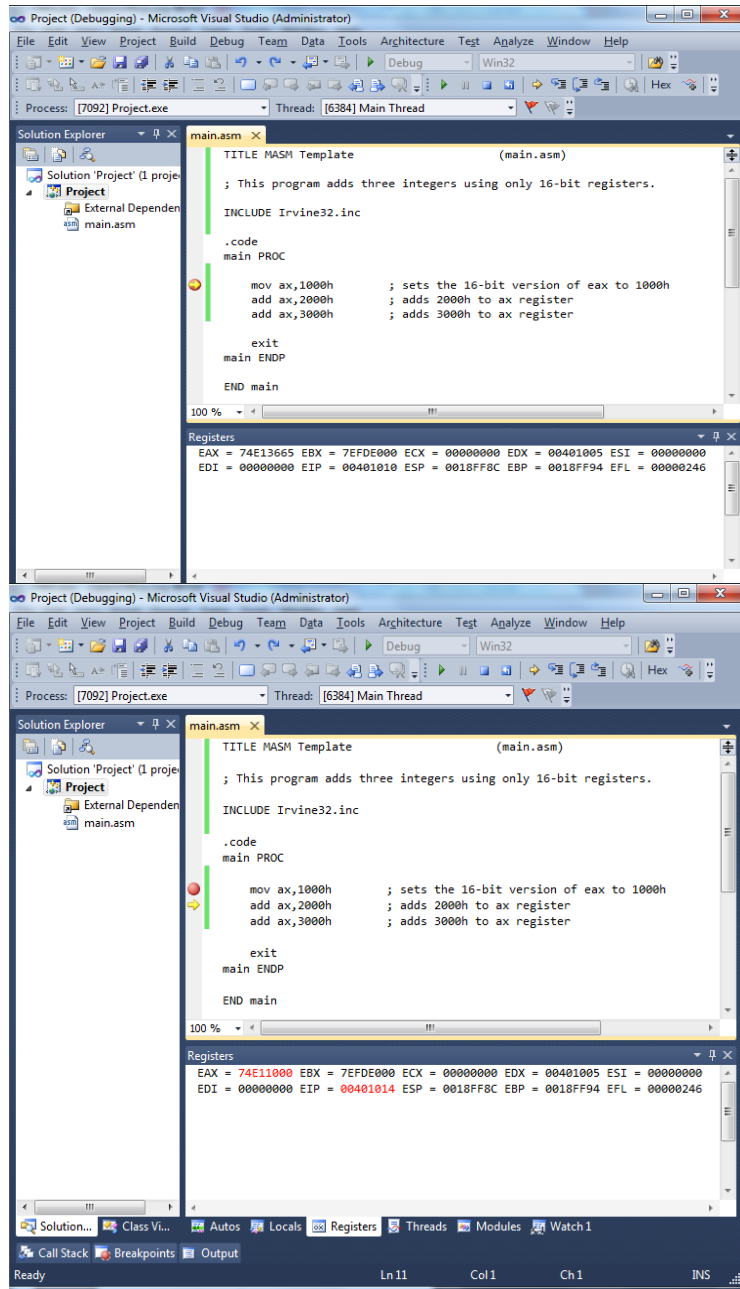
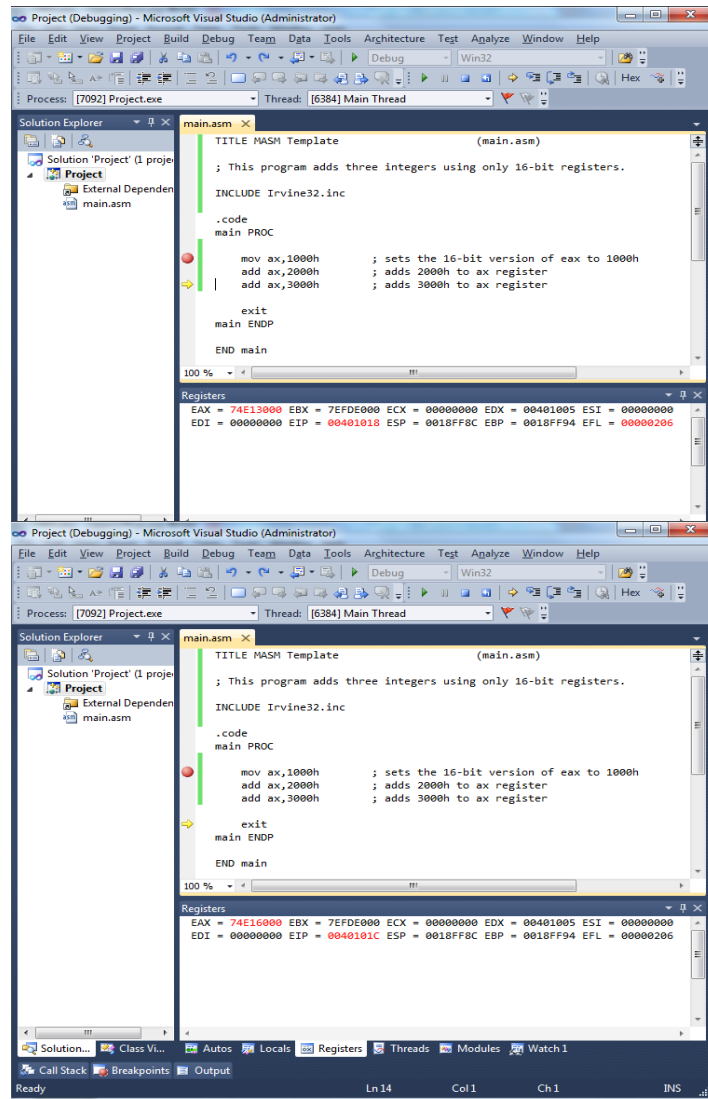


## HW # 3: Theme: Single Step, Flags, and Simple Instructions

1. Single step through the program created in the previous assignment (HW3). Submit a trace of screenshots single stepping through each line of the program.





2. Declare whether or not each flag is set ('1' for set, '0' for not set) following each instruction.

Instruction #	Instruction	CF	SF	ZF	OF
0	mov ax, 7F00h	0	0	0	0
1	add ax, 100h	0	1	0	1
2	add al, 0h	0	0	1	0
3	add al, 0FFh	0	1	0	1
4	sub ah, 080h	0	0	1	0

3. Given the following data declarations:

```
.data
Alpha  BYTE  50h, 33h
Beta   DWORD 400h
Delta  DWORD 600h
Iota   DWORD 800h
Zeta   WORD   200h
```

- A. Write an instruction that moves *Beta* into *EAX* and adds *Delta* to the same register  
***MOV EAX, [Beta]***  
***ADD EAX, [Delta]***
- B. Write a set of instructions that adds both elements of the array *Alpha* into *AL*  
***ADD AL, [Alpha]***  
***ADD AL, [Alpha + 1]***
- C. Write a set of instructions that moves *Iota* into *EAX*, adds the value stored in *Zeta* to the same register  
***MOV EAX, [Iota]***  
***ADD EAX, [Zeta]***
- D. Write an instruction that moves both bytes in *ALPHA* into *CX*.  
***MOV CH, [Alpha]***  
***MOV CL, [Alpha + 1]***
- E. What are the contents of *CX* subsequent to part D of this question?  
***CH = 0101 0000; CL = 0011 0011; CX = 0101 0000 0011 0011 = 5033h***

4. Fill in the requested register values after executions of the instructions:

```
.data
myBytes      BYTE  11h, 22h, 33h, 44h
myWords       WORD  5566h, 7788h, 8899h, AAB Bh, CCDDh
myDoubles     DWORD 0000EEFFh, 00000005h, 1020h, 3040h, 60h
myPointer     DWORD myDoubles

.code
mov esi, OFFSET myBytes
mov ax, WORD PTR [esi+2]           ; A.  AX = 4433h
mov eax, DWORD PTR myWords         ; B.  EAX = 77885566h
mov esi, myPointer
mov ax, WORD PTR [esi+8]           ; C.  AX = 1020h
mov ax, WORD PTR [esi+1]           ; D.  AX = 00EEh
mov ax, WORD PTR [esi-6]           ; E.  AX = 8899h
```

5. What is returned by the following operations in connection to the given array:  
*myArray DWORD 50 DUP(25), 1, 21, 321, 4321, 54321*

- A. TYPE myArray **DWORD = 4 bytes = 4**  
B. LENGTHOF myArray **50 + 1 + 1 + 1 + 1 + 1 = 55**  
C. SIZEOF myArray **55 \* 4 = 220**

6. Fill in the requested register values after executions of the instructions (Do not let your eyes deceive you. There are some movSx instructions and some movZx instructions.):

```
.code
mov bx, 0A703h
movzx eax, bx           ; A.  EAX = 0000A703h
movzx edx, bl           ; B.  EDX = 00000003h
movzx cx, bh            ; C.  CX = 000000A7h

mov bx, B574h
movsx eax, bx           ; D.  EAX = FFFFB574h
movsx edx, bl           ; E.  EDX = 00000074h
movsx cx, bh            ; F.  CX = FFB5h
```

**Source code for #1:**

```
TITLE MASM Template                                     (main.asm)

; This program adds three integers using only 16-bit registers.

INCLUDE Irvine32.inc

.code
main PROC

    mov ax,1000h      ; sets the 16-bit version of eax to 1000h
    add ax,2000h      ; adds 2000h to ax register
    add ax,3000h      ; adds 3000h to ax register

    exit
main ENDP

END main
```