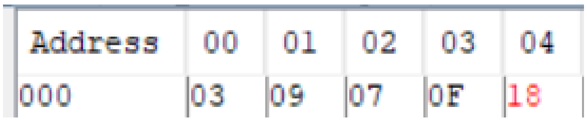
Lab 4: Requirement Description

2021/10/19

注意 ： 進階題 goto 以及 b 開頭的 branch 指令不能使用，且必須在程式中更改 PCL

- Macro & Subroutine 教學
  - 影片
    https://youtu.be/yw8xRusVn3U
  - Hackmd :
    https://hackmd.io/_HTeeEEmQqiWHKojzt4cWw?view

- **基本題 (70%)** ：
  - 題目敘述：給長方形對角頂點 A(x1,y1)、B(x2,y2)且 x1<x2,y1<y2，設計以下 2 個 macro 算出長方形面積。
    1. MOVLF literal,F
       功能 ： 將常數放到指定 register F 裡。
       Ex. MOVLF 0x06,0x25 ; [0x25] =0x06
    2. RECT addr_x1, addr_y1, addr_x2, addr_y2, F
       功能 ： 算出長方形面積，前四個參數對應頂點座標 x1,y1,x2,y2 存放的位置，F 為面積存放的 register。
       Ex. RECT 0x00,0x01,0x02,0x03, 0x04
       [0x04] = ([0x02]-[0x00]) x ([0x03]-[0x01])
  - 評分標準：
    1. 會檢查是否有建立並使用題目敘述中的兩個 macro，macro 的名稱和參數名稱需與敘述一致。
    2. 組語中最後一個指令需為 RECT。
    3. Demo 時測資為 A(0x03,0x09)、B(0x07,0x0F)，需在 data memory 0x000~0x003 顯示出 x1,y1,x2,y2 的值並且在 0x004 出示結果，如下圖一。



Address	00	01	02	03	04
000	03	09	07	0F	18

圖一

- **進階題 (30%)** ：
  - 題目敘述：寫一個名為 Fib 的 subroutine 算出費波那契數列的值，在 Fib 裡需使用迴圈並以更改 program counter(PCL)取代 goto 以及 bra 指令，將結果放入位置 0x000 中。
    費波那契數列 ：F0 = 0, F1 = 1, Fn = Fn-1 + Fn-2
  - 評分標準：
    1. 會檢查是否有名為 Fib 的 subroutine。
    2. 需使用到 rcall 指令。
    3. 需使用迴圈。

4. 不能出現 goto 以及 b 開頭的 branch 指令(ex. BRA、BZ、BN...)。
5. Demo 時請出示 F6 的值，F6 為 8。
6. 結果需放在位置 0x000。
7. 在程式中必須更改 PCL

- **加分題 (20%) ：**
  - 題目敘述：寫一個名為 Fib_recur 的 subroutine 算出費波那契數列的值，需用遞迴的方式撰寫。
  - 評分標準：
    1. 會檢查是否有名為 Fib_recur 的 subroutine。
    2. 需用遞迴撰寫。
    3. Demo 時請出示 F6 的值，F6 為 8。
    4. 結果需存在 0x000。
  - 提示：
    1. 同學可以自己建 software stack 來存變數。

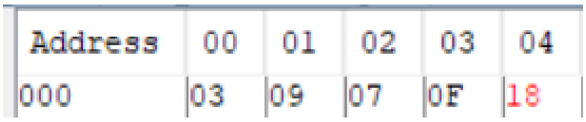
Lab 4: Requirement Description

2021/10/19

Notice: You cannot used goto and branch instructions that first character are B and you must modify PCL in your program in Advanecd.

- **Macro & Subroutine 教學**
  - **Video:** https://youtu.be/yw8xRusVn3U
  - **Hackmd:** https://hackmd.io/_HTeeEEmQqiWHKojzt4cWw?view

- **Basic (70%):**
  - Description: Give two opposite vertices of rectangle A(x1,y1)、B(x2,y2) and x1<x2,y1<y2. Design two macros below and use them to calculate the area of the rectangle.
    1. MOVLF literal,F
       Description: Put literal in register F.
       Ex. MOVLF 0x06,0x25 ; [0x25] =0x06
    2. RECT addr_x1, addr_y1, addr_x2, addr_y2, F
       Description: calculate the area of the rectangle. The first four arguments map to the address of coordinates x1,y1,x2,y2 and F stores the result.
       Ex. RECT 0x00,0x01,0x02,0x03, 0x04
       [0x04] = ([0x02]-[0x00]) x ([0x03]-[0x01])
  - Standard of grading:
    1. We will check whether you use two macros mentioned above. Macros' arguments and name must be the same as the description.
    2. The last instruction of your code must be RECT.
    3. You need to show the value of x1,y1,x2,y2 in data memory 0x000~0x003 and the result in 0x004 with A(0x03,0x09)、B(0x07,0x0f). See Figure 1 below.



Address	00	01	02	03	04
000	03	09	07	0F	18

Figure 1

- **Advanced (30%):**
  - Description: Write a subroutine called Fib to calculate Fibonacci sequence. You need to use loop in Fib, replace goto and bra instructions by changing program counter(PCL), and put result in 0x000.

Fibonacci sequence: F0 = 0, F1 = 1, Fn = Fn-1 + Fn-2

- Standard of grading:
  1. We will check whether you have a subroutine called Fib.
  2. You must use rcall instruction.
  3. You must use loop.
  4. You cannot used goto and branch instructions which first character is B(ex. BRA、BZ、BN...).
  5. You need to show F6, F6 is 8.
  6. The result must be stored in 0x000.
  7. You must modify PCL in your program.

- **Bonus (20%):**
  - Description: Write a subroutine called Fib_recur to calculate Fibonacci sequence. You need to get the answer by using recursion.
  - Standard of grading:
    1. We will check whether you have a subroutine called Fib_recur.
    2. You need to use recursion.
    3. You need to show F6, F6 is 8.
    4. The result must be stored in 0x000.
  - hint:
    1. You can create software stack by yourself to store variables.