

## ใบความรู้ที่ 1

รายวิชา ง33202 การเขียนโปรแกรมภาษาซีเบื้องต้น

ชั้นมัธยมศึกษาปีที่ 6

กลุ่มสาระการเรียนรู้การงานอาชีพและเทคโนโลยี

หน่วยการเรียนรู้ที่ 1 เรื่องแนวคิดในการเขียนโปรแกรมและการวิเคราะห์ระบบ เวลา 6 ชั่วโมง

โรงเรียนพินายวิทยา

ครูผู้สอน

นางสาวพัชรนันท์ กุลวรพิสิษฐ์

### แนวคิดในการเขียนโปรแกรม

การเขียนโปรแกรมเพื่อแก้ปัญหาได้ปัญหาหนึ่ง จนได้ผลลัพธ์ตามที่ต้องการ หากผู้เขียนโปรแกรมมีแนวคิดในการเขียนโปรแกรมที่ดี และทำการเขียนโปรแกรมตามแนวคิดที่ได้วางไว้



### ขั้นตอนการพัฒนาโปรแกรม

ในการพัฒนาโปรแกรมมีขั้นตอนหลัก 5 ขั้นตอน ซึ่งไม่ว่าจะทำการพัฒนาโปรแกรมครั้งใด จะต้องปฏิบัติตามขั้นตอนเหล่านี้

โจทย์: จงเขียนโปรแกรมรับค่าเลขจำนวนเต็ม 2 จำนวนและหาผลบวกของเลขทั้ง 2 จำนวนนั้น

#### 1. วิเคราะห์ปัญหา (Analysis)

👉 ขั้นตอนนี้ถือว่าเป็นขั้นตอนที่สำคัญที่สุด ผู้เขียนโปรแกรมต้องวิเคราะห์ปัญหาให้ออกกว่าจะต้องทำการเขียนโปรแกรมเพื่อแก้ปัญหาอะไร เพราะหากวิเคราะห์หรือมองปัญหาผิดแล้ว จะทำให้เขียนโปรแกรมได้ผลลัพธ์ออกมาผิดไปจากสิ่งที่ต้องการด้วย และวิเคราะห์ข้อมูลที่จะนำเข้ามาใช้ในโปรแกรมมีอะไรบ้าง จากโจทย์ข้างต้น สามารถแตกปัญหาได้เป็น 2 ส่วน คือ

- ต้องรับข้อมูลเลขจำนวนเต็ม 2 ตัวเข้ามาในโปรแกรม

วิเคราะห์ กำหนดให้ X เก็บเลขจำนวนเต็มที่ 1

กำหนดให้ y เก็บเลขจำนวนเต็มที่ 2

👉 เลขจำนวนเต็มที่ 1 + เลขจำนวนเต็มที่ 2 มีค่าเท่ากับเท่าไร

วิเคราะห์ กำหนดให้ sum เก็บค่าผลบวกของเลขจำนวนเต็มที่ 2 จำนวน นั้น

คือ  $sum = x + y$

## 2. วางแผนและออกแบบ (Planning & Design)

```
START
READ X
READ Y
COMPUTE SUM = X+Y
PRINT SUM
STOP
```

การวางแผน คือ การนำปัญหาที่วิเคราะห์ได้จากขั้นตอนที่ 1 มาวางแผนอย่างเป็นขั้นตอนว่าจะต้องเขียนโปรแกรมเพื่อแก้ปัญหาอย่างไร การวางแผนอย่างเป็นขั้นตอนนี้ เรียกว่า อัลกอริทึม (Algorithm) ซึ่งอัลกอริทึมแบ่งออกเป็น 2 รูปแบบ คือ








👉 ชูโดโค้ด (Pseudo code) คือ การเขียนอัลกอริทึมโดยใช้ประโยคภาษาอังกฤษที่สื่อความหมายง่ายๆ สามารถอ่านแล้วเข้าใจได้โดยทันที จากโจทย์สามารถเขียนชูโดโค้ดได้ดังนี้


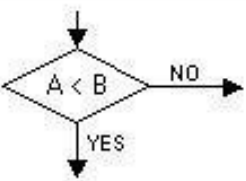


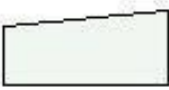
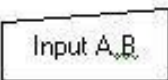



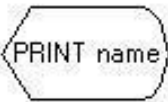

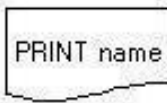


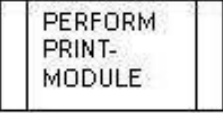



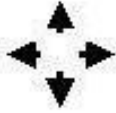
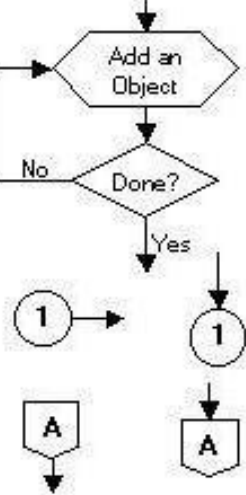
จะเห็นว่า เมื่ออ่านชูโดโค้ดแล้วสามารถเข้าใจได้ทันทีว่าขั้นตอนการเขียนโปรแกรมเป็นอย่างไร





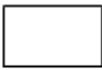
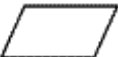




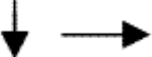
### สัญลักษณ์ โฟลวชาร์ต (Flowchart)

สัญลักษณ์ Flowchart คำว่า Flowchart มักนำไปใช้ในทางโปรแกรม กล่าวคือสำหรับคนที่ศึกษาเกี่ยวกับการเขียนโปรแกรม ก่อนอื่นจะต้องเข้าใจ Flowchart กันเสียก่อน สัญลักษณ์ต่างๆ ที่ใช้ใน Flowchart แสดงกระบวนการทำงานต่างๆ ความหมายต่างๆ ของสัญลักษณ์ใน Flowchart แต่ละอันนั้นหมายความว่าอย่างไร

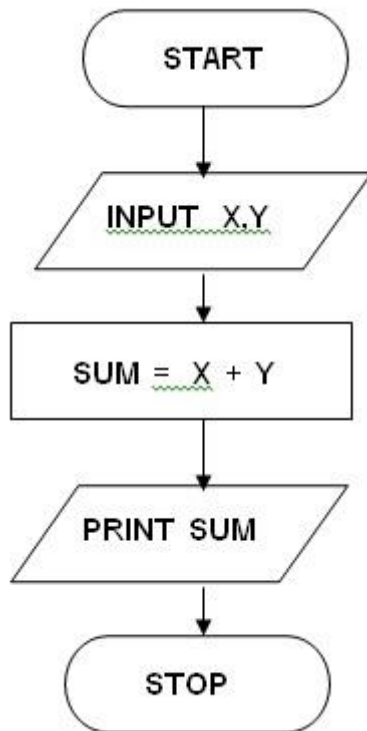


สัญลักษณ์	ชื่อ	คำอธิบาย
	สัญลักษณ์เทอร์มินัล (terminal symbol)	แสดงจุดเริ่มต้น และจุดจบของการทำงาน
	สัญลักษณ์การรับเข้าหรือแสดงผล (input / output symbol)	แสดงการรับข้อมูลเข้า หรือแสดงผลลัพธ์ โดยไม่ระบุชนิดของอุปกรณ์ที่ใช้ในการรับเข้าหรือแสดงผล
	สัญลักษณ์การนำข้อมูลเข้าด้วยมือ (manual input symbol)	แสดงการรับข้อมูลเข้าโดยมนุษย์ เช่น อาจใช้เป็นพิมพ์ (keyboard) หรือ เมาส์ (mouse)
	สัญลักษณ์บัตรเจาะรู (punched card symbol)	แสดงการรับข้อมูลเข้า หรือแสดงผล โดยใช้บัตรเจาะรูเป็นสื่อ
	สัญลักษณ์เทปกระดาษเจาะรู (punched tape symbol)	แสดงการรับข้อมูลเข้า หรือแสดงผล โดยใช้เทปกระดาษเจาะรูเป็นสื่อ
	สัญลักษณ์เทปแม่เหล็ก (magnetic tape symbol)	แสดงการรับข้อมูลเข้า หรือแสดงผล โดยใช้เทปแม่เหล็กเป็นสื่อ
	สัญลักษณ์จานแม่เหล็ก (magnetic disk symbol)	แสดงการรับข้อมูลเข้า หรือแสดงผล โดยใช้จานแม่เหล็กเป็นสื่อ

สัญลักษณ์	ความหมาย	ตัวอย่าง
Decision 	กำหนดเงื่อนไข ทางเลือก การเปรียบเทียบทางตรรกศาสตร์ เพื่อการตัดสินใจ	
Terminal, Interrupt 	แสดงจุดเริ่มต้นหรือสิ้นสุดของโปรแกรม	
Manual Input 	การป้อนข้อมูลผ่านทางแป้นพิมพ์	
Magnetic Tape 	การรับหรือ แสดงผลของข้อมูลทางเทปแม่เหล็ก	
Display 	แสดงผลจอทางจอภาพ	
Printer 	แสดงผลทางเครื่องพิมพ์	
Online Storage 	แหล่งเก็บข้อมูล Online หน่วยความจำสำรอง	
Predefined Process 	โปรแกรมย่อย หรือ โมดูล เริ่มทำงาน หลังจากจบคำสั่งในโปรแกรมย่อยแล้ว จะกลับมาทำคำสั่งต่อไป	
Preparation 	การเตรียมงานลำดับต่อไป	
Connector 	จุดเชื่อมต่อผังงานในหน้าเดียวกัน	
Offpage Connector 	จุดเชื่อมต่อผังงานที่อยู่ต่างหน้ากัน	
Arrowheads 	หัวลูกศร แสดงทิศทางการทำงานของผังงาน	

ภาพสัญลักษณ์	ความหมาย
 Start/End Symbol	เริ่มต้น/สิ้นสุด, การเริ่มต้นหรือการลงท้าย
 Connection Symbol	จุดเชื่อมต่อ ในหน้าเดียวกัน
 Connection Symbol	จุดเชื่อมต่อคนละหน้า
 Monitor	จอภาพแสดงผล
 Processing	การประมวลผลทั่วไป ยกเว้นการอ่านข้อมูลและ การแสดงผลลัพธ์
 Input/Output Data	รับหรือแสดงข้อมูล โดยไม่ระบุชนิดอุปกรณ์
 Decision Symbol	การตัดสินใจ การเปรียบเทียบ (จะมีทิศทางออก 2 ทิศทาง คือกรณีที่ผลตรวจสอบเงื่อนไขเป็นเท็จและเป็นจริง)
 Manual input	การรับข้อมูล เข้าทางแป้นพิมพ์
 Document Output	เอกสารแสดงผล, การแสดงผลทางเครื่องพิมพ์
 Preparation	ใช้กำหนดค่าต่างๆล่วงหน้า ซึ่งเป็นการทำงาน ภายในช่วงหนึ่งที่ซ้ำๆกัน
 Flow line	เส้นแสดงลำดับกิจกรรม

🌟 โฟลวชาร์ต (Flowchart) คือการเขียนอัลกอริทึมโดยใช้สัญลักษณ์รูปภาพเป็นตัวสื่อความหมายจาก โจทย์ จงหาผลรวมของ  $X + Y$  และเก็บไว้ในตัวแปรชื่อ SUM สามารถเขียนโฟลวชาร์ตได้ดังนี้

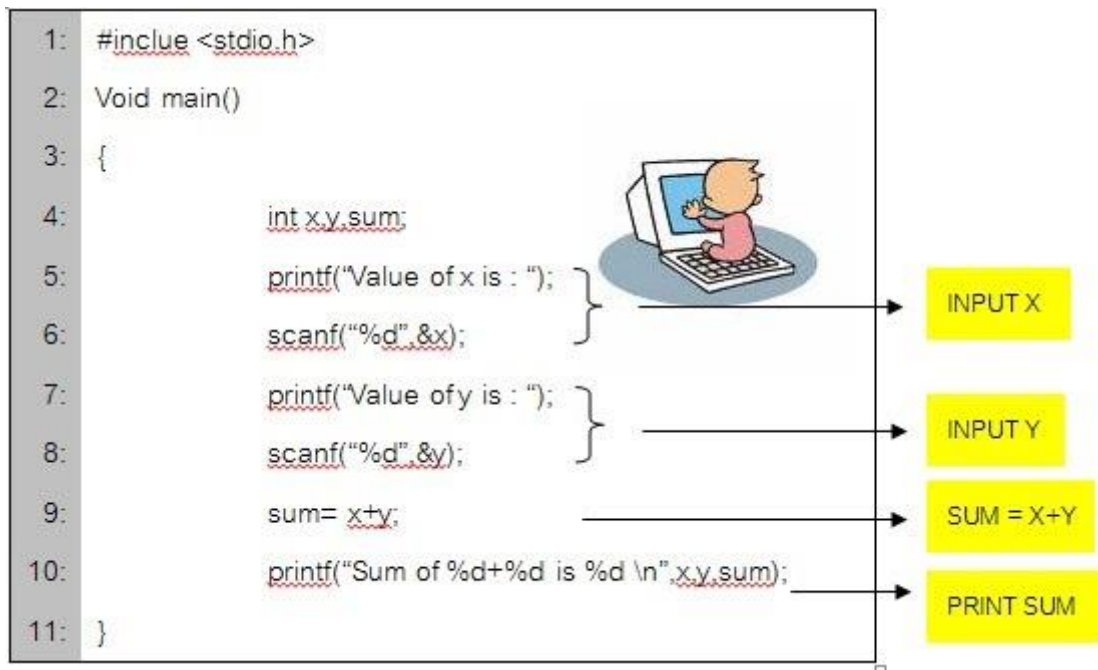


### 3. เขียนโปรแกรม (Coding)

เป็นการนำอัลกอริทึมจากขั้นตอนที่ 2 มาเขียนโปรแกรมให้ถูกต้องตามหลักไวยากรณ์ (syntax) ของภาษาซี จากโจทย์สามารถเขียนโปรแกรมได้ดังนี้  
ตัวอย่างแสดง ซอร์สโค้ด

```
1: #include <stdio.h>
2: Void main()
3: {
4:     int x,y,sum;
5:     printf("Value of x is : ");
6:     scanf("%d",&x);
7:     printf("Value of y is : ");
8:     scanf("%d",&y);
9:     sum= x+y;
10:    printf("Sum of %d+%d is %d \n",x,y,sum);
11: }
```

หากนำโปรแกรม มาพิจารณา จะพบว่าการเขียนโปรแกรมมีขั้นตอนเป็นไปตามขั้นตอนของ อัลกอริทึมที่ได้วิเคราะห์ขึ้นทุกประการ



#### 4. ทดสอบโปรแกรม

เป็นการนำผลลัพธ์จากขั้นตอนที่ 3 มาทำการรัน จากนั้นทดสอบโดยป้อนค่า x และ y เข้าไปในโปรแกรม และตรวจสอบผลลัพธ์ที่ได้ว่าถูกต้องหรือไม่ โดยทดสอบหลาย ๆ ครั้ง หากผลลัพธ์ที่ได้ถูกต้องแสดงว่าโปรแกรมที่เขียนขึ้นถูกต้องแล้ว แต่หากผลลัพธ์ถูกบ้างผิดบ้างหรือผิดทุกครั้งแสดงว่าโปรแกรมที่เขียนขึ้นผิดพลาด ผู้เขียนต้องกลับไปตรวจสอบโปรแกรม

จากโจทย์สามารถทดสอบโปรแกรมได้ดังนี้

ผลลัพธ์ของโปรแกรม

รันครั้งที่ 1

Value of x is : 5
Value of y is : 7
Sum of 5 + 7 is 12





รันครั้งที่ 2

Value of x is : 50
Value of y is : 30
Sum of 50 + 30 is 80

#### 5. จัดทำคู่มือ (Documentation)

จุดประสงค์ที่สำคัญของการทำคู่มือ คือ ช่วยให้ผู้ที่ศึกษาซอร์สโค้ดของโปรแกรม (source code) ได้ง่ายขึ้น จะเป็นประโยชน์มากสำหรับการพัฒนาโปรแกรมในอนาคต เพราะจะช่วยให้ศึกษาซอร์สโค้ดได้ง่ายและรวดเร็วขึ้น การจัดทำคู่มือไม่มีกฎเกณฑ์ระบุไว้แน่นอน แต่ผู้เขียนโปรแกรมควรจัดทำคู่มือให้มีรายละเอียดมากที่สุด

จากโจทย์ สามารถจัดทำคู่มือได้ดังนี้

ชื่อโปรแกรม	หาค่าผลบวกของเลขจำนวนเต็ม 2 จำนวน
ตัวแปรที่ใช้	x เก็บค่าจำนวนเต็มตัวที่ 1 Y เก็บค่าจำนวนเต็มตัวที่ 2 Sum เก็บค่าผลบวกของตัวเลขจำนวนเต็มทั้ง 2 จำนวน
ชนิดของข้อมูล	x,y,sum เป็นข้อมูลชนิดเลขจำนวนเต็ม (integer)
วิธีการแก้ปัญหา	ใช้สมการ $sum = x + y$