

## ใบงานการทดลองที่ 11 เรื่อง ฟังก์ชัน

### 1. จุดประสงค์ทั่วไป

- 2.1. รู้และเข้าใจหลักภาษาที่ใช้ในการพัฒนาโปรแกรมคอมพิวเตอร์

### 2. จุดประสงค์เชิงพฤติกรรม

- 2.1.29. บอกและอธิบายฟังก์ชัน  
2.1.30. ฝึกหัดและทดลองใช้ฟังก์ชัน  
2.1.31. ออกแบบแนวทางการใช้ฟังก์ชันเพื่อให้ทำงานได้มีประสิทธิภาพมากขึ้น  
2.1.32. แนะนำแนวทางการใช้ฟังก์ชันอย่างเป็นระบบ

### 3. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่องที่ติดตั้งโปรแกรม Dev-C

### 4. ทฤษฎีการทดลอง

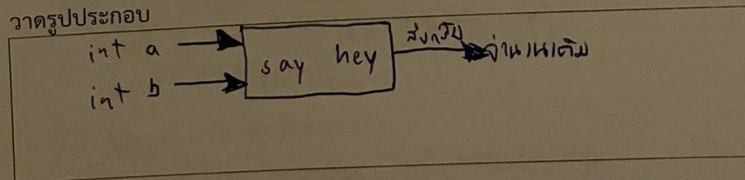
- 4.1. จงบอกและอธิบายความหมายของ "ฟังก์ชัน"

ฟังก์ชัน (function) คือ กระบวนการทำงานย่อย เป็นชุดคำสั่ง ที่ถูกนำมาใช้ซ้ำๆ กัน เพื่อลดความซ้ำซ้อนของโค้ด และทำให้โปรแกรมทำงานได้อย่างมีประสิทธิภาพมากขึ้น

- 4.2. จงระบุประโยชน์ของ "ฟังก์ชัน"

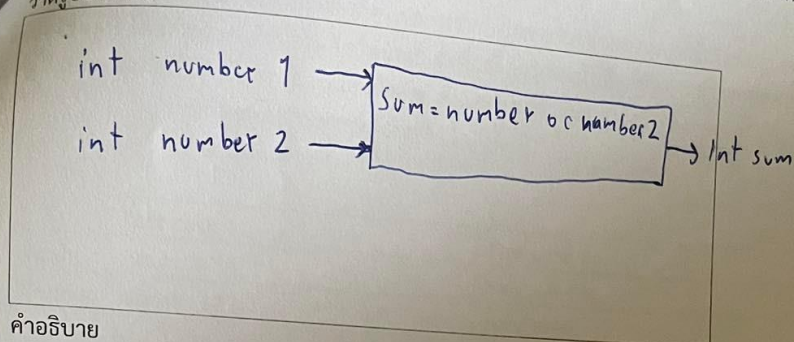
สามารถนำโค้ดที่เขียนไว้แล้วมาใช้ซ้ำได้

- 4.3. จงยกตัวอย่างการสร้างโครงสร้างของฟังก์ชัน พร้อมวาดรูปประกอบการอธิบาย  
วาดรูปประกอบ



คำอธิบาย  
ฟังก์ชันสามารถมี Input ก็ได้ แต่กลับส่งออกมาเป็น Output  
ต้องมีเพียง 1 ตัว เท่านั้น

ฟังก์ชันที่มีการส่งกลับค่า และไม่มีการส่งกลับค่าแตกต่างกันอย่างไร อธิบายพร้อมวาดรูป  
 วาดรูปประกอบ



คำอธิบาย

ฟังก์ชัน sum input ถ้า int number 1 และ number 2  
 เข้ามา เราใส่หน้าทั้งรอมค่ามาบวกกัน แล้ว Output ออกมาเป็น  
 ค่าของ sum

โค้ดโปรแกรมตัวอย่างประกอบคำอธิบาย

```

#include <stdio.h>
int sum (int);
int main() {
    int number 1=100;
    int number 2=200;
    sum (number 1, number 2);
} //end function
int sum (int number 1; int number 2) {
    int sum = number 1 + number 2;
    printf ("A", sum);
} //end function
int sum (int number 1; int number 2) {
    int sum = number 1 + number 2;
    printf ("A", sum);
} //end function
  
```



ขั้นตอนการปฏิบัติงาน

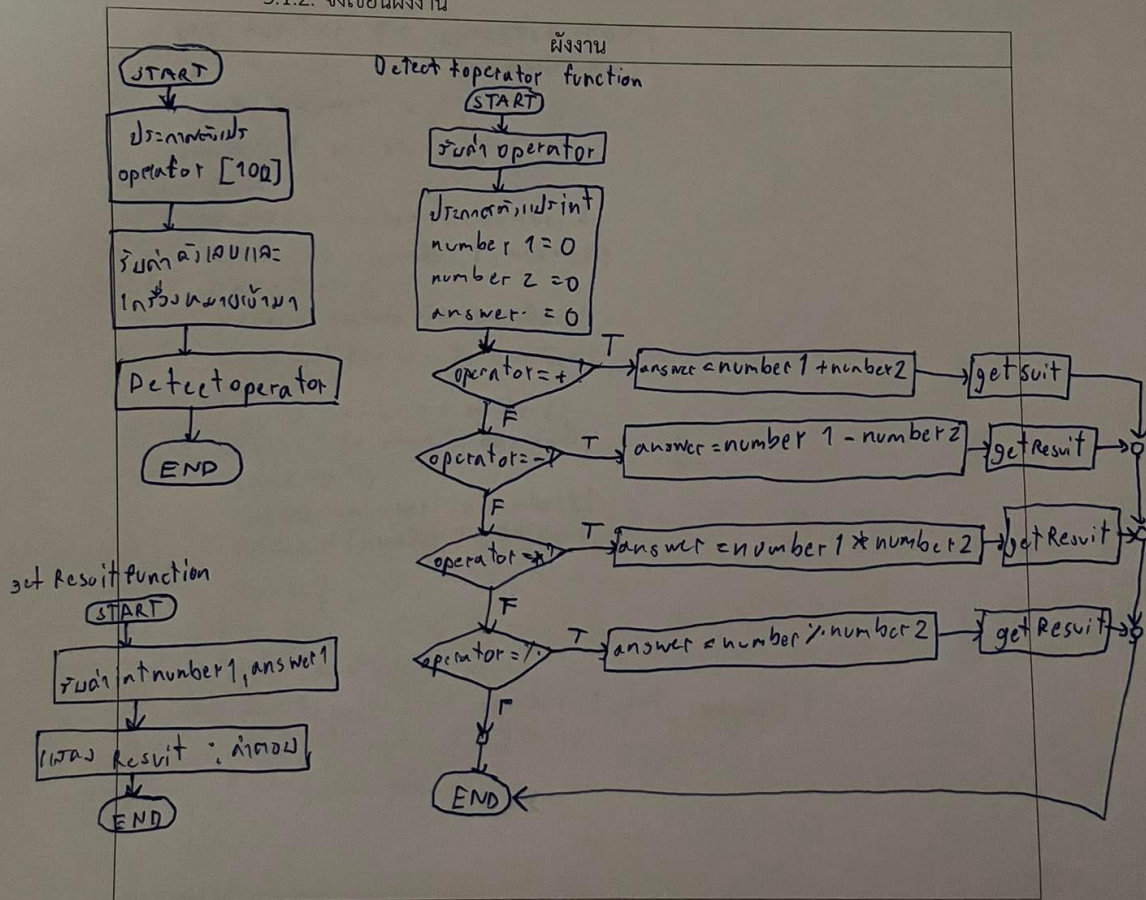
5.1. จงเขียนผังงานและโค้ดโปรแกรมเพื่อแก้ไขโจทย์ปัญหาดังต่อไปนี้

5.1.1. จงเขียนโปรแกรมเครื่องคิดเลขอย่างง่าย ด้วยการรับชุดข้อความจากผู้ใช้นำมาหาผลลัพธ์ของคำตอบ โดยกำหนดให้ภายในฟังก์ชันหลักจะต้องมีการเรียกใช้ฟังก์ชันการทำงานย่อยเพื่อรับค่าจากผู้ใช้ ประมวลผลตัวดำเนินการ และแสดงผลลัพธ์ ดังแสดงฟังก์ชันต้นแบบดังนี้

```
char DetectOperator( char [] );
int GetResult( int, int );
```

	Test case 1	Test case 2
Input	Input : 3 * 4	Input : 5 - 9
Output	Result : 12	Result : -4

5.1.2. จงเขียนผังงาน



```

#include <stdio.h>
char Detect operator (char []);
int get Result (int, int);

int main() {
    char operator[100];
    printf("Input: ");
    scanf("%d; %d", &operator[0], &operator[1], &operator[2]);
    Detect operator (operator);
    return 0;
} //end function

char Detect operator (char operator[]) {
    int number 1 = operator [0];
    int number 2 = operator [2];
    int answer = 0;
    switch (operator [1]) {
        case '+':
            answer = number 1 + number 2;
            Get Result (number 1, answer);
        case "-":
            answer = number 1 - number 2;
            Get Result (number 1 - number 2);
        case '*':
            answer = number 1 * number 2;
            Get Result (number 1, answer);
        case '/':
            answer = number 1 / number 2;
            Get Result (number 1, answer);
        break;
    }
} //end function

int Get Result (int number 1, int answer) {
    Print ("Result : %d / %d", answer);
} //end function

```



## สรุปผลการปฏิบัติงาน

วิธี Detect operator ในการใช้เครื่องหมาย  $+$ ,  $-$ ,  $*$  จากนั้นหาตำแหน่งของเครื่องหมาย และใช้  
Get Result ในการหาคำนำหน้าของค่าที่หามาไว้ก่อนจะแสดงค่าที่หามา

### 7. คำถามทางการทดลอง

7.1. จงอธิบายความหมายของฟังก์ชันต้นแบบ (Prototype function)

การประกาศฟังก์ชันในส่วนนี้ ฟังก์ชันจะถูกทูลกรัง แล้วจึงเป็นของฟังก์ชัน  
ในส่วนที่ 2 ของโปรแกรม แล้ว ฟังก์ชัน Main จึงจะเรียกใช้งาน ฟังก์ชันได้

7.2. จงอธิบายลักษณะการจัดวางโค้ดโปรแกรมที่ดีของฟังก์ชันต้นแบบ และคำอธิบายฟังก์ชัน

การจัดวางโค้ดโปรแกรม ของฟังก์ชันประกาศ ฟังก์ชันไว้ข้างบนก่อน ฟังก์ชันต้องกำหนด 9  
ชนิดให้ถูกต้อง กับการใช้งาน

7.3. จงอธิบายว่าผู้เรียนจะเลือกสร้างและฟังก์ชันเมื่อใด เพราะเหตุใด?

ถ้าฟังก์ชัน ตอนเริ่มได้โปรแกรม บวก ๆ เพราะสามารถใช้งานได้ เมื่อใดก็ได้  
และไม่ต้องเขียนโปรแกรมซ้ำ

7.4. จงอธิบายว่าหากต้องการสร้างตัวแปรเพื่อรับค่าที่ได้จากฟังก์ชัน ควรสร้างฟังก์ชันลักษณะใด  
เพราะเหตุใด?

การสร้างฟังก์ชัน Void เพราะสามารถตั้งตัวแปรชนิดใดก็ได้