

2105-2118

การเขียนโปรแกรมคอมพิวเตอร์

ตรงตามจุดประสงค์รายวิชา สมรรถนะรายวิชา และคำอธิบายรายวิชา
หลักสูตรประกาศนียบัตรวิชาชีพ พุทธศักราช 2556
สำนักงานคณะกรรมการการอาชีวศึกษา กระทรวงศึกษาธิการ



หลักการของภาษา
การวางแผนและเขียนโปรแกรม

นิทัศน์ ศรีเทียมศักดิ์

วิทยาลัยเทคนิคกำแพงเพชร
สำนักงานคณะกรรมการการอาชีวศึกษา กระทรวงศึกษาธิการ

คำนำ

เอกสารประกอบการเรียนการสอน รายวิชาการเขียนโปรแกรมคอมพิวเตอร์ รหัสวิชา 2105-2118 เล่มนี้ ได้จัดทำขึ้นเพื่อใช้ประกอบการเรียนการสอนให้กับนักเรียนในสาขาวิชา ช่างอิเล็กทรอนิกส์ ระดับชั้นประกาศนียบัตรวิชาชีพ โดยเรียบเรียงตรงตามจุดประสงค์รายวิชา สมรรถนะรายวิชา และคำอธิบายรายวิชา ของหลักสูตรประกาศนียบัตรวิชาชีพ พุทธศักราช 2556 สำนักงานคณะกรรมการการอาชีวศึกษา กระทรวงศึกษาธิการ

เอกสารประกอบการเรียนการสอน รายวิชาการเขียนโปรแกรมคอมพิวเตอร์ รหัสวิชา 2105-2118 แบ่งเนื้อหาออกเป็น 10 หน่วย สำหรับหน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม มีสาระเนื้อหาเกี่ยวกับหลักการของภาษา โปรแกรมคอมพิวเตอร์ ตัวแปลภาษา และการวางแผนและเขียนโปรแกรม นอกจากนี้ภายในเล่มยังประกอบไปด้วย แบบทดสอบก่อนการเรียนรู้ แบบฝึกหัด และแบบทดสอบหลังการเรียนรู้

ผู้เรียบเรียงหวังเป็นอย่างยิ่งว่า เอกสารประกอบการเรียนการสอนเล่มนี้ จะเป็นประโยชน์แก่นักเรียนที่เรียนในรายวิชาการเขียนโปรแกรมคอมพิวเตอร์ รหัสวิชา 2105-2118 ครูและผู้สนใจทั่วไป

หากมีข้อผิดพลาดประการใด ผู้เรียบเรียงขออภัยด้วยความยินดี โดยจะนำไปปรับปรุงและพัฒนา เพื่อให้เกิดประโยชน์ต่อการศึกษาต่อไป

นิทัศน์ ศรีเทียมศักดิ์

วิทยาลัยเทคนิคกำแพงเพชร



สารบัญ

หน้า

คำนำ	
สารบัญ	
สารบัญภาพ	
สารบัญตาราง	
สิ่งที่กำหนดให้รายวิชา.....	1
หน่วยการเรียนรู้และเวลาที่ใช้ในการจัดการเรียนรู้.....	2
คำแนะนำการใช้เอกสารประกอบการเรียนการสอน.....	3
จุดประสงค์การเรียนรู้.....	5
สมรรถนะประจำหน่วย.....	5
กิจกรรมการเรียนการสอน.....	6
แบบทดสอบก่อนการเรียนรู้.....	9
หน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม.....	1
1.1 หลักการของภาษา.....	12
1.1.1 ภาษาเครื่อง.....	12
1.1.2 ภาษาแอสเซมบลี.....	12
1.1.3 ภาษาระดับสูง.....	13
1.1.4 ภาษาระดับสูงมาก.....	13
1.1.5 ภาษาธรรมชาติ.....	13
1.2 โปรแกรมคอมพิวเตอร์.....	16
1.2.1 โปรแกรมระบบ.....	16
1.2.2 โปรแกรมประยุกต์.....	18



สารบัญ (ต่อ)

หน้า

1.3 ตัวแปลภาษา.....	20
1.3.1 ตัวแปลโปรแกรมหรือคอมไพเลอร์ (Compiler).....	20
1.3.2 ตัวแปลคำสั่งหรืออินเทอร์พรีเตอร์ (Interpreter).....	21
1.4 การวางแผนและเขียนโปรแกรม.....	22
สรุปท้ายหน่วย.....	31
แบบฝึกหัดหน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม.....	32
แบบฝึกหัดที่ 1.1 หลักการของภาษา และโปรแกรมคอมพิวเตอร์.....	32
แบบฝึกหัดที่ 1.2 การวางแผนและเขียนโปรแกรม.....	34
แบบทดสอบหลังการเรียนรู้.....	38
เกณฑ์การประเมินผลแบบฝึกหัด.....	41
แบบประเมินคุณลักษณะที่พึงประสงค์.....	42
แบบประเมินตนเอง.....	43
เอกสารอ้างอิง.....	44



สารบัญภาพ

ภาพ	หน้า
1.1 ตัวอย่างภาษาเบสิก.....	14
1.2 ตัวอย่างภาษาปาสคาล.....	15
1.3 ตัวอย่างโปรแกรมระบบปฏิบัติการ (Operating System).....	17
1.4 ตัวอย่างโปรแกรมอรรถประโยชน์ (System Utility).....	17
1.5 ตัวอย่างโปรแกรมสำเร็จรูป (Packaged software).....	18
1.6 ตัวอย่างโปรแกรมโอเพ่นซอร์ส (Open Source Software).....	19
1.7 ตัวอย่างแชร์แวร์ (Shareware).....	19
1.8 ตัวอย่างฟรีแวร์ (Freeware).....	20
1.9 การแปลโปรแกรม.....	20
1.10 การแปลคำสั่ง.....	21
1.11 ผังงานรูปแบบเรียงลำดับ (Sequence structure).....	28
1.12 ผังงานรูปแบบที่มีการกำหนดเงื่อนไขหรือให้เลือก (Decision structure).....	29
1.13 ผังงานรูปแบบที่มีการทำงานแบบวนรอบหรือทำซ้ำ (Iteration structure).....	29



สารบัญตาราง

ตาราง	หน้า
1.1 ข้อดีและข้อเสียของตัวแปลภาษา.....	22
1.2 สัญลักษณ์ของผังงาน.....	26



สิ่งที่กำหนดให้รายวิชา

หลักสูตร ประกาศนียบัตรวิชาชีพ พุทธศักราช 2556

ประเภทวิชา ช่างอุตสาหกรรม

รหัสวิชา 2105-2118 **ชื่อรายวิชา** การเขียนโปรแกรมคอมพิวเตอร์

ทฤษฎี 1 ชั่วโมง/สัปดาห์ **ปฏิบัติ** 3 ชั่วโมง/สัปดาห์ **จำนวน** 2 หน่วยกิต

จุดประสงค์รายวิชา เพื่อให้

1. เข้าใจหลักการเขียนโปรแกรมด้วยคอมพิวเตอร์
2. มีทักษะในการวางแผน เขียน ตรวจสอบ และแก้ไขโปรแกรมคอมพิวเตอร์
3. มีกิจนิสัยที่ดีในการทำงานด้วยความรับผิดชอบ รอบคอบ และปลอดภัย

สมรรถนะรายวิชา

1. แสดงความรู้เกี่ยวกับหลักการเขียนโปรแกรมด้วยคอมพิวเตอร์ โดยใช้ภาษาซีและโปรแกรมอื่นๆ
2. เขียนโปรแกรมด้วยคอมพิวเตอร์

คำอธิบายรายวิชา

ศึกษาและปฏิบัติเกี่ยวกับหลักการของภาษา องค์ประกอบของโปรแกรม คำสั่ง ตัวแปร โฟลชาร์ต ฟังก์ชันโปรแกรมย่อย การวางแผนและเขียนโปรแกรม ตรวจสอบ แก้ไขโปรแกรม โดยเลือกใช้โปรแกรมภาษาซี และโปรแกรมภาษาอื่นๆ



หน่วยการเรียนรู้และเวลาที่ใช้ในการจัดการเรียนรู้

รหัสวิชา 2105-2118 ชื่อรายวิชา การเขียนโปรแกรมคอมพิวเตอร์

ทฤษฎี 1 ชั่วโมง/สัปดาห์ ปฏิบัติ 3 ชั่วโมง/สัปดาห์ จำนวน 2 หน่วยกิต

หน่วยที่	ชื่อหน่วยการเรียนรู้	สัปดาห์ที่	ชั่วโมงที่
1	ภาษาคอมพิวเตอร์และการพัฒนาโปรแกรม 1.1 โปรแกรมคอมพิวเตอร์ 1.2 ภาษาคอมพิวเตอร์ 1.3 ตัวแปลภาษา 1.4 ขั้นตอนการพัฒนาโปรแกรม	1	1-4
2	ความรู้พื้นฐานเกี่ยวกับโปรแกรมภาษาซี	2	5-8
3	ชนิดของข้อมูล	3	9-12
4	ตัวดำเนินการและนิพจน์	4	13-16
5	การรับและแสดงผลข้อมูล	5-6	17-24
6	คำสั่งควบคุมเงื่อนไข	7-9	25-36
7	การทำงานแบบวนรอบทำซ้ำ	10-12	37-48
8	ตัวแปรอาร์เรย์	13-14	49-56
9	ข้อมูลสตริง	15	57-60
10	ฟังก์ชันในภาษาซี	16-17	61-68
รวม		17	68



คำแนะนำการใช้เอกสารประกอบการเรียนการสอน

เอกสารประกอบการเรียนการสอน การเขียนโปรแกรมคอมพิวเตอร์ รหัสวิชา 2105-2118 จัดทำขึ้นเพื่อให้ครูใช้เป็นคู่มือประกอบการสอน และให้นักเรียนใช้ศึกษาค้นคว้าด้วยตนเอง นอกจากนี้ยังสามารถใช้เพื่อซ่อมเสริมให้กับนักเรียน ในกรณีที่นักเรียนทำกิจกรรมการเรียนรู้ไม่ผ่านเกณฑ์ โดยมีคำแนะนำการใช้ ดังนี้

สำหรับครู

1. ใช้เป็นสื่อการเรียนการสอน เรื่องภาษาคอมพิวเตอร์และการพัฒนาโปรแกรม โดยก่อนใช้เอกสารประกอบการเรียนการสอน ครูควรชี้แจงให้นักเรียนทราบถึงขั้นตอนการใช้เอกสารประกอบการเรียนการสอน และความซื่อสัตย์ในการทำแบบทดสอบก่อนการเรียนรู้ แบบฝึกหัด และแบบทดสอบหลังการเรียนรู้
2. ใช้เป็นแนวทางในการจัดกิจกรรมการเรียนรู้ให้กับนักเรียน เรื่องภาษาคอมพิวเตอร์และการพัฒนาโปรแกรม ทั้งในภาคทฤษฎีและภาคปฏิบัติ โดยให้นักเรียนทำแบบทดสอบก่อนการเรียนรู้ ศึกษาเนื้อหา ทำแบบฝึกหัด และทำแบบทดสอบหลังการเรียนรู้
3. ในการจัดกิจกรรมภาคปฏิบัติ ครูอาจจัดกิจกรรมให้เหมาะสมกับนักเรียน โดยให้นักเรียนจับคู่ แบ่งกลุ่ม เพื่อฝึกให้นักเรียนสามารถทำงานร่วมกับผู้อื่น หรือทำเป็นรายบุคคลก็ได้
4. เมื่อจบกิจกรรมการเรียนรู้ ควรให้นักเรียนร่วมกันสรุปเนื้อหาที่ศึกษาและช่วยกันประเมินผลการทำงานร่วมกันกับครู โดยครูต้องบันทึกผลคะแนนทดสอบก่อนการเรียนรู้ คะแนนระหว่างเรียนรู้ และคะแนนทดสอบหลังการเรียนรู้ของนักเรียน เพื่อให้นักเรียนได้ทราบถึงผลของการพัฒนาของตนเอง หากนักเรียนทำคะแนนได้ไม่ถึงเกณฑ์ ควรให้นักเรียนย้อนกลับไปศึกษาซ้ำ หรือครูจะสอนซ่อมเสริมให้แก่ นักเรียน จนกระทั่งทดสอบผ่านเกณฑ์ที่กำหนดไว้

สำหรับนักเรียน

1. ใช้เอกสารประกอบการเรียนการสอนนี้ สำหรับศึกษาค้นคว้าด้วยตนเอง
2. ในการศึกษาเอกสารประกอบการเรียนการสอนนี้ หากนักเรียนไม่เข้าใจให้จดบันทึกไว้เพื่อสอบถามครูในเวลาเรียนหรือนอกเวลาเรียน
3. ขั้นตอนการใช้เอกสารประกอบการเรียนการสอน
 - 3.1 ศึกษาจุดประสงค์การเรียนรู้ และกระบวนการเรียนรู้
 - 3.2 ทำแบบทดสอบก่อนการเรียนรู้ เพื่อทดสอบความรู้ของนักเรียนก่อนที่จะศึกษาเนื้อหา



3.3 ศึกษาเนื้อหาในเอกสารประกอบการเรียนการสอนตามกระบวนการเรียนรู้ จนกว่าจะเข้าใจ

3.4 ฝึกกิจกรรมการเรียนรู้ทุกกิจกรรม และทำแบบฝึกหัดเพื่อทบทวนความรู้ให้ครบตามที่กำหนดให้

3.5 ในกรณีที่นักเรียนตอบคำถามผิด ให้กลับไปศึกษาเนื้อหาเดิมซ้ำอีกครั้ง และแก้ไขคำตอบที่ผิดให้ถูก ก่อนที่จะไปศึกษาเนื้อหาใหม่

3.6 ส่งงานให้กับครูเพื่อประเมินผลร่วมกันกับนักเรียน

3.7 ทำแบบทดสอบหลังการเรียนรู้ เพื่อทดสอบความรู้และความเข้าใจ

3.8 ตรวจสอบคำตอบแบบทดสอบก่อนการเรียนรู้ และแบบทดสอบหลังการเรียนรู้ เพื่อให้ทราบผลของการพัฒนาตนเอง

4. ในการศึกษา นักเรียนควรได้รับความรู้ ทักษะ และฝึกนิสัยที่ดีในการทำงาน จะต้องปฏิบัติงานด้วยความรอบคอบและเป็นระเบียบเรียบร้อย โดยคำนึงถึงของหลักความปลอดภัย การใช้ทรัพยากรอย่างประหยัด และการรักษาสิ่งแวดล้อม รวมทั้งสร้างคุณธรรมให้เกิดขึ้นกับตนเอง ได้แก่ ความสนใจใฝ่รู้ ความรับผิดชอบ ความซื่อสัตย์ ความมีวินัย และการตรงต่อเวลา เพื่อให้เป็นคนดี คนเก่ง และอยู่ในสังคมได้อย่างมีความสุข



จุดประสงค์การเรียนรู้

หลังจากศึกษาเอกสารประกอบการเรียนการสอนนี้จบแล้ว นักเรียนมีความรู้ ความสามารถ และทักษะ ดังนี้

จุดประสงค์ทั่วไป

นักเรียนมีความรู้ ความเข้าใจและทักษะปฏิบัติ เกี่ยวกับหลักการของภาษา โปรแกรมคอมพิวเตอร์ ตัวแปลภาษา การวางแผนและเขียนโปรแกรม

จุดประสงค์เชิงพฤติกรรม

ด้านพุทธิพิสัย

1. บอกระดับของภาษาคอมพิวเตอร์และประเภทของโปรแกรมคอมพิวเตอร์ได้ถูกต้อง
2. บอกลักษณะของตัวแปลภาษาได้ถูกต้อง
3. บอกขั้นตอนการวางแผนและเขียนโปรแกรมได้ถูกต้อง
4. บอกลักษณะโครงสร้างของผังงานและสัญลักษณ์ของผังงานได้ถูกต้อง

ด้านทักษะพิสัย

5. ปฏิบัติการเขียนผังงานโปรแกรมได้ถูกต้อง

ด้านจิตพิสัย

6. มีคุณลักษณะที่พึงประสงค์ที่ดีในการเรียนและงานอาชีพ ได้แก่
 - 6.1 มีความเสียสละ และความซื่อสัตย์สุจริต
 - 6.2 ความมีวินัย มีความรับผิดชอบ
 - 6.3 การมีมนุษยสัมพันธ์ ความรักสามัคคี ขยัน ประหยัด อดทน
 - 6.4 มีความสนใจใฝ่เรียนรู้ และความคิดริเริ่มสร้างสรรค์

สมรรถนะประจำหน่วย

แสดงความรู้เกี่ยวกับ หลักการของภาษา โปรแกรมคอมพิวเตอร์ ตัวแปลภาษา ขั้นตอนการวางแผนและเขียนโปรแกรม



กิจกรรมการเรียนการสอน

กิจกรรมการเรียนการสอนหรือการเรียนรู้	
ขั้นตอนการสอนหรือกิจกรรมของครู	ขั้นตอนการเรียนรู้หรือกิจกรรมของนักเรียน
ขั้นเตรียม (15 นาที) - ครูจัดเตรียมเอกสาร สื่อการเรียนการสอน แบบทดสอบก่อนการเรียนรู้ แบบฝึกหัด แนวการตอบแบบฝึกหัด (5 นาที) - ครูอธิบายรายละเอียดการจัดการเรียนการสอน การวัดผลประเมินผล และเช็คชื่อนักเรียน - ครูแจกแบบทดสอบก่อนการเรียนรู้ หน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม (10 นาที)	ขั้นเตรียม - นักเรียนจัดเตรียมเอกสาร อุปกรณ์การเรียนตามที่ครูกำหนด - นักเรียนฟังและจดบันทึกรายละเอียดต่างๆ ที่ครูอธิบาย - นักเรียนทำแบบทดสอบก่อนการเรียนรู้ หน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม
MIAP 1 (225 นาที) ขั้นสนใจปัญหา (Motivation) (5 นาที) - ครูตั้งคำถามให้นักเรียนตอบเกี่ยวกับการติดต่อสื่อสารระหว่างมนุษย์กับคอมพิวเตอร์ ว่าสื่อสารกันได้อย่างไร - ครูตั้งคำถามให้นักเรียนบอกเหตุผลว่าคอมพิวเตอร์เข้าใจภาษามนุษย์และทำตามคำสั่งได้อย่างไร	ขั้นสนใจปัญหา (Motivation) - นักเรียนฟังครูและคิดตาม แล้วตอบคำถามว่า มนุษย์กับคอมพิวเตอร์ติดต่อสื่อสารกันได้อย่างไร - นักเรียนบอกเหตุผลว่าคอมพิวเตอร์เข้าใจภาษามนุษย์และทำตามคำสั่งได้อย่างไร



กิจกรรมการเรียนการสอนหรือการเรียนรู้	
ขั้นตอนการสอนหรือกิจกรรมของครู	ขั้นตอนการเรียนรู้หรือกิจกรรมของนักเรียน
ขั้นศึกษาหาข้อมูล (Information) (20 นาที) 1. ครูบรรยายเนื้อหาตามหัวข้อเรื่อง พร้อมถามนักเรียนเกี่ยวกับ หลักการของภาษา การวางแผนและเขียนโปรแกรม 2. ครูให้นักเรียนแลกเปลี่ยนเรียนรู้ซึ่งกันและกัน ขั้นทำแบบฝึกหัดแก้ปัญหา (Application) (200 นาที) 1. ครูมอบหมายงานตามแบบฝึกหัดที่ 1.1 หลักการของภาษาและโปรแกรมคอมพิวเตอร์ (60 นาที) - กระตุ้นและตั้งคำถามให้นักเรียนสรุป พร้อมทั้งอธิบายเพิ่มเติม และสรุปเนื้อหาตามจุดประสงค์เชิงพฤติกรรม 2. ครูมอบหมายงานตามแบบฝึกหัดที่ 1.2 การวางแผนและเขียนโปรแกรม (120 นาที) - ครูกระตุ้นและตั้งคำถามให้นักเรียนสรุป พร้อมทั้งอธิบายเพิ่มเติม และสรุปเนื้อหาตามจุดประสงค์เชิงพฤติกรรม 3. แจกแบบทดสอบหลังการเรียนรู้ หน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม (15 นาที)	ขั้นศึกษาหาข้อมูล (Information) 1. นักเรียนฟังครูบรรยายเนื้อหา เรื่อง หลักการของภาษา การวางแผนและเขียนโปรแกรม 2. นักเรียนแลกเปลี่ยนเรียนรู้ซึ่งกันและกัน เพื่อจะได้เข้าใจเรื่อง การวางแผนและเขียนโปรแกรม มากยิ่งขึ้น ขั้นทำแบบฝึกหัดแก้ปัญหา (Application) 1. นักเรียนทำแบบฝึกหัดที่ 1.1 หลักการ ของภาษาและโปรแกรมคอมพิวเตอร์ - ใช้เหตุผลในการคิดคำตอบ อธิบายเพิ่ม และร่วมกันสรุปเนื้อหาตามจุดประสงค์เชิงพฤติกรรม 2. นักเรียนทำแบบฝึกหัดที่ 1.2 การวางแผนและเขียนโปรแกรม - ใช้เหตุผลในการคิดคำตอบ อธิบายเพิ่ม และร่วมกันสรุปเนื้อหาตามจุดประสงค์เชิงพฤติกรรม 3. นักเรียนทำแบบทดสอบหลังการเรียนรู้ หน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม



กิจกรรมการเรียนการสอนหรือการเรียนรู้	
ขั้นตอนการสอนหรือกิจกรรมของครู	ขั้นตอนการเรียนรู้หรือกิจกรรมของนักเรียน
ขั้นทราบผลการฝึกหัด (Progress) (5 นาที) 1. เฉลยแบบฝึกหัดที่ 1.1-1.2 2. เฉลยแบบทดสอบหลังการเรียนรู้ หน่วยที่ 1	ขั้นทราบผลการฝึกหัด (Progress) 1. ร่วมกันเฉลยแบบฝึกหัดที่ 1.1-1.2 2. ร่วมกันเฉลยแบบทดสอบหลังการเรียนรู้ การเรียนรู้หน่วยที่ 1



แบบทดสอบก่อนการเรียนรู้
หน่วยที่ 1 ภาษาคอมพิวเตอร์และการพัฒนาโปรแกรม
รายวิชาการเขียนโปรแกรมคอมพิวเตอร์ รหัสวิชา 2105-2118
สาขาวิชาช่างอิเล็กทรอนิกส์ ระดับประกาศนียบัตรวิชาชีพ
จำนวน 10 ข้อ 10 คะแนน

คำชี้แจง ให้นักเรียนทำเครื่องหมาย X ทับตัวอักษรหน้าคำตอบที่ถูกต้องที่สุด

1. ข้อใด เป็นโปรแกรมสำเร็จรูป

ก. DOS	ข. WinZip
ค. Windows	ง. Microsoft Excel
2. โปรแกรมที่ทำหน้าที่ควบคุมการทำงานของเครื่องคอมพิวเตอร์ คือข้อใด ?

ก. โปรแกรมเฉพาะ	ข. โปรแกรมระบบ
ค. โปรแกรมประยุกต์	ง. โปรแกรมสำเร็จรูป
3. ข้อใด เป็นภาษาระดับสูงทั้งหมด ?

ก. ภาษาซี ภาษาแอสเซมบลี	ข. ภาษาโคบอล ภาษาฟอร์เทน
ค. ภาษา SQL ภาษาปาสคาล	ง. ภาษาแอสเซมบลี ภาษาเบสิก
4. ข้อใด เป็นลักษณะของภาษาเครื่อง ?

ก. เป็นภาษาที่ใช้กับงานขั้นสูง	
ข. เป็นภาษาคอมพิวเตอร์ยุคใหม่	
ค. ทำงานได้โดยไม่ต้องมีโปรแกรมระบบ	
ง. เป็นภาษาที่คอมพิวเตอร์เข้าใจและปฏิบัติตามคำสั่งทันที	
5. ภาษาคอมพิวเตอร์ภาษาใดที่ต้องใช้ คอมไพเลอร์เป็นตัวแปลภาษา ?

ก. ภาษาเบสิก	ข. ภาษา SQL
ค. ภาษาปาสคาล	ง. ภาษาแอสเซมเบอร์
6. การแปลภาษาคอมพิวเตอร์ให้เป็นรหัสภาษาเครื่องที่มีการแปลทีละบรรทัด เรียกว่า ?

ก. รันไทม์	ข. แอสเซมเบอร์
ค. คอมไพเลอร์	ง. อินเทอร์พรีเตอร์



7. หากต้องการพัฒนาโปรแกรมต้องทำอะไรก่อนเป็นอันดับแรก ?

- | | |
|-------------------|-----------------------------|
| ก. เขียนผังงาน | ข. เขียนโปรแกรม |
| ค. วิเคราะห์ปัญหา | ง. เลือกภาษาที่ต้องใช้เขียน |

8. การเขียนผังงานเพื่อใช้แสดงผลข้อมูลทางจอภาพ ต้องใช้สัญลักษณ์ใด ?

- | | |
|--|---|
| ก.  | ข.  |
| ค.  | ง.  |

9. การเขียนผังงานที่มีสัญลักษณ์ตัดสินใจ ต้องใช้ผังงานที่มีโครงสร้างรูปแบบใด ?

- | | |
|------------------------------|---|
| ก. โครงสร้างรูปแบบเรียงลำดับ | ข. โครงสร้างรูปแบบมีทางเลือก |
| ค. โครงสร้างรูปแบบมีการทำซ้ำ | ง. โครงสร้างรูปแบบทำซ้ำจนเงื่อนไขเป็นจริง |

10. ถ้าต้องการเขียนโปรแกรมหาอายุเฉลี่ยของนักเรียน 3 คน ข้อใด คือข้อมูลนำเข้า (Input) ?

- | | |
|-------------------------------|---|
| ก. คำนวณหาอายุเฉลี่ย | ข. ค่าอายุเฉลี่ยที่คำนวณได้ |
| ค. ตัวเลขอายุของนักเรียน 3 คน | ง. $\text{อายุเฉลี่ย} = (\text{อายุ1} + \text{อายุ2} + \text{อายุ3}) / 3$ |



1

หลักการของภาษา การวางแผนและเขียนโปรแกรม



หัวข้อเรื่อง

- 1.1 หลักการของภาษา
- 1.2 โปรแกรมคอมพิวเตอร์
- 1.3 ตัวแปลภาษา
- 1.4 การวางแผนและเขียนโปรแกรม

สาระสำคัญ

คอมพิวเตอร์จะสามารถทำงานได้ ผู้ใช้ต้องป้อนคำสั่งให้กับคอมพิวเตอร์ โดยมีตัวแปลคำสั่ง (Interpreter) หรือตัวแปลโปรแกรม (Compiler) ทำการแปลคำสั่งให้เป็นภาษาที่คอมพิวเตอร์เข้าใจ และทำงานตามคำสั่งที่ต้องการ

การพัฒนาโปรแกรมเพื่อให้โปรแกรมที่สร้างขึ้นมีลักษณะที่ดี มีขั้นตอนการพัฒนาโปรแกรม 5 ขั้นตอน ได้แก่ กำหนดและวิเคราะห์ปัญหา การเขียนผังงาน การเขียนโปรแกรม การทดสอบและแก้ไขโปรแกรม การเขียนทำเอกสารประกอบและบำรุงรักษาโปรแกรม



1.1 หลักการของภาษา

ภาษาคอมพิวเตอร์ เป็นภาษาที่มนุษย์ใช้ติดต่อสื่อสารกับคอมพิวเตอร์ เพื่อให้คอมพิวเตอร์เข้าใจและสามารถปฏิบัติตามคำสั่งเราได้ หรือ หมายถึง โปรแกรมหรือชุดคำสั่งที่ผู้เขียนโปรแกรมเขียนขึ้นมา เพื่อใช้ให้คอมพิวเตอร์ทำงานตามรูปแบบและโครงสร้างของภาษา

คอมพิวเตอร์จะทำงานได้ด้วยภาษาเครื่องซึ่งเป็นรหัสเลขฐานสอง และถ้ามนุษย์ต้องป้อนโปรแกรมให้กับคอมพิวเตอร์เป็นเลขฐานสองนั้นจะทำได้ยากมาก เพราะเป็นภาษาที่มนุษย์เข้าใจยาก จึงมีการพัฒนาภาษาที่เป็นกลาง และเป็นภาษาที่ใกล้เคียงกับคำที่มนุษย์รู้จักกันดี แล้วแปลงกลับไปเป็นเลขฐานสองให้คอมพิวเตอร์อีกครั้งหนึ่ง ซึ่งวิธีนี้จะทำให้โปรแกรมคอมพิวเตอร์ทำงานได้ง่ายขึ้น จึงทำให้มีการคิดค้นภาษาคอมพิวเตอร์ออกมาอย่างหลากหลาย แบ่งได้เป็น 5 ระดับ คือ ภาษาเครื่อง, ภาษาแอสเซมบลี, ภาษาระดับสูง, ภาษาระดับสูงมาก และภาษาธรรมชาติ

1.1.1 ภาษาเครื่อง (Machine language)

เป็นภาษาระดับต่ำสุด เครื่องคอมพิวเตอร์สามารถเข้าใจและปฏิบัติตามคำสั่งได้ในทันที แต่เป็นภาษาที่มนุษย์เข้าใจได้ยาก เพราะใช้เลขฐานสองในการแทนคำสั่งและข้อมูลต่างๆ ทำให้นักเขียนและพัฒนาโปรแกรมต้องกำหนดชุดตัวเลขขึ้นมาใช้แทนคำสั่ง ตัวอย่างเช่น การเขียนคำสั่งเพื่อทำการบันทึกข้อมูล 61H (61 ฐานสิบหก) เก็บไว้ในหน่วยความจำภายในซีพียูที่เป็นรีจิสเตอร์ AL โดยใช้ภาษาเครื่องคอมพิวเตอร์รุ่น x86/IA-32 จะเขียนดังนี้

```
10110000 01100001
```

1.1.2 ภาษาแอสเซมบลี (Assembly language)

เป็นภาษาระดับต่ำ พัฒนาขึ้นมาเพื่อช่วยให้จดจำคำสั่งได้ง่ายขึ้น โดยใช้ตัวอักษรภาษาอังกฤษแทนคำสั่งที่เป็นเลขฐานสอง โดยเรียกอักษรสัญลักษณ์ที่ใช้แทนคำสั่งนี้ว่า นิโมนิกโค้ด (Mnemonic code) ตัวอย่างการใช้ภาษาแอสเซมบลีเขียนเป็นโปรแกรม เช่น

```
MOV AL, 61H
```



1.1.3 ภาษาระดับสูง (High-level language)

เป็นภาษาที่พัฒนามาจากภาษาแอสเซมบลี ใช้คำภาษาอังกฤษมาสั่งงานและควบคุมคอมพิวเตอร์ โดยมีจุดประสงค์เพื่อให้ใช้งานและเรียนรู้คำสั่งได้ง่าย และสะดวกต่อการเขียนโปรแกรม เรียกภาษาระดับสูงยุคนี้ว่า ภาษายุคที่สาม ภาษาระดับสูงที่เกิดขึ้นในยุคนี้ ได้แก่ ภาษาเบสิก, ภาษาโคบอล, ภาษาฟอร์แทรน ภาษาปาสคาล และภาษาซี

1.1.4 ภาษาระดับสูงมาก (Very high-level language)

เป็นภาษายุคที่สี่ หรือ 4GLs โดยใช้หลักการเขียนโปรแกรมแบบไม่ใช่พรซีเยอร์ (Procedural Language) ซึ่งแบบเก่าจะต้องเขียนโปรแกรมสร้างพรซีเยอร์แต่ละตัว เพื่อให้ทำงานในด้านต่างๆ แต่ยุคนี้โปรแกรมการทำงานในส่วนต่างๆ จะถูกสร้างไว้ให้พร้อมแล้ว นักพัฒนาโปรแกรมจึงไม่จำเป็นต้องเขียนโปรแกรมขึ้นใหม่ทั้งหมด สามารถเรียกชุดการทำงานที่มีอยู่แล้ว มาสร้างเป็นโปรแกรมใหม่ที่ต้องการได้ทันที ภาษาที่รู้จักกันดีในยุคนี้ คือ ภาษาเรียกค้นข้อมูล SQL (Structured Query Language) ซึ่งภาษายุคที่สี่มีข้อดี ดังนี้

- การเขียนโปรแกรมจะมุ่งหวังที่จะได้ผลลัพธ์ของงานเป็นหลักว่า อยากได้อะไรและไม่สนใจวิธีการทำมากนัก
- ส่งเสริมต่อการพัฒนาเนื้องาน เนื่องจากสามารถเขียนและแก้ไขโปรแกรมได้โดยง่าย
- ลดเวลาในการอบรมและพัฒนาผู้เขียนโปรแกรม เพราะถ้าไม่เชี่ยวชาญในการเขียนโปรแกรมก็สามารถทำได้
- ผู้เขียนโปรแกรมไม่จำเป็นต้องรู้และศึกษาถึงโครงสร้างของโปรแกรมและระบบฮาร์ดแวร์ของเครื่อง

1.1.5 ภาษาธรรมชาติ (Natural language)

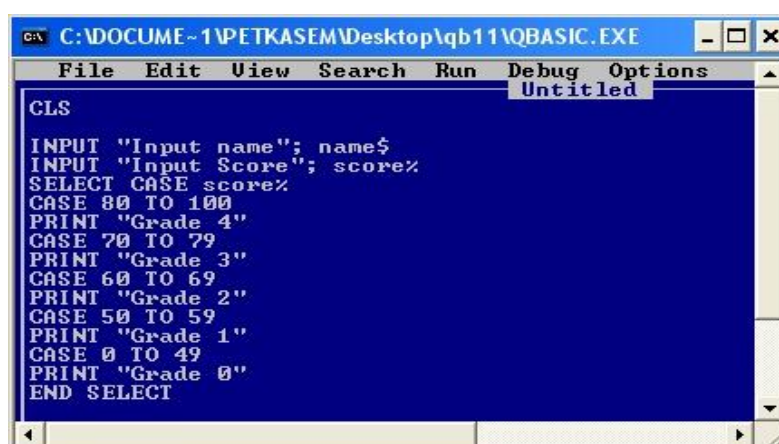
เป็นภาษาในยุคที่ห้า หรือ 5GLs ซึ่งมีความใกล้เคียงกับภาษาธรรมชาติของมนุษย์ ไม่สนใจเรื่องไวยากรณ์ของภาษา เพียงแค่พิมพ์คำสั่งลงไป ผู้ใช้แต่ละคนอาจต้องใช้คำศัพท์และรูปประโยคที่ต่างกัน จากนั้นคอมพิวเตอร์จะแปลความหมายและทำงานตามคำสั่งเหล่านั้นเอง การแปลความหมายของคำสั่งในภาษาธรรมชาติ จะใช้ระบบฐานความรู้ (Knowledge Base System) ที่มีความใกล้เคียงกับการใช้ภาษาของมนุษย์



ทั้งนี้ภาษาคอมพิวเตอร์ที่ใช้สำหรับการพัฒนาโปรแกรมมีอย่างหลากหลายซึ่งในแต่ละภาษาจะมีโครงสร้างของภาษาและความสามารถที่ต่างกัน โดยภาษาคอมพิวเตอร์ที่นิยมใช้เขียนโปรแกรมได้แก่

1. **ภาษาฟอร์แทรน (FORTRAN)** เป็นภาษาคอมพิวเตอร์ระดับสูง ย่อมาจากคำว่า FORMular TRANslator เหมาะสำหรับงานด้านคำนวณ เช่น งานวิจัยต่างๆ งานด้านวิศวกรรมศาสตร์ และงานด้านวิทยาศาสตร์

2. **ภาษาเบสิก (BASIC)** เป็นภาษาคอมพิวเตอร์ระดับสูง ย่อมาจากคำว่า Beginner's All purpose Symbolic Instruction Code เหมาะสำหรับผู้ที่เริ่มต้นเขียนโปรแกรม เนื่องจากมีรูปแบบคำสั่งที่ง่าย สามารถนำไปประยุกต์สร้างโปรแกรมได้ทั่วไป ทั้งงานด้านธุรกิจและงานด้านวิทยาศาสตร์ แต่ความสามารถน้อยกว่าภาษาอื่นๆ เนื่องจากเป็นภาษาที่พัฒนามานานแล้ว



```

C:\DOCUMENTS\1\PETKASEM\Desktop\qbb1\QBASIC.EXE
File Edit View Search Run Debug Options
Untitled
CLS
INPUT "Input name"; name$
INPUT "Input Score"; score%
SELECT CASE score%
CASE 80 TO 100
PRINT "Grade 4"
CASE 70 TO 79
PRINT "Grade 3"
CASE 60 TO 69
PRINT "Grade 2"
CASE 50 TO 59
PRINT "Grade 1"
CASE 0 TO 49
PRINT "Grade 0"
END SELECT
  
```

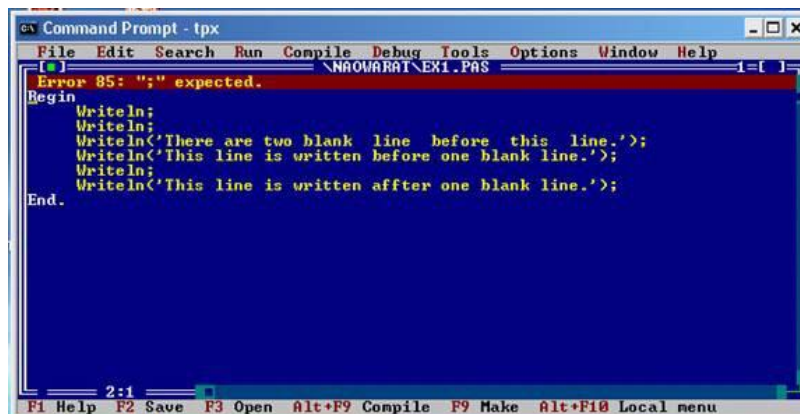
ภาพ 1.1 ตัวอย่างภาษาเบสิก

ที่มา : ภาพ <http://konpretty.blogspot.com>. สืบค้นเมื่อวันที่ 12 มกราคม 2557.

3. **ภาษาโคบอล (COBOL)** เป็นภาษาคอมพิวเตอร์ระดับสูง ที่เหมาะสำหรับงานด้านธุรกิจ งานทางด้านบัญชี การจัดเก็บข้อมูล และการเชื่อมต่อคอมพิวเตอร์ภายในองค์กร

4. **ภาษาปาสคาล (PASCAL)** เป็นภาษาคอมพิวเตอร์ระดับสูง ที่ใช้สอนหลักการเขียนโปรแกรม เพราะจะช่วยให้เขียนโปรแกรมได้เร็ว แก้ไขปรับปรุงและค้นหาที่ผิดพลาดได้ง่ายและรวดเร็ว เหมาะสำหรับงานด้านธุรกิจ คณิตศาสตร์ การควบคุมต่างๆ งานกราฟิก ภาพและเสียง





ภาพ 1.2 ตัวอย่างภาษาปาสคาล

ที่มา : ภาพ http://www.thaigoodview.com/library/teachershow/lopburi/naowarta_s/programming/sec05p07.html. สืบค้นเมื่อวันที่ 12 มกราคม 2557.

5. **ภาษาซี (C)** เป็นภาษาคอมพิวเตอร์ที่เป็นทั้งระดับต่ำและระดับสูงที่จะรวมข้อมูลของภาษา ระดับต่ำและภาษาระดับสูงเข้าไว้ด้วยกัน ซึ่งเป็นที่นิยมของนักเขียนโปรแกรม เพราะเป็นภาษาที่มีไวยากรณ์เข้าใจง่าย มีประสิทธิภาพและความเร็วในการทำงานดีกว่าภาษาระดับสูงทั่วไป ควบคุมฮาร์ดแวร์ได้มากกว่าภาษาระดับสูงอื่นๆ เหมาะสำหรับพัฒนาโปรแกรมประยุกต์ และระบบปฏิบัติการ เช่น ระบบปฏิบัติการยูนิกซ์

6. **ภาษาซีพลัสพลัส (C++)** ภาษานี้พัฒนามาจากภาษาซี นิยมใช้สำหรับการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming : OOP) ซึ่งเป็นการเขียนโปรแกรมที่มองสิ่งที่เห็นและเกิดขึ้นทุกอย่างเป็นวัตถุที่บอกลักษณะของวัตถุนั้นได้

7. **วิชวลเบสิก (VISUAL BASIC)** เป็นภาษาที่คล้ายกับภาษา BASIC ได้รับความนิยมมากสำหรับการเขียนโปรแกรมบนระบบปฏิบัติการวินโดวส์ เหมาะกับผู้พัฒนาโปรแกรม แต่ไม่เหมาะกับผู้ที่เริ่มต้นเขียนโปรแกรม

8. **ภาษาจาวา (JAVA)** เป็นภาษาที่สามารถทำงานได้บนเครื่องคอมพิวเตอร์ทุกระบบ และเหมาะกับการพัฒนาโปรแกรม หรือเกมที่ต้องการเข้าถึงผู้ใช้ได้ในทุกระบบ



1.2 โปรแกรมคอมพิวเตอร์

คอมพิวเตอร์ เป็นอุปกรณ์อิเล็กทรอนิกส์ชนิดหนึ่งที่ใช้ในการประมวลผลข้อมูล ไม่สามารถทำงานเองได้ ต้องอาศัยผู้ใช้งาน ควบคุม เปิด ปิด หรือสั่งให้คอมพิวเตอร์ทำงานตามต้องการ โดยใช้ชุดคำสั่ง ซึ่งเรียกว่า โปรแกรมหรือซอฟต์แวร์ และการใช้คำสั่งเพื่อให้คอมพิวเตอร์ทำงานนั้น ต้องใช้ภาษาที่คอมพิวเตอร์เข้าใจได้ ภาษาที่คอมพิวเตอร์เข้าใจเรียกว่า ภาษาเครื่อง (Machine Language) ซึ่งจะเป็นรหัสเลขฐานสอง เมื่อป้อนภาษานี้เข้าไปในเครื่องคอมพิวเตอร์ รหัสนี้จะถูกแปลงให้เป็นสัญญาณทางไฟฟ้าที่คอมพิวเตอร์เข้าใจ

โปรแกรมคอมพิวเตอร์ คือ ชุดคำสั่งที่มีขั้นตอนตามลำดับหรือตามเงื่อนไขที่กำหนด เพื่อให้คอมพิวเตอร์ทำงานตามขั้นตอนที่ได้กำหนดไว้ ถ้าหากไม่มีโปรแกรมเครื่องคอมพิวเตอร์ก็จะไม่สามารถทำงานได้ โปรแกรมแบ่งออกได้เป็น 2 ประเภท คือ โปรแกรมระบบ และโปรแกรมประยุกต์ ดังนี้

1.2.1 โปรแกรมระบบ

โปรแกรมระบบ เป็นโปรแกรมที่ทำหน้าที่ควบคุมการทำงานของเครื่องคอมพิวเตอร์ให้สามารถทำงานต่างๆ สะดวกมากขึ้น สามารถแบ่งออกเป็น โปรแกรมระบบปฏิบัติการ (Operating System) โปรแกรมอรรถประโยชน์ (System Utility) และโปรแกรมดีไวซ์ไดรเวอร์ (Device Driver)

1) โปรแกรมระบบปฏิบัติการ (Operating System)

โปรแกรมระบบปฏิบัติการ เป็นโปรแกรมที่ถูกออกแบบมาเพื่อทำงานร่วมกับเครื่องคอมพิวเตอร์ ดูแลระบบ และควบคุมการทำงานของส่วนประกอบต่างๆ ในคอมพิวเตอร์ เช่น ควบคุมการทำงานของแป้นพิมพ์ จอภาพ ระบบอ่านและบันทึกข้อมูล ตัวอย่างของโปรแกรมประเภทนี้ ได้แก่ ดอส (DOS), วินโดวส์ (Windows), ลินุกซ์ (Linux) และ ยูนิกซ์ (UNIX)





ภาพ 1.3 ตัวอย่างโปรแกรมระบบปฏิบัติการ (Operating System)

ที่มา : ภาพ <http://news.techgenie.com>. สืบค้นเมื่อวันที่ 12 มกราคม 2557

2) โปรแกรมอรรถประโยชน์ (System Utility)

โปรแกรมอรรถประโยชน์ เป็นโปรแกรมที่ช่วยอำนวยความสะดวกให้กับผู้ใช้ เพื่อใช้ปรับปรุงคอมพิวเตอร์ให้ทำงานดีขึ้น เช่น โปรแกรมบีบอัดข้อมูล โปรแกรมแบ็คอัพข้อมูล โปรแกรมตรวจสอบการทำงานของระบบต่างๆ โปรแกรมตรวจสอบไวรัส เป็นต้น



ภาพ 1.4 ตัวอย่างโปรแกรมอรรถประโยชน์ (System Utility)

ที่มา : ภาพ <http://www.scriptux.com>. สืบค้นเมื่อวันที่ 12 มกราคม 2557



3) โปรแกรมดีไวซ์ไดรเวอร์ (Device Driver)

โปรแกรมดีไวซ์ไดรเวอร์ เป็นโปรแกรมที่ออกแบบมาสำหรับฮาร์ดแวร์ต่างๆ ที่ใช้ในเครื่องคอมพิวเตอร์ เนื่องจากฮาร์ดแวร์บางประเภทออกแบบมาหลากหลายรุ่น บางรุ่นอาจแตกต่างกันไปจากรุ่นอื่นๆ หรือมีความสามารถพิเศษจึงต้องมีโปรแกรมดีไวซ์ไดรเวอร์ เพื่อช่วยในการจัดการต่างๆ

1.2.2 โปรแกรมประยุกต์

โปรแกรมประยุกต์ เป็นโปรแกรมที่พัฒนาขึ้นมาสำหรับงานเฉพาะอย่าง การทำงานของโปรแกรมประยุกต์นั้น ต้องทำงานภายใต้ระบบปฏิบัติการ โดยโปรแกรมประยุกต์ที่พบเห็นกันทั่วไป มีดังนี้

1) โปรแกรมสำเร็จรูป (Packaged software)

เป็นโปรแกรมที่ผลิตออกมาใช้งานทั่วไปและใช้งานง่าย เช่น โปรแกรมตารางทำงาน โปรแกรมประมวลผลคำ โปรแกรมวาดภาพ โปรแกรมตกแต่งภาพ เป็นต้น



ภาพ 1.5 ตัวอย่างโปรแกรมสำเร็จรูป (Packaged software)

ที่มา : ภาพ <http://loadprome.blogspot.com>. สืบค้นเมื่อวันที่ 10 ตุลาคม 2557.

2) โปรแกรมเฉพาะ (Custom software)

เป็นโปรแกรมที่ผลิตขึ้นมาเฉพาะในงานด้านธุรกิจ หรืออุตสาหกรรม หรือเขียนขึ้นมาเพื่อใช้ในหน่วยงานใดหน่วยงานหนึ่งโดยเฉพาะ ผู้ใช้สามารถนำมาใช้งานได้ทันที



3) โปรแกรมโอเพ่นซอร์ส (Open Source Software)

เป็นโปรแกรมที่ผู้ใช้สามารถนำมา พัฒนาเพิ่มเติม แก้ไข แจกจ่ายได้



ภาพ 1.6 ตัวอย่างโปรแกรมโอเพ่นซอร์ส (Open Source Software)

ที่มา : ภาพ <http://saiainfosoft.com/open-source-technologies.html>.

สืบค้นเมื่อวันที่ 12 มกราคม 2557

4) แชร์แวร์ (Shareware)

เป็นโปรแกรมที่ออกแบบมา เพื่อให้ผู้ใช้ได้นำมาทดลองใช้ก่อนได้โดยไม่คิดค่าใช้จ่าย ถ้าหากผู้ใช้พอใจในโปรแกรม สามารถขอซื้อหรือขอรหัสผ่านเพื่อให้อาจสามารถใช้โปรแกรมได้เต็มประสิทธิภาพ ซึ่งบางโปรแกรมจะมีกำหนดระยะเวลาให้ใช้งาน หรือใช้ได้เพียงบางฟังก์ชันเท่านั้น



ภาพ 1.7 ตัวอย่างแชร์แวร์ (Shareware)

ที่มา : ภาพ <http://software.thaiware.com/3758-IDM.html>.

สืบค้นเมื่อวันที่ 12 มกราคม 2557.



5) ฟรีแวร์ (Freeware)

เป็นโปรแกรมที่สามารถหาดาวน์โหลดได้จากอินเทอร์เน็ตหรือหามาใช้ได้ฟรี ส่วนใหญ่เป็นโปรแกรมที่เขียนมาเพื่อใช้งานเฉพาะด้าน และนำออกมาให้ผู้อื่นลองใช้ร่วมกัน ไม่ต้องเสียค่าใช้จ่ายใดๆ



ภาพ 1.8 ตัวอย่างฟรีแวร์ (Freeware)

ที่มา : ภาพ <http://thailandfreeware.com>. สืบค้นเมื่อวันที่ 12 มกราคม 2557

1.3 ตัวแปลภาษา

การเขียนโปรแกรมคอมพิวเตอร์ด้วยภาษาระดับสูงหรือภาษาระดับต่ำ จะต้องแปลภาษาเหล่านั้นให้เป็นภาษาเครื่องที่คอมพิวเตอร์เข้าใจก่อน คอมพิวเตอร์จึงจะสามารถทำงานได้ตามต้องการ ซึ่งตัวแปลภาษามี 2 ประเภท คือ

1.3.1 ตัวแปลโปรแกรมหรือคอมไพเลอร์ (Compiler)

คอมไพเลอร์ จะทำการแปลโปรแกรมที่เขียนขึ้น ด้วยภาษาระดับสูงที่เรียกกันว่า “โปรแกรมต้นฉบับ” (Source program) ให้เป็นภาษาเครื่อง (Object program) โดยจะตรวจสอบความถูกต้องของไวยากรณ์ภาษาก่อน หากมีข้อผิดพลาด ผู้ใช้สามารถสั่งประมวลผลโปรแกรม และสามารถเก็บโปรแกรมที่แปลภาษาเครื่องไว้ใช้งานต่อไปได้ โดยไม่ต้องทำการแปลโปรแกรมซ้ำอีก



ภาพ 1.9 การแปลโปรแกรม



ชื่อเรียกของตัวแปลโปรแกรมภาษาต่างๆ จะเรียกตามภาษานั้น ๆ เช่น

- Assembler เรียก ตัวแปลโปรแกรมภาษาแอสเซมบลี
- C compiler เรียก ตัวแปลโปรแกรมภาษาซี
- COBOL compiler เรียก ตัวแปลโปรแกรมภาษาโคบอล

1.3.2 ตัวแปลคำสั่งหรืออินเทอร์พรีเตอร์ (Interpreter)

อินเทอร์พรีเตอร์ จะทำการแปลคำสั่งและประมวลผลคำสั่งของโปรแกรมต้นฉบับ (Source program) ทีละคำสั่งให้เป็นคำสั่งภาษาเครื่องก่อนแล้วทำงานตามคำสั่งนั้น และจะทำการแปลคำสั่งต่อไปเรื่อย ๆ จนจบโปรแกรม ตัวแปลคำสั่งจะเหมาะกับโปรแกรมที่ไม่ยาวมากและต้องการผลลัพธ์ในทันที ถ้าหากนำโปรแกรมนั้นมาใช้งานอีกจะต้องทำการแปลโปรแกรมใหม่ทุกๆ ครั้ง ตัวแปลคำสั่งแบบนี้จะมีชื่อเรียกตามภาษานั้นๆ เช่น ตัวแปลคำสั่งภาษาเบสิก (BASIC interpreter)



ภาพ 1.10 การแปลคำสั่ง

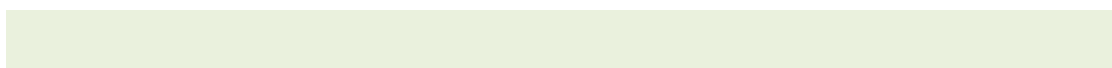


ตาราง 1.1 ข้อดีและข้อเสียของตัวแปลภาษา

ตัวแปลภาษา	ข้อดี	ข้อเสีย
คอมไพเลอร์ (Compiler)	- ทำงานได้เร็ว เนื่องจากทำการแปลผลทีเดียว แล้วจึงทำงานตามคำสั่งของโปรแกรมในภายหลัง - เมื่อทำการแปลผลแล้ว ในครั้งต่อไปไม่จำเป็นต้องทำการแปลผลใหม่อีก เนื่องจากภาษาเครื่องที่แปลได้จะถูกเก็บไว้ที่หน่วยความจำสามารถเรียกใช้งานได้ทันที	- เมื่อเกิดข้อผิดพลาดกับโปรแกรมจะตรวจสอบหาข้อผิดพลาดได้ยาก เพราะทำการแปลผลทีเดียวทั้งโปรแกรม
อินเทอร์พรีเตอร์ (Interpreter)	- หาข้อผิดพลาดของโปรแกรมได้ง่าย เนื่องจากทำการแปลผลทีละบรรทัด - เนื่องจากทำงานทีละบรรทัด ดังนั้นจึงสั่งให้โปรแกรมทำงานตามคำสั่งเฉพาะจุดที่ต้องการได้ไม่เสียเวลารอการแปลโปรแกรมเป็นเวลานาน	- ช้า เนื่องจากทำงานทีละบรรทัด

1.4 การวางแผนและเขียนโปรแกรม

การเขียนโปรแกรมเพื่อให้คอมพิวเตอร์ทำงานตามที่เราต้องการนั้น ผู้เขียนโปรแกรมจะต้องรู้ว่าจะให้โปรแกรมทำอะไร มีข้อมูลอะไรที่ต้องให้กับโปรแกรมบ้าง และต้องการเอาผลลัพธ์อะไรจากโปรแกรม รวมไปถึงรูปแบบการแสดงผลอีกด้วย ผู้เขียนโปรแกรมจะต้องทราบถึงขั้นตอนการแก้ปัญหาโปรแกรมด้วยว่าจะต้องทำอะไร โดยเขียนเป็นลำดับขั้นตอนขึ้นมาก่อน แล้วจดบันทึกไว้ จากนั้นจึงนำลำดับขั้นตอนที่เขียนไว้มาพัฒนาเป็นโปรแกรม โดยทั่วไปแล้วการวางแผนและเขียนโปรแกรม มีขั้นตอนดังนี้



ขั้นตอนที่ 1 การกำหนดและวิเคราะห์ปัญหา

ขั้นตอนนี้เป็นขั้นตอนแรกที่คุณเขียนโปรแกรมจะต้องทำ เพราะถ้าหากผู้เขียนโปรแกรมไม่สามารถทำความเข้าใจกับปัญหาที่ต้องการแก้ไขได้ การนำคอมพิวเตอร์มาใช้ในการแก้ปัญหาต่างๆ ก็ไม่สามารถทำได้ ดังนั้นเราจะต้องมีแนวทางการแก้ปัญหาที่เหมาะสมให้กับคอมพิวเตอร์ เพื่อให้การทำงานเป็นไปอย่างมีประสิทธิภาพ การกำหนดและวิเคราะห์ปัญหามีขั้นตอนย่อยๆ ดังนี้

1. กำหนดขอบเขตของปัญหา โดยต้องกำหนดรายละเอียดให้ชัดเจนว่าจะให้คอมพิวเตอร์ทำอะไร ตัวแปร ค่าคงที่ ที่จะต้องใช้เป็นลักษณะใด หากไม่มีการกำหนดขอบเขตของปัญหา จะทำให้คอมพิวเตอร์ตัดสินใจได้ยากกว่าข้อมูลต่างๆ ที่เกิดขึ้นนั้นถูกหรือผิด

2. กำหนดลักษณะของข้อมูลเข้าและออกจากระบบ ผู้เขียนโปรแกรมต้องรู้ว่าข้อมูลที่จะส่งเข้าไปเป็นอย่างไร มีอะไรบ้าง เพื่อให้โปรแกรมทำการประมวลผลข้อมูลและแสดงผลลัพธ์ เช่น การใช้เมาส์ การรับค่าจากแป้นพิมพ์ การกำหนดปุ่มต่างๆ รูปแบบการแสดงผลทางหน้าจอ จะให้มีรูปร่างอย่างไร การออกแบบต้องคำนึงถึงผู้ใช้เป็นหลัก เช่น หากต้องการรับข้อมูลเข้าไปเพื่อประมวลผลก็ต้องพิจารณาว่าข้อมูลนั้นเป็นตัวอักษรหรือตัวเลข ถ้าเป็นตัวเลขก็ต้องพิจารณาว่าเป็นเลขจำนวนเต็มหรือเลขทศนิยม เอาต์พุตที่แสดงออกทางจอภาพจะแสดงผลทศนิยมกี่ตำแหน่ง เป็นต้น

3. กำหนดวิธีการประมวลผล โดยต้องรู้ว่าจะให้คอมพิวเตอร์ประมวลผลอย่างไร จึงจะได้ผลลัพธ์ตามต้องการ

ตัวอย่างเช่น ต้องการออกแบบโปรแกรมให้คอมพิวเตอร์คำนวณหาพื้นที่สี่เหลี่ยมผืนผ้า และแสดงค่าพื้นที่สี่เหลี่ยมผืนผ้าออกทางจอภาพ เราอาจกำหนดและวิเคราะห์ปัญหาได้ดังนี้

วัตถุประสงค์ของงาน

- คำนวณหาพื้นที่สี่เหลี่ยมผืนผ้า

ออกแบบผลลัพธ์ (Output)

- แสดงผลข้อมูลด้านกว้าง ด้านยาว และผลลัพธ์ที่ได้จากคำนวณพื้นที่สี่เหลี่ยมผืนผ้า

ข้อมูลนำเข้า (Input)

- ด้านยาวของสี่เหลี่ยม
- ด้านสูงของสี่เหลี่ยม

ชื่อตัวแปรที่ใช้

- ด้านยาว ใช้ชื่อตัวแปร Width
- ด้านสูง ใช้ชื่อตัวแปร Length
- พื้นที่ ใช้ชื่อตัวแปร Area



ขั้นตอนวิธีการประมวลผล/ลำดับงาน

- ป้อนข้อมูลด้านกว้าง
- ป้อนข้อมูลด้านยาว
- คำนวณหาพื้นที่สี่เหลี่ยมผืนผ้า $\text{Area} = \text{Width} \times \text{Length}$
- แสดงผลลัพธ์
- จบการทำงาน

ขั้นตอนที่ 2 การเขียนผังงาน

ขั้นตอนนี้ เป็นการนำเอารายละเอียดที่ได้จากการวิเคราะห์งาน มาเขียนเป็นแผนภาพหรือสัญลักษณ์ต่างๆ เพราะถ้าเป็นงานใหญ่ๆ ที่มีขั้นตอนมาก การเขียนโปรแกรมจากการเขียนวิเคราะห์งานโดยตรงจะทำได้ลำบาก การใช้ข้อความหรือคำอธิบายไม่ชัดเจนอาจทำให้งานผิดพลาดได้ ดังนั้นจึงต้องเปลี่ยนจากคำพูดหรือข้อความในการวิเคราะห์โจทย์ที่ได้มาเป็นรูปภาพหรือสัญลักษณ์เสียก่อน โดยอาจมีคำพูดหรือข้อความสั้นๆ อธิบายเพิ่มเติมในรูป ซึ่งรูปภาพหรือสัญลักษณ์ดังกล่าวนี้เรียกว่า “ผังงาน” (Flowchart)

การเขียนผังงานจะเป็นการถ่ายทอดความเข้าใจที่ได้จากการวิเคราะห์งานมาเป็นภาพหรือสัญลักษณ์ที่มีความหมายตามลำดับขั้นตอนของวิธีการประมวลผล ผู้ที่เขียนโปรแกรมจะเข้าใจลำดับขั้นตอนการเขียนโปรแกรมได้อย่างรวดเร็วและง่ายขึ้น โดยดูจากผังงาน นอกจากนี้ผังงานยังช่วยให้การตรวจสอบความถูกต้องของลำดับขั้นตอนในวิธีการประมวลผลกระทำได้อย่างสะดวก ในกรณีที่โปรแกรมมีข้อผิดพลาดอาจตรวจสอบดูที่ผังงานได้อีกครั้ง

ในวิธีการทางคอมพิวเตอร์ได้แบ่งผังงานตามลักษณะการใช้งานออกเป็น 2 ประเภท คือ ผังงานระบบ (System flowchart) และผังงานโปรแกรม (Program flowchart)

1. ผังงานระบบ (System flowchart)

ผังงานระบบ หมายถึง ผังงานที่แสดงถึงขั้นตอน วิธีการทำงานของระบบงานใดระบบหนึ่ง ซึ่งในผังงานระบบจะแสดงให้เห็นถึงสื่อที่รับข้อมูล บันทึกข้อมูล วิธีการประมวลผลขั้นตอนการทำงานและความสัมพันธ์ของส่วนต่างๆ ในระบบงานอย่างกว้างๆ เพื่อให้เห็นภาพรวมของระบบงานใดระบบงานหนึ่ง

2. ผังงานโปรแกรม (Program flowchart)

ผังงานโปรแกรม หมายถึง ผังงานที่แสดงถึงขั้นตอน และวิธีการทำงานของโปรแกรมที่ต้องการเขียนขึ้นภายในผังงานโปรแกรม จะแสดงให้เห็นขั้นตอนวิธีการทำงานของโปรแกรมอย่างละเอียด เพื่อให้คอมพิวเตอร์สามารถทำงานตามที่เขียนไว้ในผังงานโปรแกรมได้อย่างถูกต้อง โดยเรา



สามารถนำขั้นตอน วิธีการทำงานของโปรแกรมในขั้นการวิเคราะห์งาน มาเขียนเป็นผังงานโปรแกรม ได้ทันทีจากนั้นก็สมารถนำผังงานโปรแกรมไปเขียนโปรแกรมตามที่ต้องการได้

2.1 ประโยชน์ของผังงาน

- ทำให้เข้าใจและแยกแยะปัญหาต่างๆ ได้ง่ายขึ้น
- ผู้เขียนโปรแกรมมองเห็นลำดับการทำงาน รู้ว่าสิ่งใดควรทำก่อน สิ่งใดควรทำ

หลัง

- สามารถหาข้อผิดพลาดของโปรแกรมได้ง่าย
- ทำให้ผู้อื่นเข้าใจการทำงานได้ง่ายกว่าดูจาก source code
- ไม่ขึ้นกับภาษาคอมพิวเตอร์ภาษาใดภาษาหนึ่ง ผู้อื่นสามารถเรียนรู้และ

เข้าใจง่าย

2.2 ข้อจำกัดของผังงาน

ผู้เขียนโปรแกรมบางคนจะไม่นิยมเขียนผังงานก่อนการเขียนโปรแกรม เพราะ เห็นว่าเสียเวลา นอกจากนี้แล้วยังมีข้อจำกัดอื่นๆ อีก คือ

- ผังงานเป็นการสื่อความหมายระหว่างบุคคลกับบุคคลมากกว่าการสื่อความหมาย ระหว่างบุคคลกับเครื่อง เพราะผังงานไม่ขึ้นกับภาษาคอมพิวเตอร์ภาษาใดภาษาหนึ่ง ทำให้เครื่องไม่สามารถรับรู้และเข้าใจได้ว่า ในผังงานนั้นต้องการให้ทำอะไร

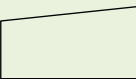
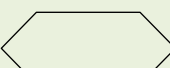
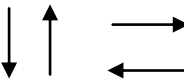
- ในบางครั้งเมื่อพิจารณาจากผังงาน จะไม่สามารถทราบได้ว่า ขั้นตอนการทำงานใดสำคัญกว่ากัน เพราะทุกๆ ขั้นตอนจะใช้รูปภาพหรือสัญลักษณ์ในลักษณะเดียวกัน

- การเขียนผังงานจะเป็นการสิ้นเปลือง เพราะต้องใช้กระดาษและอุปกรณ์อื่นๆ เขียนภาพ ซึ่งไม่สามารถเขียนด้วยมืออย่างเดียวได้ และในบางครั้งการเขียนผังงานอาจต้องใช้กระดาษมากกว่า 1 แผ่น หรือ 1 หน้า ซึ่งถ้าเป็นข้อความอธิบายอาจจะใช้เพียง 2-3 บรรทัดเท่านั้น

การเขียนผังงานต้องใช้ภาพสัญลักษณ์ต่างๆ มาเรียงต่อกัน เพื่อแสดงลำดับขั้นตอนการทำงานโดยมีลูกศรเชื่อมระหว่างภาพต่างๆ สัญลักษณ์ที่จะใช้ในการเขียนผังงานที่นิยมใช้กันทั่วไปแสดงดังตาราง 1.2



ตาราง 1.2 สัญลักษณ์ของผังงาน

สัญลักษณ์	ชื่อเรียก	ความหมาย
	เริ่มต้นและสิ้นสุด	แทนจุดเริ่มต้นและลงท้ายของผังงาน โปรแกรมหลักและโปรแกรมย่อย
	การปฏิบัติงาน	แทนจุดที่มีการปฏิบัติงานอย่างใดอย่างหนึ่ง
	การนำข้อมูลเข้า และนำออกโดยทั่วไป	แทนจุดที่จะนำข้อมูลเข้าหรือออกจากระบบคอมพิวเตอร์โดยไม่ระบุอุปกรณ์
	การนำเข้าข้อมูลด้วยมือ	แทนจุดที่จะนำข้อมูลเข้าด้วยมือ
	การตัดสินใจ	แทนจุดที่จะต้องเลือกปฏิบัติอย่างใดอย่างหนึ่ง
	การทำเอกสาร	แทนจุดที่มีข้อมูลเป็นเอกสารหรือแสดงข้อมูลด้วยเครื่องพิมพ์
	การแสดงผลข้อมูล	แทนจุดที่แสดงข้อมูลด้วยจอภาพ
	การเตรียมการ	แทนจุดกำหนดชื่อข้อมูลหรือค่าเริ่มต้นต่างๆ
	ฟังก์ชันย่อย	แทนจุดที่มีการเรียกใช้ฟังก์ชันย่อย
	ทิศทาง	แทนทิศทางขั้นตอนการดำเนินงาน ซึ่งจะปฏิบัติต่อเนื่องกันตามหัวลูกศรชี้
	จุดเชื่อมต่อ	แทนจุดเชื่อมต่อของผังงานที่อยู่หน้าเดียวกัน
	จุดเชื่อมต่อหน้ากระดาษ	แทนจุดเชื่อมต่อผังงานที่อยู่คนละหน้ากระดาษ



2.3 หลักในการเขียนผังงาน

การเขียนผังงานนั้น ไม่มีวิธีการที่แน่ชัดว่าจะต้องใช้คำสั่งอะไรบ้าง ขึ้นอยู่กับลักษณะของงานที่ทำ ซึ่งลักษณะงานจะมีอยู่ 3 ขั้นตอน คือ การรับข้อมูล (Input) การประมวลผล (Process) และการแสดงผล (Output) การศึกษาลำดับขั้นตอนในการทำงานของผังงาน ให้สังเกตจากลูกศรแสดงทิศทางการไหลของข้อมูลในผังงานเป็นหลัก โดยการเขียนผังงานต้องคำนึงถึงสิ่งต่างๆ ต่อไปนี้

- ใช้สัญลักษณ์ที่มีรูปแบบเป็นมาตรฐาน
- ขนาดของสัญลักษณ์ขึ้นอยู่กับความเหมาะสม
- ควรเขียนทิศทางการไหลของข้อมูล โดยเริ่มเขียนจากด้านบนลงด้านล่าง หรือเขียนจากด้านซ้ายไปด้านขวา และควรทำหัวลูกศรกำกับทิศทางไว้ด้วย
- การเขียนคำอธิบาย ให้เขียนภายในสัญลักษณ์ โดยใช้ข้อความที่เข้าใจง่าย สั้นๆ กระชับและชัดเจน
- พยายามให้เกิดจุดตัดน้อยที่สุด หรืออาจใช้สัญลักษณ์ "ตัวเชื่อม" (Connector) เขียนแทน เพื่อหลีกเลี่ยงข้อผิดพลาดที่อาจเกิดขึ้น
- หากเป็นไปได้ ควรเขียนผังงานให้จบภายในหน้าเดียวกัน
- ผังงานที่ดีควรเขียนให้เป็นระเบียบเรียบร้อย สะอาด ชัดเจน เข้าใจและติดตามขั้นตอนได้ง่าย
- จุดเริ่มต้นและสิ้นสุดของงาน ควรมีเพียงจุดเดียว

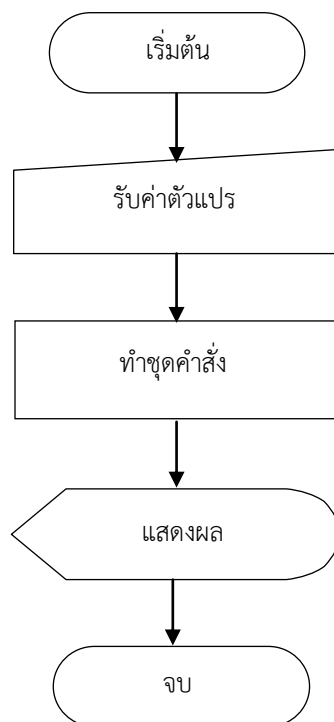
2.4 ลักษณะโครงสร้างของผังงาน

โดยทั่วไปโครงสร้างของผังงาน มีอยู่ 3 รูปแบบ ได้แก่ รูปแบบเรียงลำดับ รูปแบบที่มีการกำหนดเงื่อนไขหรือให้เลือก และรูปแบบที่มีการทำงานแบบวนรอบหรือทำซ้ำ

1) รูปแบบเรียงลำดับ (Sequence structure)

เป็นการทำงานแบบเรียงลำดับตั้งแต่ต้นจนจบ ไม่มีการเปรียบเทียบใดๆ โดยจะมีทิศทางการไหลของข้อมูลเพียงทางเดียว อาจจะเป็นแบบบนลงล่าง หรือจากซ้ายไปขวา สามารถเขียนเป็นผังงาน ดังภาพ 1.11



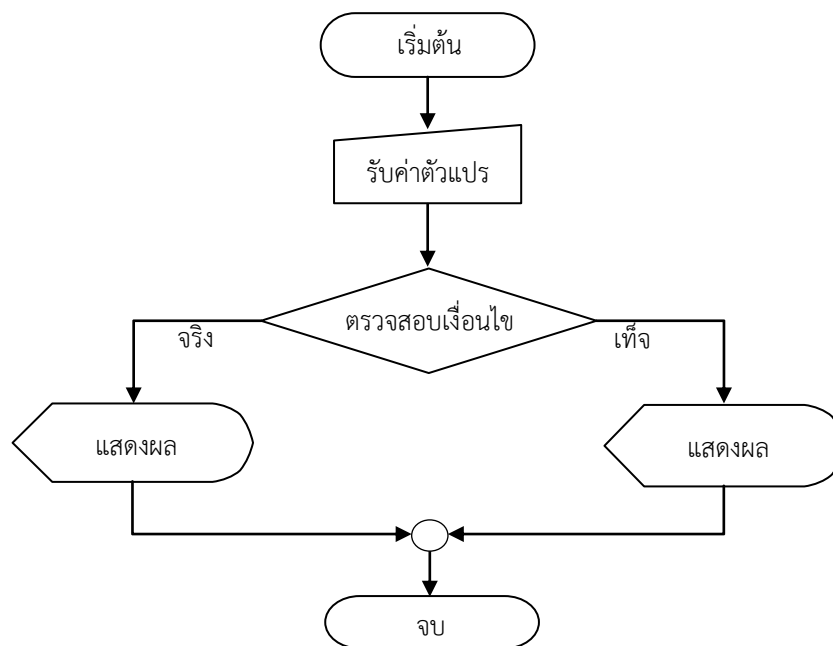


ภาพ 1.11 ผังงานรูปแบบเรียงลำดับ (Sequence structure)

2) รูปแบบที่มีการกำหนดเงื่อนไขหรือให้เลือก (Decision structure)

รูปแบบนี้จะมีการสร้างเงื่อนไขเพื่อให้เลือกทำงาน หากเลือกทางใดก็จะไปทำงานในเงื่อนไขที่เลือก ซึ่งเงื่อนไขที่กำหนดขึ้นนี้จะเขียนอยู่ในสัญลักษณ์ "การตัดสินใจ" สามารถเขียนเป็นผังงาน ดังภาพ 1.12

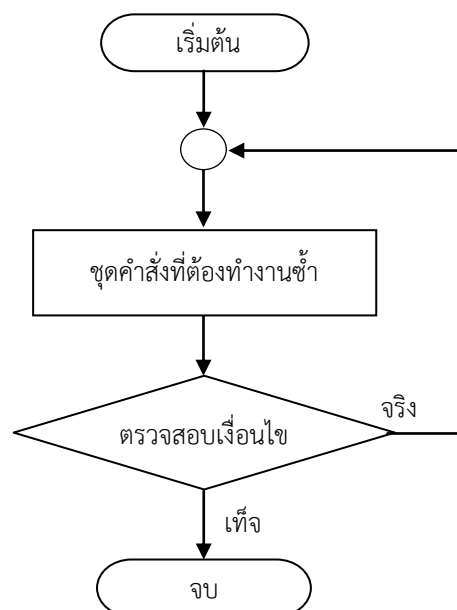




ภาพ 1.12 ผังงานรูปแบบที่มีการกำหนดเงื่อนไขหรือให้เลือก (Decision structure)

3) รูปแบบที่มีการทำงานแบบวนรอบหรือทำซ้ำ (Iteration structure)

การทำงานของรูปแบบนี้ จะเป็นการทำงานซ้ำกันหลายๆ ครั้ง เท่าที่เราต้องการ ซึ่งการทำงานนี้จะขึ้นอยู่กับเงื่อนไขที่กำหนดให้ และจะมีการนำเอาลักษณะของการตัดสินใจมาช่วยว่าจะมีการทำงานซ้ำอีกหรือไม่ สามารถเขียนเป็นผังงาน ดังภาพ 1.13



ภาพที่ 1.13 ผังงานรูปแบบที่มีการทำงานแบบวนรอบหรือทำซ้ำ (Iteration structure)



ขั้นตอนที่ 3 การเขียนโปรแกรม

หลังจากผ่านขั้นตอนการวิเคราะห์ปัญหาและเขียนผังงานแล้ว ขั้นตอนต่อไปคือ จะต้องเขียนโปรแกรมเพื่อให้คอมพิวเตอร์สามารถประมวลผลได้ การเขียนโปรแกรมจะต้องเขียนตามภาษาที่คอมพิวเตอร์เข้าใจ โดยอาจใช้ภาษาระดับสูงหรือระดับต่ำ ซึ่งสามารถเลือกได้หลายภาษา ทั้งนี้การเขียนโปรแกรมแต่ละภาษาจะต้องทำตามหลักไวยากรณ์ (Syntax) ที่กำหนดไว้ในภาษานั้น นอกจากนี้การเลือกใช้ภาษาจะต้องพิจารณาถึงความถนัดของผู้เขียนโปรแกรมอีกด้วย

ขั้นตอนที่ 4 การทดสอบและแก้ไขโปรแกรม

เป็นขั้นตอนการทดลองนำโปรแกรมที่เขียนขึ้น มาทดลองใช้และตรวจหาข้อผิดพลาด (Bug) เมื่อพบข้อผิดพลาด ผู้เขียนโปรแกรมจะต้องแก้ไขข้อผิดพลาดทันที ซึ่งการแก้ไขนี้ จะเรียกว่า ดีบั๊ก (Debug)

โปรแกรมที่ไม่สามารถทำงานได้ตามวัตถุประสงค์จะเรียกว่า ความผิดพลาด (error) และ การเกิดความผิดพลาดของโปรแกรมนั้นมีที่มาจากสาเหตุต่อไปนี้

1. ความผิดพลาดทางไวยากรณ์ภาษา (Syntax error) เป็นความผิดพลาดที่เกิดขึ้นจากการเขียนคำสั่งของภาษาคอมพิวเตอร์ผิด เช่น ฟังก์ชัน `printf( )` ในภาษาซีต้องเขียนด้วยอักษรตัวเล็ก แต่เขียนเป็น `PRINTF()` เป็นต้น โดยส่วนมากความผิดพลาดทางไวยากรณ์จะถูกตรวจสอบพบ เมื่อมีการแปลโปรแกรม (compile) ให้เป็นภาษาเครื่อง ซึ่งสามารถแก้ไขได้โดยการเขียนคำสั่งให้ถูกต้องตามไวยากรณ์ของภาษานั้นๆ

2. ความผิดพลาดทางตรรกะ (Logical error) เป็นความผิดพลาดที่เกิดขึ้นจากการลำดับการทำงานผิดหรือป้อนสูตรผิด เช่น ต้องการหาค่า $X = X + Y$ แต่ป้อนสูตรเป็น $X = X * Y$ เป็นต้น วิธีการตรวจหาความผิดพลาดแบบนี้ คือ ต้องตรวจสอบการคำนวณผลลัพธ์ของโปรแกรมว่าตรงกับผลลัพธ์ที่คำนวณด้วยมือหรือคำนวณด้วยเครื่องคิดเลขหรือไม่ ถ้าไม่ตรงกันแสดงว่าเกิดความผิดพลาดขึ้นทางตรรกะ วิธีแก้ไขคือ แก้ไขสูตรให้ถูกต้อง หรือแก้ไขลำดับการทำงานให้ถูกต้อง



ขั้นตอนที่ 5 การทำเอกสารและบำรุงรักษาโปรแกรม

ในขั้นตอนนี้ จะช่วยให้ผู้ใช้โปรแกรมใช้โปรแกรมได้อย่างมีประสิทธิภาพและสะดวกต่อการตรวจสอบข้อผิดพลาด โดยเขียนเป็นเอกสารประกอบโปรแกรมขึ้นมา แบ่งได้ 2 ประเภท ดังนี้

1. คู่มือการใช้ หรือ User document หรือ User guide ซึ่งจะอธิบายวิธีการใช้โปรแกรม
2. คู่มือโปรแกรมเมอร์ หรือ Program document หรือ Technical reference ซึ่งจะ

อำนวยความสะดวกในการแก้ไขโปรแกรม และพัฒนาโปรแกรมในอนาคต โดยมีรายละเอียดต่างๆ เกี่ยวกับโปรแกรม เช่น ชื่อของโปรแกรม การรับข้อมูล การพิมพ์ผลลัพธ์ของขั้นตอนต่างๆ ในโปรแกรม เป็นต้น

สำหรับการบำรุงรักษาโปรแกรม (Maintenance) เป็นการที่ผู้เขียนโปรแกรมจะต้องคอยตรวจสอบการใช้โปรแกรมจริง เพื่อแก้ไขข้อผิดพลาดซึ่งอาจจะเกิดขึ้นได้ในภายหลัง รวมทั้งพัฒนาโปรแกรมให้ทันสมัยอยู่เสมอเมื่อเวลาผ่านไป

สรุปท้ายหน่วย

- ภาษาคอมพิวเตอร์ แบ่งได้ 5 ระดับ คือ ภาษาเครื่อง ภาษาแอสเซมบลี ภาษาระดับสูง ภาษาระดับสูงมาก และภาษาธรรมชาติ
- ซอฟต์แวร์หรือโปรแกรม แบ่งออกเป็น 2 ประเภท คือ โปรแกรมระบบ และโปรแกรมประยุกต์
- ตัวแปลภาษาระดับสูงให้เป็นภาษาที่เครื่องคอมพิวเตอร์สามารถเข้าใจ มี 2 ประเภท คือ ตัวแปลโปรแกรมหรือคอมไพเลอร์ (Compiler) จะแปลโปรแกรมทั้งโปรแกรม และตัวแปลคำสั่งหรืออินเทอร์พรีเตอร์ (Interpreter) จะแปลทีละประโยคคำสั่ง
- ขั้นตอนการวางแผนเขียนและโปรแกรม มี 5 ขั้นตอน ได้แก่ กำหนดและวิเคราะห์ปัญหา การเขียนผังงาน การเขียนโปรแกรม การทดสอบและแก้ไขโปรแกรม การทำเอกสารประกอบและบำรุงรักษาโปรแกรม



แบบฝึกหัด

หน่วยที่ 1 หลักการของภาษา การวางแผน และเขียนโปรแกรม

แบบฝึกหัดที่ 1.1 หลักการของภาษาและโปรแกรมคอมพิวเตอร์

จุดประสงค์การเรียนรู้

1. บอกระดับของภาษาคอมพิวเตอร์และประเภทของโปรแกรมคอมพิวเตอร์ได้
2. บอกลักษณะของตัวแปลภาษาได้
3. มีคุณลักษณะที่พึงประสงค์ที่ดีในด้านการเรียนและงานอาชีพ

เวลาทำกิจกรรม : 60 นาที

คำชี้แจง แบบฝึกหัดมีทั้งหมด 2 ตอน ให้นักเรียนทำทุกข้อ

ตอนที่ 1 ให้นักเรียนตอบคำถามต่อไปนี้ให้ได้ใจความที่สมบูรณ์ และถูกต้อง

1. จงบอกประเภทของโปรแกรมคอมพิวเตอร์ว่ามีกี่ประเภท อะไรบ้าง (1 คะแนน)

ตอบ _____

2. จงบอกระดับของภาษาคอมพิวเตอร์ว่ามีกี่ระดับ อะไรบ้าง (1 คะแนน)

ตอบ _____

3. จงบอกความแตกต่างของภาษาระดับสูงกับภาษาเครื่อง (1 คะแนน)

ตอบ _____

4. จงเขียนชื่อโปรแกรมภาษาระดับสูงที่นิยมใช้มาอย่างน้อย 3 ชื่อ (1 คะแนน)

ตอบ _____



5. จงบอกความแตกต่างของอินเทอร์พรีเตอร์และคอมไพเลอร์ (1 คะแนน)

ตอบ _____

ตอนที่ 2 ให้นักเรียนขีดเครื่องหมาย ✓ หน้าข้อที่ถูก และขีดเครื่องหมาย ✗ หน้าข้อที่ผิด
 (5 คะแนน)

- 1 _____ วินโดวส์ (Windows) จัดว่าเป็นโปรแกรมระบบปฏิบัติการ
- 2 _____ โปรแกรม MacAfee Virus Scans เป็นซอฟต์แวร์สำเร็จรูป
- 3 _____ ฟรีแวร์ (Freeware) เป็นโปรแกรมที่สามารถดาวน์โหลดได้จากอินเทอร์เน็ตหรือหามาใช้ได้ฟรี
- 4 _____ ภาษาระดับสูง คือ ภาษาคอมพิวเตอร์ที่ใกล้เคียงกับภาษาของมนุษย์
- 5 _____ ภาษาแอสเซมบลี (Assembly) เป็นภาษาเครื่อง
- 6 _____ ภาษาเบสิก (Basic) เป็นภาษาระดับต่ำ
- 7 _____ ตัวอย่างของภาษาระดับสูง ได้แก่ ภาษาปาสคาล (Pascal) และภาษาซี เป็นต้น
- 8 _____ แอสแซมเบอร์ เป็นตัวแปลภาษาของภาษาแอสเซมบลี
- 9 _____ อินเทอร์พรีเตอร์ เป็นตัวแปลคำสั่ง ที่จะแปลคำสั่งทั้งโปรแกรม ให้เป็นคำสั่งภาษาเครื่อง แล้วทำงานตามคำสั่งนั้น
- 10 _____ ภาษาซี ต้องใช้คอมไพเลอร์ เป็นตัวแปลคำสั่งให้เป็นคำสั่งภาษาเครื่อง



แบบฝึกหัด

หน่วยที่ 1 หลักการของภาษา การวางแผน และเขียนโปรแกรม

แบบฝึกหัดที่ 1.2 การวางแผนและเขียนโปรแกรม

จุดประสงค์การเรียนรู้

1. บอกขั้นตอนการวางแผนเขียนและโปรแกรมได้ถูกต้อง
2. บอกลักษณะโครงสร้างของผังงานและสัญลักษณ์ของผังงานได้ถูกต้อง
3. ปฏิบัติการเขียนผังงานโปรแกรมได้ถูกต้อง
4. มีคุณลักษณะที่พึงประสงค์ที่ดีในด้านการเรียนและงานอาชีพ

เวลาทำกิจกรรม : 120 นาที

คำชี้แจง ให้นักเรียนตอบคำถามต่อไปนี้ให้ได้ใจความที่สมบูรณ์ และถูกต้อง

1. จงบอกขั้นตอนการวางแผนและเขียนโปรแกรมมีกี่ขั้นตอน อะไรบ้าง (2 คะแนน)

ตอบ _____

2. จงบอกลักษณะโครงสร้างของผังงานว่ามีกี่รูปแบบ อะไรบ้าง (2 คะแนน)



3. จงบอกชื่อเรียกและความหมายของสัญลักษณ์ผังงานต่อไปนี้ (2 คะแนน)

สัญลักษณ์	ชื่อเรียก	ความหมาย
		
		
		
		
		
		
		
		
		
		
		



4. จงวิเคราะห์งาน เพื่อเขียนโปรแกรมคำนวณหาค่าเฉลี่ยของตัวเลข 3 จำนวน (2 คะแนน)

- วัตถุประสงค์ของงาน

ตอบ _____

- ผลลัพธ์ (Output)

ตอบ _____

- ข้อมูลนำเข้า (Input)

ตอบ _____

- ชื่อตัวแปรที่ใช้

ตอบ _____

- ขั้นตอนวิธีการประมวลผล/ลำดับงาน

ตอบ _____



5. จงเขียนผังงานโปรแกรมคำนวณหาค่าเฉลี่ยของตัวเลข 3 จำนวน (2 คะแนน)



แบบทดสอบหลังการเรียนรู้

หน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม
รายวิชาการเขียนโปรแกรมคอมพิวเตอร์ รหัสวิชา 2105-2118
ระดับประกาศนียบัตรวิชาชีพ สาขาวิชาช่างอิเล็กทรอนิกส์
จำนวน 15 ข้อ 10 คะแนน เวลา 15 นาที

ตอนที่ 1 ข้อสอบแบบปรนัย เลือกตอบ 4 ตัวเลือก

คำชี้แจง ให้นักเรียนทำเครื่องหมาย X ทับตัวอักษรหน้าคำตอบที่ถูกต้องที่สุด (5 คะแนน)

- โปรแกรมที่ทำหน้าที่ควบคุมการทำงานของเครื่องคอมพิวเตอร์ คือข้อใด ?

ก. โปรแกรมเฉพาะ	ข. โปรแกรมระบบ
ค. โปรแกรมประยุกต์	ง. โปรแกรมสำเร็จรูป
- ข้อใด เป็นโปรแกรมสำเร็จรูป ?

ก. DOS	ข. WinZip
ค. Windows	ง. Microsoft Excel
- ข้อใด เป็นลักษณะของภาษาเครื่อง ?

ก. เป็นภาษาที่ใช้กับงานขั้นสูง
ข. เป็นภาษาคอมพิวเตอร์ยุคใหม่
ค. ทำงานได้โดยไม่ต้องมีโปรแกรมระบบ
ง. เป็นภาษาที่คอมพิวเตอร์เข้าใจและปฏิบัติตามคำสั่งทันที
- ข้อใด เป็นภาษาระดับสูงทั้งหมด ?

ก. ภาษาซี ภาษาแอสเซมบลี	ข. ภาษาโคบอล ภาษาฟอร์เทน
ค. ภาษา SQL ภาษาปาสคาล	ง. ภาษาแอสเซมบลี ภาษาเบสิก
- ลักษณะการแปลภาษาคอมพิวเตอร์ให้เป็นภาษาเครื่องที่มีการแปลทีละบรรทัด คือข้อใด ?

ก. รันไทม์	ข. แอสเซมเบอร์
ค. คอมไพเลอร์	ง. อินเทอร์พรีเตอร์
- ภาษาคอมพิวเตอร์ภาษาใดที่ต้องใช้ตัวแปลภาษาที่เรียกว่า “คอมไพเลอร์” ?

ก. ภาษาเบสิก	ข. ภาษา SQL
ค. ภาษาปาสคาล	ง. ภาษาแอสเซมเบอร์



9. 

ตอนที่ 2 ข้อสอบแบบอัตนัย

คำชี้แจง ให้นักเรียนเขียนคำตอบลงในช่องว่างที่กำหนดให้ (5 คะแนน)

1. จงบอกภาษาคอมพิวเตอร์ แบ่งออกเป็นกี่ระดับ อะไรบ้าง ? (1 คะแนน)

2. จงยกตัวอย่างโปรแกรมที่เป็นโปรแกรมประยุกต์ มาอย่างน้อย 3 ตัวอย่าง ? (1 คะแนน)

3. จงยกตัวอย่างโปรแกรมภาษาที่เป็นภาษาระดับสูงมา 3 โปรแกรม ? (1 คะแนน)

4. จงอธิบายความแตกต่างของคอมไพเลอร์ (Compiler) และอินเทอร์พรีเตอร์ (Interpreter) ?
(1 คะแนน)

5. จงบอกขั้นตอนการวางแผนเขียนและโปรแกรม มีกี่ขั้นตอน อะไรบ้าง ? (1 คะแนน)



เกณฑ์การประเมินผลแบบฝึกหัด

หน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม

เกณฑ์การประเมินผลแบบฝึกหัดที่ 1.1 (10 คะแนน)

ตอนที่ 1 (5 คะแนน)

ข้อที่	ระดับคะแนน		
	3 (1 คะแนน)	2 (0.5 คะแนน)	1 (0 คะแนน)
1-5	ตอบถูกต้อง ครบถ้วน สมบูรณ์ ทุกประเด็น คำถาม	ตอบถูกต้องเป็นส่วนใหญ่ แต่มีบางประเด็นที่ตอบผิด	ตอบผิดเป็นส่วนใหญ่ หรือตอบไม่ถูกเลย

ตอนที่ 2 ข้อที่ 1-10 (5 คะแนน)

-	ตอบถูก	ให้คะแนน
0.5 คะแนน		
-	ตอบผิดหรือไม่ตอบ	ให้
คะแนน 0.0 คะแนน		

เกณฑ์การประเมินผลแบบฝึกหัดที่ 1.2 (10 คะแนน)

ข้อที่	ระดับคะแนน		
	3 (2 คะแนน)	2 (1 คะแนน)	1 (0 คะแนน)
1-5	ตอบถูกต้อง ครบถ้วน สมบูรณ์ ทุกประเด็น คำถาม	ตอบถูกต้องเป็นส่วนใหญ่ แต่มีบางประเด็นที่ตอบผิด	ตอบผิดเป็นส่วนใหญ่ หรือตอบไม่ถูกเลย

หมายเหตุ คะแนนรวม (10 คะแนน) = (คะแนนแบบฝึกหัดที่ 1.1+คะแนนแบบฝึกหัดที่ 1.2) ÷ 2



แบบประเมินคุณลักษณะที่พึงประสงค์

หน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม

คำชี้แจง ทำเครื่องหมาย ✓ ลงในช่องที่ตรงกับพฤติกรรมที่นักเรียนแสดงออก

รายการ	ลำดับ	พฤติกรรมที่แสดงออก	ระดับคะแนน		
			3	2	1
มีความเสียสละ ความซื่อสัตย์ สุจริต ความมีวินัย ความ รับผิดชอบ การมีมนุษยสัมพันธ์ ความรักสามัคคี ขยัน ประหยัด อดทน ความสนใจใฝ่เรียนรู้ และ ความคิดริเริ่มสร้างสรรค์	1	เข้าเรียนตรงเวลา			
	2	แต่งกายถูกต้องตามระเบียบของวิทยาลัย			
	3	มีความรับผิดชอบ			
	4	มีความละเอียด รอบคอบ และปลอดภัย			
	5	ให้คำแนะนำช่วยเหลือผู้อื่น			
	6	มีส่วนร่วมในการทำงาน			
	7	มีความสนใจใฝ่เรียนรู้			
	8	มีความคิดริเริ่มสร้างสรรค์			
	9	ใช้พลังงานอย่างประหยัด			
	10	กล้าแสดงออกและยอมรับอย่างมีเหตุผล			
คะแนนรวม					
ค่าเฉลี่ย (คะแนนรวม ÷ 3)					

เกณฑ์การประเมิน

- 3 หมายถึง มีพฤติกรรมที่เหมาะสมมาก
- 2 หมายถึง มีพฤติกรรมที่เหมาะสมปานกลาง
- 1 หมายถึง มีพฤติกรรมที่ควรปรับปรุง

เกณฑ์การผ่านการประเมิน ได้คะแนน 5 ขึ้นไป

ลงชื่อ.....ผู้ประเมิน
(นายนิทัศน์ ศรีเทียมศักดิ์)



แบบประเมินตนเอง

หน่วยที่ 1 หลักการของภาษา การวางแผนและเขียนโปรแกรม

คำชี้แจง ให้นักเรียนประเมินตนเองและหาค่าเฉลี่ยของคะแนน

ลำดับ ที่	รายการ	คะแนนที่ได้	การประเมินคุณภาพ
1	การทดสอบก่อนการเรียนรู้ (คะแนนเต็ม 10 คะแนน)		-
2	การทดสอบหลังการเรียนรู้ (คะแนนเต็ม 10 คะแนน)		ผ่าน ☐ ดีมาก ☐ ดี ☐ พอใช้ ไม่ผ่าน ☐ ต้องปรับปรุง
3	การประเมินผลขณะเรียนรู้ 3.1 แบบฝึกหัด (คะแนนเต็ม 10 คะแนน) 3.2 คุณลักษณะที่พึงประสงค์ (คะแนนเต็ม 10 คะแนน) ค่าคะแนนเฉลี่ย (10 คะแนน) หมายเหตุ คะแนนเฉลี่ย = (คะแนนรวมข้อ 3.1+3.2) ÷ 2		ผ่าน ☐ ดีมาก ☐ ดี ☐ พอใช้ ไม่ผ่าน ☐ ต้องปรับปรุง
คะแนนตามค่าน้ำหนักคะแนนการเรียนรู้ตามสมรรถนะ (5 คะแนน) หมายเหตุ คะแนนสมรรถนะ = ((คะแนนที่ได้ในข้อ 2+ ข้อ 3)) x5 ÷20			-

เกณฑ์การประเมินคุณภาพ

9-10 คะแนน หมายถึง ดีมาก

7-8 คะแนน หมายถึง ดี

5-6 คะแนน หมายถึง พอใช้

1-4 คะแนน หมายถึง ต้องปรับปรุง

ลงชื่อ.....ผู้ประเมิน
(.....)



เอกสารอ้างอิง

- คำตัน ธรรมวงศ์. (2557). **โหลดโปรมี**. สืบค้นเมื่อวันที่ 10 ตุลาคม 2557,
จาก <http://loadprome.blogspot.com>.
- จุฬาลักษณ์ ถายไยลา. (2551). **บทที่ 7 การเขียนผังงาน**. สืบค้นเมื่อวันที่ 24 สิงหาคม 2557,
จาก <http://www.sttc.ac.th/~computerbc/backup/elearning/GUI/4.pdf>.
- ชานัญญา บุญรัตนเสถียร. (2555). **การเขียนโปรแกรม**. สืบค้นเมื่อวันที่ 24 สิงหาคม 2557,
จาก <http://ptomoechaniya.blogspot.com/2012/07/blog-post.html>.
- ธนาวุฒิ ประกอบผล. (2554). **การเขียนโปรแกรมเบื้องต้นด้วย ภาษา C**. กรุงเทพฯ:
ซีคเซสมิเดีย.
- เนาวรัตน์ ใจการุณ. (2550). **หลักการโปรแกรมเบื้องต้น**. สืบค้นเมื่อวันที่ 12 มกราคม 2557,
จาก http://www.thaigoodview.com/library/teachershow/lopburinaowarta_s/programming/sec05p07.html.
- พัฒพงษ์ อมรวงศ์. (2554). **การเขียนโปรแกรมภาษาซี**. ปทุมธานี: มีเดีย อินเทลลิเจนซ์เทคโนโลยี.
มหาวิทยาลัยนเรศวร. (2551). **คู่มือการจัดทำสารนิพนธ์ มหาวิทยาลัยนเรศวร ฉบับ พ.ศ. 2550**
(พิมพ์ครั้งที่ 2). พิษณุโลก: รัตนสุวรรณ 3.
- วัชรภรณ์ สุริยาภวัฒน์. (2540). **คอมพิวเตอร์เบื้องต้นและเทคนิคการเขียนโปรแกรม**. กรุงเทพฯ:
ไทยเจริญการพิมพ์.
- อรพิน ประวัติบริสุทธิ. (2554). **คู่มือเรียนภาษาซี ฉบับปรับปรุงใหม่**. กรุงเทพฯ: โปริวิชั่น.
nu gad. (2011). **ภาษาเบสิก**. สืบค้นเมื่อวันที่ 12 มกราคม 2557, จาก <http://konpretty.blogspot.com>.
- SAIAM iDeas. (2014). **Open Source Technologies**. สืบค้นเมื่อวันที่ 12 มกราคม 2557,
จาก <http://saiaminfosoft.com/open-source-technologies.html>.
- scriptux.com. (2014). **Download**. สืบค้นเมื่อวันที่ 12 มกราคม 2557,
จาก <http://www.scriptux.com>.

เอกสารอ้างอิง (ต่อ)

- techGenie. (2014). **How to Compare Windows & Linux Security**. สืบค้นเมื่อวันที่ 12 มกราคม 2557, จาก <http://news.techgenie.com>.
- Thailand Freeware. (2014). **โหลดโปรแกรมฟรี โหลดซอฟต์แวร์ฟรี**. สืบค้นเมื่อวันที่ 12 มกราคม 2557, จาก <http://thailandfreeware.com>.
- THAIWARE. (2014). **ดาวน์โหลดโปรแกรม**. สืบค้นเมื่อวันที่ 12 มกราคม 2557, จาก <http://software.thaiware.com/3758-IDM.html>.