

# 期末 Project

110 學年度第 1 學期

老師：朱守禮 老師

資訊二乙 第 32 組

蒲品憶 10927207

吳凱鈺 10927234

姜美羚 10927256

## 一、程式說明

1.Name：印出組員語組別名單

2.Id：輸入學號，計算學號總和，印出組員學號及學號總和。

3.JuliaSet: 使用暫存器存放變數，利用迴圈不斷透過各個指令來計算 pixel 存至陣列中，繪製一個個碎形圖案，因迴圈不斷變化使圖像也不斷變化，形成動畫。

4.Main：結合前三個函數的功能，除了各函數原本的功能外，最後印出組別、全組員之學號和姓名，以及總和。

## 二、設計重點說明

結合 minterm project，並改寫 juliaSet.c 為 juliaSet.s，最後用 main.c 結合在一起。

1. 定址模式：

(1) ldr r0, [r0], #2	@id.s load r0 with word at the @address in r0 then $r0 = r0 + 2$
(2) ldr r2, [r0], r3	@id.s load r2 with word at the @address in r0 then $r2 = r0 + r3$
(3) ldr r0, [r0]	@id.s load r0 with word at the @address in r0
(4) str r0, [r1, #0]	@id.s store r0 with word at the @address in $r1 + 0$

## 2. Condition execution

(1) mov <b>hs</b> r3, sp	@name.s if (hs) r3 = sp
(2) ldreq r0, =msg5	@id.s if (eq) r0 = address of msg5
(3) add <b>vc</b> r1, r1, r2	@id.s if (vc) r1 = r1 + r2
(4) add <b>pl</b> r0, r0, r1	@id.s if (pl) r0 = r0 + r1
(5) addvs r0, r1, r2, lsl #0	@drawJuliaSet.s if (vs) r0 = r1 + r2
(6) subne r0, r1, r2, lsr r1	@drawJuliaSet.s
	@if (ne) r0 = r1 - r2>>r1
(7) suble r0, r1, #0	@drawJuliaSet.s if (le) r0 = r1 - 0

## 3. Operand2

(1) add <b>vc</b> r1, r1, r2	@id.s if (vc) r1 = r1 + r2
(2) add r0, r1, r2, lsl #0	@name.s r0 = r1 + r2 右旋 0
(3) sub r0, r1, r2, lsr r1	@name.s r0 = r1 - r2 右移 r1
(4) sub r0, r1, #0	@name.s r0 = r1 - 0
(5) addvs r0, r1, r2, lsl #0	@drawJuliaSet.s if (vs) r0 = r1 + r2
(6) subne r0, r1, r2, lsr r1	@drawJuliaSet.s
	@if (ne) r0 = r1 - r2>>r1
(7) suble r0, r1, #0	@drawJuliaSet.s if (le) r0 = r1 - 0

## 4.

```

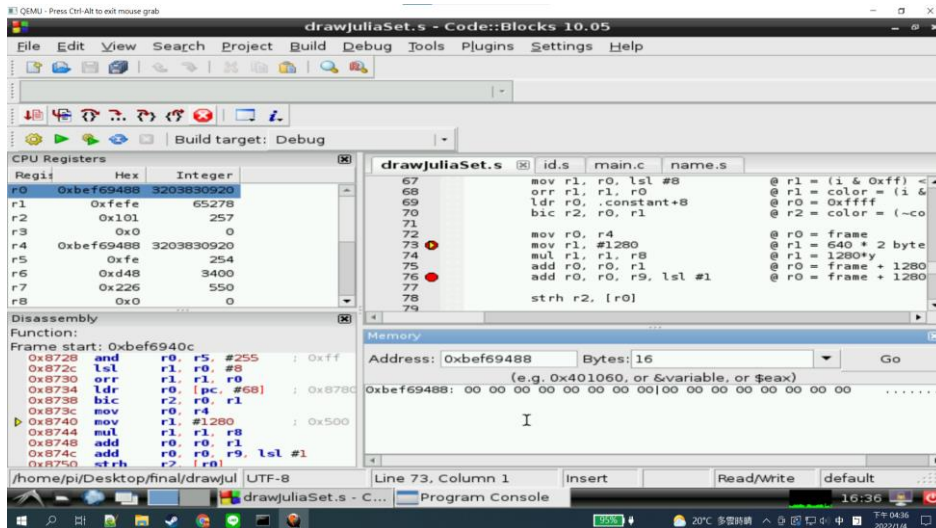
top/mmai/drawJuliaSet.s  Desktop/mmai/id.s  Desktop/mmai/name.s
.text
.global drawJuliaSet

drawJuliaSet:
    stmfd sp!, {r4-r11,lr}      @ backup r4-r11 & return address
    mov r11, r0                 @ r11 = cx
    mov r10, r1                 @ r10 = cy
    mov r4, r2                  @ r4 = frame
    mov r9, #0                  @ x=0
    add r14, r0, r15             @ !!!!!!! I
loopx: cmp r9, #640              @ if ( x < width )
    bge finishx

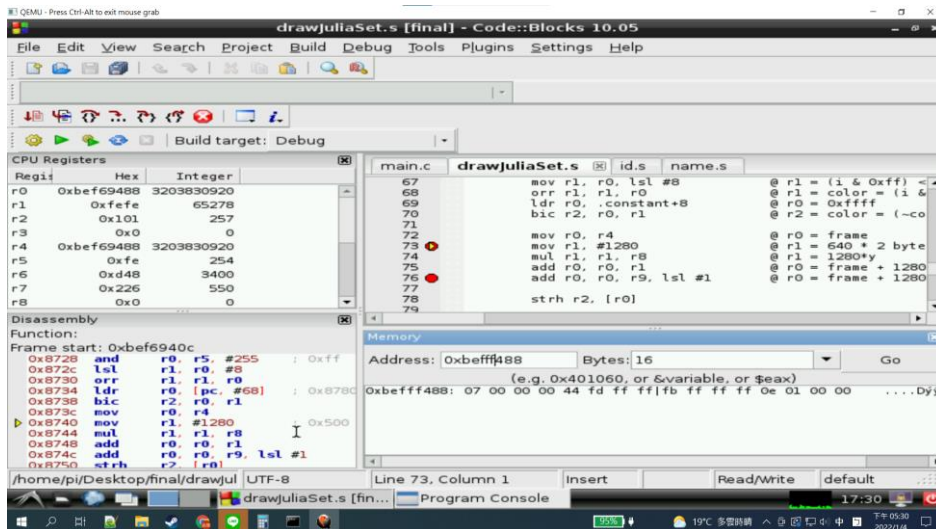
```

## 三、結果

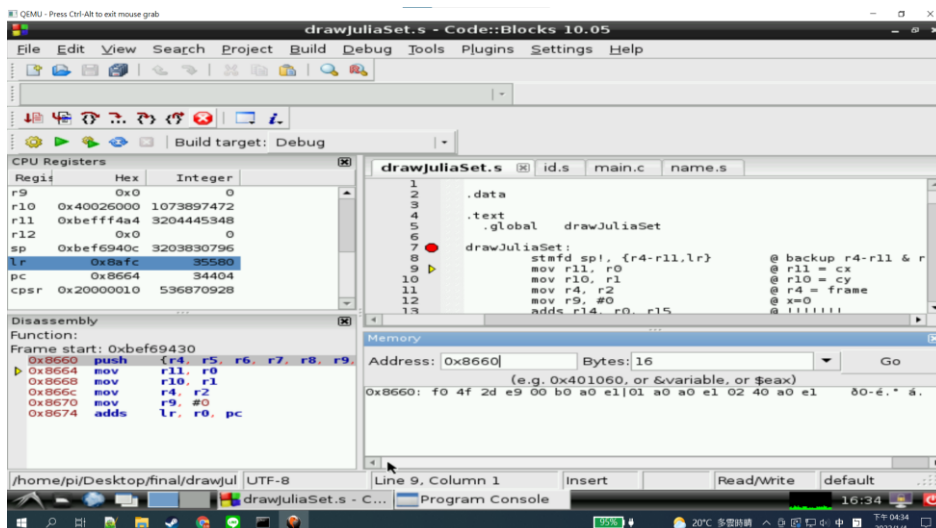
drawJuliaSet.s : frame 陣列起始位址 0xbef69488 (初始陣列無存放資料)



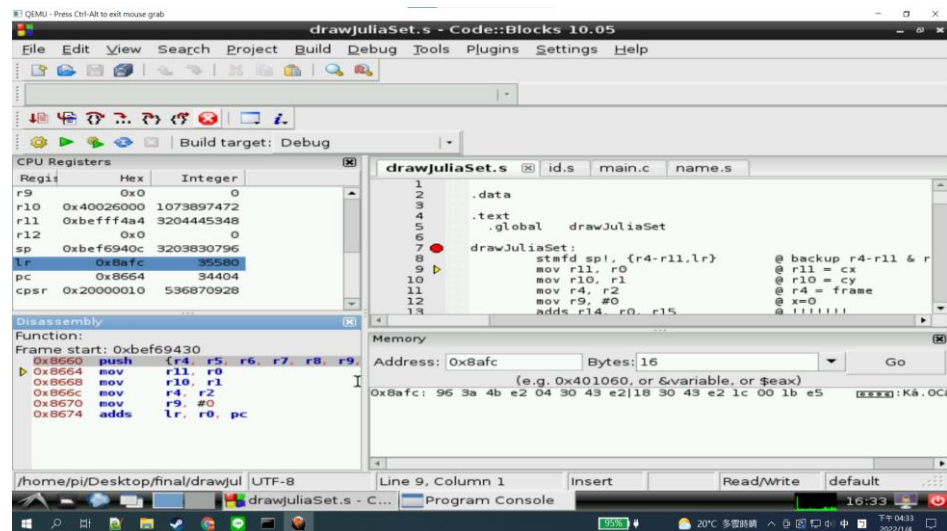
drawJuliaSet.s : frame 陣列結束位址 0xbef69488 (初始陣列無存放資料)



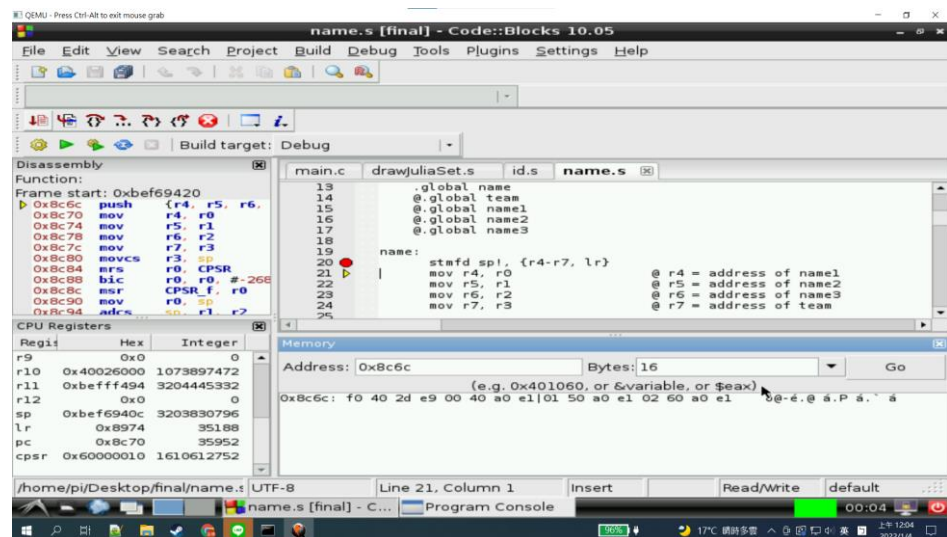
drawJuliaSet.s : drawJuliaSet 函數所在位址 0x8660



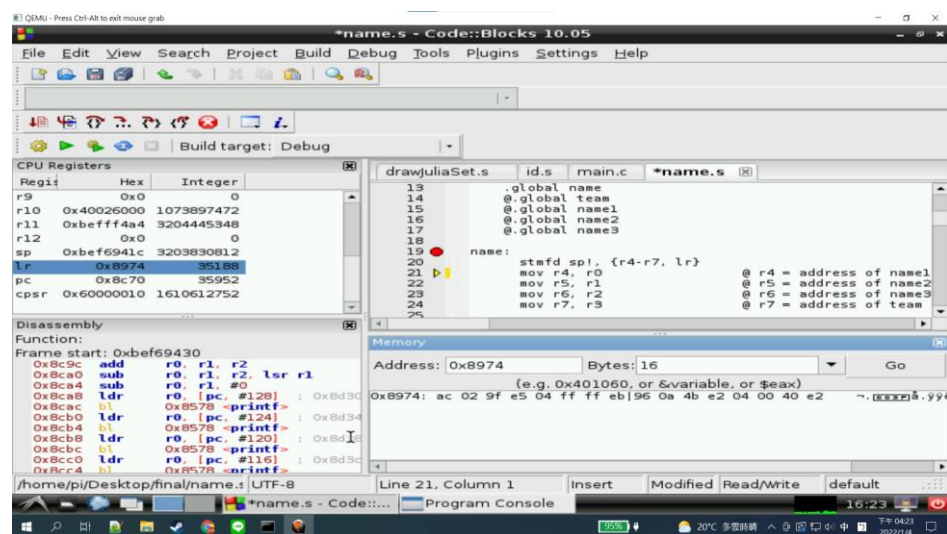
drawJuliaSet.s : drawJuliaSet 函數返回位址 0x8660



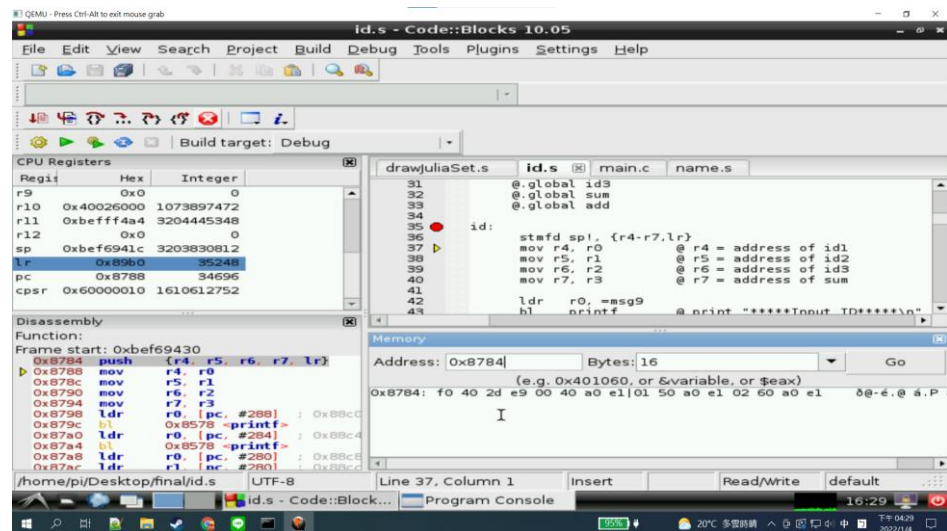
name.s : name 函数所在位址 0x8c6c



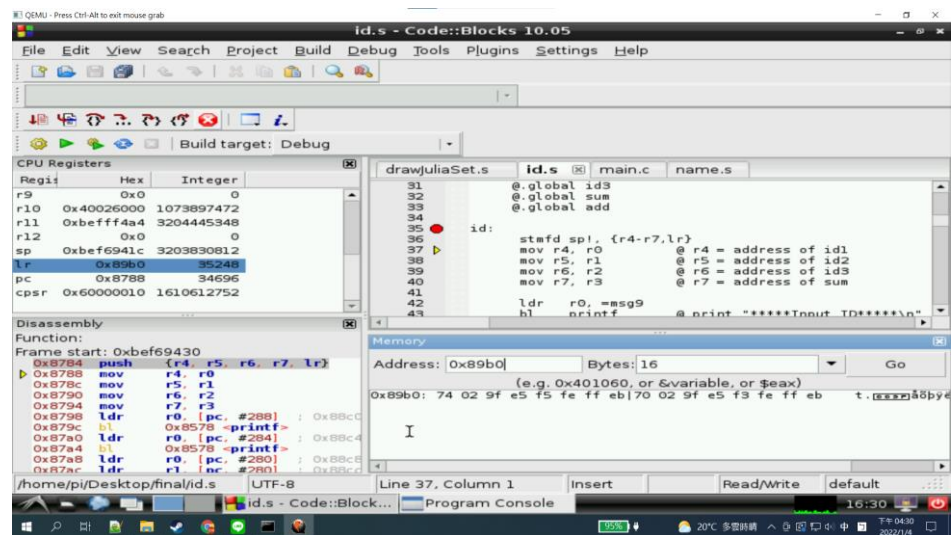
name.s : name 函数返回位址 0x8974



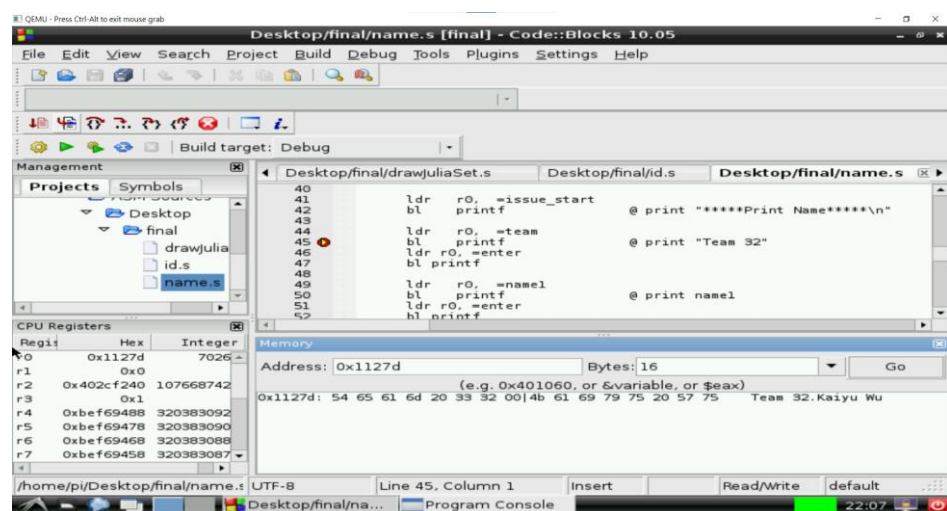
id.s : id 函数所在位址 0x8784



id.s : id 函数返回位址 0x89b0

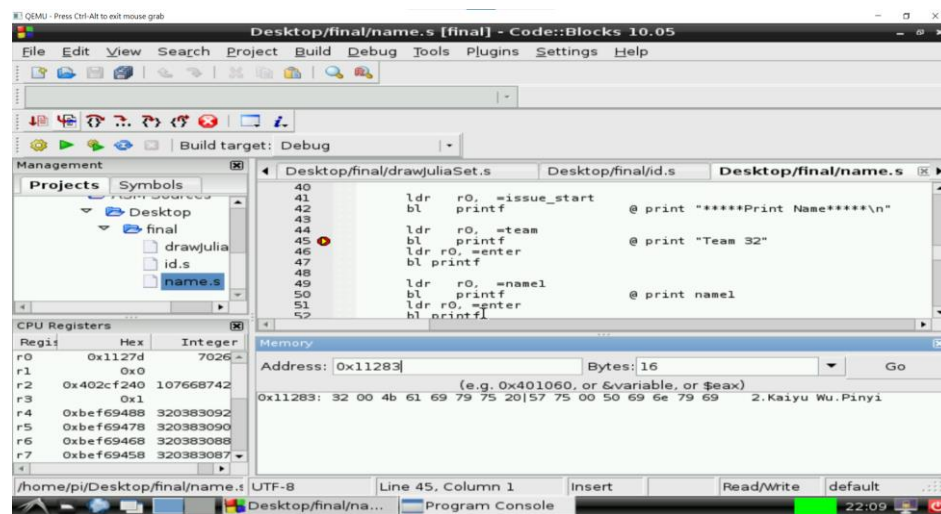


name.s : Team 32 起始位址 0x1127d

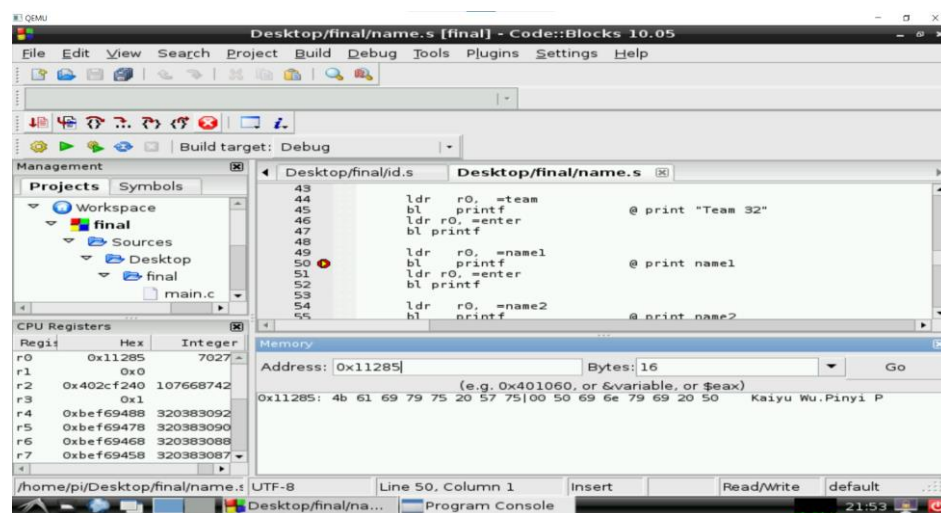




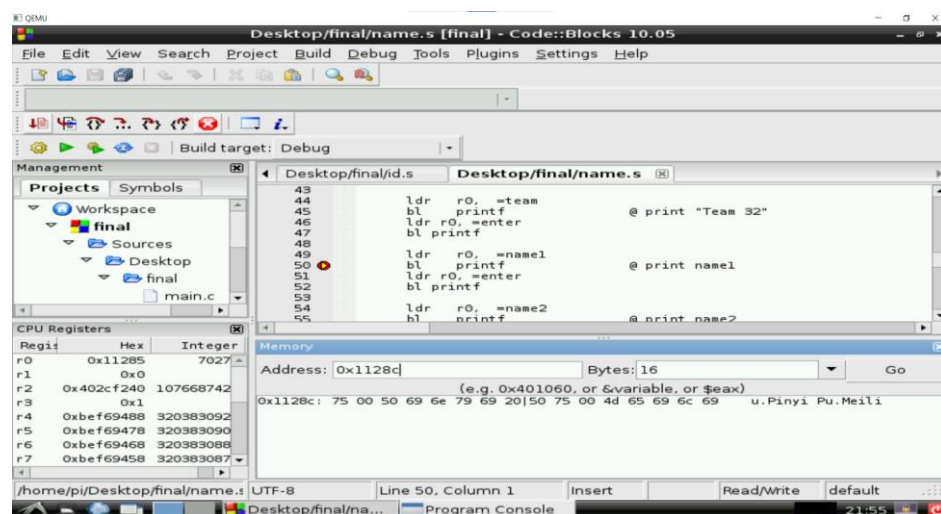
name.s : Team 32 結束位址 0x11283



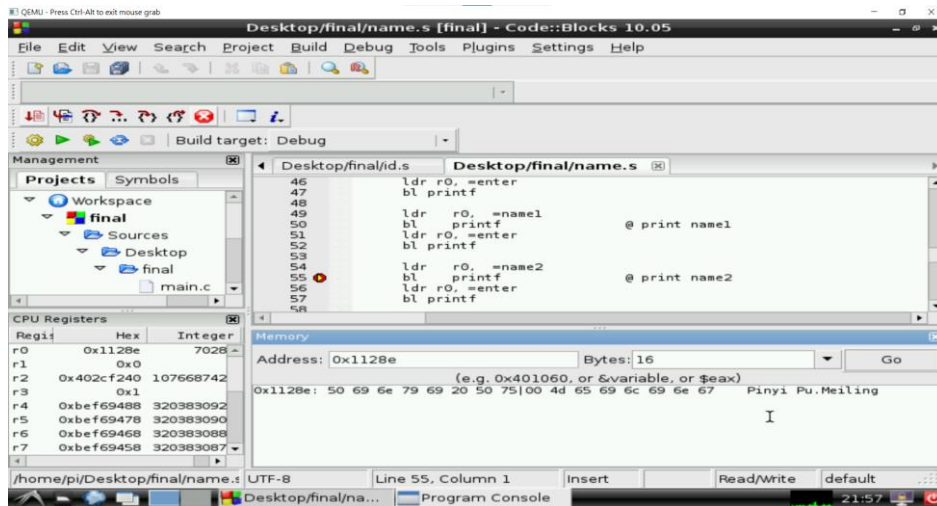
name.s : name1 起始位址 0x11285



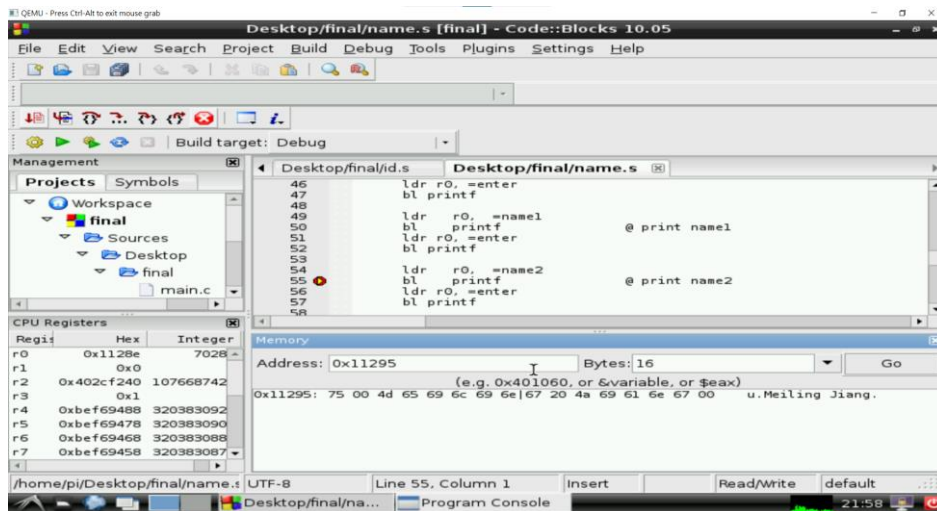
name.s : name1 結束位址 0x1128c



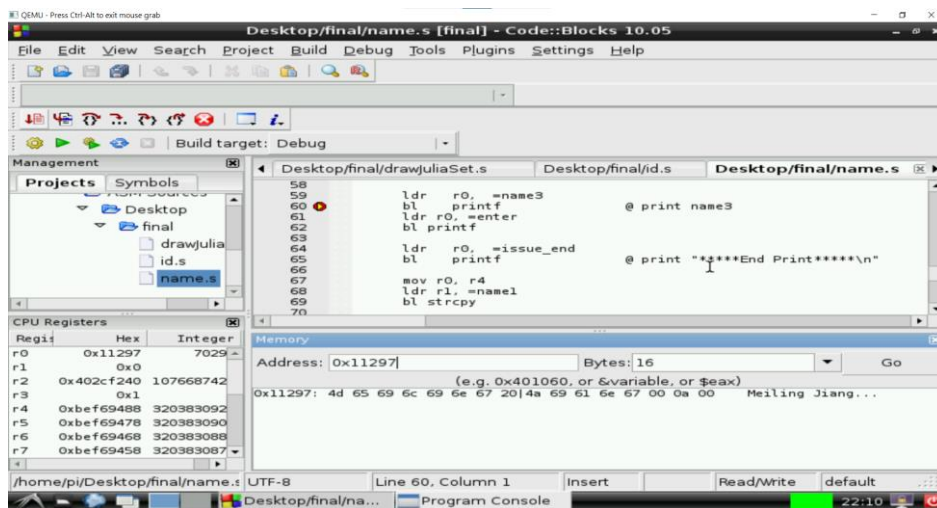
name.s : name2 起始位址 0x1128e



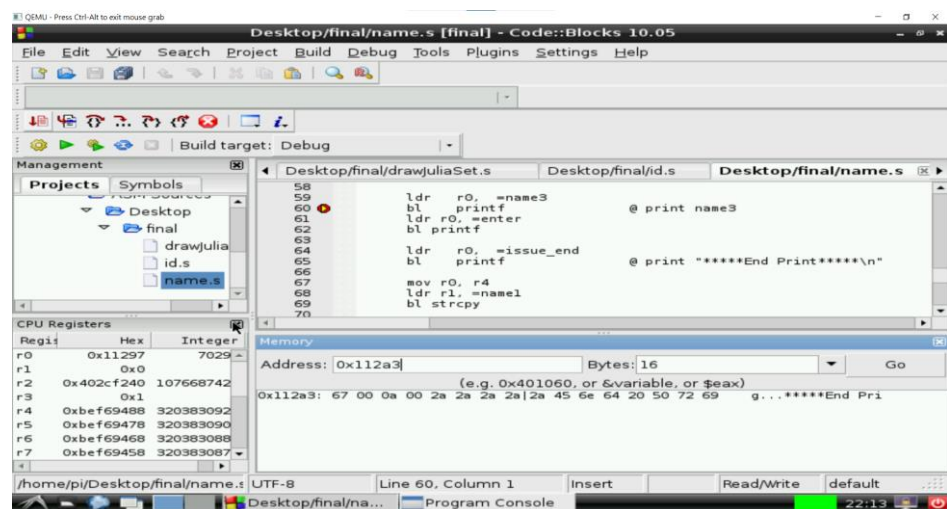
name.s : name2 結束位址 0x1195



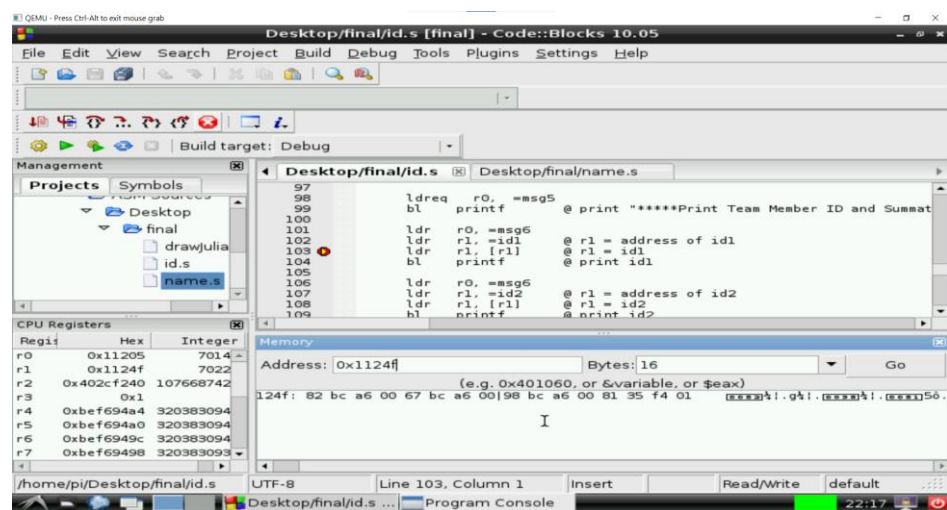
name.s : name3 起始位址 0x11297



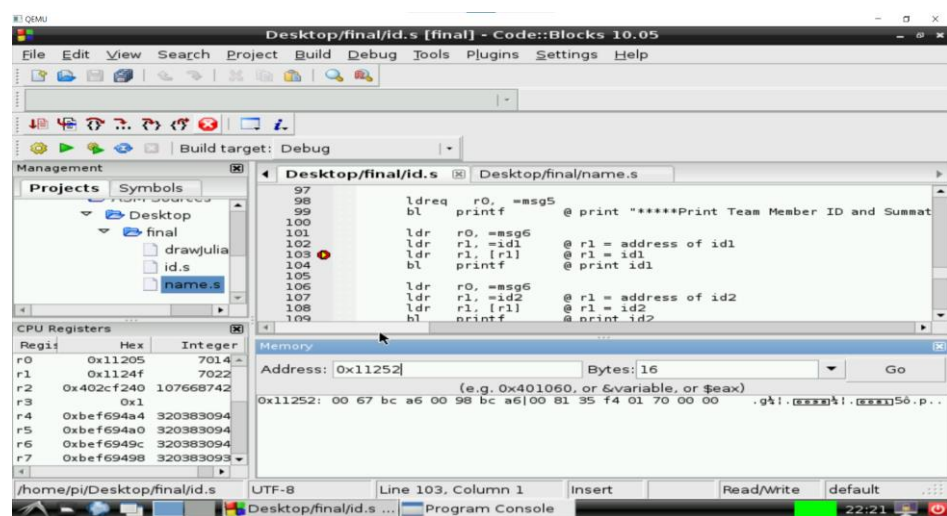
name.s : name3 結束位址 0x112a3



id.s : id1 起始位址 0x1124f

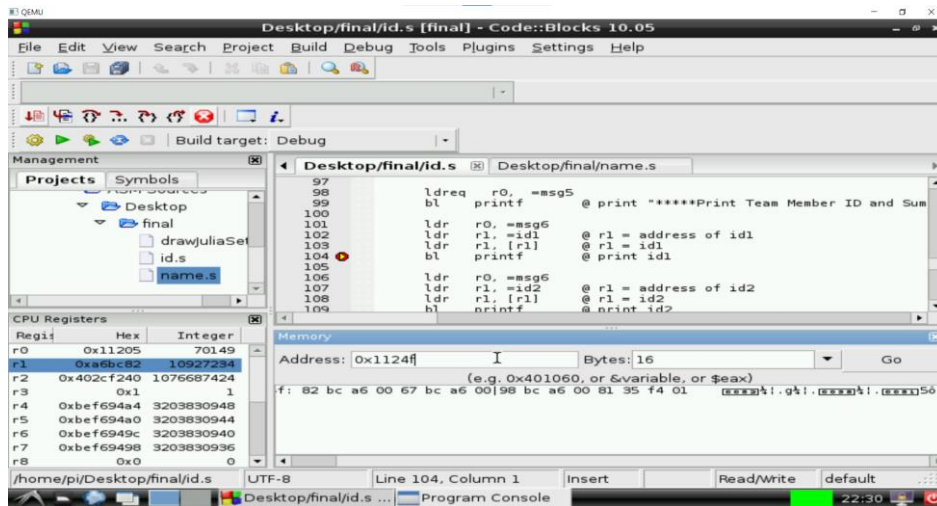


id.s : id1 結束位址 0x11252

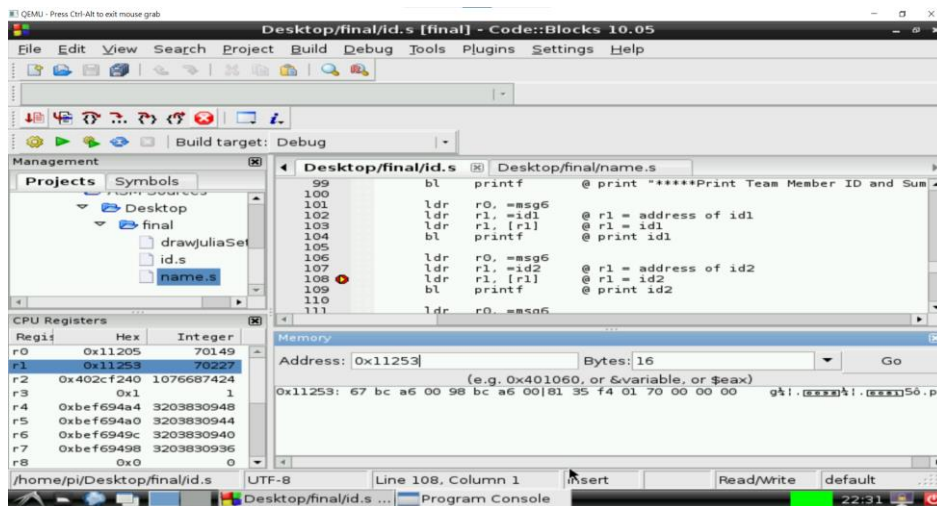


id.s : id1 驗證位址 0x1124f

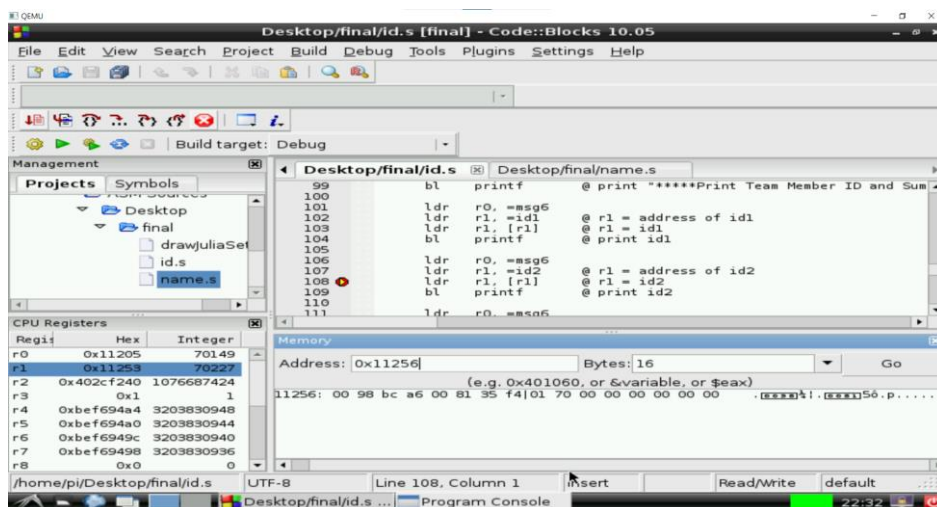




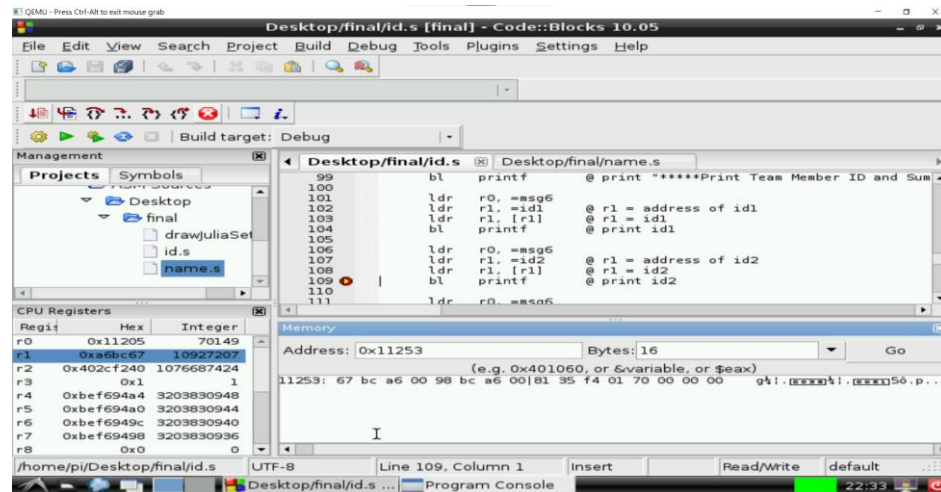
id.s : id2 起始位址 0x11253



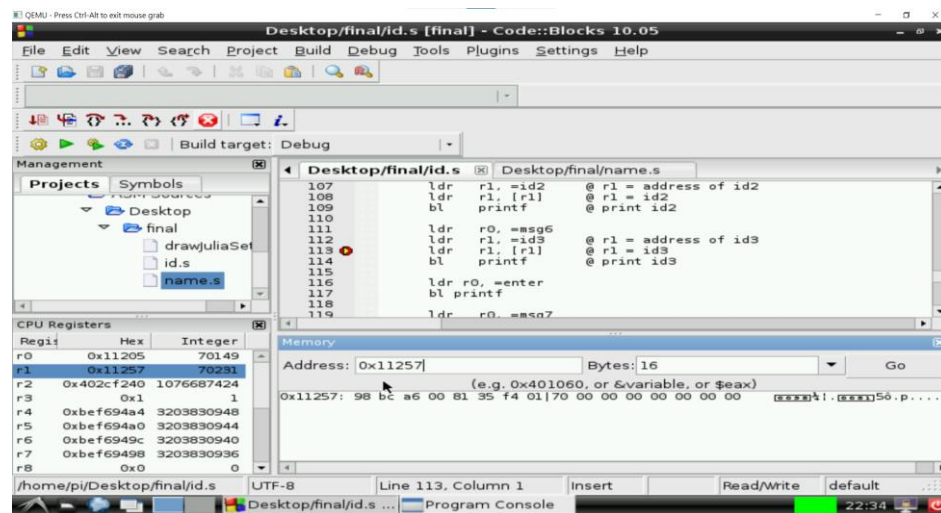
id.s : id2 結束位址 0x11256



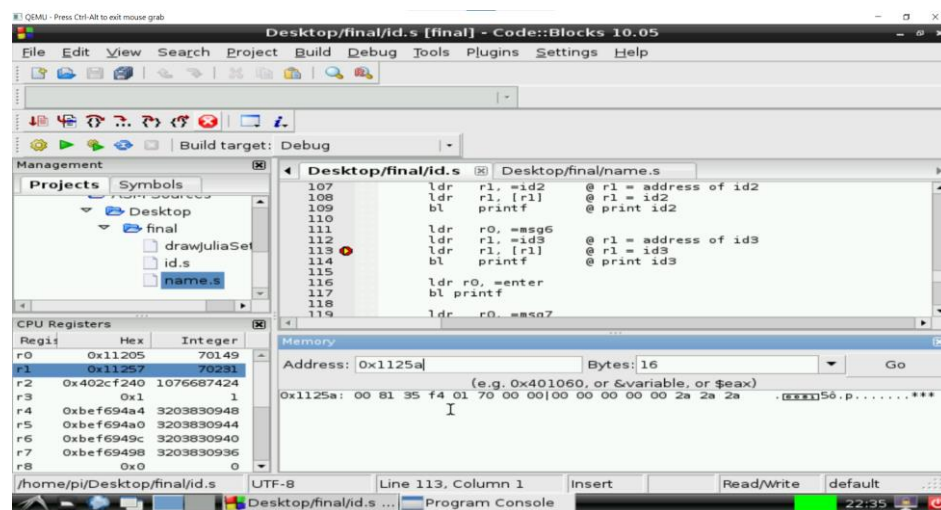
id.s : id2 驗證位址 0x11253



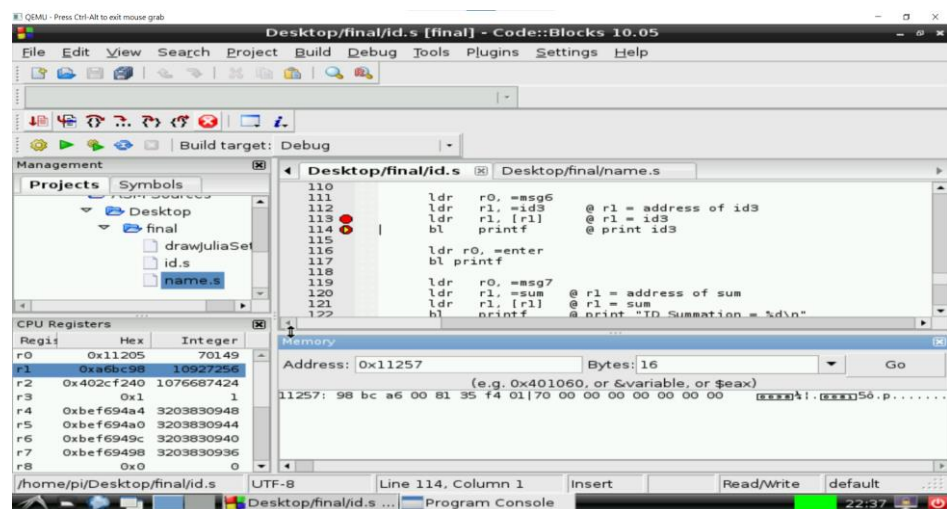
id.s : id3 起始位址 0x11257



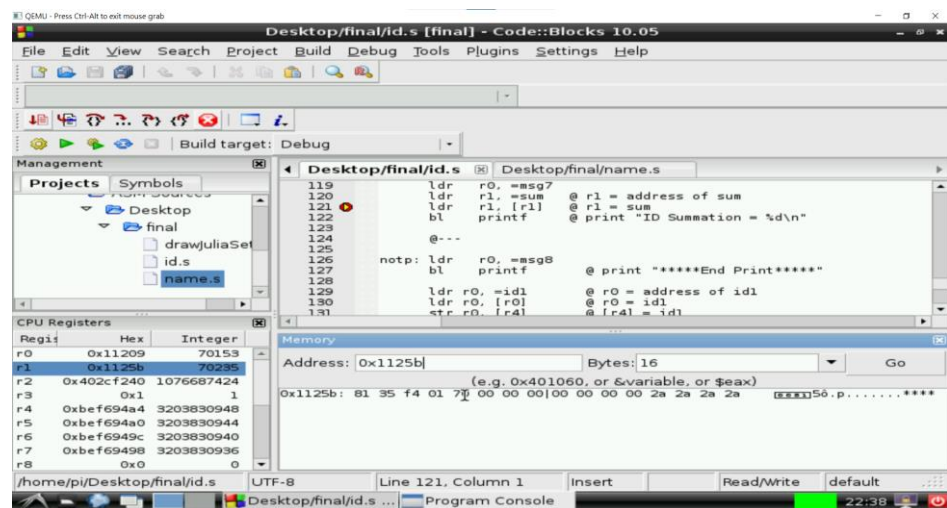
id.s : id3 結束位址 0x1125a



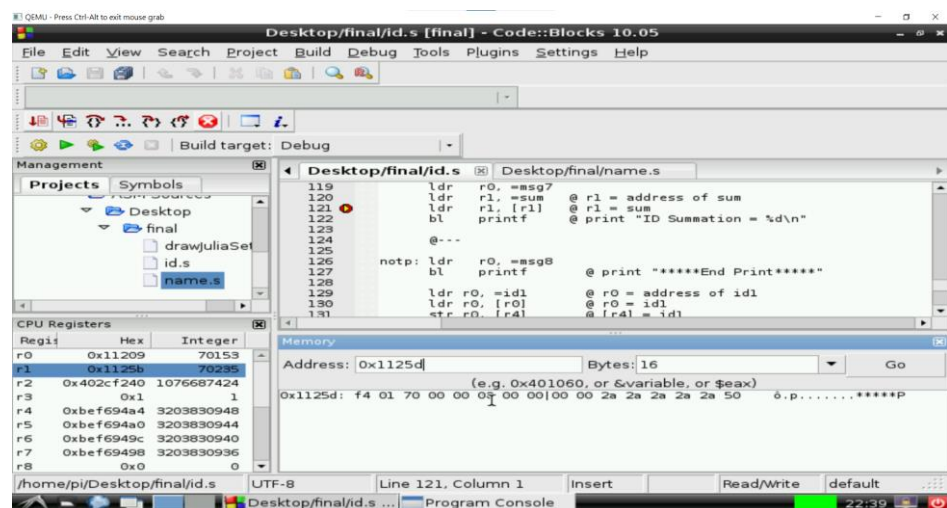
id.s : id3 驗證位址 0x11257



id.s : sum 起始位址 0x1125b

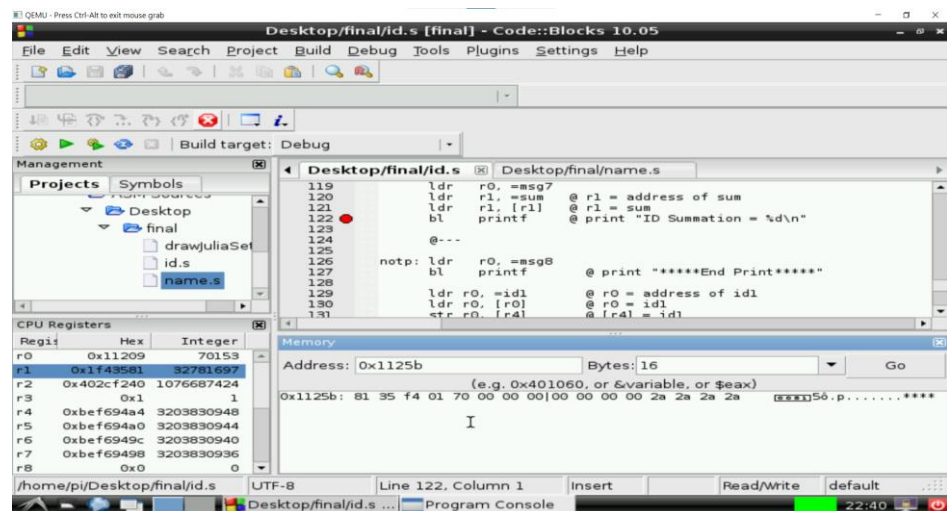


id.s : sum 結束位址 0x1125d

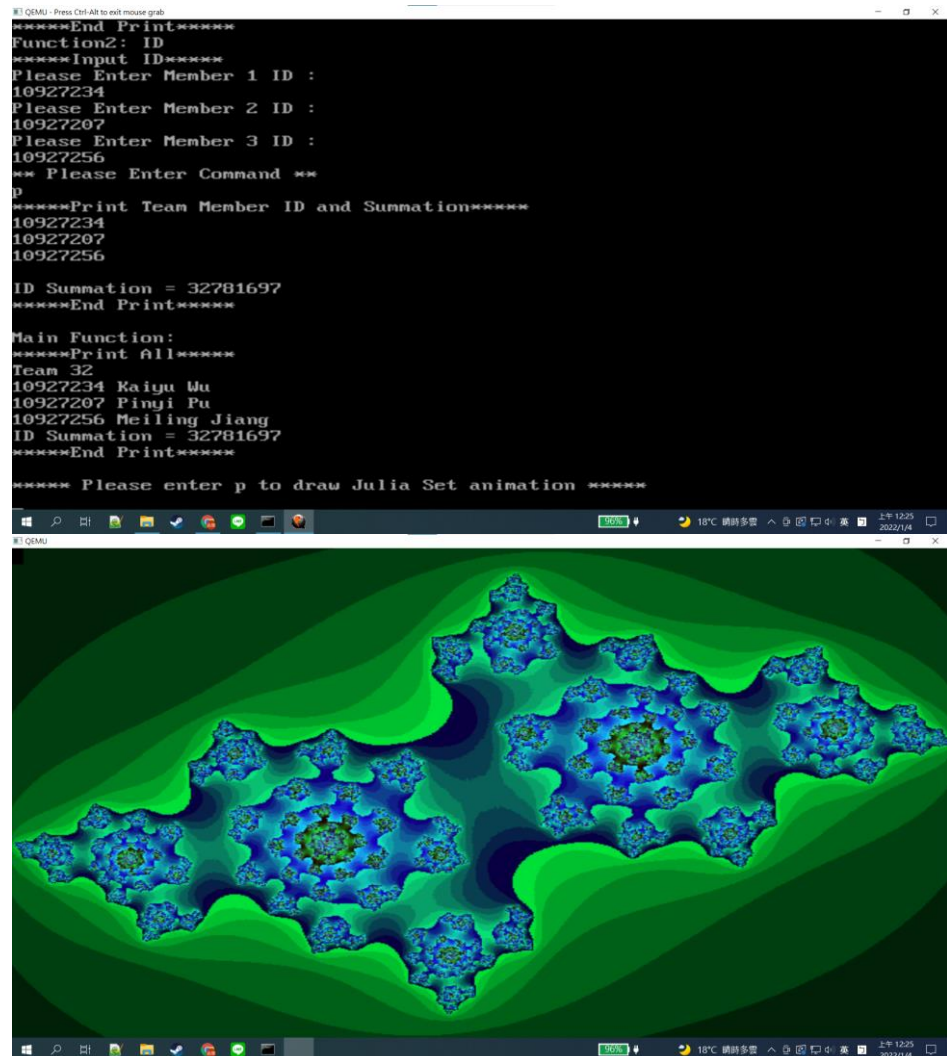


id.s : sum 驗證位址 0x1125b

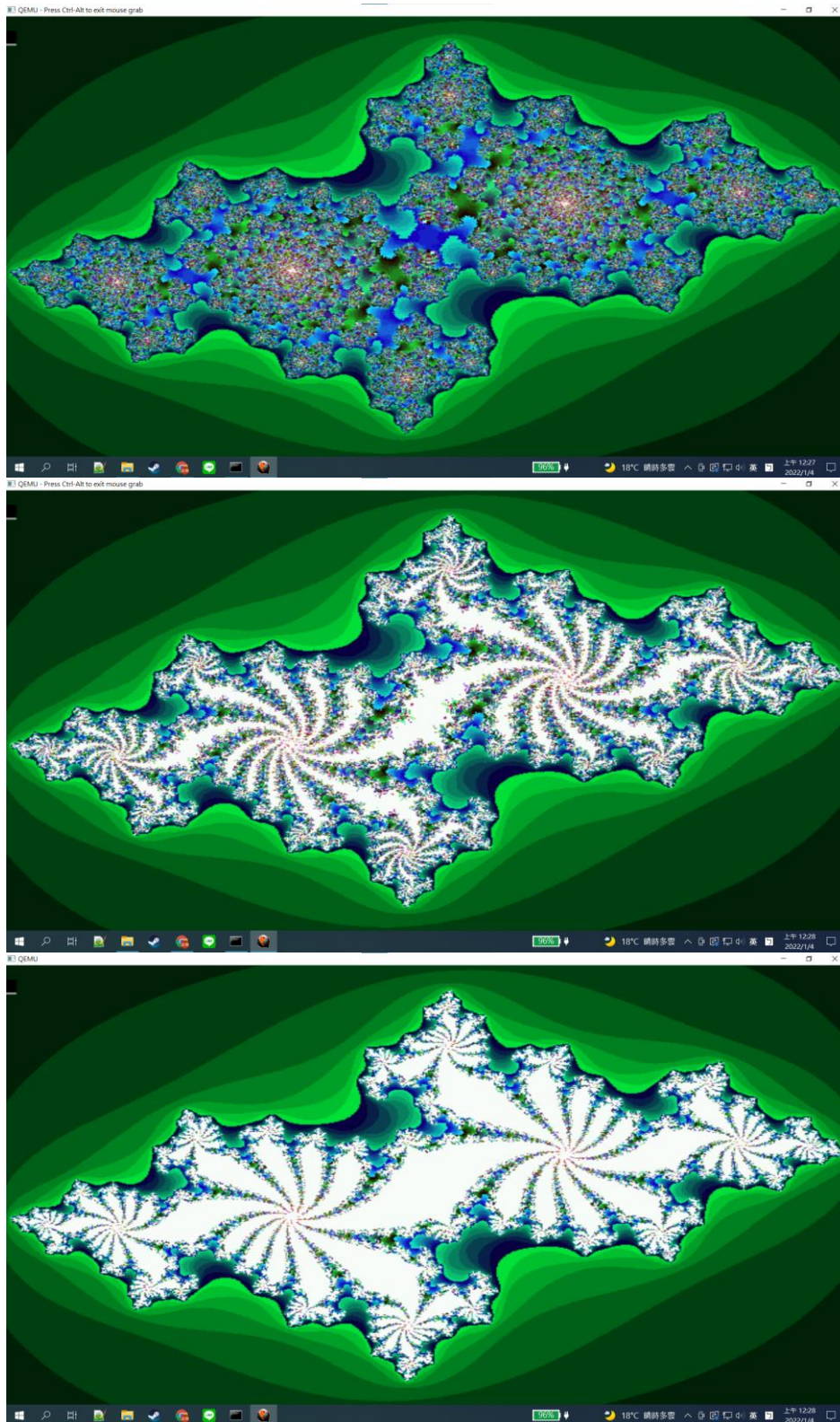


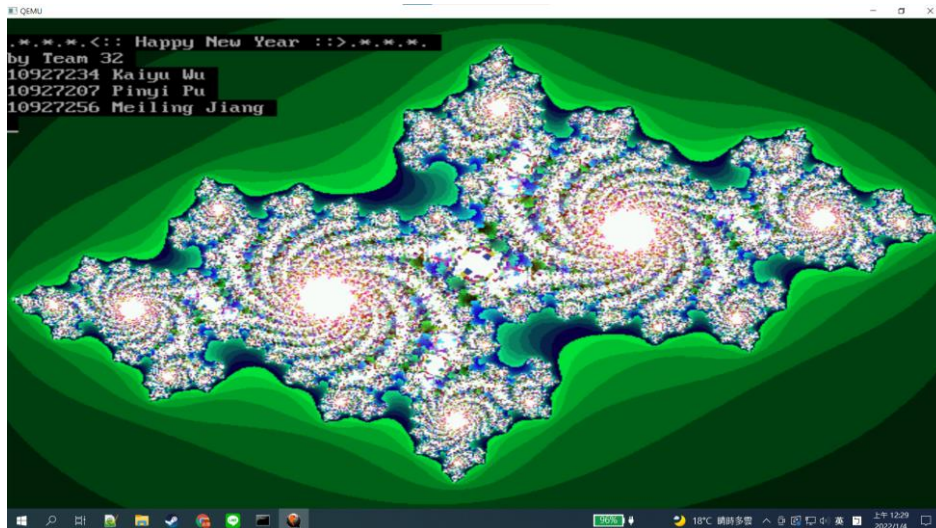


## 程式結果









#### 四、感想

這次作業比 minterm 更加有挑戰性，因為要把 c 語言和 ARM 結合在一起，還要翻譯 drawJuliaSet.c 翻譯成 drawJuliaSet.s，其中有三個迴圈，一個包著一個，十分複雜，還有因為 main 是改為用 c 語言寫，所以在傳變數進 function 的時候費盡了心思。由於這次變數很多，使用的暫存器也多，所以要事先規劃好暫存器位置給變數，並要記得，小心不要改變到變數，不然很難 debug，感覺這次成長了很多。

#### 五、分工

程式碼：姜美玲

報告：蒲品憶、吳凱鈺