้เรียนมีความรู้ ความเข้าใจและมีทักษะการเขียนโปรแกรมมากยิ่งขึ้น

หนังสือเรียน พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ Visual Basic 2015 นี้มีเนื้อหาครอบคลุมตั้งแต่ หลักการเขียนโปรแกรม องค์ประกอบของโปรแกรม Visual Basic 2015 กระบวนการเขียนโปรแกรม การเขียนโปรแกรม (Code) ให้ตัวควบคุม การเขียนโปรแกรมคำนวณ การเขียนโปรแกรมตามเงื่อนไข การเขียนโปรแกรมทำซ้ำ การใช้ตัวแปรชุด (Array) และการออกแบบและเขียนโปรแกรมอย่างง่ายในงานธุรกิจ ซึ่งทั้งหมดนี้เป็นพื้นฐานที่สำคัญในการเขียนโปรแกรม Microsoft Visual Basic 2015 เหมาะสำหรับนักเรียน นักศึกษา และผู้ที่สนใจศึกษาเกี่ยวกับการเขียนโปรแกรม Microsoft Visual Basic 2015 โดยไม่จำเป็นต้องมีพื้นฐานมาก่อน

ผู้เขียนหวังเป็นอย่างยิ่งว่า หนังสือเรียนเล่มนี้จะเป็นประโยชน์กับทั้งนักเรียน นักศึกษา ครูผู้สอน และผู้สนใจทั่วไป จะได้รับความรู้ในเชิงปฏิบัติการที่ทันสมัย และเป็นแนวทางให้ผู้เรียนได้พัฒนาศักยภาพในการเขียนโปรแกรม Microsoft Visual Basic 2015 ให้ก้าวหน้ามากขึ้นต่อไป ทั้งนี้หากมีข้อผิดพลาด หรือข้อเสนอแนะเพิ่มเติมประการใด กรุณาติดต่อไปยังผู้เขียนได้โดยตรง จักเป็นพระคุณอย่างยิ่ง

พัฒพงษ์ อมรวงศ์
apatpong@gmail.com

สารบัญ

บทที่ 1 หลักการเขียนโปรแกรม 9

1.1 คอมพิวเตอร์และส่วนประกอบ	10
1.2 โปรแกรมคอมพิวเตอร์ (Computer Program)	11
1.3 ภาษาคอมพิวเตอร์ (Computer Languages)	12
1.4 ตัวแปลภาษา (Translator)	14
1.5 ขั้นตอนการพัฒนาโปรแกรม	15
1.6 การวิเคราะห์งาน (Job Analysis)	17
1.7 ขั้นตอนการแก้ปัญหา (Algorithm)	20
1.8 การเขียนผังงาน (Flowchart)	24
1.9 การเขียนรหัสเทียม (Pseudo Code)	28
การวัดและประเมินผลการเรียนรู้ บทที่ 1	31

บทที่ 2 องค์ประกอบของโปรแกรม Visual Basic 2015 32

2.1 ความเป็นมาของ Visual Basic 2015	33
2.2 ลักษณะการสร้างโปรแกรมใช้งานบนวินโดวส์ด้วย Visual Basic 2015	33
2.3 ความต้องการของระบบ (System Requirements)	34
2.4 การดาวน์โหลดและการติดตั้งโปรแกรม	35
2.5 การเข้าสู่โปรแกรม Visual Studio 2015	37

2.6 ส่วนประกอบของหน้าต่างเริ่มต้นของโปรแกรม Visual Studio 2015	38
2.7 การออกจากโปรแกรม Visual Studio 2015	39
การวัดและประเมินผลการเรียนรู้ บทที่ 2	40

บทที่ 3 กระบวนการเขียนโปรแกรม 41

3.1 การสร้างโปรแกรมใช้งานบนวินโดวส์	42
3.2 การกำหนดความต้องการของโปรแกรม	42
3.3 การสร้างโครงงานใหม่ (New Project)	42
3.4 ส่วนประกอบต่างๆ บน IDE ของ Visual Studio 2015	44
3.5 การออกแบบส่วนติดต่อกับผู้ใช้ (User Interface Design)	48
3.6 การกำหนดคุณสมบัติประจำตัว (Properties) ของฟอร์ม	49
3.7 การนำตัวควบคุมต่างๆ มาจัดวางบนฟอร์ม	51
3.8 การกำหนดคุณสมบัติประจำตัว (Properties) ของตัวควบคุม	54
การวัดและประเมินผลการเรียนรู้ บทที่ 3	62

บทที่ 4 การเขียนโปรแกรม (Code) ให้ตัวควบคุม 63

4.1 วิธีการเขียนโปรแกรมให้ตัวควบคุม	64
4.2 การบันทึกโครงงาน	70
4.3 การทดสอบโปรแกรมและแก้ไขที่ผิด	71
4.4 การสร้างโปรแกรมแบบเรียกใช้งานได้โดยตรง	74
4.5 การเริ่มต้นสร้างโครงงานใหม่	75
4.6 การเรียกโครงงานที่เก็บไว้ในดิสก์มาปรับปรุงแก้ไข	77
4.7 การตกแต่งฟอร์มและตัวควบคุมต่างๆ	78
การวัดและประเมินผลการเรียนรู้ บทที่ 4	83

บทที่ 5 การเขียนโปรแกรมคำนวณ 84

5.1 ข้อมูล (Data)	85
5.2 ตัวแปร (Variables)	88
5.3 ค่าคงที่ (Constant)	98
5.4 การแปลงค่าข้อมูล	100
5.5 การสร้างโปรแกรมใช้งานคำนวณ	102
การวัดและประเมินผลการเรียนรู้ บทที่ 5	109

บทที่ 6 การเขียนโปรแกรมตามเงื่อนไขกรณี 110

6.1 การใช้ประโยคคำสั่ง If	111
6.2 การใช้ประโยคคำสั่ง Select ... Case	121
6.3 การใช้ประโยคคำสั่ง Try ... Catch ... Finally	125
6.4 การใช้กล่องข่าวสาร (MessageBox) อย่างมีเงื่อนไข	128
6.5 การใช้กล่องป้อนข้อมูล (InputBox)	133
6.6 การใช้กล่องรวมกลุ่ม (GroupBox)	134
6.7 การใช้ปุ่มเลือก (RadioButton)	135
6.8 การใช้กล่องเลือก (CheckBox)	139
6.9 การเพิ่มฟอร์ม	143
การวัดและประเมินผลการเรียนรู้ บทที่ 6	145

บทที่ 7 การเขียนโปรแกรมทำซ้ำ 146

7.1 การใช้ประโยคคำสั่ง While ... End While	147
7.2 การใช้ประโยคคำสั่ง Do ... Loop	150
7.3 การใช้ประโยคคำสั่ง For ... Next	158
7.4 การวนรอบทำงานแบบใช้เงื่อนไขที่ไม่กำหนดรอบ	163

7.5 การออกจากการวนรอบทำงานโดยไม่ขึ้นอยู่กับเงื่อนไขการวนรอบ	169
7.6 การเขียนโปรแกรมวนรอบแบบซ้อนกัน	172
การวัดและประเมินผลการเรียนรู้ บทที่ 7	179

บทที่ 8 การใช้ตัวแปรชุด (Array) 178

8.1 ตัวแปรชุด (Array)	179
8.2 การประกาศตัวแปรชุดแบบ 1 มิติ	180
8.3 การกำหนดค่าให้ตัวแปรชุดแบบ 1 มิติ	181
8.4 การใช้ตัวแปรชุดแบบ 1 มิติ	186
8.5 การประกาศตัวแปรเพื่อใช้ร่วมกันหลายฟอร์ม	196
8.6 การประมวลผลข้อมูลด้วยตัวแปรชุดแบบ 1 มิติ	199
การวัดและประเมินผลการเรียนรู้ บทที่ 8	204

บทที่ 9 การออกแบบและเขียนโปรแกรมอย่างง่ายในงานธุรกิจ 205

9.1 การวิเคราะห์ความต้องการของผู้ใช้	206
9.2 การออกแบบโปรแกรมเงินเดือนอย่างง่าย	207
9.3 การเขียนโปรแกรมเงินเดือนอย่างง่าย	213
9.4 การพัฒนาโปรแกรมใช้งานอย่างง่าย	218
การวัดและประเมินผลการเรียนรู้ บทที่ 9	219

บรรณานุกรม 220

ภาคผนวก 221

คำสงวน (Reserved Keyword) ของ Visual Basic 2015	222
---	-----

บทที่ 1



หลักการเขียนโปรแกรม

ผลการเรียนรู้ที่คาดหวัง

- 1.1 บอกความหมายของคอมพิวเตอร์และหน้าที่ของส่วนประกอบต่างๆ ได้
- 1.2 บอกความหมายและประเภทของโปรแกรมคอมพิวเตอร์ได้
- 1.3 ยกตัวอย่างภาษาคอมพิวเตอร์ได้
- 1.4 อธิบายลักษณะของตัวแปลภาษาแต่ละแบบได้
- 1.5 บอกขั้นตอนการพัฒนาโปรแกรมได้
- 1.6 อธิบายการวิเคราะห์งานได้
- 1.7 อธิบายขั้นตอนการแก้ปัญหาได้
- 1.8 เขียนผังงานได้
- 1.9 เขียนรหัสเทียมได้

สาระสำคัญ

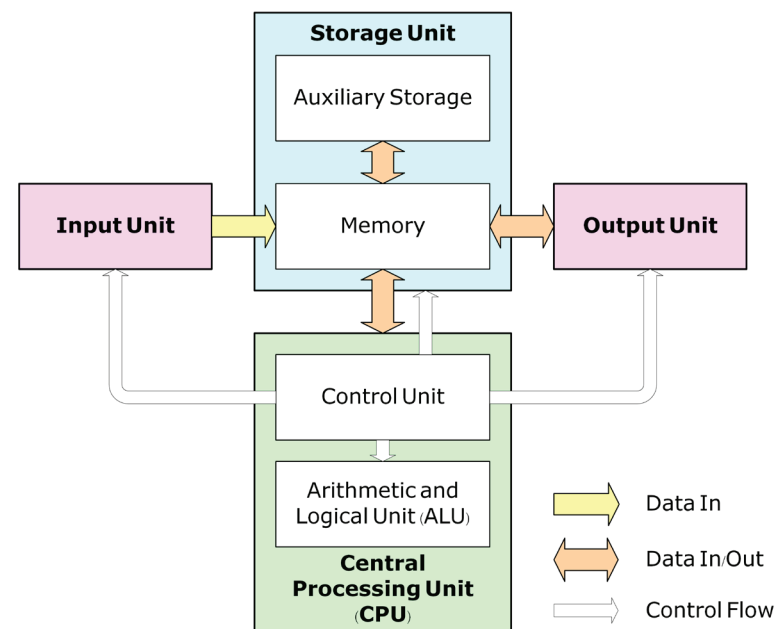
- 1.1 คอมพิวเตอร์และส่วนประกอบ
- 1.2 โปรแกรมคอมพิวเตอร์ (Computer Program)
- 1.3 ภาษาคอมพิวเตอร์ (Computer Languages)
- 1.4 ตัวแปลภาษา (Translator)
- 1.5 ขั้นตอนการพัฒนาโปรแกรม
- 1.6 การวิเคราะห์งาน (Job Analysis)
- 1.7 ขั้นตอนการแก้ปัญหา (Algorithm)
- 1.8 การเขียนผังงาน (Flowchart)
- 1.9 การเขียนรหัสเทียม (Pseudo Code)

การเขียนโปรแกรมคอมพิวเตอร์เป็นงานที่สร้างสรรค์และท้าทายความสามารถ การที่จะเขียนโปรแกรมคอมพิวเตอร์ได้ดีควรมีความรู้พื้นฐานเกี่ยวกับคอมพิวเตอร์และหลักการเขียนโปรแกรมเบื้องต้นก่อน

1.1 คอมพิวเตอร์และส่วนประกอบ

คอมพิวเตอร์ (Computer) คือเครื่องมืออิเล็กทรอนิกส์ชนิดหนึ่งที่ใช้ในการประมวลผลข้อมูลและทำงานต่างๆ ตามคำสั่งที่กำหนดไว้

เครื่องคอมพิวเตอร์หรือฮาร์ดแวร์ (Computer Hardware) ไม่ว่าจะเป็นขนาดเล็กหรือใหญ่ ประกอบด้วยส่วนทำงานหลัก 4 ส่วน ดังภาพ 1-1



ภาพ 1-1 ส่วนประกอบของเครื่องคอมพิวเตอร์

ส่วนประกอบของเครื่องคอมพิวเตอร์แต่ละส่วนทำหน้าที่ดังต่อไปนี้

1. **หน่วยรับข้อมูล (Input Unit)** ทำหน้าที่รับข้อมูลเข้าไปเก็บไว้ในหน่วยความจำหลัก อุปกรณ์นำเข้าข้อมูล ได้แก่ แป้นพิมพ์ (Keyboard) เมาส์ (Mouse) หรืออุปกรณ์อย่างอื่น เช่น จอภาพแบบสัมผัส (Touch Screen Monitor)
2. **หน่วยประมวลผลกลาง (CPU: Central Processing Unit)** ประกอบด้วย หน่วยควบคุม (Control Unit) สำหรับควบคุมการทำงานของส่วนต่างๆ ทั้งหมด และหน่วยคำนวณและตรรกะ (ALU: Arithmetic and Logical Unit) สำหรับการประมวลผลทางตัวเลขและตรรกะ

หน่วยประมวลผลกลาง ทำหน้าที่คำนวณหรือประมวลผลข้อมูลตามโปรแกรมที่เก็บไว้ในหน่วยความจำหลักแล้วอาจเก็บข้อมูลหรือผลการประมวลไว้ในหน่วยความจำหลักหรือบันทึกข้อมูลไว้ในหน่วยความจำสำรองหรือส่งผลออกทางหน่วยแสดงผล

3. **หน่วยเก็บข้อมูล (Storage Unit)** ทำหน้าที่เก็บข้อมูลและโปรแกรมที่ใช้ในการประมวลผล ประกอบด้วย หน่วยความจำหลัก (Main Storage) หรือเมมโมรี (Memory) และหน่วยความจำสำรอง (Auxiliary Storage) เช่น จานแม่เหล็ก (Hard Disk) ยูเอสบีแฟลชไดรฟ์ (USB flash drive)
4. **หน่วยแสดงผล (Output Unit)** ทำหน้าที่แสดงข้อมูลหรือสารสนเทศ (ข้อมูลที่ประมวลผลแล้ว) ออกทางอุปกรณ์แสดงผลต่างๆ เช่น จอภาพ (Monitor) เครื่องพิมพ์ (Printer) หรืออุปกรณ์แสดงผลอย่างอื่น

คอมพิวเตอร์สามารถทำงานตามคำสั่งได้รวดเร็วมากแต่ไม่สามารถเริ่มทำงานเองได้ ต้องอาศัยผู้ใช้งาน (Users) เปิด ปิด ควบคุม หรือสั่งให้ทำงานตามต้องการโดยใช้ชุดคำสั่งซึ่งเรียกว่า โปรแกรมหรือซอฟต์แวร์ (Program or Software)

1.2 โปรแกรมคอมพิวเตอร์ (Computer Program)

โปรแกรมคอมพิวเตอร์ คือชุดคำสั่งที่มีขั้นตอนตามลำดับหรือตามเงื่อนไขที่กำหนดไว้เพื่อให้คอมพิวเตอร์ทำงานตามวัตถุประสงค์ แบ่งออกเป็น 2 ประเภทใหญ่ๆ คือ

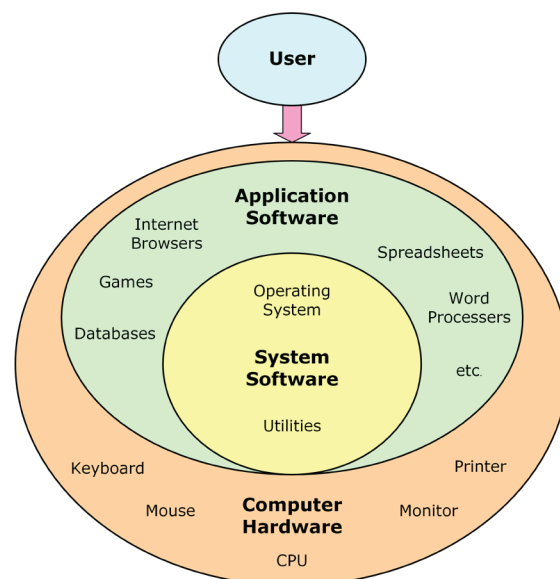
1. **โปรแกรมระบบ (System Software)** ประกอบด้วยการทำงานของ 2 ระบบคือ ระบบปฏิบัติการ (Operating System) และโปรแกรมอรรถประโยชน์ (Utility Program) ระบบปฏิบัติการ (OS) ใช้สำหรับควบคุมการทำงานหลักเกี่ยวกับเครื่องคอมพิวเตอร์ทั้งฮาร์ดแวร์และซอฟต์แวร์ เช่น โปรแกรมวินโดวส์ (Windows) ลินุกซ์ (Linux) แมคโอเอส (Mac OS) หรือระบบปฏิบัติการของสมาร์ทโฟน เช่น แอนดรอยด์ (Android) ไอโอเอส (iOS) และวินโดวส์โฟน (Windows Phone)

โปรแกรมอรรถประโยชน์ใช้สำหรับช่วยในการจัดการคอมพิวเตอร์ เช่น โปรแกรมบีบอัดแฟ้มข้อมูล โปรแกรมป้องกันไวรัส

2. **โปรแกรมใช้งานหรือโปรแกรมประยุกต์ (Application Software)** ที่นิยมเรียกกันสั้นๆ ว่า “แอป” ใช้สำหรับประมวลผลข้อมูลต่างๆ เช่น โปรแกรมประมวลผลคำ (Word Processor) หรือโปรแกรมใช้งานด้านต่างๆ โดยอาจเป็นโปรแกรมสำเร็จรูป เช่น ไมโครซอฟต์เวิร์ด หรือโปรแกรมที่เขียนเพื่อใช้งานเฉพาะอย่าง เช่น โปรแกรมควบคุมระบบงานผลิตสินค้า

ในการใช้งานคอมพิวเตอร์ปกติทั่วไปเป็นการทำงานอย่างมีระบบ เรียกว่า ระบบคอมพิวเตอร์ (Computer System) ซึ่งประกอบด้วยเครื่องคอมพิวเตอร์ (Hardware) โปรแกรม (Software) และผู้ใช้งาน (User)

เครื่องคอมพิวเตอร์จะต้องมีทั้งระบบปฏิบัติการและโปรแกรมใช้งานติดตั้งอยู่ภายในเครื่อง ผู้ใช้งานจึงจะสามารถใช้โปรแกรมสำเร็จรูปหรือโปรแกรมใช้งานนั้นได้ ดังภาพ 1-2



ภาพ 1-2 ระบบคอมพิวเตอร์

ผู้ใช้งานทั่วไปไม่จำเป็นต้องเขียนโปรแกรมขึ้นเอง การเขียนโปรแกรมคอมพิวเตอร์ (Computer Programming) ทำได้โดยนักเขียนโปรแกรม (Programmer) ซึ่งต้องเขียนอย่างมีขั้นตอนถูกต้องตามหลักการเขียนโปรแกรมที่ดี ตรงตามวัตถุประสงค์ของการใช้งาน และเลือกภาษาที่ใช้เขียนโปรแกรม (Programming Languages) ภาษาใดภาษาหนึ่งหรือหลายภาษาก็ได้ ตามความเหมาะสม

1.3 ภาษาคอมพิวเตอร์ (Computer Languages)

โดยพื้นฐานแล้วคอมพิวเตอร์ทำงานด้วยภาษาเครื่อง (Machine Language) ซึ่งทำความเข้าใจยาก เนื่องจากเป็นรหัสตัวเลขแบบต่างๆ ไม่สะดวกต่อการเขียนโปรแกรม จึงได้มีการพัฒนาภาษาที่ใช้กับคอมพิวเตอร์ขึ้นมาหลายภาษา แบ่งได้เป็น 2 ระดับ คือ

1. ภาษาระดับต่ำ (Low-level Language) เป็นภาษาที่เข้าถึงการทำงานในระดับเครื่อง เช่น ภาษาแอสเซมบลี (Assembly Language) ซึ่งใช้รหัสตัวอักษรสำหรับใช้แทนภาษาเครื่อง แต่ก็ยังไม่สะดวกสำหรับผู้ใช้งานทั่วไป

2. ภาษาระดับสูง (High-level Language) เป็นภาษาที่สามารถศึกษาและทำความเข้าใจได้ง่ายขึ้นเหมาะสำหรับการใช้งานในลักษณะต่างกัน มีหลายภาษาตามวัตถุประสงค์ของการพัฒนาภาษาเพื่อใช้งาน ดังตัวอย่างต่อไปนี้

ภาษา	ใช้สำหรับ
BASIC	โปรแกรมใช้งานทั่วไปสำหรับผู้เริ่มต้น
C	โปรแกรมระบบ (System Programming) และการเขียนโปรแกรมแบบโครงสร้าง (Structured Programming)
C++	โปรแกรมแบบโครงสร้าง และแบบเชิงวัตถุ (Object-Oriented Programming) เพื่อใช้งานทั่วไป
COBOL	โปรแกรมใช้งานด้านธุรกิจ
FORTRAN	โปรแกรมใช้งานด้านวิทยาศาสตร์และวิศวกรรม
Java	โปรแกรมใช้งานบนเว็บ (Web Application)
Pascal	ศึกษาการเขียนโปรแกรมแบบโครงสร้าง
PHP	เขียนโปรแกรมบนเว็บ (Web Programming)
Python	โปรแกรมแบบเชิงวัตถุ
SQL	โปรแกรมใช้งานด้านฐานข้อมูล
Visual Basic	โปรแกรมแบบเชิงวัตถุที่ใช้ส่วนประสานกับผู้ใช้แบบกราฟิก (GUI: Graphical User Interface) เพื่อใช้งานทั่วไป

หมายเหตุ บางภาษาเป็นคำย่อมาจากชื่อเต็มดังนี้

ภาษา	ชื่อเต็ม
BASIC COBOL FORTRAN SQL	Beginner's All-purpose Symbolic Instruction Code Common Business Oriented Language FORMula TRANslator Structured Query Language

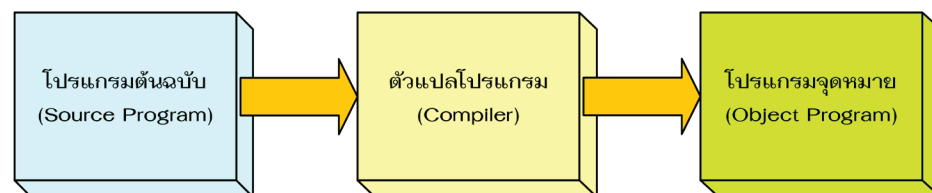
1.4 ตัวแปลภาษา (Translator)

การเขียนโปรแกรมด้วยภาษาใดก็ตามจะต้องใช้ตัวแปลภาษาเพื่อแปลโปรแกรมที่เขียนขึ้นด้วยภาษานั้นให้เป็นภาษาเครื่องก่อนจึงจะใช้งานโปรแกรมนั้นได้เพราะเครื่องคอมพิวเตอร์รับรู้ภาษาเครื่องได้เพียงอย่างเดียวเท่านั้น

ตัวแปลภาษามี 2 แบบ คือ

1. **ตัวแปลโปรแกรม หรือคอมไพเลอร์ (Compiler)** จะทำการแปลโปรแกรมให้เป็นภาษาเครื่อง

โปรแกรมที่เขียนด้วยภาษาใดๆ เรียกว่า โปรแกรมต้นฉบับ (Source Program) หรือรหัสต้นฉบับ (Source Code) เมื่อทำการแปลโดยตัวแปลโปรแกรมแล้วจะได้ผลเป็นโปรแกรมภาษาเครื่อง ที่เรียกว่า โปรแกรมจุดหมาย (Object Program) หรือรหัสจุดหมาย (Object Code) ดังภาพ 1-3

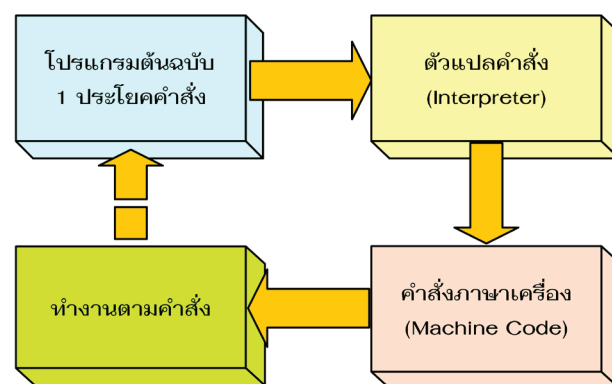


ภาพ 1-3 การแปลโปรแกรม

ตัวแปลโปรแกรมภาษาต่างๆ มีชื่อเรียกตามภาษานั้น เช่น

Assembler	ตัวแปลโปรแกรมภาษาแอสเซมบลี
C Compiler	ตัวแปลโปรแกรมภาษาซี
COBOL Compiler	ตัวแปลโปรแกรมภาษาโคบอล

2. **ตัวแปลคำสั่ง หรืออินเทอร์พรีเตอร์ (Interpreter)** จะทำการแปลโปรแกรมต้นฉบับทีละประโยคคำสั่ง (Statement) ให้เป็นคำสั่งภาษาเครื่องแล้วทำงานตามคำสั่งนั้น ดังภาพ 1-4



ภาพ 1-4 ตัวแปลคำสั่ง

เมื่อทำงานตามคำสั่งเสร็จแล้วก็จะทำการแปลประโยคคำสั่งต่อไปทีละประโยคจนจบโปรแกรมตัวแปลคำสั่งเหมาะสำหรับโปรแกรมที่ไม่ยาวมากและต้องการผลลัพธ์ทันที แต่การทำงานทั้งโปรแกรมจะช้ากว่าแบบตัวแปลโปรแกรม ตัวแปลแบบนี้จะมีชื่อเรียกตามภาษานั้น เช่น ตัวแปลคำสั่งภาษาจาวาสคริปต์ (JavaScript Interpreter)

1.5 ขั้นตอนการพัฒนาโปรแกรม

ภาษาคอมพิวเตอร์กำลังเป็นที่นิยมในปัจจุบันแม้ว่าจะเป็นแบบดูง่าย พัฒนาง่าย หรือที่เรียกว่าแบบวิซวล เช่น Visual Basic, Visual C# ซึ่งเป็นลักษณะการเขียนโปรแกรมเชิงวัตถุที่ทำให้ใช้งานง่ายขึ้น อย่างไรก็ตาม ถ้าไม่มีพื้นฐานการเขียนโปรแกรมที่ดีพอก็ไม่สามารถพัฒนาโปรแกรมใช้งานได้ ดังนั้น ไม่ว่าจะเขียนโปรแกรมโดยใช้ภาษาใดก็ตามจำเป็นต้องศึกษาขั้นตอนการเขียนและพัฒนาโปรแกรมให้ดีกว่า

นักเขียนโปรแกรมที่ดีเมื่อต้องการพัฒนาโปรแกรมใช้งานจะไม่เขียนหรือพิมพ์โปรแกรมเข้าเครื่องคอมพิวเตอร์ทันทีแต่จะทำตามขั้นตอนที่เรียกว่า “วัฏจักรการพัฒนาโปรแกรม”

วัฏจักรการพัฒนาโปรแกรม (Program Development Life Cycle: PDLC) หรือ Software Development Life Cycle: SDLC) เป็นขั้นตอนที่ใช้ในการพัฒนาโปรแกรมใช้งาน คล้ายกันกับขั้นตอนการพัฒนาระบบงาน (System Development Life Cycle: SDLC) ที่นักวิเคราะห์ระบบ (Systems Analyst) ใช้ในการพัฒนาระบบสารสนเทศ

วัฏจักรการพัฒนาโปรแกรม ประกอบด้วย 6 ขั้นตอน ดังภาพ 1-5



ภาพ 1-5 วัฏจักรการพัฒนาโปรแกรม (Software Development Life Cycle: SDLC)

รายละเอียดของแต่ละขั้นตอน มีดังต่อไปนี้

1. วิเคราะห์ความต้องการของผู้ใช้ (Requirement analysis)

การวิเคราะห์ความต้องการของผู้ใช้หรือการวิเคราะห์ปัญหา (Problem Analysis) หรือการวิเคราะห์งาน (Job Analysis) เป็นหัวใจสำคัญของการเขียนโปรแกรม เมื่อนักเขียนโปรแกรมได้รับมอบหมายงานจากนักวิเคราะห์ระบบ หรือต้องการเขียนโปรแกรมตามความต้องการของผู้ใช้ก็ต้องทำความเข้าใจปัญหา ศึกษาและสอบถามความต้องการของผู้ใช้อย่างละเอียดและให้ผู้ใช้ยืนยันว่านักเขียนโปรแกรมเข้าใจปัญหาและความต้องการถูกต้องดีแล้ว

2. ออกแบบโปรแกรม (Design the program)

ในการออกแบบโปรแกรมที่ดีควรทำดังต่อไปนี้

2.1 กำหนดรายละเอียดของโปรแกรม (Program Specification)

เป็นการกำหนดความสามารถและขีดจำกัดของโปรแกรมให้ชัดเจน เช่น กำหนดรายละเอียดเกี่ยวกับข้อมูลเข้า ขั้นตอนการประมวลผล และรายละเอียดเกี่ยวกับการแสดงผลตามที่ต้องการ

2.2 กำหนดขั้นตอนการทำงานของโปรแกรม

เป็นการกำหนดลำดับขั้นตอนการทำงานของโปรแกรม หรือวิธีการที่จะใช้ในการแก้ปัญหาต่างๆ ซึ่งเรียกว่า “อัลกอริทึม” (Algorithm) โดยอาจเขียนเป็นผังงานหรือรหัสเทียม

3. เขียนโปรแกรม (Code the program)

เป็นการเขียนโปรแกรมจากผังงานหรือรหัสเทียมที่ได้ออกแบบมาแล้ว โดยใช้ภาษาที่เหมาะสมกับงานนั้นๆ เช่น ภาษาโคบอล โดยอาจเขียนบนกระดาษลงรหัส (Coding Sheet) ก่อน หรือพิมพ์เข้าเครื่องคอมพิวเตอร์เลยโดยใช้โปรแกรมจัดพิมพ์ข้อความ (Text editor) ใดก็ได้ เช่น Notepad, EditPlus หรือโปรแกรมพิมพ์ข้อความที่มีอยู่ในชุดพัฒนาโปรแกรมของตัวแปลภาษาที่เรียกว่า IDE (Integrated Development Environment) ซึ่งจะสะดวกกว่าแล้วทำการบันทึกเก็บไว้เป็นแฟ้มโปรแกรม (Program File) หรือ Source File โดยใช้นามสกุลตามข้อกำหนดของภาษาที่ใช้ เช่น .VB หมายถึง โปรแกรมภาษา Visual BASIC

4. ทดสอบโปรแกรม (Testing) และแก้ไขข้อผิดพลาด (Debugging)

เป็นการตรวจสอบโปรแกรมด้วยสายตา ก่อนเพื่อดูว่าโปรแกรมที่พิมพ์ตรงกับที่เขียนไว้หรือไม่ โดยตรวจสอบบนเอดิเตอร์หรือพิมพ์โปรแกรมออกมาตรวจสอบบนโต๊ะ (Desk-checking) แล้วให้ตัวแปลภาษาในคอมพิวเตอร์ทำการแปลโปรแกรมหรือที่เรียกว่า คอมไพล์ (Compile) เพื่อหาว่ามีข้อผิดพลาดอะไรบ้าง เช่น ผิดรูปแบบคำสั่ง (Syntax Error) หรืออื่นๆ ถ้ามีที่ผิดก็ต้องทำการแก้ไข แล้วแปลโปรแกรมใหม่จนกว่าจะคอมไพล์ผ่านได้เป็น Object File ซึ่งมีนามสกุล .OBJ

เมื่อไม่มีข้อผิดพลาดจากการแปลโปรแกรมแล้ว จึงทำการสร้างแฟ้มโปรแกรมใช้งาน (Application Program) ที่เรียกกันติดปากว่า “แอป” ซึ่งเป็นแฟ้มที่ทำงานได้ (Executable File) และมีนามสกุล .EXE โดยใช้ฟังก์ชัน Build (ตัวแปลภาษาบางตัวใช้ Make) ซึ่งจะมีการเชื่อมโยงแฟ้มอื่นที่เกี่ยวข้องกับโปรแกรมที่เขียน ถ้ามีความผิดพลาดก็ต้องทำการแก้ไขก่อน

เมื่อได้แฟ้ม .EXE แล้วก็สามารถทำการทดสอบการทำงานของโปรแกรม (Execute) เพื่อหาข้อผิดพลาดขณะทำงาน (Run-time Error) โดยใช้ข้อมูลทดสอบ (Test Data) เพื่อตรวจสอบผลการทำงานของโปรแกรม และทดสอบตรรกะของโปรแกรมว่าผิดหรือไม่ (Program Logic Error) โดยตรวจสอบและแก้ไขจนแน่ใจว่าโปรแกรมทำงานได้ถูกต้องตามที่ผู้ใช้งานต้องการ

5. นำโปรแกรมไปใช้งานจริง (Deployment)

การนำโปรแกรมไปใช้งานจริง โดยให้ผู้ใช้งานทำงานต่างๆ ตามโปรแกรมที่จัดทำไว้ เช่น การป้อนข้อมูล การคำนวณ และการแสดงผลต่างๆ หากพบข้อผิดพลาด ก็ต้องทำการปรับปรุงแก้ไขโปรแกรมให้ถูกต้อง

การจัดทำเอกสารประกอบโปรแกรม (Documentation)

เพื่อให้ผู้ใช้งานสามารถใช้โปรแกรมได้สะดวก ควรจัดทำคู่มือการใช้โปรแกรมและเอกสารประกอบโปรแกรมด้วย โดยคู่มือการใช้โปรแกรมควรจะอธิบายขั้นตอนต่างๆ ที่จำเป็นในการใช้งาน ความสามารถและขีดจำกัดของโปรแกรม ตัวอย่างการใช้งานต่างๆ เช่น การนำข้อมูลเข้า การแสดงผลหรือรายงานที่ได้จากการทำงานของโปรแกรม เป็นต้น

ส่วนเอกสารประกอบโปรแกรมจะเป็นรายละเอียดทั้งหมดของโปรแกรมตั้งแต่เริ่มพัฒนา วิธีการต่างๆ ที่ใช้ในการพัฒนาโปรแกรม ผังโครงสร้าง รหัสเทียม ผังงาน และโปรแกรมทั้งหมด ซึ่งจะเป็นประโยชน์ในการแก้ไขหรือพัฒนาโปรแกรมให้ดีขึ้นต่อไป

6. การปรับปรุงและพัฒนาโปรแกรม (Program Maintenance)

การปรับปรุงและพัฒนาโปรแกรม ทำให้โปรแกรมใช้งานมีความถูกต้อง ทันสมัย และตรงกับความต้องการของผู้ใช้ (ที่อาจมีเพิ่มเติมอยู่เสมอ) โดยทั่วไปโปรแกรมใช้งานรุ่นแรก เรียกว่า รุ่นทดสอบ (Beta Version) เมื่อปรับปรุงและพัฒนาให้ดีขึ้นก็จะเป็นรุ่นที่ใช้งานจริง (Release Version) และถ้ามีการปรับเปลี่ยนโปรแกรมให้ดีขึ้นอีกอาจตั้งชื่อรุ่นเป็นตัวเลขต่อไป เช่น โปรแกรมรุ่นแรก (Version 1.0) เมื่อมีการเพิ่มเติม ปรับปรุง แก้ไขโปรแกรมเป็นรุ่น 1.1 เป็นต้น

1.6 การวิเคราะห์งาน (Job Analysis)

หลักเกณฑ์ในการวิเคราะห์งานเพื่อให้ได้ผลดี ควรพิจารณาดังต่อไปนี้

1. สิ่งที่ต้องการ เช่น ต้องการให้เขียนโปรแกรมเกี่ยวกับอะไร มีการคำนวณหรือประมวลผลเพื่อหาอะไรบ้าง หรือต้องการให้ใช้ภาษาใดเขียน