

題目：期末 protect

111 學年度第一學期

老師：朱守禮 老師

學生：徐梓薰 11027114

李若菱 11027133

游婕歆 11027149

● 程式說明

1. Name.s 和 id.s 都是沿用期中 project 的程式碼
2. 必要程式碼 add lr, r0, pc, 執行這行前先把 lr 備份起來 mov r2, lr(r2=lr), 再讓 lr = r0+pc, 再 mov lr, r2(lr=r2), 讓 lr 不要跑掉)
3. 一開始我們先把 lr 和 r4-r11 備份到 stack(借位置)。將 r0、r1 的位置複製到 r10、r7 使得 cy、frame 是可回傳值。寫一個大迴圈跑寬，裡面再包一個小迴圈跑高。

跑高的迴圈：先將 y 初始為 0 並存到 r6，接著比較 x 跟 640，若 $x < 640$ 就繼續往下執行高的迴圈 loopheight， \geq 的話就跳出迴圈 loopwidth 然後跳到 endloopwidth。

endloopwidth：設定 $r0 = 0$ ，因結束了最外圈的迴圈，所以要把一開始借的東西還回去。最後還原 lr 回到原本指令的下一行。跑寬的迴圈：先比較 y 跟 480，如果 \geq 就跳到 endloopheight，否則就繼續往下執行，將 r0 的值設為 640(width)，之後再將 width 右旋 1 存到 r2，再將 $r5(x)$ 減 r2，再將 r0 的值設為 1500，將 r0 的值與 r2 相乘存到 r0，再將 width 右旋 1 存到 r1，再利用 __aeabi_idiv 將 r0 與 r1 相除，最後存到 r8(zx)。zy=1000*(y - (height>>1)) / (height>>1) 再以同樣的方式進行。將 r11 的值設為 255(i)，將 r8(zx) 乘以 r8 後存到 r0，接著把將 r9(zy) 乘以 r9 後存到 r1，把 r0 與 r1 相加後可得 $zx * zx + zy * zy$ 。比較 r0 與 constant pool 裡的 4000000，若 ≥ 4000000 就跳到 endwhile，在比較，若 > 0 就跳到 endwhile。

while：將 r8(zx) 乘以 r8 後存到 r0，接著把將 r9(zy) 乘以 r9 後存到 r1，把 r0 與 r1 相減後存入 r0。然後把 r1 設為 1000，利用 __aeabi_idiv 將 r0 與 r1 相除，接著把 $r0 - 700$ 存到 r4 可得 $tmp = (zx * zx - zy * zy) / 1000 + cX$ 。再以相同的方式得到 $zy = (2 * zx * zy) / 1000 + cY$ 、 $zx = tmp$ 。將 r11(i)-1 存回 r11。將 r8(zx) 乘以 r8 後存到 r0，接著把將 r9(zy) 乘以 r9 後存到 r1，把 r0 與 r1 相加後可得 $zx * zx + zy * zy$ ，比較 r0 與 constant pool 裡的 4000000，若 ≥ 4000000 就跳到 endwhile，在比較，若 > 0 就跳到 endwhile，否則就繼續 while。

endwhile：將 r11 AND 0xff 後設定 CPSR，將 r0 設成 8，然後把 r11 跟 (r0 左移一後) 的值之後存入 r11。將 constant pool 裡的 0xffff 存到 r0，再把 r0 NOT AND r11 後存到 r11。把 r7(frame) 存到 r0，再把 r1 設定為 1280，然後再將 r1 和 r6 相乘並存到 r1，再把 $r0 + r1$ 存到 $r0(frame + 1280y)$ ，將 r5 左移一位後加上 r0 後存到 r0，最後把 r11(color) 的值 store 到 r0(frame) 裡。跳到 loopheight。

● 設計重點說明

1. 在 JuliaSet 必須加入作業要求的條件 add lr, r0, pc
2. 呼叫 label、如何設定 nzcv
3. 非 Branch 指令的 Conditional Execution:

```
cmplt r11, #0
```

```
subne r0, r0, r1
```

4. operand2:

```
mov r1, r1, asr #1
```

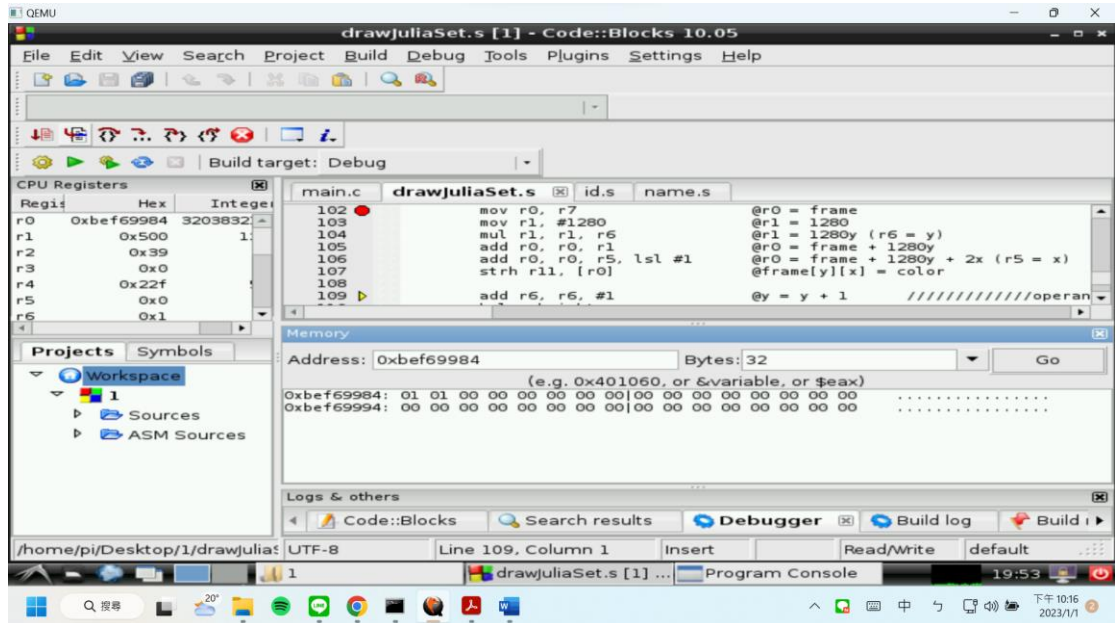
```
add r9, r0, r10
```

```
add r6, r6, #1
```

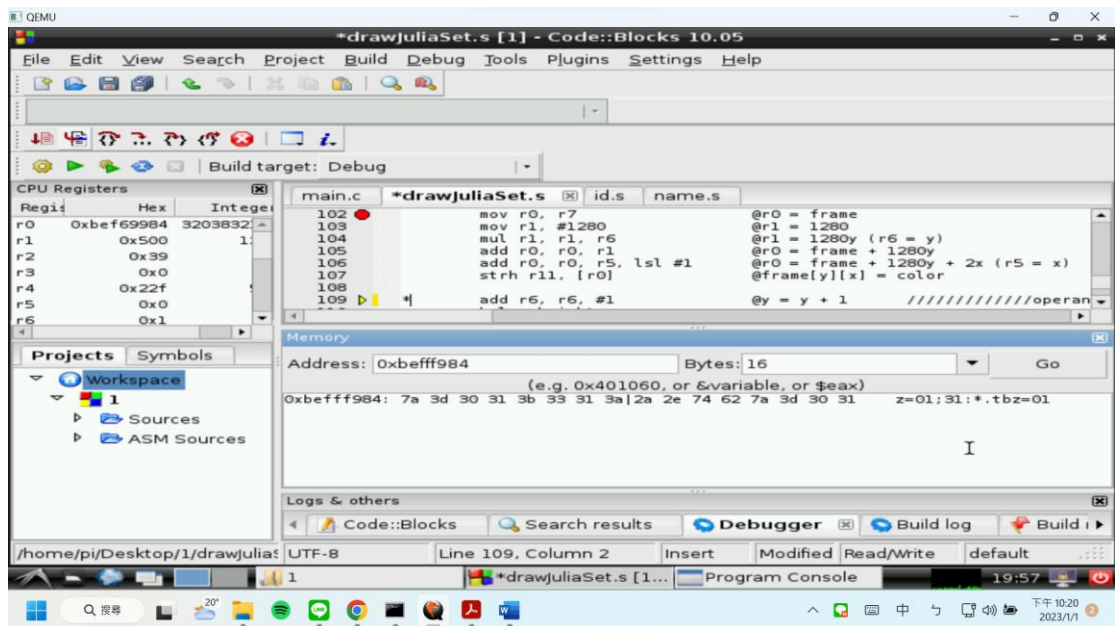
```
add r0, r0, r5, lsl #1
```

結果

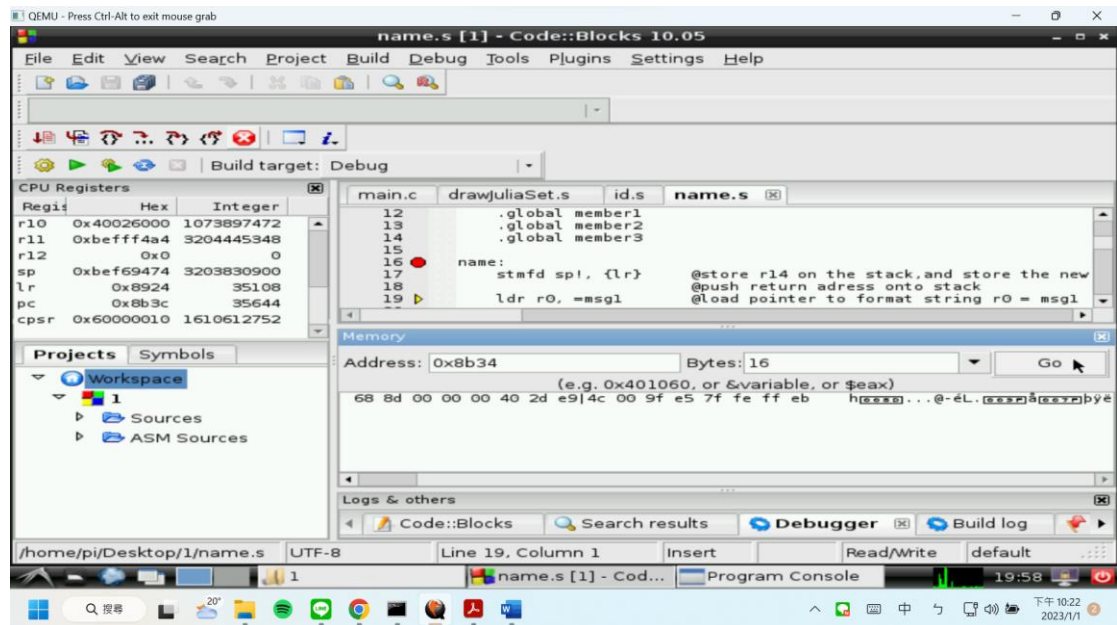
● Frame 起始位址



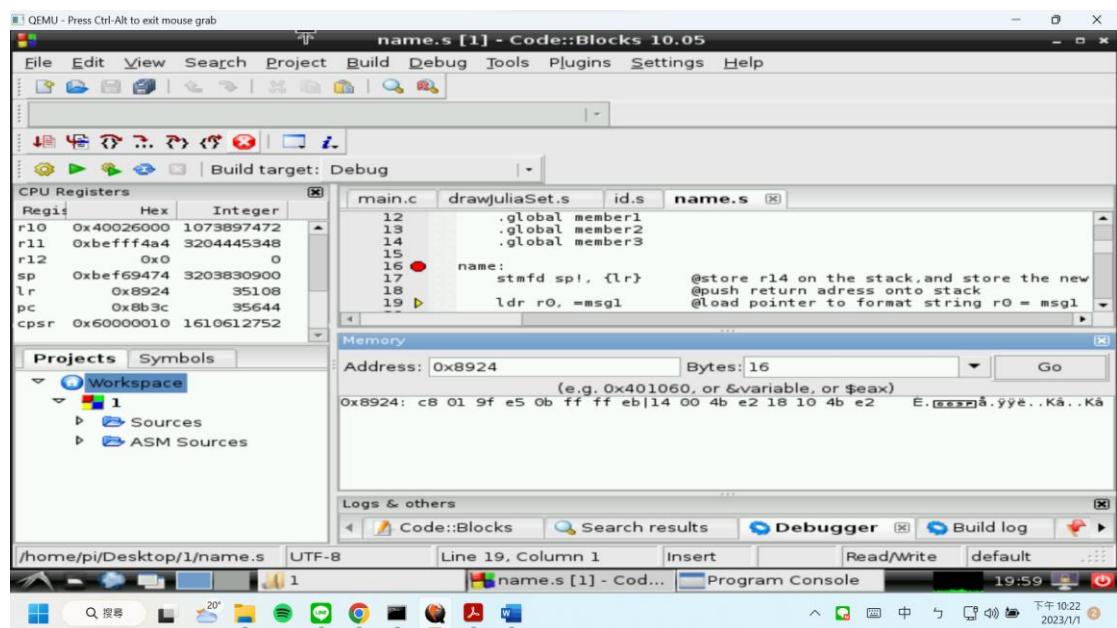
結束位址



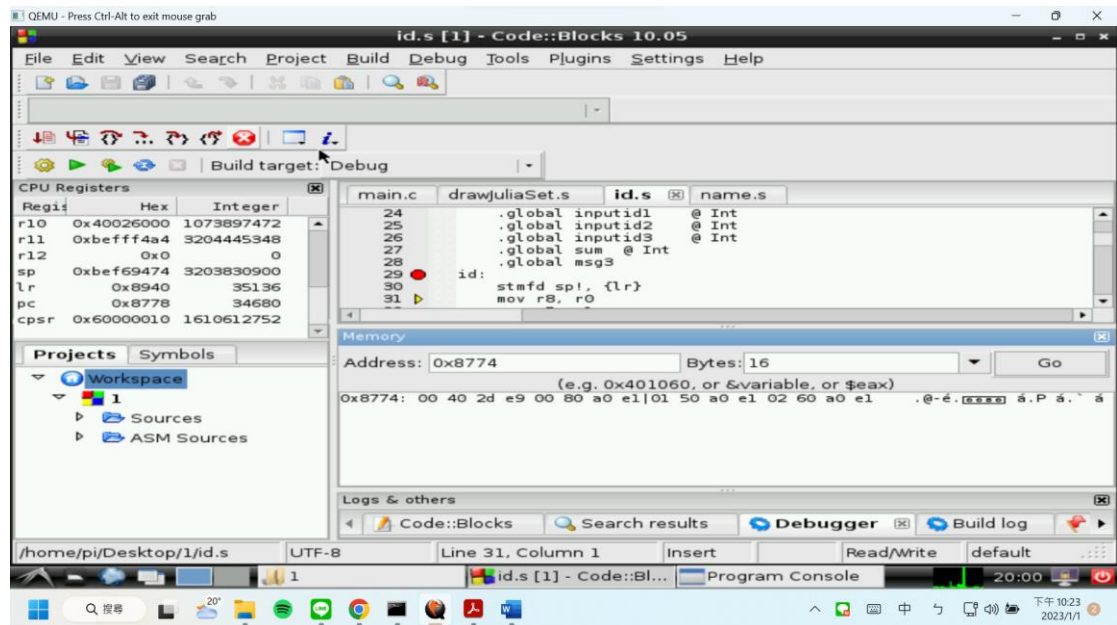
● Name 起始位址



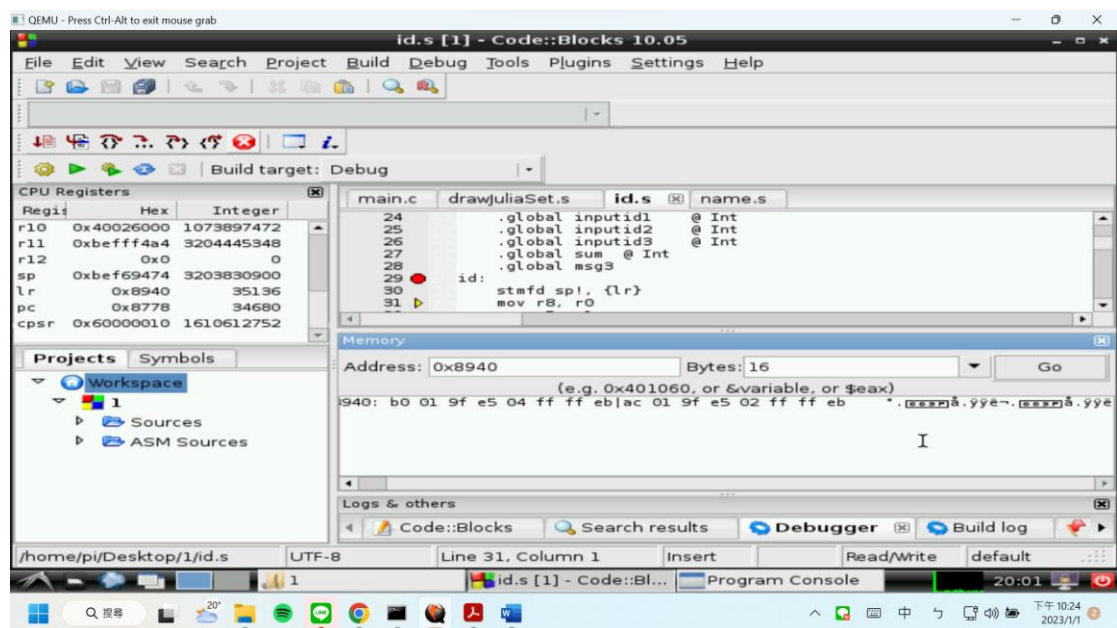
返回位址



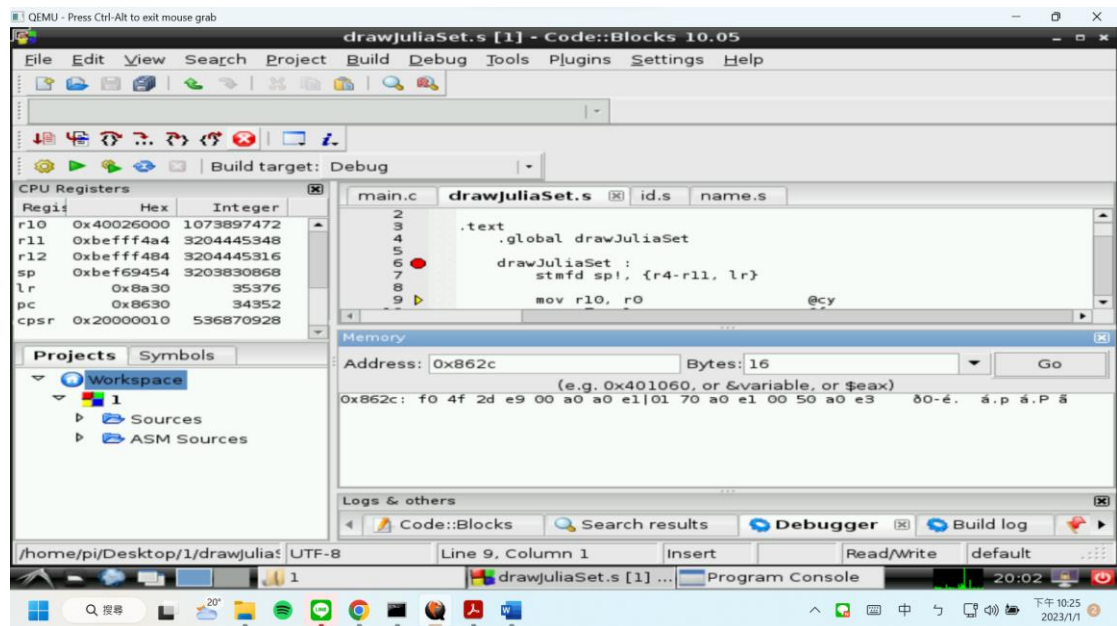
● Id 起始位址



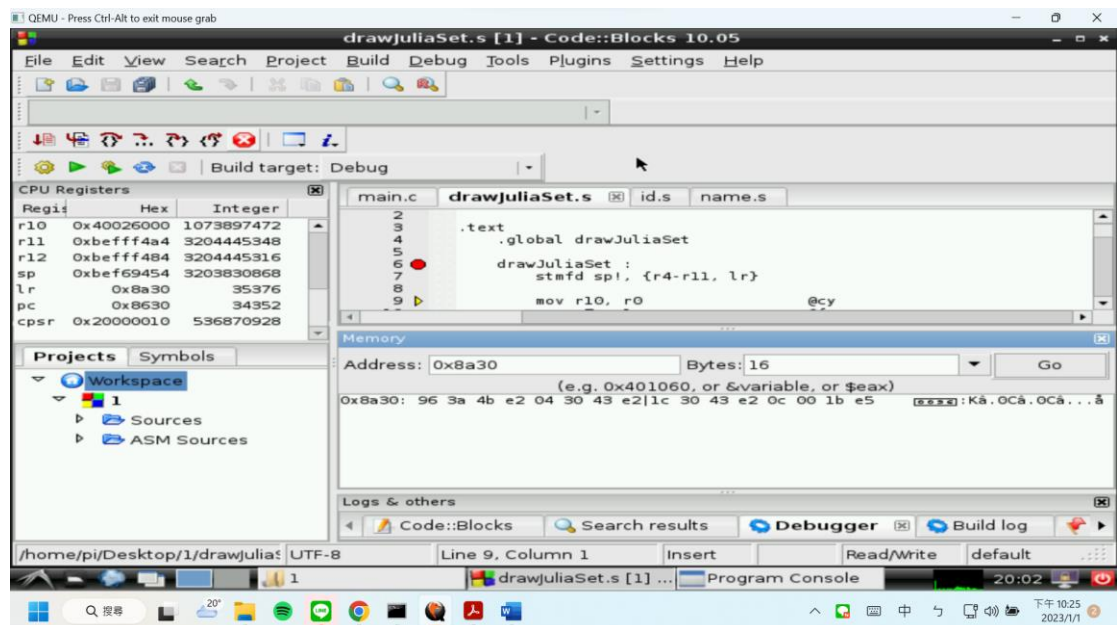
返回位址




● Drawjuliasset 起始位址



返回位址




```
QEMU

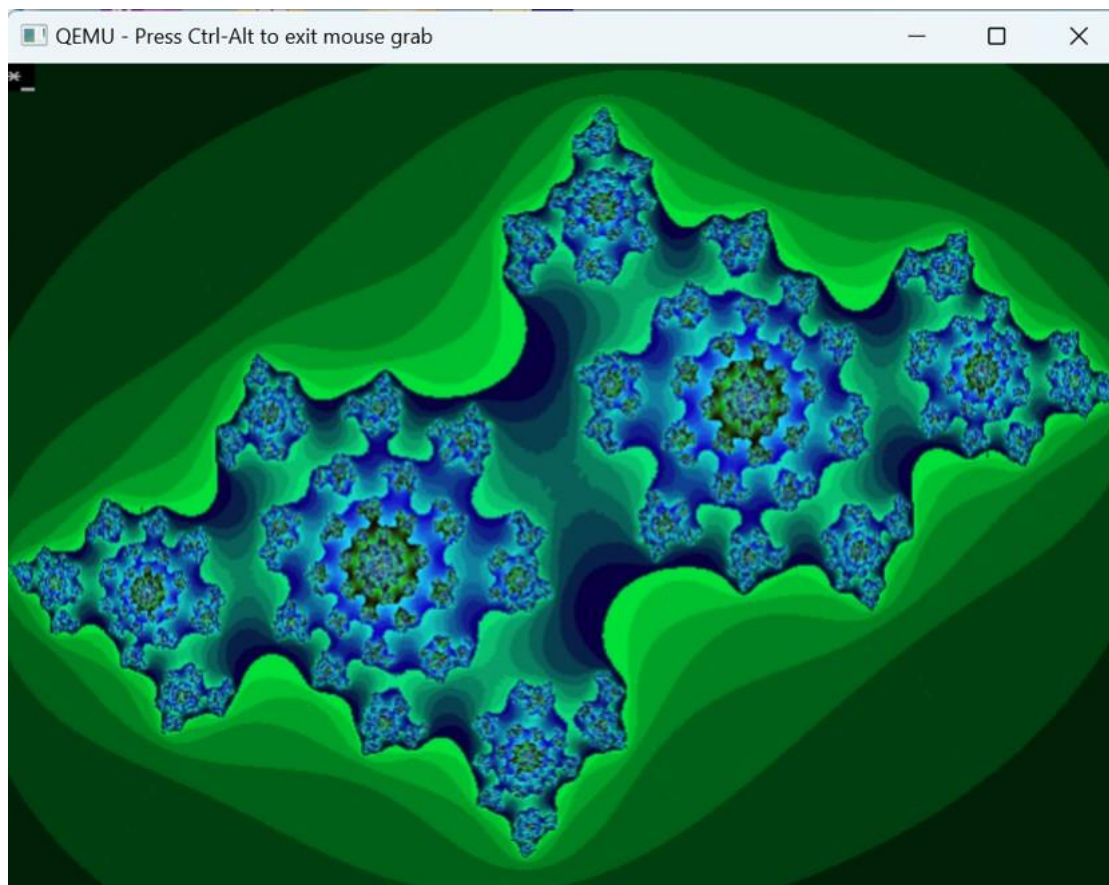


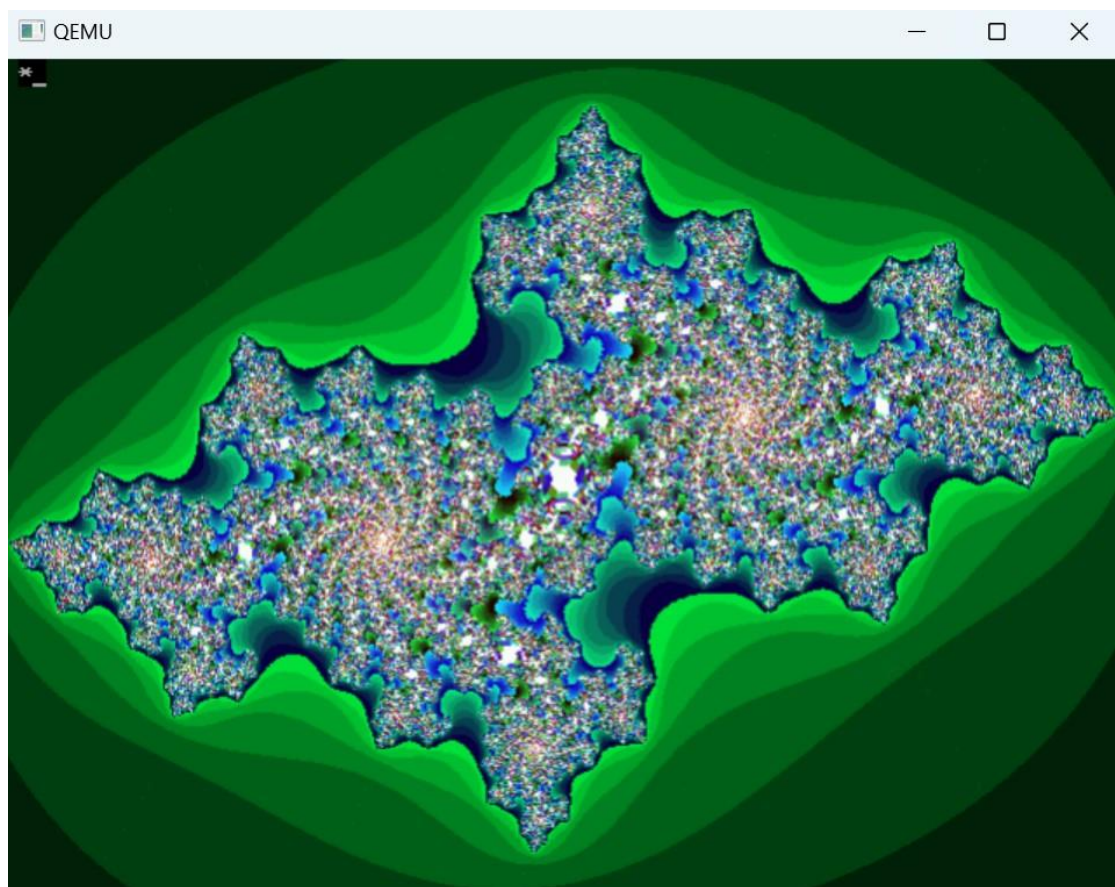
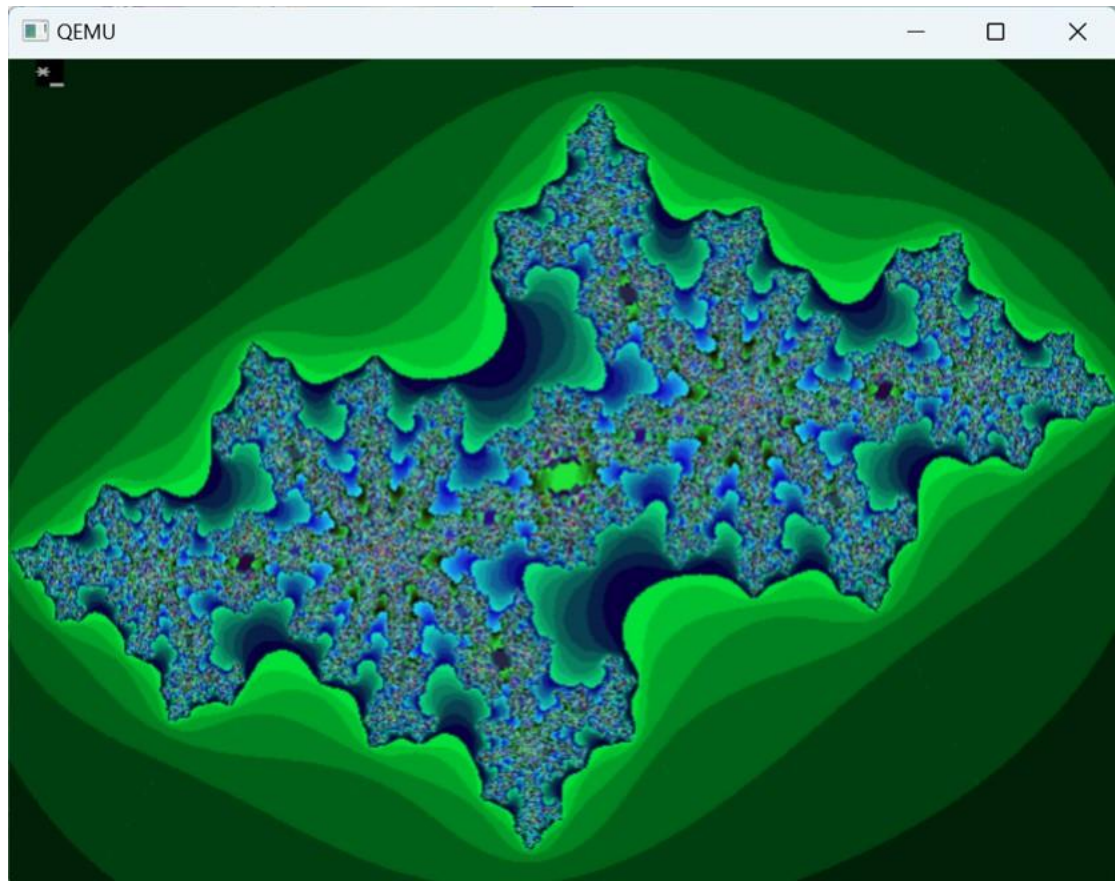
**please enter member2 id : **
11027114
**please enter member3 id : **
11027133
**please enter command : **
p
*****print team member id and id summation*****
11027149
11027114
11027133

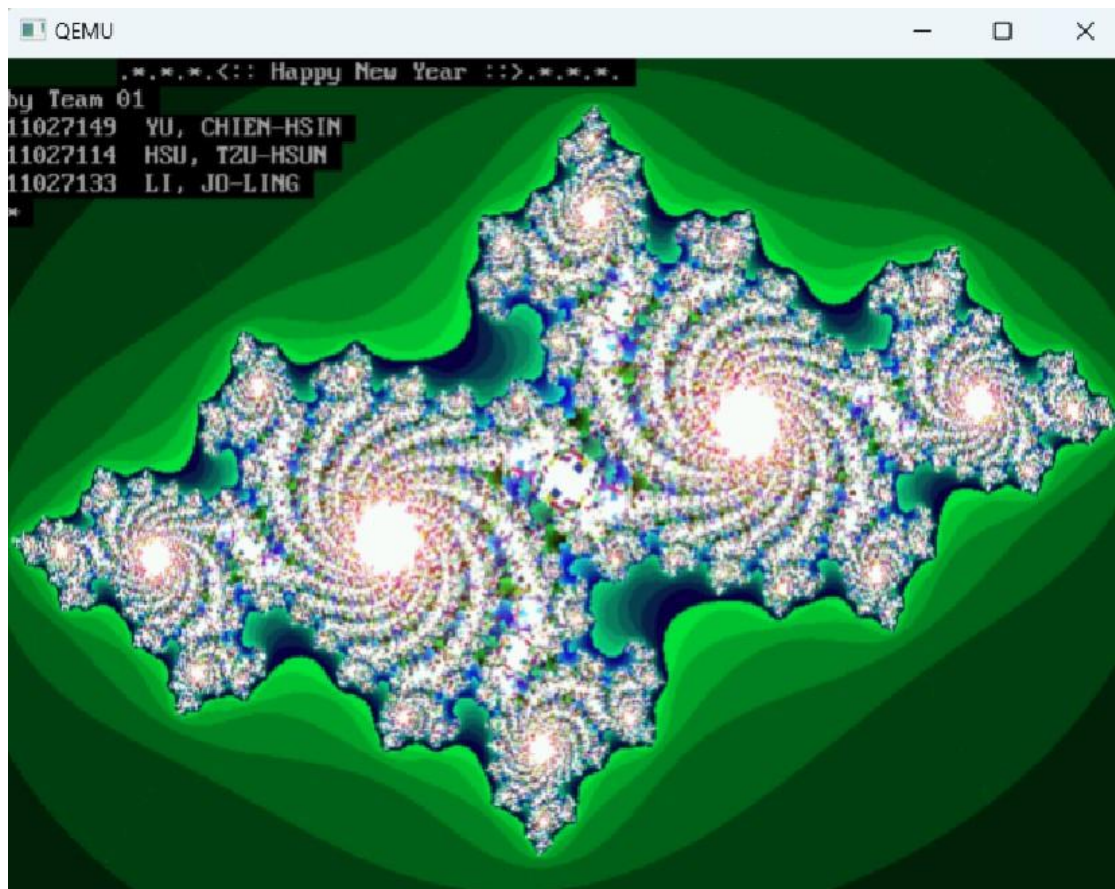
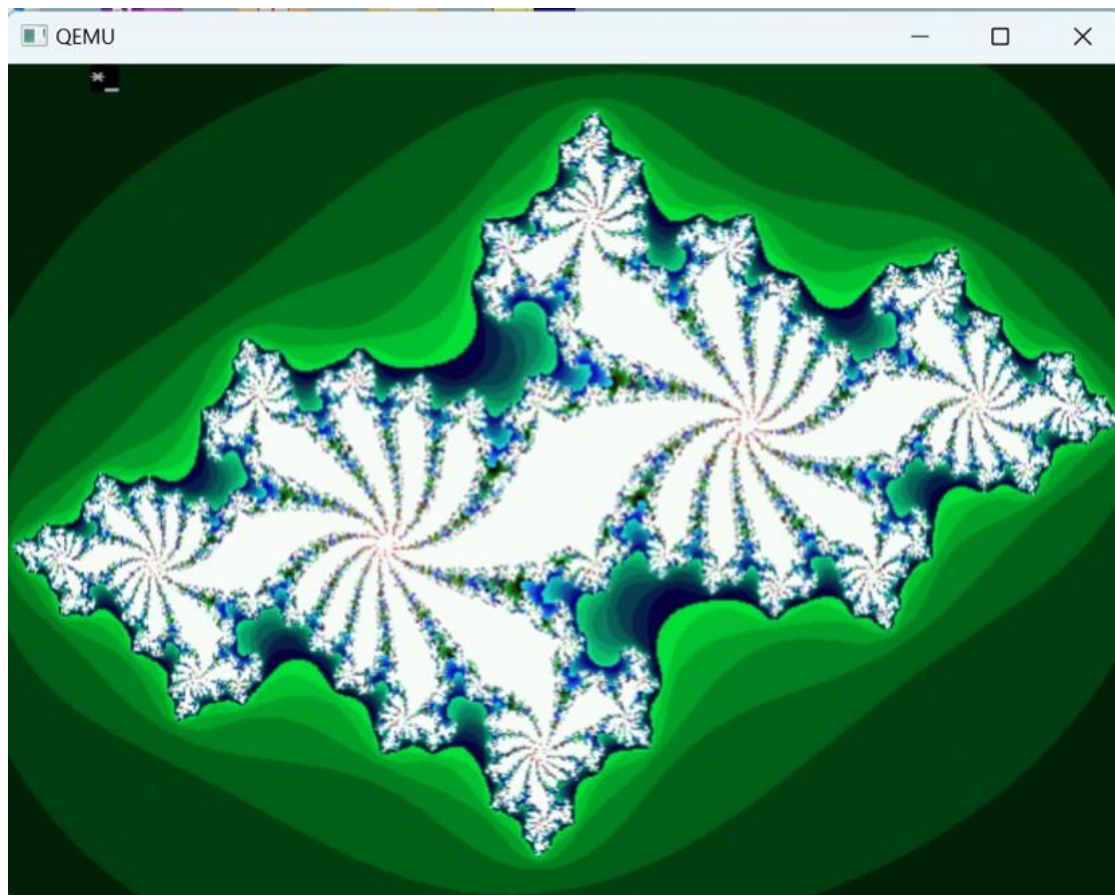
id summation = 33081396
*****end print*****

Main Function :
*****Print All*****
Team 01
11027149 YU, CHIEN-HSIN
11027114 HSU, TZU-HSUN
11027133 LI, JO-LING
ID Summation = 33081396
*****End Print*****

***** Please enter p to draw Julia Set animation*****
```







1. frame 的結束位置要利用 $640 * 2 * 480$ 他總共記憶體的空間，所以要將他的初始位址加上 614400。
2. 在找每個 function 的起始位址時，是要看每個 function 還沒執行的前一條指令，而一條指令佔 4bytes，所以將 PC-4 就是每個 function 的起始位址。
3. frame 的起始位址裡的 0101 內容為 color 的內容。

● 心得

這次作業相較於期中難很多，雖然只是寫一個 function 而已，但裡面多了很多迴圈讓人寫的時候會腦袋打結不知道該如何下手，但和助教同學一起討論問問完問題後就比較了解該怎麼做。

● 分工 大家一起共同完成

