

ASM Language	ใบงานที่ 3 Addition Instructions และ Subtraction Instructions
	สัปดาห์ที่ 1 ระยะเวลา ในชั้นเรียน 01418233 แอสเซมบลีและสถาปัตยกรรมคอมพิวเตอร์ ชื่อ ..... นายสุเมธ มณีจันทร์ ..... รหัสนิสิต ..... 6440204052 .....

เวลา : 2 ชั่วโมง

#### จุดประสงค์การเรียนรู้

1. ทำการเรียนรู้การเขียนโปรแกรมภาษา Assembly
2. ทำการเรียนรู้การกำหนดชนิดข้อมูลและ Symbolic

#### เนื้อหาการเรียนรู้

1. Addition และ Subtraction

#### เนื้อหา

##### 1. Addition และ Subtraction

Integer addition และ subtraction เป็นคำสั่งพื้นฐานที่สำคัญของ processor ในส่วนนี้จะทำการเรียนรู้คำสั่ง INC, DEC, ADD, SUB, และ NEG และนิสิตจะได้เรียนรู้เกี่ยวกับผลกระทบกับ flags หากมีการใช้งานคำสั่งทางคณิตศาสตร์:

INC destination	Increment <i>destination</i> ขึ้น 1.
DEC destination	Decrement <i>destination</i> ลง 1.
ADD destination, source	Add <i>source</i> กับ <i>destination</i> .
SUB destination, source	Subtract <i>source</i> จาก <i>destination</i> .
NEG destination	Negate <i>destination</i> โดยการคำนวณ 2's complement.
CF	Carry Flag: set ถ้า unsigned ผลลัพธ์คือ out of range.
OF	Overflow Flag: set ถ้า signed ผลลัพธ์คือ out of range.
SF	Sign Flag: set ถ้า sign bit ของผลลัพธ์คือ negative (or 1).
ZF	Zero Flag: set ถ้าผลลัพธ์คือ zero (all bits are zero).
PF	Parity Flag: set ถ้า low byte มีผลลัพธ์มีเลข 1 เป็นคู่.

ADD, SUB, และ NEG จะกระทบกับทุก flags ที่แสดงด้านบน และ INC กับ DEC instructions จะกระทบกับทุก flags ที่แสดงด้านบน ยกเว้น carry flag ไม่ถูกแก้ไข

ให้เขียนโปรแกรม SimpleArith.asm ด้านล่างนี้ เพื่อนำไปประกอบกับการฝึกปฏิบัติในหัวข้อต่อไป

section .data ; No data

```

section .text
    global _start
_start:

    ; ADD
    mov eax, 91ab0748h
    mov ebx, 3f54f8f2h
    add eax, ebx

    ; SUB
    mov eax, 91ab0748h
    sub eax, ebx

    ; NEG
    mov eax, 91ab0748h
    neg eax

    ; INC
    clc                ; clear carry flag to show that it is not affected
    mov eax, 7fffffffh
    inc eax

    ; DEC
    mov eax, 0
    dec eax

    ;Exit
    mov eax, 60
    mov rdi, 0
    syscall

```

## 2.1 ให้ทำการ Assemble และ Link SimpleArith.asm

## 2.2 ให้ทำการ Debugger เพื่อหาค่าที่ถูกประมวลผลจาก ./SimpleArith

เขียนค่าของ **eax** register (เป็น hexadecimal) และค่าของ **cf**, **of**, **sf**, **zf**, และ **pf** flags หลังจากประมวลผลคำสั่ง **add**, **sub**, **neg**, **inc**, **dec** ลงในตาราง:

1) EAX after ADD (hex) = 0xd100003a	CF= 0	OF= 0	SF= 1	ZF= 0	PF= 1
2) EAX after SUB (hex) = 0x52560e56	CF= 0	OF= 1	SF= 1	ZF= 0	PF= 1

