



實驗二 ARM Assembly I

1. 實驗目的

熟悉基本 ARMv7 組合語言語法使用。

在這次實驗中需要同學了解

- 如何利用條件跳躍指令完成程式迴圈的操作
- 算數與邏輯操作指令使用
- 暫存器(Register)使用與基本函式參數傳遞
- 記憶體與陣列存取

2. 實驗原理

請參考上課 Assembly 部分講義。

3. 實驗步驟

3.1. Hamming distance

計算兩個數長度為 half-word(2bytes)的漢明距離，並將結果存放至 result 變數中。

```
.data
    result: .byte 0
.text
.global main
.equ X, 0x55AA
.equ Y, 0xAA55

hamm:
    //TODO
    bx lr
main:
    movs R0, #X //This code will cause assemble error. Why? And how
to fix.
    movs R1, #Y
    ldr R2, =result
    bl hamm
L: b L
```

Note: 漢明距離主要是利用 XOR 計算兩數 bit 間差異個數，計算方式可參考下列連結。

Reference: https://en.wikipedia.org/wiki/Hamming_distance



3.2. Fibonacci serial

宣告一數值 N ($1 \leq N \leq 100$)，計算 $\text{Fib}(N)$ 並將回傳值存放至 R4 暫存器

```
.text
.global main
.equ N, 20

fib:
    //TODO
    bx lr
main:
    movs R0, #N
    bl fib
L: b L
```

Note: 當 N 數值出過範圍時 fib 回傳-1，計算方式可參考下列連結

Reference: https://it.wikipedia.org/wiki/Successione_di_Fibonacci

3.3. Bubble sort

利用組合語言完成長度為 8byte 的 8bit 泡沫排序法。

實作要求：完成 do_sort 函式，其中陣列起始記憶體位置作為輸入參數 R0，程式結束後需觀察 arr1 與 arr2 記憶體內容是否有排序完成。

```
.data
    arr1: .byte 0x19, 0x34, 0x14, 0x32, 0x52, 0x23, 0x61, 0x29
    arr2: .byte 0x18, 0x17, 0x33, 0x16, 0xFA, 0x20, 0x55, 0xAC
.text
.global main
do_sort:
    //TODO
    bx lr
main:
    ldr r0, =arr1
    bl do_sort
    ldr r0, =arr2
    bl do_sort
L: b L
```

Note: 注意記憶體存取需使用 byte alignment 指令，例如：STRB, LDRB