

องค์ประกอบของโปรแกรม ภาษาคอมพิวเตอร์

หน่วยที่

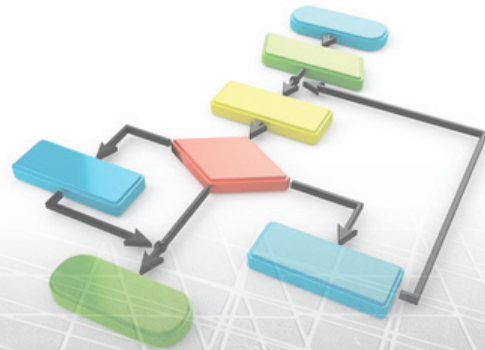
2

สาระการเรียนรู้

1. ภาษาคอมพิวเตอร์
2. โปรแกรมแปลภาษาคอมพิวเตอร์
3. องค์ประกอบของโปรแกรมภาษาคอมพิวเตอร์

สมรรถนะการเรียนรู้

1. บอกภาษาคอมพิวเตอร์ที่รู้จักได้
2. อธิบายลักษณะต่าง ๆ ของโปรแกรมแปลภาษาคอมพิวเตอร์ได้
3. บอกองค์ประกอบของโปรแกรมภาษาคอมพิวเตอร์ได้



หน่วยที่ 2

องค์ประกอบของโปรแกรมภาษาคอมพิวเตอร์

ภาษาโปรแกรม (Programming Languages) หมายถึง ภาษาประดิษฐ์ชนิดหนึ่งที้ออกแบบขึ้นมาเพื่อสื่อสารชุดคำสั่งแก่เครื่องจักรโดยเฉพาะคอมพิวเตอร์ ภาษาโปรแกรมสามารถใช้สร้างโปรแกรมที่ควบคุมพฤติกรรมของเครื่องจักรหรือการแสดงออกด้วยขั้นตอนวิธี (Algorithm) อย่างตรงไปตรงมา

ภาษาคอมพิวเตอร์ (Computer Language)

ภาษาคอมพิวเตอร์ (Computer Language) หมายถึง ชุดคำสั่งต่างๆ ที่จะสั่งให้เครื่องคอมพิวเตอร์ทำงาน ซึ่งการเขียนคำสั่งเหล่านั้นเรียกว่า โปรแกรม (Program) ถ้ามีหลาย ๆ คำสั่ง จะเรียกว่า ชุดคำสั่ง (Statement) ซึ่งปัจจุบันมีอยู่หลายภาษาตามความถนัดของผู้ใช้ เพื่อให้ผู้เขียนโปรแกรมสามารถเลือกได้ว่า จะเขียนโปรแกรมด้วยภาษาใด เพื่อให้เหมาะกับงานที่ต้องการ โดยลักษณะที่คล้ายคลึงกันของภาษาคอมพิวเตอร์ทุกภาษามีดังนี้

1. คำสั่งรับข้อมูลและแสดงผล (Receive Data and Display) คำสั่งประเภทนี้จำเป็นอย่างยิ่งที่จะต้องมีการใช้ในภาษา ทั้งยังต้องแจกแจงรายละเอียดต่อไปด้วยว่ารับผ่านอุปกรณ์ใดและแสดงผลทางอุปกรณ์ใด
2. คำสั่งคำนวณ (Calculated) โปรแกรมหรือคำสั่งที่เขียนจะหนีไม่พ้นคำสั่งที่สั่งให้ประมวลผลประเภท บวก ลบ คูณ และหาร
3. คำสั่งที่มีการเลือกทิศทาง (Direction) หมายถึง สั่งให้มีการเปรียบเทียบ เช่น ถ้ามากกว่า ให้ทำอย่างหนึ่ง ถ้าเท่ากันให้ทำอย่างหนึ่ง หรือน้อยกว่าให้ทำอีกอย่างหนึ่ง เป็นต้น นอกจากนั้นอาจมีคำสั่งประเภทให้ทำงานเป็นวงซ้ำแล้วซ้ำอีกจนกว่าจะมีการเปรียบเทียบค่าจึงจะให้หยุดได้
4. คำสั่งเข้าออกของข้อมูล (Input and Output Data) หมายถึง การนำโปรแกรมหรือข้อมูลออกหรือส่งเข้าไปเก็บในสื่ออย่างใดอย่างหนึ่งเพื่อเรียกมาใช้ใหม่ได้

ภาษาเครื่อง (Machine Language)

ภาษาเครื่องเป็นภาษารุ่นแรก ๆ ที่ผู้ใช้งานคอมพิวเตอร์ใช้เขียนคำสั่งเพื่อสั่งให้เครื่องคอมพิวเตอร์ทำงาน ภาษาเครื่องเป็นภาษาที่เขียนเป็นรหัสเลขฐานสอง (Binary Code) คือจะใช้เฉพาะตัวเลข 0 และ 1 เท่านั้น เขียนสลับกันไปมาเพื่อใช้เป็นรหัสสั่งงาน เป็นคำสั่งที่ให้เครื่องคอมพิวเตอร์ทำงานตามที่ผู้ใช้ต้องการ ภาษาเครื่องเป็นภาษาที่เขียนยากที่สุดเพราะผู้เขียน

ต้องจำรหัสคำสั่งต่าง ๆ เป็นจำนวนมาก สมัยก่อนผู้ที่เขียนโปรแกรมคำสั่งด้วยภาษาเครื่องหรือโปรแกรมเมอร์ (Programmer) จึงมีจำนวนน้อยมาก ต้องเป็นผู้ที่รู้เรื่องของคอมพิวเตอร์และระบบเลขฐานสองจริง ๆ เท่านั้นจึงจะสามารถเขียนโปรแกรมได้ ภาษาเครื่องจึงไม่เป็นที่นิยมในการเขียนโปรแกรมมากนัก แต่ภาษาเครื่องเป็นภาษาเดียวที่คอมพิวเตอร์สามารถเข้าใจคำสั่งที่ผู้ใช้เขียนและปฏิบัติตามได้ทันที โดยไม่ต้องมีตัวกลางในการแปลภาษาอีกทีหนึ่ง

ตัวอย่าง การเขียนชุดคำสั่งด้วยภาษาเครื่อง

ถ้าต้องการเขียนชุดคำสั่ง ให้เครื่องบวกเลขสองจำนวน เช่น $9+3$ ต้องเขียนดังนี้

เลข 9 แทนด้วยรหัส คือ 00111001

เลข 3 แทนด้วยรหัส คือ 00110011

เครื่องหมาย + แทนด้วยรหัส คือ 00101011

ดังนั้น คำสั่ง $9+3$ เมื่อเขียนเป็นชุดคำสั่งภาษาเครื่องจะได้ดังนี้

00111001 00101011 00110011

เนื่องจากการเขียนชุดคำสั่งต่าง ๆ ให้เครื่องคอมพิวเตอร์ทำงานนั้นเป็นเรื่องที่ยุ่งยากและไม่ค่อยได้รับความนิยม จึงมีผู้คิดค้นสร้างภาษาใหม่ขึ้นมาเพื่อให้เขียนชุดคำสั่งต่าง ๆ ได้ง่ายขึ้นกว่าเดิม โดยจะต้องมีตัวกลางในการแปลชุดคำสั่งต่าง ๆ ที่เขียนให้เป็นภาษาที่เครื่องเข้าใจ เพื่อเครื่องจะได้ปฏิบัติตามได้ถูกต้อง การพัฒนาเช่นนี้ทำให้นักุษย์กับคอมพิวเตอร์เข้าใจและสื่อสารกันได้ง่ายขึ้น การสั่งงานจึงทำได้ง่ายและสะดวกขึ้นทุกที โดยภาษาคอมพิวเตอร์นี้สามารถแบ่งได้เป็น 2 ระดับ ดังนี้

ภาษาระดับต่ำ (Low Level Language)

ภาษาระดับต่ำที่ใช้อยู่ในปัจจุบันนี้มีอยู่เพียงภาษาเดียวก็คือ **ภาษาแอสเซมบลี (Assembly)** โดยภาษานี้จะมีการนำตัวอักษรย่อหรือสัญลักษณ์ต่าง ๆ มาใช้แทนชุดคำสั่งที่เป็นเลขฐานสอง ซึ่งทำให้ผู้ใช้สามารถจำชุดคำสั่งได้ง่ายขึ้น เช่น ADD หมายถึง บวก, SUB หมายถึง ลบ เช่น $A \text{ SUB } X, Y$ หมายถึง ให้นำค่า A ลบออกจากค่า X และนำค่าที่ได้ไปเก็บไว้ที่ Y เป็นต้น

ตัวอย่าง การเขียนโปรแกรมด้วยภาษา Assembly

ถ้าต้องการเขียนชุดคำสั่งให้เครื่องบวกเลขสองจำนวน เช่น $9+3$ ต้องเขียนดังนี้

MOV AX, 9

MOV BX, 3

ADD AX, BX

ถึงแม้ว่าภาษา Assembly จะเขียนง่ายกว่าภาษาเครื่อง แต่ก็ยังคงยุ่งยากอยู่เพราะต้องมีการอ้างอิงถึงส่วนประกอบของตัวเครื่องคอมพิวเตอร์หลายอย่าง เช่น Register, Memory เป็นต้น ดังนั้นจึงมีผู้คิดค้นสร้างภาษาระดับสูงขึ้นมา เพื่อให้ผู้คนทั่วไปสามารถเขียนโปรแกรมหรือชุดคำสั่งได้ง่ายยิ่งขึ้นนั่นก็คือ ภาษาระดับสูง (High Level Language)

ภาษาระดับสูง (High Level Language)

ภาษาระดับสูงเป็นภาษาที่เขียนได้ง่าย เพราะมีการนำคำศัพท์และข้อความต่าง ๆ ที่บุคคลทั่วไปใช้ในชีวิตประจำวันอยู่แล้วมาเขียนเป็นชุดคำสั่งเพื่อให้เครื่องคอมพิวเตอร์ทำงาน เช่น คำว่า Read, Print, Goto, End เป็นต้น ส่วนการคำนวณต่าง ๆ ก็เขียนเป็นคำสั่งที่มีลักษณะคล้ายกับสูตรทางคณิตศาสตร์ ทำให้ผู้ใช้สามารถศึกษาและเขียนชุดคำสั่งได้ง่ายขึ้น เช่น

ถ้าต้องการหาค่าผลบวกเลขสองจำนวน เช่น $9+3$ แล้วพิมพ์ผลลัพธ์แสดงออกมาทางจอภาพ จะสามารถเขียนเป็นคำสั่งได้ดังนี้

$$X = 9 + 3$$

Print X

ตัวอย่าง ภาษาระดับสูง

FORTRAN	ย่อมาจาก	Formular Translater
BASIC	ย่อมาจาก	Beginner's All Purpose Symbolic Instruction Code
PL/1	ย่อมาจาก	Programming Language Version1
ALGOL	ย่อมาจาก	ALGOarithmic Language
COBOL	ย่อมาจาก	Common Business Oriented Language
RPG	ย่อมาจาก	Report Program Generator
PASCAL		
Ada	ย่อมาจาก	Add Lovelace
Java		Java Programming Language
C,C++		C C++ Programming Language
VB	ย่อมาจาก	Visual Basic
Delphi		

ความแตกต่างระหว่างภาษาระดับต่ำและภาษาระดับสูง

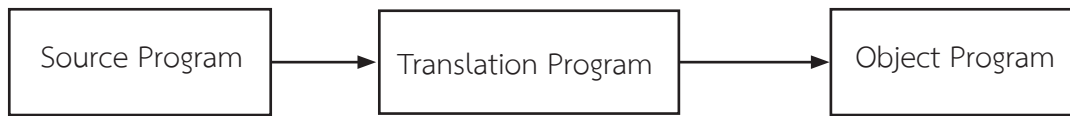
ภาษาระดับต่ำ	ภาษาระดับสูง
1. ขึ้นอยู่กับ Hardware ของแต่ละเครื่อง	1. ไม่ขึ้นอยู่กับ Hardware เครื่องใดเครื่องหนึ่ง
2. จำเป็นต้องศึกษาและเข้าใจระบบการทำงานภายในของเครื่องคอมพิวเตอร์	2. ไม่จำเป็นต้องศึกษาระบบการทำงานภายในของเครื่องคอมพิวเตอร์
3. การเขียนโปรแกรมใช้เวลานาน เพราะต้องเขียนขั้นตอนการทำงานอย่างละเอียด	3. การเขียนโปรแกรมใช้เวลาน้อยเพราะคำสั่งเข้าใจได้ง่าย
4. โปรแกรมอ่านยากและคนอื่นเข้าใจได้ยาก	4. โปรแกรมอ่านง่ายและคนอื่นสามารถเข้าใจได้ง่าย
5. การพัฒนาโปรแกรมทำได้ยาก	5. การพัฒนาโปรแกรมทำได้ง่าย
6. โปรแกรมที่เขียนด้วยภาษาระดับต่ำไม่สามารถใช้กับเครื่องต่างระบบกันได้	6. โปรแกรมที่เขียนด้วยภาษาระดับสูงสามารถนำไปใช้กับเครื่องคอมพิวเตอร์ต่างระบบได้

โปรแกรมแปลภาษาคอมพิวเตอร์ 

หากเขียนโปรแกรมด้วยภาษาระดับสูง จำเป็นต้องมีตัวกลางในการแปลภาษาระดับสูงที่เขียนให้เป็นภาษาเครื่อง เพื่อให้เครื่องเข้าใจชุดคำสั่งต่าง ๆ ที่เขียน เครื่องจึงจะสามารถปฏิบัติตามชุดคำสั่งต่าง ๆ ที่เขียนได้ โปรแกรมแปลภาษามีอยู่ด้วยกัน 2 ประเภท ดังนี้

1. **อินเทอร์พรีเตอร์ (Interpreter)** เป็นโปรแกรมแปลภาษาระดับสูง เป็นภาษาเครื่อง ซึ่งจะแปลคำสั่งทีละคำสั่ง เมื่อแปลเสร็จก็จะทำงานตามคำสั่งนั้นจึงจะแปลคำสั่งอื่นต่อไปจนกว่าจะจบโปรแกรม เมื่อพบข้อผิดพลาดของคำสั่งใด ๆ ก็หยุดแปลที่คำสั่งนั้นจนกว่าจะมีการแก้ไขใหม่ให้ถูกต้องเสียก่อนจึงจะแปลคำสั่งอื่น ๆ ต่อไป การแปลแต่ละครั้งไม่มีการจำหรือบันทึกไว้ เมื่อต้องการจะใช้งานอีกก็จะต้องทำการแปลใหม่ทุกครั้ง

2. **คอมไพเลอร์ (Compiler)** เป็นโปรแกรมแปลภาษาซึ่งทำการแปลภาษาระดับสูงเป็นภาษาเครื่อง โดยจะทำการแปลคำสั่งทั้งหมดให้เสร็จก่อนทั้งโปรแกรม แล้วจึงรายงานข้อผิดพลาดของคำสั่งให้แก้ไข แต่ในขณะที่แปลจะยังไม่มีการทำงานของโปรแกรมที่เขียน และเมื่อนำโปรแกรมที่ผ่านการแปลเรียบร้อยแล้วไปใช้งานจะทำให้เครื่องทำงานได้เร็วขึ้น การแปลแบบนี้มีการบันทึกโปรแกรมที่ผ่านการแปลไว้แล้ว เมื่อต้องการนำโปรแกรมไปใช้งานเมื่อใดก็สามารถใช้งานได้โดยทันทีโดยไม่ต้องแปลอีก



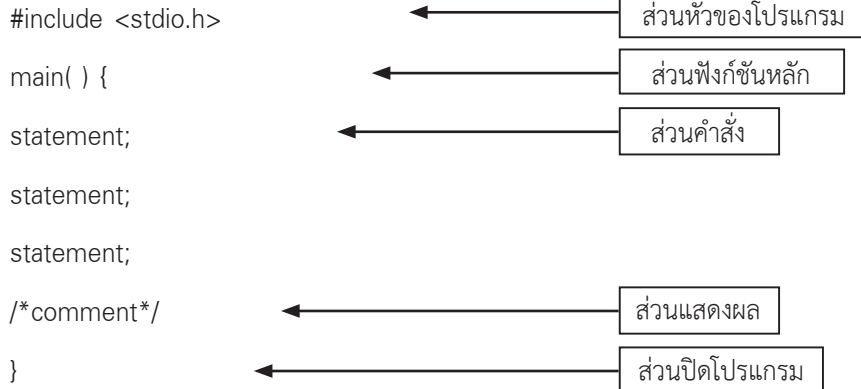
กระบวนการเขียนโปรแกรมภาษาคอมพิวเตอร์

องค์ประกอบของโปรแกรมภาษาคอมพิวเตอร์

ในการเขียนโปรแกรมแต่ละภาษาจะมีการใช้คำสั่งที่ต่างกัน แต่ส่วนใหญ่จะมีองค์ประกอบที่คล้ายกัน โดยจะมีองค์ประกอบที่สำคัญ 2 ส่วนคือ ไฟล์ส่วนหัวโปรแกรมและไฟล์โปรแกรม ไฟล์ส่วนหัวโปรแกรมเป็นไฟล์ที่ใช้เก็บไลบรารีเพื่อใช้รวม (include) ในการคอมไพล์โปรแกรมซึ่งจะมีส่วนขยายเป็น *.h มีชื่อเรียกว่า **Compiler Directive** ไฟล์โปรแกรมจะเริ่มต้นด้วยฟังก์ชัน main() และตามด้วยเครื่องหมายปีกกาเปิด ({) เพื่อเริ่มต้นเขียนโปรแกรม การเขียนโปรแกรมจะต้องเขียนด้วยอักษรภาษาอังกฤษตัวพิมพ์เล็กเสมอ และเมื่อจบประโยคคำสั่งจะใช้เครื่องหมายเซมิโคลอน (;) ในการคั่นแต่ละคำสั่ง ภายในโปรแกรมจะประกอบด้วยฟังก์ชันและส่วนของคำอธิบาย เมื่อเขียนคำสั่งเสร็จจะปิดท้ายโปรแกรมด้วยเครื่องหมายปีกกาปิดเสมอ (}) โดยองค์ประกอบของโปรแกรมภาษาคอมพิวเตอร์มีดังนี้

1. ส่วนหัวของโปรแกรม (Header) เป็นส่วนที่จะต้องเขียนไว้เพื่อให้ใช้งานฟังก์ชันต่าง ๆ ในกรณีที่ต้องการทราบว่าฟังก์ชันใดถูกนิยามไว้ที่ใด โดยให้ทำแถบสีที่ฟังก์ชันนั้น ๆ และกดปุ่ม Ctrl+f1
2. ส่วนฟังก์ชันหลัก (Function) เป็นฟังก์ชันหลักของโปรแกรม โดยภายในวงเล็บนั้น (main()) เป็นค่าพารามิเตอร์ที่จะส่งผ่านไปทำงานยังฟังก์ชันอื่น ๆ ถ้าไม่มีการใส่ค่าแสดงว่าไม่ต้องการมีค่าพารามิเตอร์
3. ส่วนคำสั่ง (Statement) เป็นส่วนที่ใช้เขียนประโยคคำสั่ง
4. ส่วนแสดงผล (Comment) เป็นส่วนที่ใช้สำหรับแสดงข้อความเพื่ออธิบายสิ่งที่ต้องการในโปรแกรม ซึ่งจะใช้เครื่องหมาย /* และ */ ปิดหัวและปิดท้ายของข้อความที่ต้องการ
5. ส่วนปิดโปรแกรม (End) ใช้ปีกกาปิด (}) แสดงการจบการเขียนโปรแกรม

องค์ประกอบของโปรแกรมภาษาซี
โครงสร้างของภาษาซี



ตัวอย่าง การเขียนโปรแกรมภาษาซีตามโครงสร้าง

	โปรแกรม	คำอธิบาย
1.	#include<stdio.h>	ส่วนหัวของโปรแกรม
2.	void main()	ส่วนฟังก์ชันหลัก
3.	{	แสดงการเริ่มต้นเขียนโปรแกรม
4.	int a,b,area;	ส่วนคำสั่ง
5.	printf ("Enter long:");	ส่วนแสดงผล
6.	scanf ("%d",&a);	
7.	printf ("Enter high:");	
8.	scanf ("%d",&b);	
9.	area=a*b;	
10.	printf ("area%d\n",area);	
11.	scanf ("%d",area);	ส่วนปิดโปรแกรม แสดงการจบการเขียนโปรแกรม
12.	}	

องค์ประกอบของโปรแกรมภาษา Visual Basic 2010

โครงสร้างของ ภาษา Visual Basic 2010

```

Form1.vb Form1 [Design]
Form1
(Declarations)
End Sub
Private Sub TextBox3_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles TextBox3.TextChanged
End Sub
End Sub
1 Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
2 If TextBox7.Text = "" Then
    MessageBox.Show("กรุณากรอกชื่อและนามสกุล", "name", MessageBoxButtons.OK, MessageBoxIcon.Information)
End If
3 End If
    TextBox3.Text = Val(TextBox1.Text) + Val(TextBox2.Text) + Val(TextBox5.Text) 'คำนวณหาคะแนนรวม
    Dim score As Integer 'ประกาศตัวแปรชื่อ Score เป็น Integer (เลขจำนวนเต็ม)
    score = Val(TextBox3.Text) 'ใส่ค่าให้ score เท่ากับคะแนนรวม
    4 Select Case score 'คำสั่ง Select case
        Case Is >= 80 'ถ้า Score มากกว่าหรือเท่ากับ 80
            TextBox4.Text = "สอบ 4" 'ให้แสดงค่า "สอบ 4" ที่ TextBox4.text
        Case Is >= 70 'ถ้า Score มากกว่าหรือเท่ากับ 70
            TextBox4.Text = "สอบ 3" 'ให้แสดงค่า "สอบ 3" ที่ TextBox4.text
        Case Is >= 60 'ถ้า Score มากกว่าหรือเท่ากับ 60
            TextBox4.Text = "สอบ 2" 'ให้แสดงค่า "สอบ 2" ที่ TextBox4.text
        Case Is >= 50 'ถ้า Score มากกว่าหรือเท่ากับ 50
            TextBox4.Text = "สอบ 1" 'ให้แสดงค่า "สอบ 1" ที่ TextBox4.text
        Case Else 'ถ้าไม่ตรงตามผลไปก็ถือว่า
            TextBox4.Text = "สอบ 0" 'ให้แสดงค่า "สอบ 0" ที่ TextBox4.text
    End Select
    5 TextBox6.Text = TextBox7.Text + vbCrLf + TextBox4.Text
End Sub
    
```

จากตัวอย่างการเขียนโปรแกรมประกอบด้วย

- ส่วนที่ 1 เป็นส่วนหัวของโปรแกรม เป็นส่วนที่เปิดการทำงาน ซึ่งโปรแกรมจะมีมาให้แล้ว
- ส่วนที่ 2 จะประกาศตัวแปร (Dim score As Integer)
- ส่วนที่ 3 เป็นส่วนของการเขียนโปรแกรม ซึ่งในตัวอย่างใช้คำสั่ง Select Case
- ส่วนที่ 4 เป็นการกำหนดให้แสดงผล ในตัวอย่างให้แสดงผลลัพธ์ที่ TextBox6.Text
- ส่วนที่ 5 เป็นการจบโปรแกรมในส่วนนี้

ดังนั้นในการเขียนโปรแกรมใด ๆ ก็ตามจะต้องมีความรู้พื้นฐานเกี่ยวกับองค์ประกอบของโปรแกรมนั้น ๆ ก่อน เมื่อมีความรู้พื้นฐานแล้วก็จะสามารถเรียนรู้การเขียนโปรแกรมภาษาอื่น ๆ ได้ง่ายขึ้น

สรุปสาระการเรียนรู้หน่วยที่ 2

ภาษาคอมพิวเตอร์ หมายถึง ชุดคำสั่งต่าง ๆ ที่จะสั่งให้เครื่องคอมพิวเตอร์ทำงาน ซึ่งการเขียนคำสั่งเหล่านั้นเรียกว่า “โปรแกรม” ถ้ามีหลายคำสั่งจะเรียกว่า “ชุดคำสั่ง” โดยลักษณะที่คล้ายคลึงกันของภาษาคอมพิวเตอร์ทุกภาษาคือ คำสั่งรับข้อมูลและแสดงผล คำสั่งคำนวณ คำสั่งที่มีการเลือกทิศทาง และคำสั่งเข้าออกของข้อมูล โดยภาษาคอมพิวเตอร์นี้สามารถแบ่งได้เป็น 2 ระดับ คือ ภาษาระดับต่ำและภาษาระดับสูง ซึ่งหากผู้ใช้เขียนโปรแกรมด้วยภาษาระดับสูง จำเป็นจะต้องมีตัวกลางในการแปลภาษาให้เป็นภาษาเครื่องเสียก่อน โดยจะมีโปรแกรมอินเทอร์พรีเตอร์ ซึ่งจะแปลคำสั่งทีละคำสั่ง และโปรแกรมคอมไพเลอร์ที่จะแปลคำสั่งทั้งหมดให้เสร็จก่อน แล้วจึงรายงานข้อผิดพลาดทีหลัง ส่วนองค์ประกอบของโปรแกรมนั้นจะประกอบไปด้วย ส่วนหัว ส่วนฟังก์ชันหลัก ส่วนคำสั่ง ส่วนแสดงผล และส่วนปิดโปรแกรม


คำศัพท์ท้ายหน่วยที่ 2 

Header	ส่วนหัวโปรแกรม
Function	เป็นโปรแกรมย่อยที่ทำงานร่วมกับโปรแกรมหลัก
Statement	คำสั่งที่ใช้ในการเขียนโปรแกรม
Comment	คำอธิบายโปรแกรม
Computer Language	ภาษาคอมพิวเตอร์
Machine Language	ภาษาเครื่อง
Binary Code	รหัสเลขฐานสอง
Programmer	ผู้เขียนคำสั่งหรือโปรแกรมให้คอมพิวเตอร์ทำงานได้
Programming Language	ภาษาโปรแกรม
Receive Data and Display	การรับและแสดงผลข้อมูล
Calculated	คำนวณ
Direction	ทิศทาง
Low Level Language	ภาษาระดับต่ำ
High Level Language	ภาษาระดับสูง