組合語言與嵌入式系統 Midterm Project

107學年度第1學期

老師：

朱守禮　老師

學生：

10620114 龔照博

10620123 胡鎮東

10620124 張惟綱

一、心得

這次的project主要是熟悉裡面的指令與環境。雖然老師給的資料已經很多了但因為是三個人打得所以再整合的時候發現很多問題，需要利用其他的方式來呈現其結果，所以也發現很多現在還不太懂得地方，像是scanf存入的值再此程式執行後要怎麼將其值帶到另一個程式繼續利用，因此組員之間的分工還有一起討論也顯得相當中要。經過這次的活動，讓我更了解組合語言的運作方式，了解到程式中記憶體的如何配置，也更加熟悉了語法，更深深地認識到組合語言跟C語言的不同之處。

二、方法

一二題需要理解一些簡單的指令還有簡單的程式架構，宣告一些全域的變數之後在main裡面進行，只要充分的了解記憶體語暫存器之間的轉換，呼叫C Function就可以輕鬆完成，而第三題利用指令跳躍到其他兩個function中執行程式碼，並將所需要的標籤設成globl 使其可以運用在主程式中。因為需要印出的數據較多，所以利用stack將要印出的數據倒敘放路，再呼叫printf指令印出結果。

三、結果

結果發現利用堆疊印出的方式雖然較為麻煩但能減少呼叫printf指令的次數有助於程式執行的速度，但需要注意回復stack的address以免發生錯誤。而其他的部分較為簡單，只是單存利用bl指令執行其他兩個檔案。

四、討論

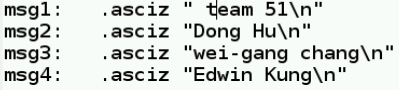
因為再呼叫其他的程式碼時，發現id.s檔案中的summation id因為不是用label存取所以再呼叫的時候發生無法印印出其值的問題。

五、程式說明

***Task1***

1543949354552

在data區段



規劃4 個label，用來存放組別與組員的英文姓名。

1543949403245

在text區段

1543949682732

宣告一個全域變數main,

1543949708059

先將回傳地址存入stack

1543949522146

讀取msg1進r1

1543949778534

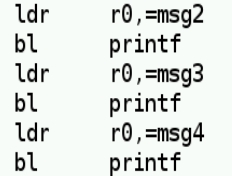
將R2設為空

1543949903262

將r1+r2指定到r0

1543949940751

使用printf



將一個一個msg印出來

1543950035223

將r0設為0

1543950060501

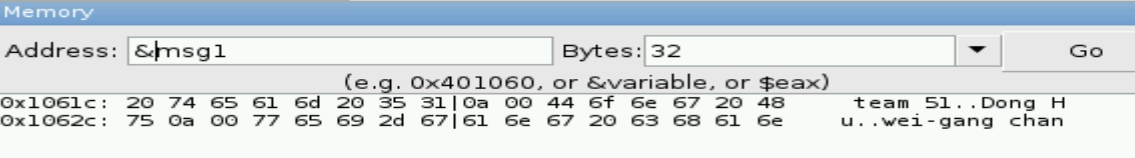
將回傳地址拿出stack

1543950078024

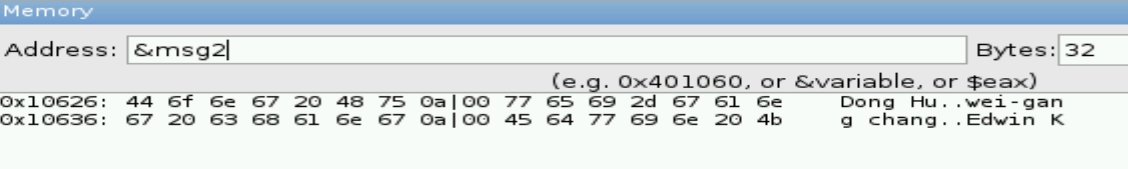
回到程式原點

Memory Dump :

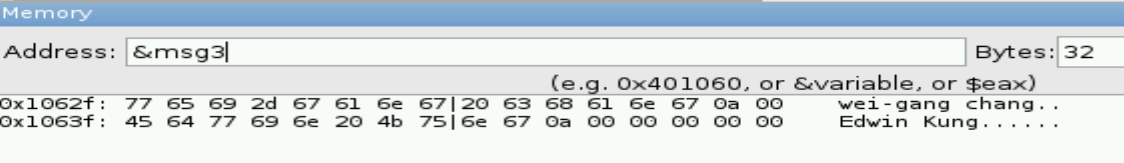
起始位置 :0x1061c 最終位置 :0x10624



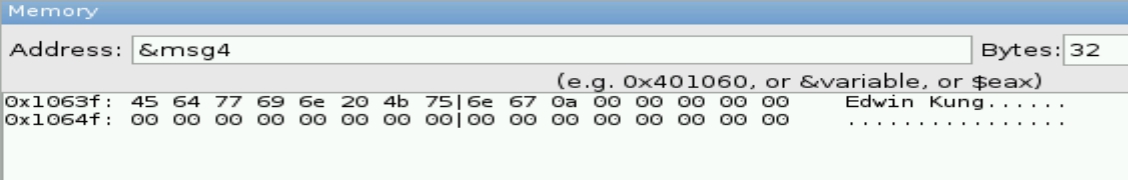
起始位置 :0x10626 最終位置 :0x1062d



起始位置 :0x1062f 最終位置 :0x1063d



起始位置 :0x1063f 最終位置 :0x10649



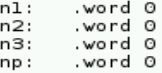
***Task2***



在data區段



規劃2 個label，用來存放型別。



規劃4個label，用來存放輸入的學號 和 指令P

1543949403245

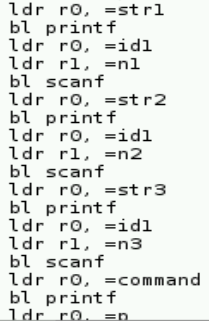
在text區段

1543949682732

宣告一個全域變數main,

1543949708059

先將回傳地址存入stack



讀取str1到暫存器R0



呼叫C Function Printf

印出str1的字串

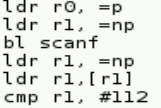


呼叫C Function Scanf

把字串讀入n1中

這裡要特別注意的就是Scanf會先去找r0暫存器，也就是型別

再來找到r1暫存器，也就是使用者輸入的值



這部分的程式碼，用來判斷使用者按下的到底是不是p鍵



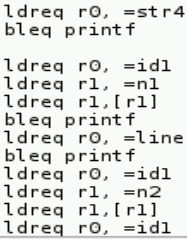
比較特別的部分在這裡

首先，我們要讓r1暫存器存放r1的值

再來，判斷r1的值是否為112

然而為甚麼是112

是因為p的ascii值恰為112



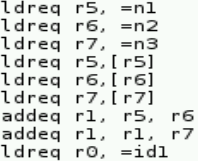
(此部分只擷取一小段程式碼)

因應題目的要求

按下P鍵才會顯示學號以及學號的加總

因此執行下面程式碼需滿足cmp的條件

R1 equal #112



取n1, n2, n3,的值(也就是前段程式碼讀入的值)

將其加總

並印出加總後的數字

1543950060501

將回傳地址拿出stack

1543950078024

回到程式原點

Memory Dump :

Address: &str0

起始位置 :0x10738 最終位置 :0x10749



Address: &str1

起始位置 :0x1074c 最終位置 :0x1075c



Address: &str2

起始位置 :0x1076c 最終位置 :0x1077c

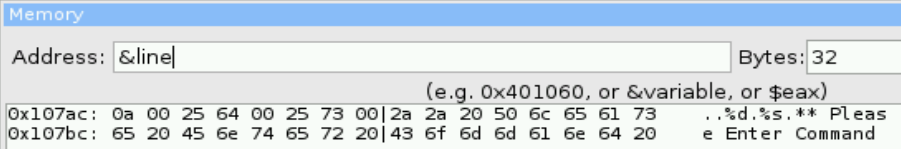


Address: &str3

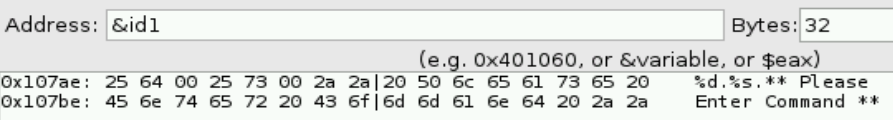
起始位置 :0x1078c 最終位置 :0x1079c



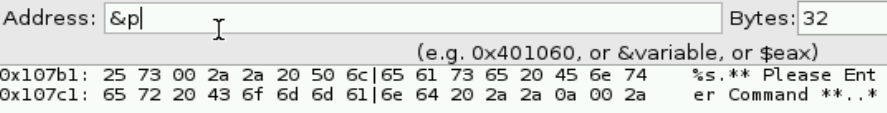
起始位置 :0x107ac 最終位置 :0x107ae



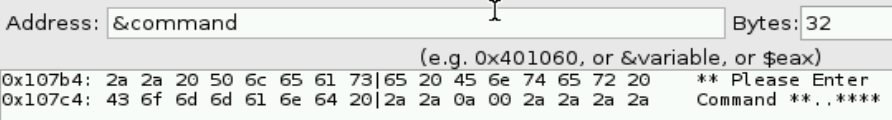
起始位置 :0x107ae 最終位置 :0x107b1



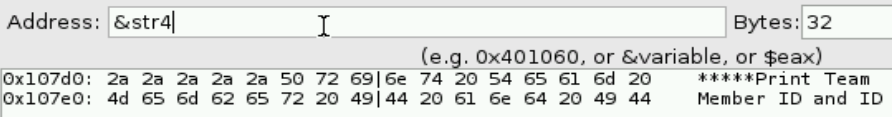
起始位置 :0x107b1 最終位置 :0x107b2



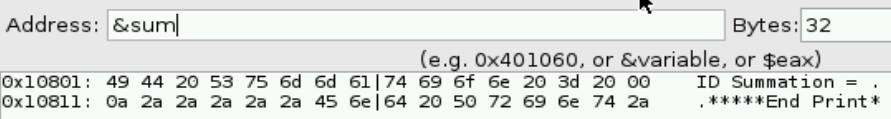
起始位置 :0x107b4 最終位置 :0x107cd



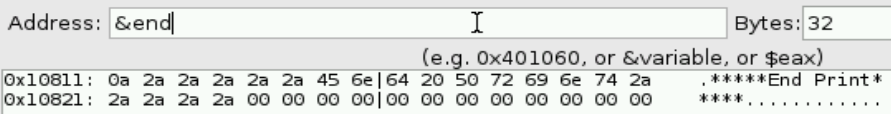
起始位置 :0x107d0 最終位置 :0x107fd



起始位置 :0x10801 最終位置 :0x1080f



起始位置 :0x10811 最終位置 :0x10824



***Task3***

將兩個副程式利用bl指令執行，然後取其資料利用堆疊的方式一次性的印出。最後再還原堆疊空間的記憶體

六、結論

利用stack堆疊資料再印出能更有效率執行程式碼。使用bl指令呼叫其他function需要注意是否設為globl且要注意有無重複的名稱。

分工:

10620114 龔照博 第二題，第二題文件10620123 胡鎮東 第一題，第一題文件

10620124 張惟綱 第三題，第三題文件