

# ในความรู้ที่ 3

## เรื่อง ขั้นตอนการพัฒนาโปรแกรม

\*\*\*\*\*



การเขียนโปรแกรมเพื่อแก้ปัญหาได้ปัญหานี้ จึงได้ผลลัพธ์ตามที่ต้องการ หากผู้เขียนโปรแกรมมีขั้นตอนในการพัฒนาโปรแกรมที่ดี และทำการเขียนโปรแกรมตามแนวคิดที่ได้วางไว้

### ขั้นตอนการพัฒนาโปรแกรม

ในการพัฒนาโปรแกรมมีขั้นตอนหลัก 5 ขั้นตอน ซึ่งไม่ว่าจะทำการพัฒนาโปรแกรมครั้งใดจะต้องปฏิบัติตามขั้นตอนเหล่านี้

โจทย์ : จงเขียนโปรแกรมรับค่าเลขจำนวนเต็ม 2 จำนวน แล้วหาค่าผลบวกของเลข 2 จำนวนนั้น

#### 1. วิเคราะห์ปัญหา (Analysis)

ขั้นตอนนี้ถือว่าเป็นขั้นตอนที่สำคัญที่สุด ผู้เขียนโปรแกรมต้องวิเคราะห์ปัญหาให้อกกว่าจะต้องทำการเขียนโปรแกรมเพื่อแก้ปัญหาอะไร เพราะหากวิเคราะห์หรือมองปัญหาผิดแล้ว จะทำให้เขียนโปรแกรมได้ผลลัพธ์ออกมาผิดไปจากสิ่งที่ต้องการด้วย และวิเคราะห์ข้อมูลที่จะนำเข้ามาใช้ในโปรแกรมมีอะไรบ้าง จากโจทย์ข้างต้น สามารถแยกปัญหาได้เป็น 2 ส่วน คือ

\* ต้องรับข้อมูลเลขจำนวนเต็ม 2 ตัวเข้ามาในโปรแกรม

วิเคราะห์ กำหนดให้  $x$  เก็บเลขจำนวนเต็มที่ 1

กำหนดให้  $y$  เก็บเลขจำนวนเต็มที่ 2

\* “เลขจำนวนเต็มที่ 1” + “เลขจำนวนเต็มที่ 2” มีค่าเท่ากันเท่าไร

วิเคราะห์ กำหนดให้  $sum$  เก็บค่าผลบวกของเลขจำนวนเต็มทั้ง 2 จำนวน นั่นคือ

$$sum = x + y$$

#### 2. วางแผนและออกแบบ (Planning & Design)

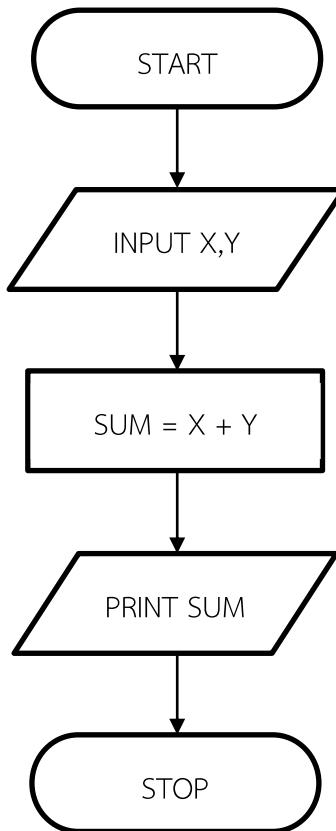
การวางแผน คือ การนำปัญหาที่วิเคราะห์ได้จากตอนที่ 1 มาวางแผนอย่างเป็นขั้นตอนว่าจะต้องเขียนโปรแกรมเพื่อแก้ปัญหาอย่างไร การวางแผนอย่างเป็นขั้นตอนนี้ เรียกว่า อัลกอริทึม (Algorithm) ซึ่ง อัลกอริทึมแบ่งออกเป็น 2 รูปแบบ

1. ชูโดโค้ด (Pseudo code) คือการเขียนอัลกอริทึมโดยใช้ภาษาอังกฤษที่สื่อความหมายง่ายๆ สามารถอ่านแล้วเข้าใจได้โดยทันที จากโจทย์สามารถเขียนชูโดโค้ดได้ดังนี้

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

จะเห็นว่า เมื่ออ่านชูโดโค้ดแล้วสามารถเข้าใจได้ทันทีว่าขั้นตอนการเขียนโปรแกรมเป็นอย่างไร

2. ผังงาน (Flowchart) คือ การเขียนอัลกอริทึม โดยใช้สัญลักษณ์รูปภาพเป็นตัวสื่อความหมาย จากโจทย์ สามารถเขียนผังงานได้ดังนี้

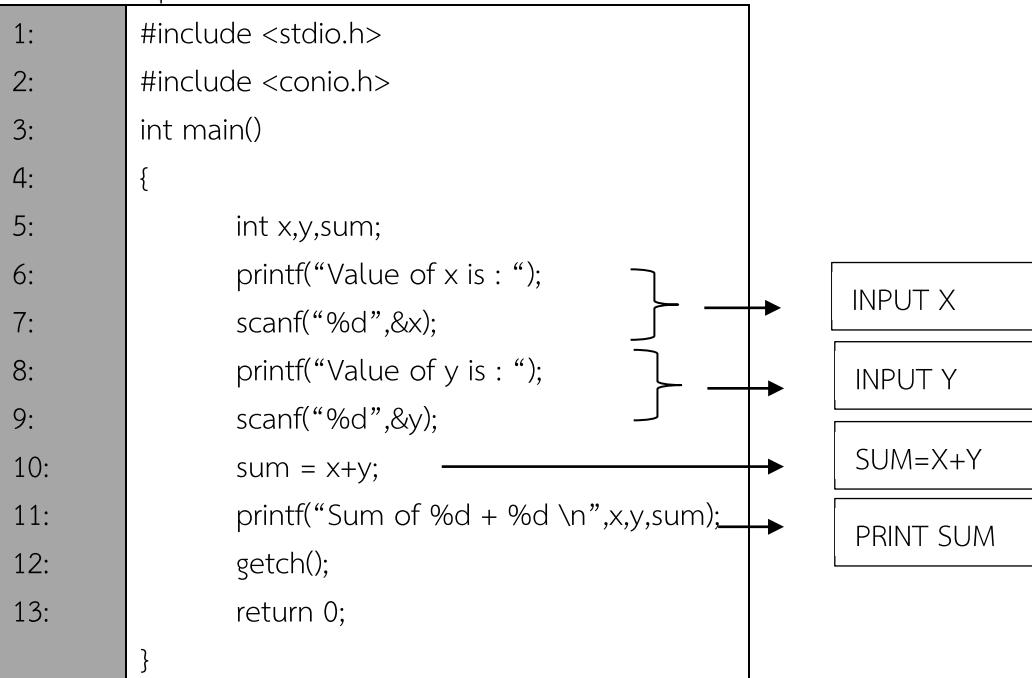


### 3. เขียนโปรแกรม (Coding)

เป็นการนำอัลกอริทึมจากขั้นตอนที่ 2 มาเขียนโปรแกรมให้ถูกต้องตามหลักไวยากรณ์ (syntax) ของภาษาซี จากโจทย์สามารถเขียนโปรแกรมได้ดังนี้

```
1: #include <stdio.h>
2: #include <conio.h>
3: int main()
4: {
5:     int x,y,sum;
6:     printf("Value of x is : ");
7:     scanf("%d",&x);
8:     printf("Value of y is : ");
9:     scanf("%d",&y);
10:    sum = x+y;
11:    printf("Sum of %d + %d \n",x,y,sum);
12:    getch();
13:    return 0;
}
```

หากนำโปรแกรมมาพิจารณา จะพบว่าการเขียนโปรแกรมมีขั้นตอนไปตามขั้นตอนของอัลกอริทึมที่ได้วิเคราะห์ขึ้นทุกประการ



#### 4. ทดสอบโปรแกรม

เป็นการนำผลลัพธ์จากขั้นตอนที่ 3 มาทำการรัน จากนั้นทดสอบโดยป้อนค่า x และ y เข้าไปในโปรแกรมและตรวจสอบผลลัพธ์ที่ได้รับถูกต้องหรือไม่ โดยทดสอบหลาย ๆ ครั้ง หากผลลัพธ์ที่ได้รับถูกต้อง แสดงว่าโปรแกรมที่เขียนขึ้นถูกต้องแล้ว แต่หากผลลัพธ์ถูกบ้าง ผิดบ้าง หรือผิดทุกครั้ง แสดงว่าโปรแกรมที่เขียนขึ้นผิดพลาด ผู้เขียนต้องกลับไปตรวจสอบโปรแกรม

จากโจทย์สามารถทดสอบโปรแกรมได้ดังนี้

#### ผลลัพธ์ของโปรแกรม

รันครั้งที่ 1	รันครั้งที่ 2
Value of x is : 5	Value of x is : 50
Value of y is : 7	Value of y is : 30
Sum of 5 + 7 is 12	Sum of 5 + 7 is 80

#### 5. จัดทำคู่มือ (Documentation)

จุดประสงค์ที่สำคัญของการทำคู่มือ คือ ช่วยให้ผู้ที่ศึกษา source code ของโปรแกรม ได้ง่ายขึ้น จะเป็นประโยชน์มากสำหรับการพัฒนาโปรแกรมในอนาคต เพราะจะช่วยให้ศึกษาซอฟต์แวร์ได้ง่ายและรวดเร็วขึ้น การจัดทำคู่มือไม่มีกฎเกณฑ์ที่แน่นอน 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