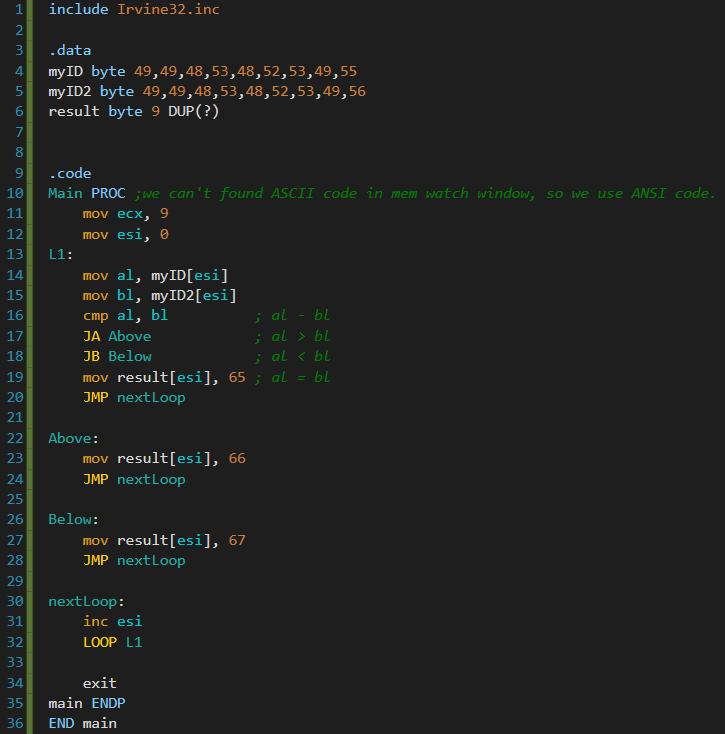
**Week 5 Lab Assembler**

**Group 65**

**Integrants: 110504517 李睿穎 (Leader)，110504518 鍾秉均 (Member)**

**Full Code**

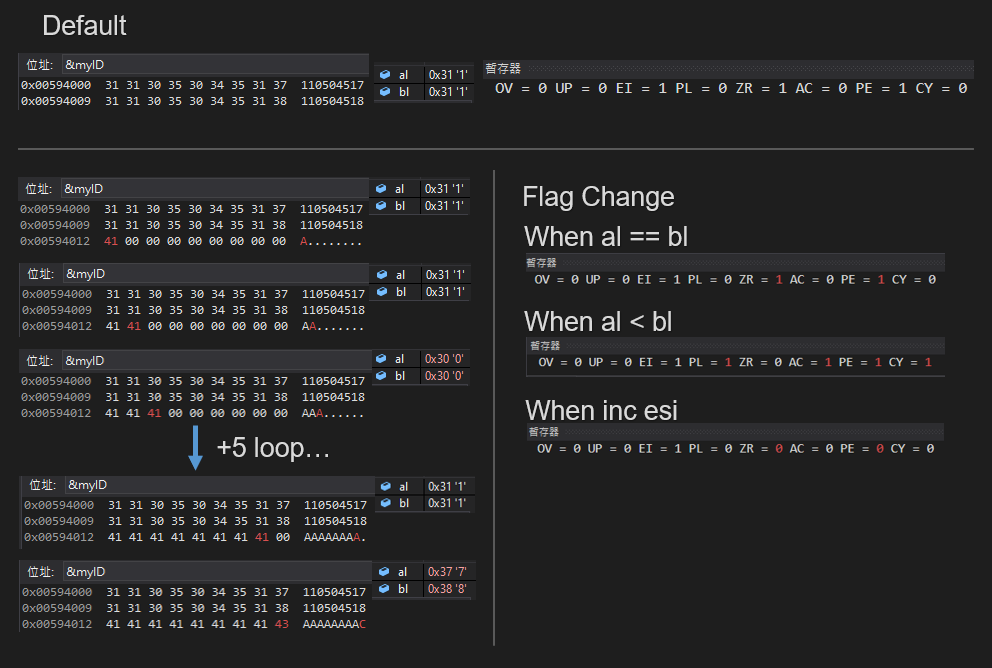
**L16:**

**Here we compare the values of al and bl, what the command does is it performs a subtraction of al - bl, if the result is negative (meaning that bl is greater, cf = 1) then we jump to “Below”, otherwise (al greater is than bl, cf = 0, zf = 0) then we jump to “Above”**

**L17: Here we jump to “Above” if al > bl  
L18: Here we jump to “Below” if al < bl**

**L20: Here we need to specify that after L19 we need to directly jump to “nextLoop”, otherwise we will go directly to “Above”**

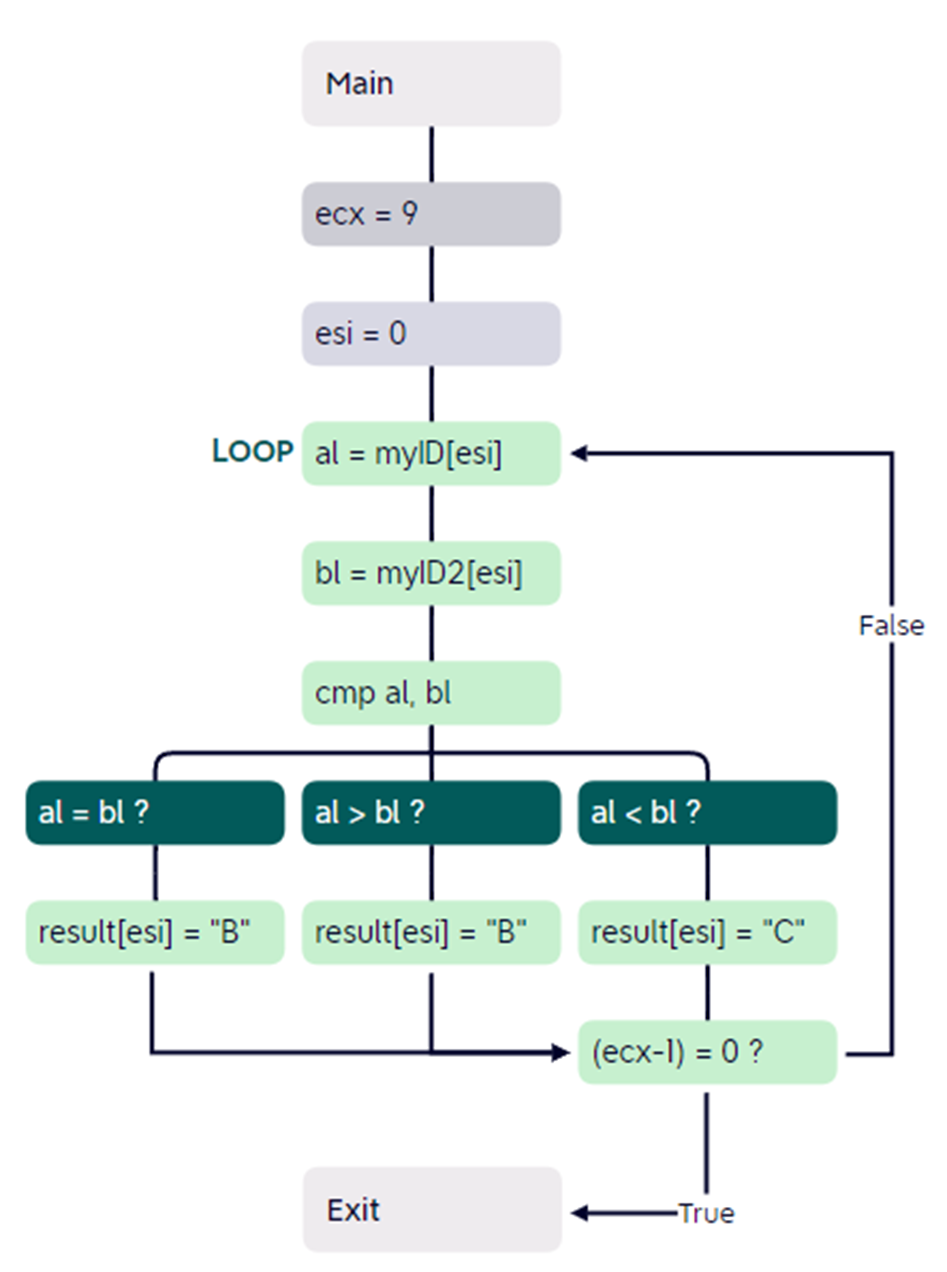
**Register and Flags**

****

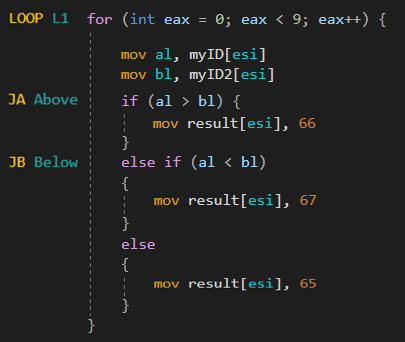
In this image we can see what are the changes in our memory, when we finish comparing the first byte, then we assign the corresponding character to our result (We used ANSI code because we couldn’t find the ASCII code option in the memory window). As you can see both our student numbers are basically the same except the last digit, when both digits are the same, then we can see that when we do the comparison, the zero flag turns into 1 (same number subtraction equals zero).

At the last digit, bl is greater than al, so when we do the subtraction our carry flag turns into 1.

**Execution flow**

****

**Logic in high level language**

****

**Reflection**

For today's class was mainly focused on how to use conditionals, first we learned how to use the Boolean operators AND OR XOR to compare inputs.

Next, at class and at this week's lab, we learned how to use the “greater than” and “less than” operators in assembler, and after comparing we also learned that we can use the JUMP command to do whatever we want depending on the output of the comparison.  This is really important because this lesson permits us to use the “if else” in assembler, which is crucial for us to create more complex programs in the future.