組合語言

QUIZ No.1(PART II) 30%

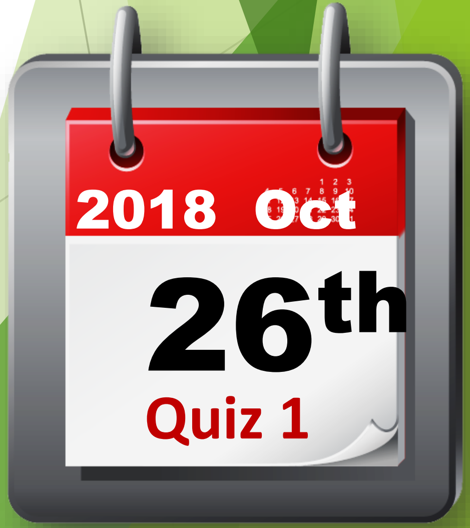
Assembly program

Dept. of CSIE, Oct 26th, 2018

Fu Jen Catholic University,

Hsin Chuang, Taipei

周賜福



Golden Rules to follow during your exercise

- Keep your cell phone and USB drive in your bag.
- When you finish your exercise, do not talk to others or help others.
- Do not copy others work. Do it by yourself.
- No drinks or water should be placed on the table. Place keep it in your bag for the safety of your computer.
- Try to write the code first before you start using computer to prove that your work is correct.

After you finish your assembly program, Demo it to the Assistant to get the signature.

Then, write the Code in your answer sheet for your grade and return the answer sheet to the Instructor.

ODD Seat numbered students Program on the next Page.

ODD Number seats write the following Program in x86 Assembly language

- Ask the user to enter two numbers, N1 and N2. Store the numbers in starting from N1 to N2 in a BYTE Array called ARRAYBYTE. Now reverse the array and write it in an another array called REVERSEBYTE. ADD the total and display the result.
- Ex: N1= 27, N2=43.
- ARRAYBYTE BYTE 200 DUP(0) // Declare this array BYTE addressed.
- REVERSEBYTE BYTE 200 DUP(0) // This also must be an array of BYTE.
- Use a loop (1) to write as follows:
- ARRAYBYTE {27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43}
- Use a loop (2) to reverse the array:
- Reverse the array and write it in REVERSEBYTE.
- Use a loop (3) to print out the reversed array and finally the total.
- You must use 3 loop separate loops.

Use the following DATA section for ODD program.

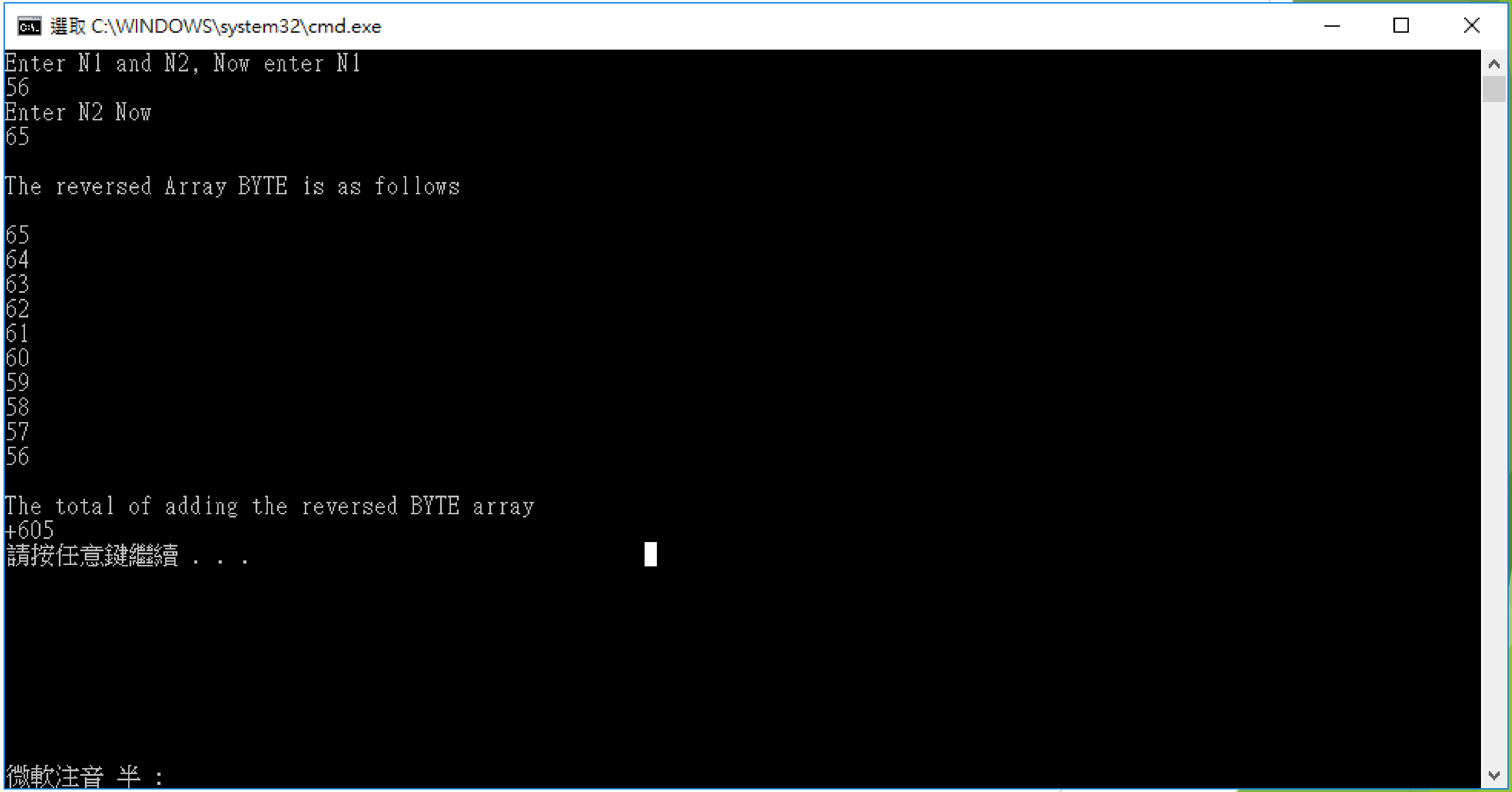
```
INCLUDE Irvine32.inc
.data
arrayBYTE BYTE 200 DUP(0)             // very important to use BYTE ARRAy
count EQU (LENGTHOF arrayBYTE)
reversearraybyte BYTE count DUP(0)   // very important to use BYTE ARRAY
N1 DWORD 0
N2 DWORD 0
runvalue DWORD 0
Total DWORD 0
MSG0 BYTE "Enter N1 and N2, Now enter N1",0dh,0ah,0
MSG1 BYTE "Enter N2 Now",0dh,0ah,0
MSG2 BYTE "The reversed Array BYTE is as follows",0dh,0ah,0
MSG3 BYTE "The total of adding the reversed BYTE array",0dh,0ah,0
```

```
.code
main PROC

mov edx, OFFSET MSG0     ; move the string ask the user to INPUT N1 and N2
call writestring         ; write the string to display.
call readint             ; read the N1 here.
mov N1, eax              ; move to the Storage N1

mov edx, OFFSET MSG1     ; move the string ask the user to INPUT N1 and N2
call writestring         ; write the string to display.
call readint             ; read the N2 here.
mov N2, eax              ; move to the Storage N2

mov eax, N2              ; Ready to move N2 in eax
sub eax, N1              ; calculate N2 – N1 to find the value to run
inc eax                  ; add one to N2-N1
mov runvalue, eax        ; Move that value into runvalue
mov ecx, eax             ; Move that value into ecx for loop
mov esi, OFFSET arrayBYTE    ;move the OFFSET of arrayBYTE
mov  al, BYTE PTR [N1]   ; Use Ptr to move the first value
```

```
; Two loops are given below. First loop to write the value from N2 to N1
L1:
   mov [esi], al
     add esi, TYPE arrayBYTE
     inc al
    loop l1

mov ecx,runvalue
mov esi, 0
mov edi, runvalue
mov eax, runvalue
; The following loop tries to reverse the array
; The loop reverses from behind and writes the value in ReversearryByte
L2:
        mov al, arrayBYTE[edi-1]
          mov reversearraybyte[esi], al
          dec edi
        inc esi
     loop l2
```

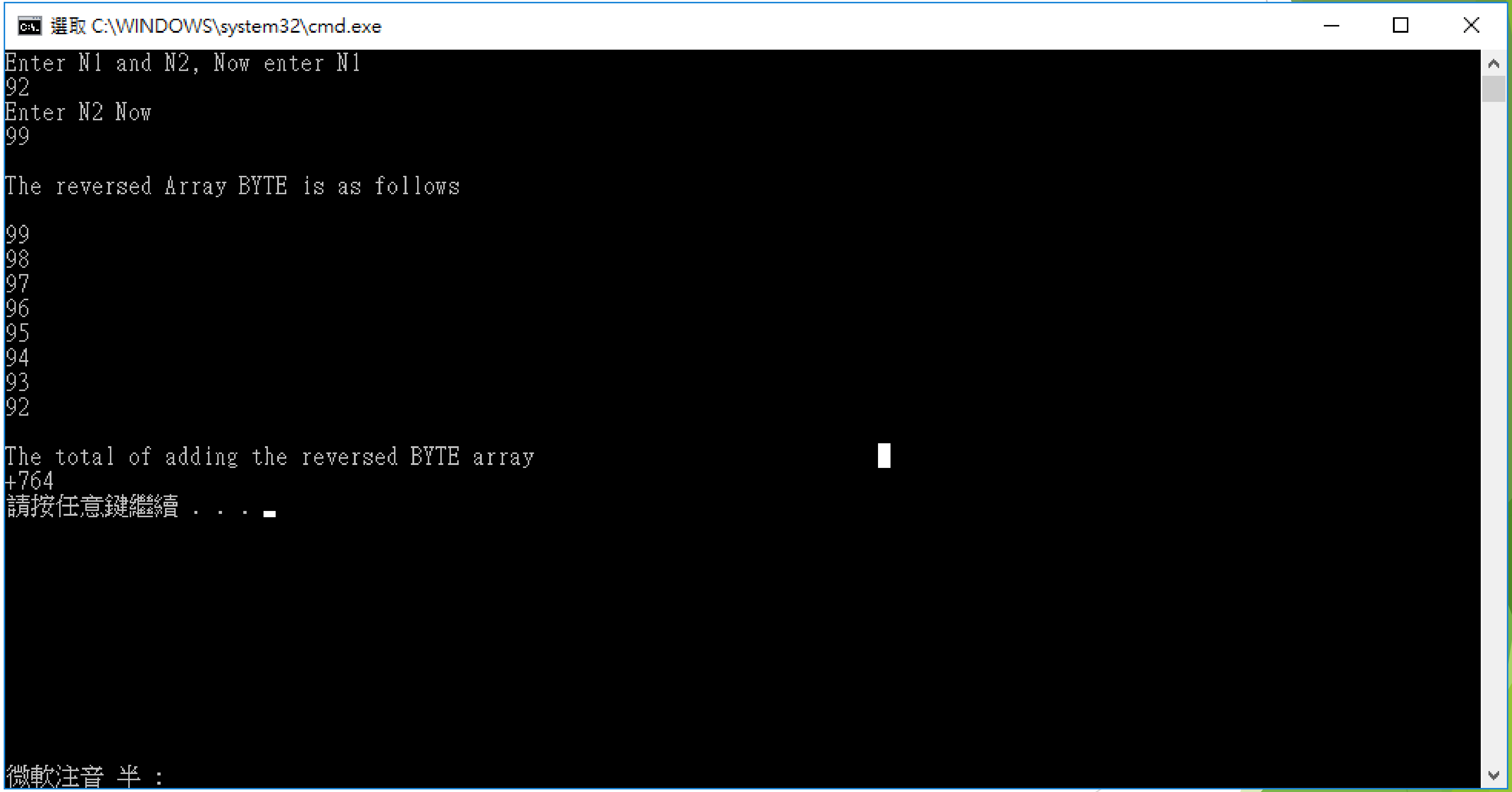
```
;Finally print the message and print out the reversed array one by one.
mov edx, OFFSET MSG2
call writestring
call crlf
mov esi, OFFSET reversearraybyte
mov ecx, runvalue
;             This loop tries to print out the reversed array one by one
L3 :
     mov al, [esi]
     add esi, TYPE arrayBYTE
     call writedec
     call crlf
   add total, eax
     loop l3
mov edx, OFFSET MSG3
call writestring
mov eax, total
call writeint
call crlf
exit
main ENDP
end main
```

Run of a program for ODD program



```
Enter N1 and N2, Now enter N1
56
Enter N2 Now
65

The reversed Array BYTE is as follows

65
64
63
62
61
60
59
58
57
56

The total of adding the reversed BYTE array
+605
請按任意鍵繼續 . . .
```

Another run of the program for ODD program



```
Enter N1 and N2, Now enter N1
92
Enter N2 Now
99

The reversed Array BYTE is as follows

99
98
97
96
95
94
93
92

The total of adding the reversed BYTE array
+764
請按任意鍵繼續 . . .
```

EVEN Seat numbered students Program on the next Page.

EVEN Number seats write the following program in x86 Assembly language

- Ask the user to enter two numbers, N1 and N2. Store the numbers in starting from N1 to N2 in a WORD Array called ARRAYWORD. Now reverse the array and write it in an another array called REVERSEWORD. ADD the total and display the result.
- Ex: N1= 27, N2=43.
- ARRAYWORD WORD 200 DUP(0)   // Declare this array WORD addressed.
- REVERSEWORD WORD 200 DUP(0) // This also must be an array of WORD.
- Use a loop (1) to write as follows:
- ARRAYWORD {27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43}
- Use a loop (2) to reverse the array:
- Reverse the array and write it in REVERSEWORD.
- Use a loop (3) to print out the reversed array and finally the total.
- You must use 3 loop separate loops.

Use the following DATA section for EVEN program

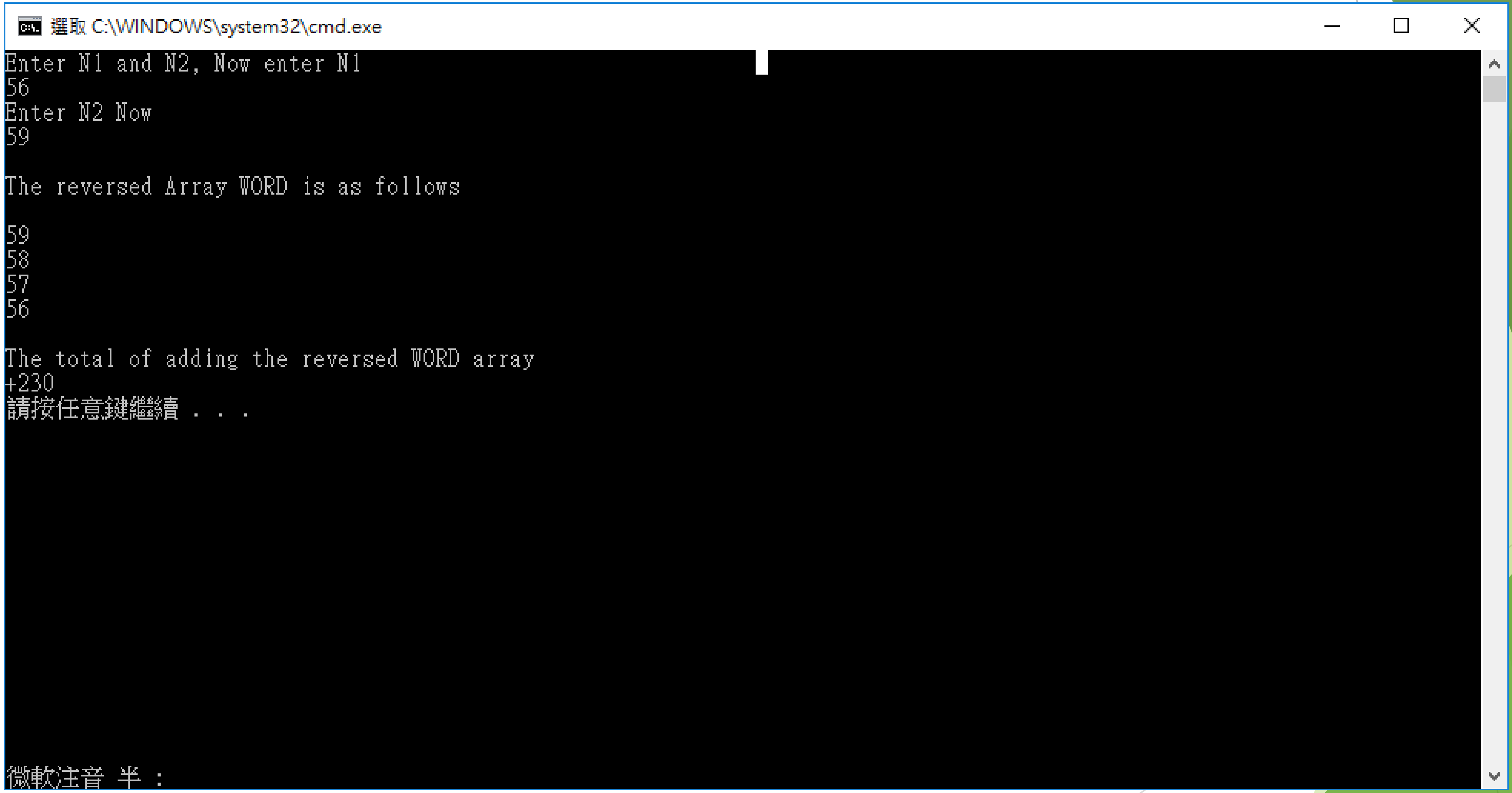
```
.data
arrayWORD WORD 200 DUP(0)
count EQU(LENGTHOF arrayWORD)
reversearrayWORD WORD count DUP(0)
N1 DWORD 0
N2 DWORD 0
runvalue DWORD 0
Total DWORD 0
MSG0 BYTE "Enter N1 and N2, Now enter N1", 0dh, 0ah, 0
MSG1 BYTE "Enter N2 Now", 0dh, 0ah, 0
MSG2 BYTE "The reversed Array WORD is as follows", 0dh, 0ah, 0
MSG3 BYTE "The total of adding the reversed WORD array", 0dh, 0ah, 0
```

```
.code
main PROC

mov edx, OFFSET MSG0   ; move the string ask the user to INPUT N1 and N2
call writestring                       ; write the string to display.
call readint                        ; read the N1 here.
mov N1, eax                       ; move to the Storage N1

mov edx, OFFSET MSG1     ; move the string ask the user to INPUT N1 and N2
call writestring          ; write the string to display.
call readint                   ; read the N2 here.
mov N2, eax                  ; move to the Storage N2

mov eax, N2          ; move N2 to calculate N2-N1
sub eax, N1           ; Minus the N1 value from N2
inc eax                      ; increment one to the index
mov runvalue, eax  ; move that into the runvalue
mov ecx, eax          ; move into ecx to loop through
mov esi, OFFSET arrayWORD       ; move the OFFSET to loop through the array
mov  eax, N1
```

```
; The first loop that reads in the number
L1:
mov[esi], ax
add esi, TYPE arrayWORD
inc eax
loop l1

mov ecx, runvalue
mov esi, 0
mov edi, runvalue
add edi, 2
; The second loop reverses the array.
L2:
mov ax, arrayWORD[edi]
mov reversearrayWORD[esi], ax
;call writedec
;call crlf
       sub edi, 2
     add esi, 2
   loop l2
```

```
call crlf
mov edx, OFFSET MSG2
call writestring
call crlf
mov esi, OFFSET reversearrayWORD
mov ecx, runvalue
; This loop prints the result on the screen one by one.
L3 :
      mov ax, [esi]
      add esi, TYPE arrayWORD
      call writedec
      call crlf
   add total, eax
      loop l3
call crlf
mov edx, OFFSET MSG3
call writestring
mov eax, total
call writeint
call crlf
exit
main ENDP
end main
```

SAMPLE RUN of the program for EVEN program



```
Enter N1 and N2, Now enter N1
56
Enter N2 Now
59

The reversed Array WORD is as follows

59
58
57
56

The total of adding the reversed WORD array
+230
請按任意鍵繼續 . . .
```

Good Luck to you for a successful completion of the code