Assembly Language Report (Week 4)

Group 7

104502517戴逸任

104502518劉冠聲

Program CODE:

.data

ninenine BYTE 9 dup(?)

.code

start@0 PROC

mov ecx,9 ;ecx=9h

mov al,0 ;al=0

mov esi, OFFSET ninenine ;use esi to point to ninenine address

L: ;loop

add al,9 ;al=al+9

mov [esi],al ;the value of esi = al

inc esi ;esi+1

loop L ;loop

Program Step & Register state:

mov ecx,9 ;al=0x19 ecx=0x00000009 esi=0x00401005

mov al,0 ;al=0x00 ecx=0x00000009 esi=0x00401005

mov esi, OFFSET ninenine ;al=0x00 ecx=0x00000009 esi=0x00404000

L:

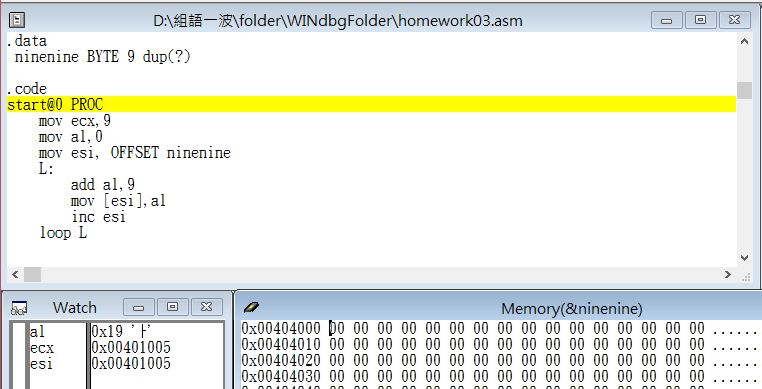
add al,9 ;al=0x00🡪0x09……🡪0x51

mov [esi],al ;ninenine=09 00……00🡪09 12……51

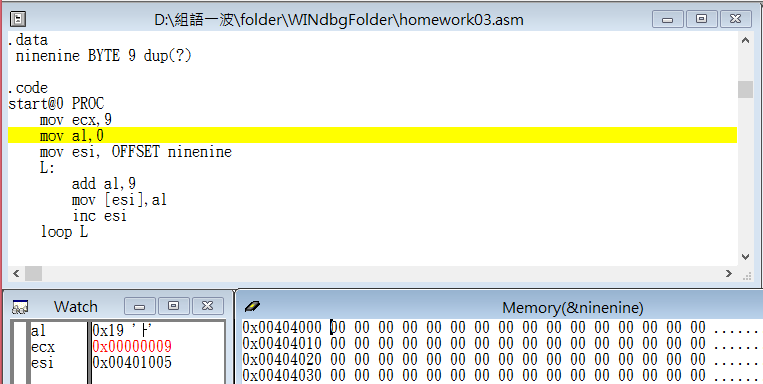
inc esi ;ecx=0x00000009🡪0x00000000

loop L

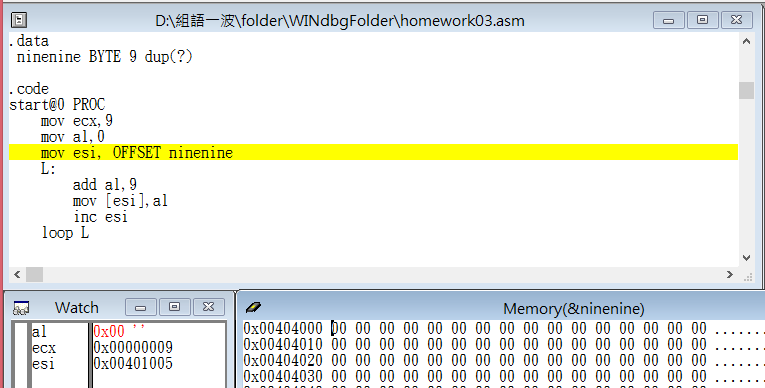
Picture & Discription:



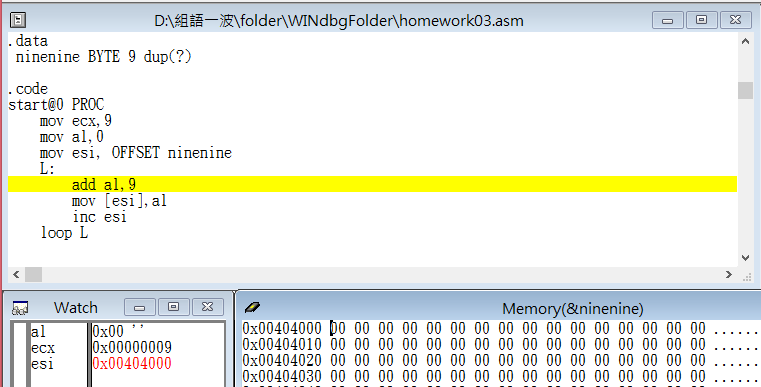
Step1: Start the program



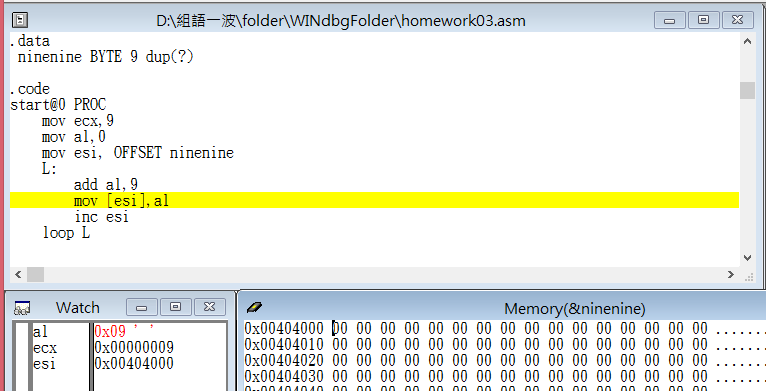
Step2: move 9 into register ecx



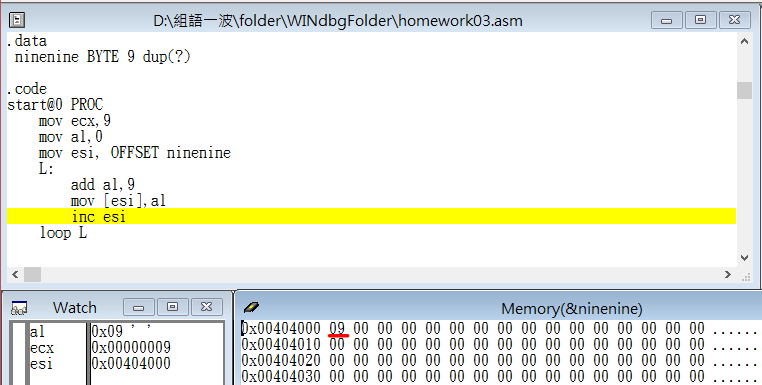
Step3: move 0 into register al



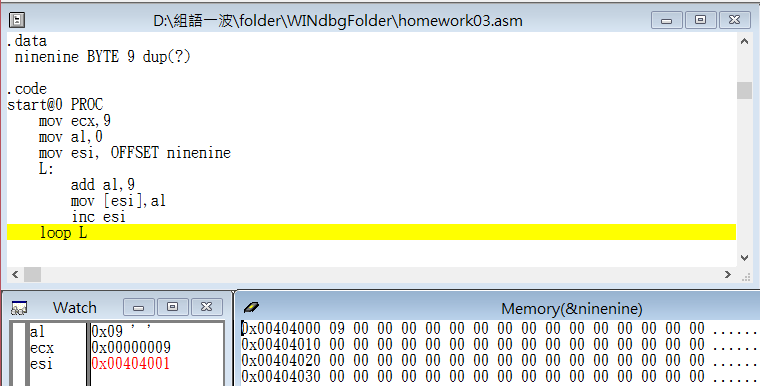
Step4: use esi to point to ninenine address



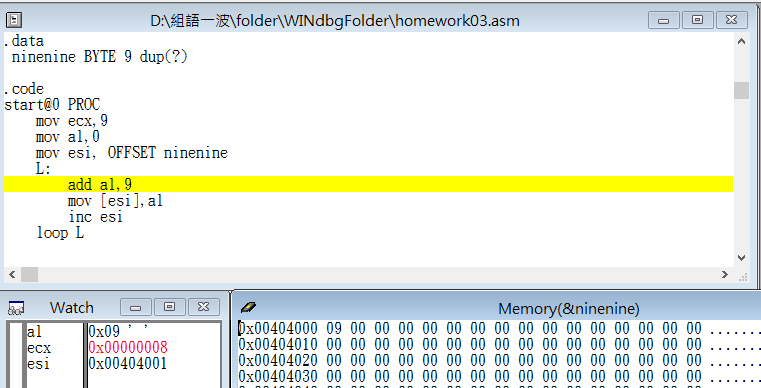
Step5: al=al+9



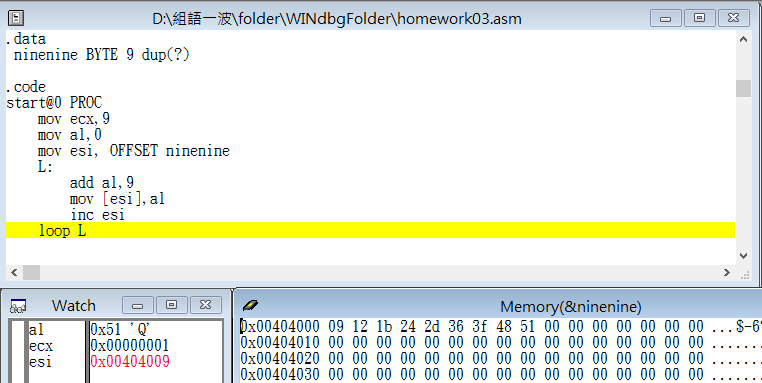
Step6: the value of esi = al



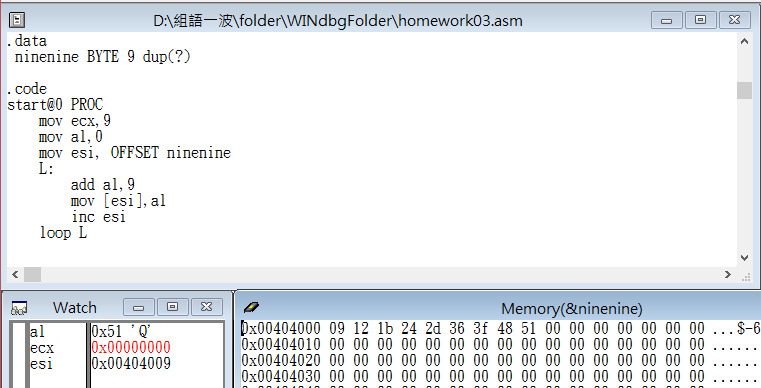
Step7: shift eax 1 time left again, it means that eax become four times of its initial value ((Val1+Val2)\*4)



Step8: return the beginning of the loop and start the second loop(total 9 loops)



Step9: the last command in total loop



Step10: ecx = 0 and loop end(the end)

Review:

The exercise this week is about calculating one column of a ninenine table, in the beginning, I didn’t clearly realize the definition of DUP and ESI, after running error for many times then I finally knew how to use it. It’s my first time to use a loop in assembly language, and it’s pretty different from the experience which I use a loop in high level language.

The news about TA changes the deadline from Thursday 10 p.m. to Friday noon is really kind to us, and that lets us have more time to practice and finish the assembly language report.