**COAL LAB 03 - 13 Feb 2025**

**Student Name:** Ibrahim Johar Farooqi

**Student ID:** 23K-0074

**Task 1:**

include irvine32.inc

.data

val1 SWORD ? ; 16 bit signed integer (not initialized)

val2 SBYTE -11 ; 8 bit signed integer initialized w/ -11

.code

main PROC

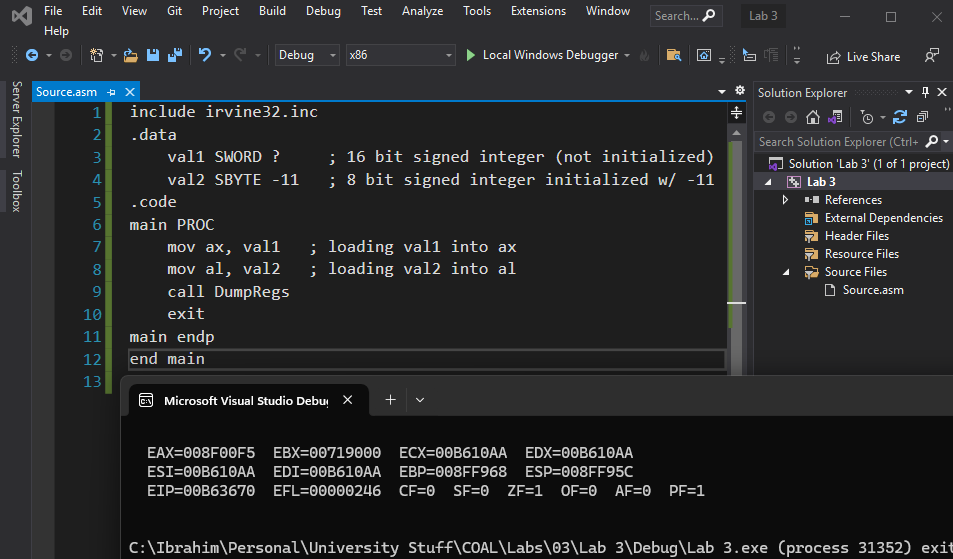
mov ax, val1 ; loading val1 into ax

mov al, val2 ; loading val2 into al

call DumpRegs

exit

main endp

end main

**Task 2:**

include irvine32.inc

.data

val3 SDWORD -2147483648 ; smallest 32bit signed integer

.code

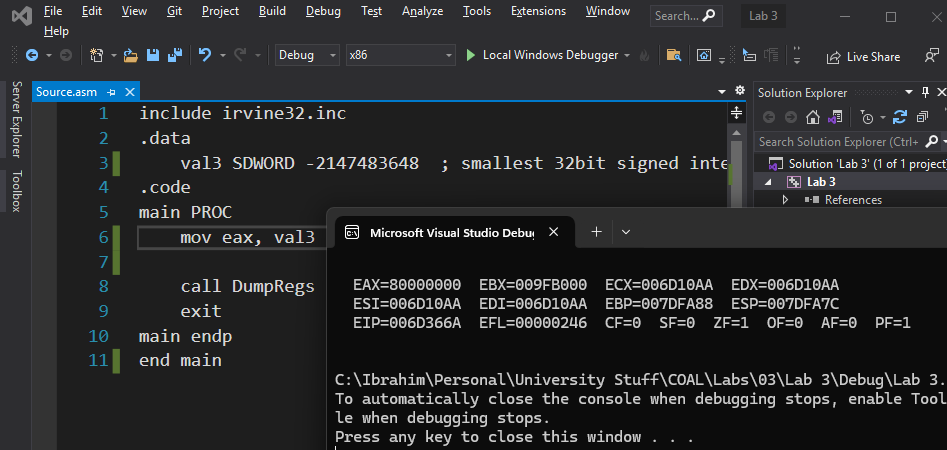
main PROC

mov eax, val3

call DumpRegs

exit

main endp

end main

**Task 3:**

include irvine32.inc

.data

val4 WORD 5, 25, 75

.code

main PROC

mov ax, val4

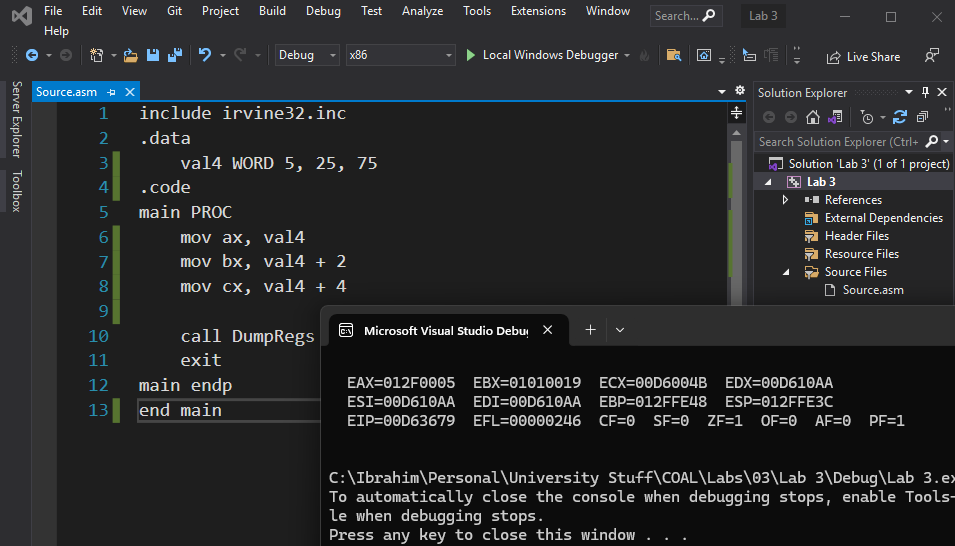
mov bx, val4 + 2

mov cx, val4 + 4

call DumpRegs

exit

main endp

end main

**Task 4:**

include irvine32.inc

.data

favcolor BYTE "Red", 0

A WORD 12

B WORD 2

C\_VAL WORD 13

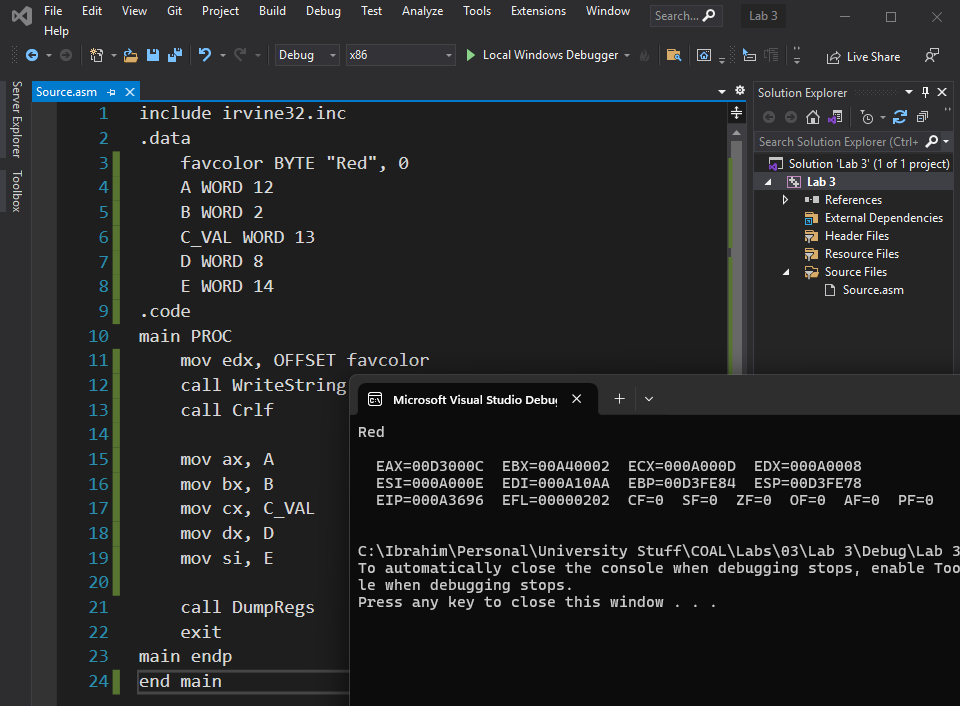
D WORD 8

E WORD 14

.code

main PROC

mov edx, OFFSET favcolor

call WriteString

call Crlf

mov ax, A

mov bx, B

mov cx, C\_VAL

mov dx, D

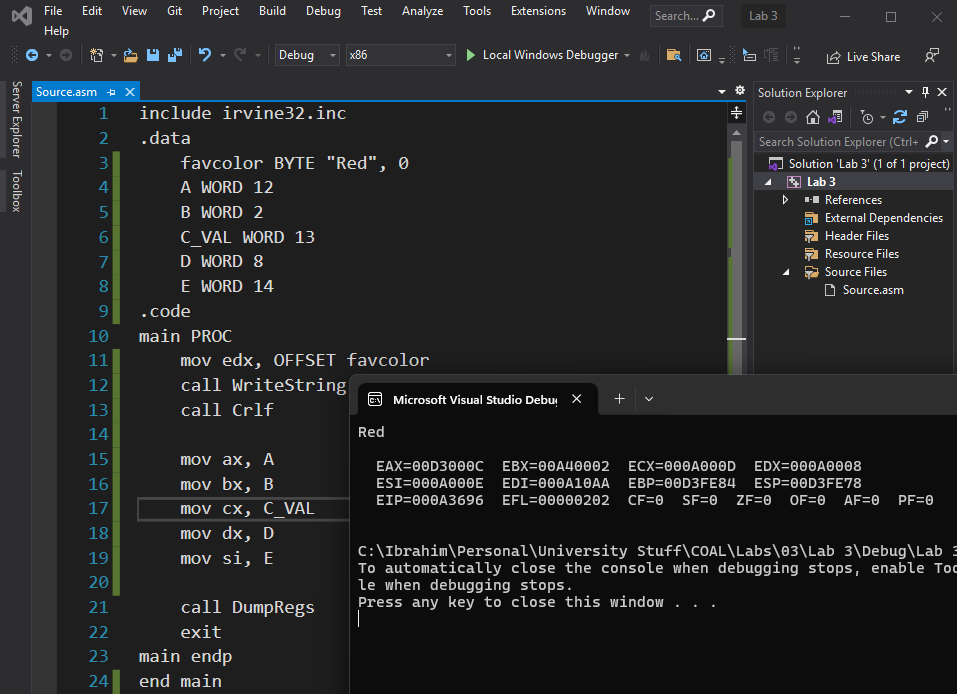
mov si, E

call DumpRegs

exit

main endp

end main



**Task 5:**

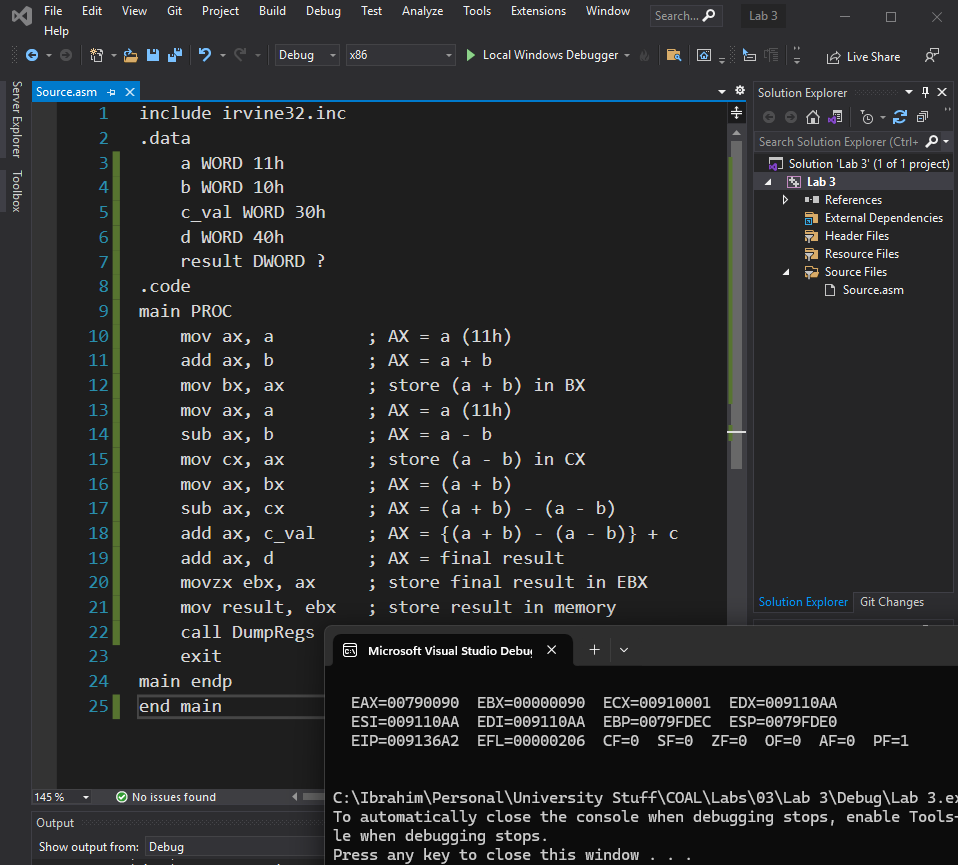
include irvine32.inc

.data

a WORD 11h

b WORD 10h

c\_val WORD 30h

d WORD 40h 

result DWORD ?

.code

main PROC

mov ax, a ; AX = a (11h)

add ax, b ; AX = a + b

mov bx, ax ; store (a + b) in BX

mov ax, a ; AX = a (11h)

sub ax, b ; AX = a - b

mov cx, ax ; store (a - b) in CX

mov ax, bx ; AX = (a + b)

sub ax, cx ; AX = (a + b) - (a - b)

add ax, c\_val ; AX = {(a + b) - (a - b)} + c

add ax, d ; AX = final result

movzx ebx, ax ; store final result in EBX

mov result, ebx ; store result in memory

call DumpRegs

exit

main endp

end main

**Task 6:**

include irvine32.inc

.data

a\_ DWORD 00010001b

b\_ DWORD 00010000b

c\_ DWORD 00110000b

d\_ DWORD 01000000b

.code

main PROC

mov ebx, a\_

mov eax, b\_

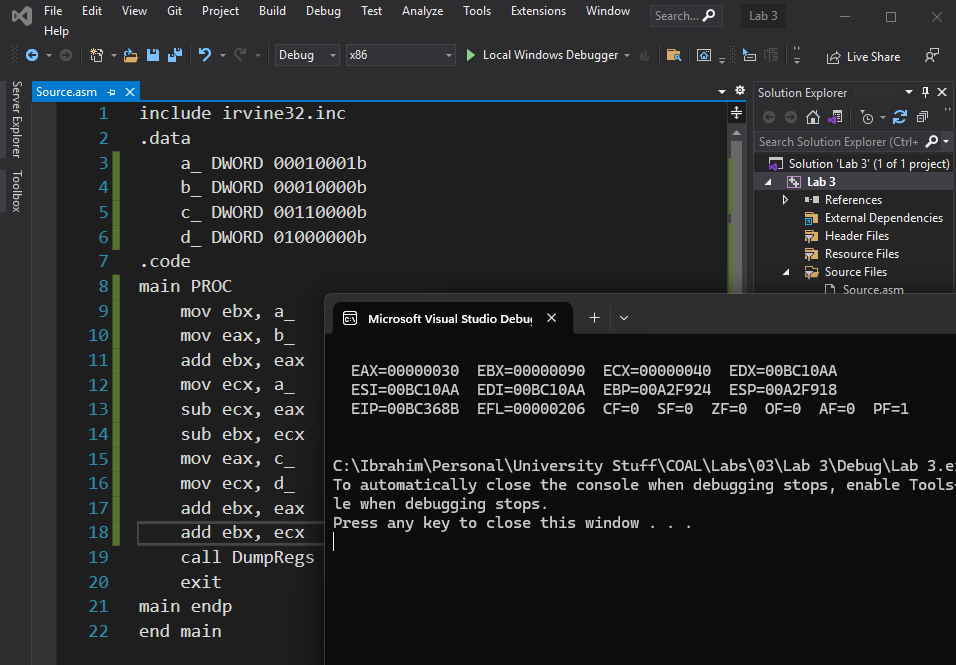
add ebx, eax

mov ecx, a\_

sub ecx, eax

sub ebx, ecx

mov eax, c\_

mov ecx, d\_

add ebx, eax

add ebx, ecx

call DumpRegs

exit

main endp

end main

**Task 7:**

include irvine32.inc

.data

wArray WORD 1, 2, 3

.code

main PROC

mov ax, wArray

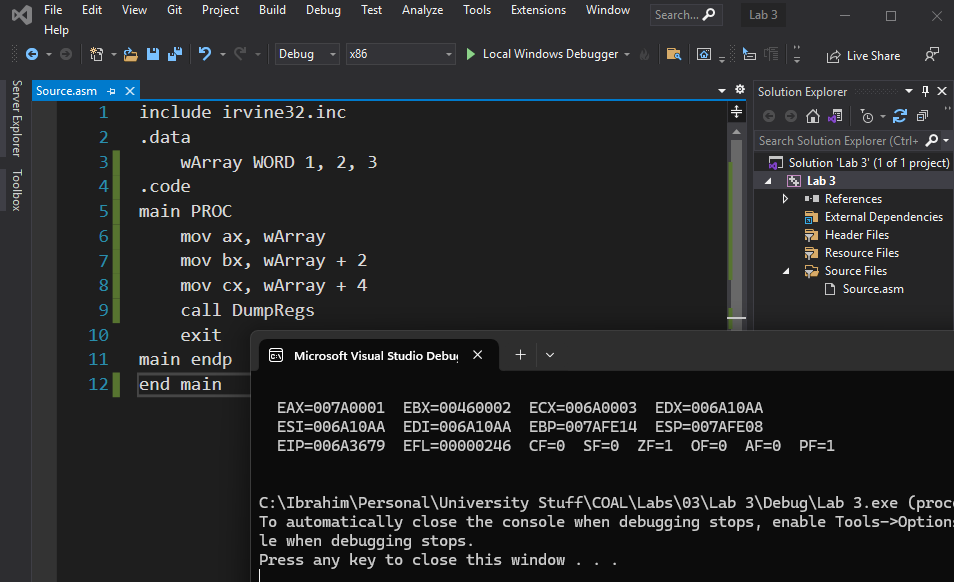
mov bx, wArray + 2

mov cx, wArray + 4

call DumpRegs

exit

main endp

end main

**Task 8:**

include irvine32.inc

.data

dArray DWORD 50 DUP(?)

.code

main PROC

call DumpRegs

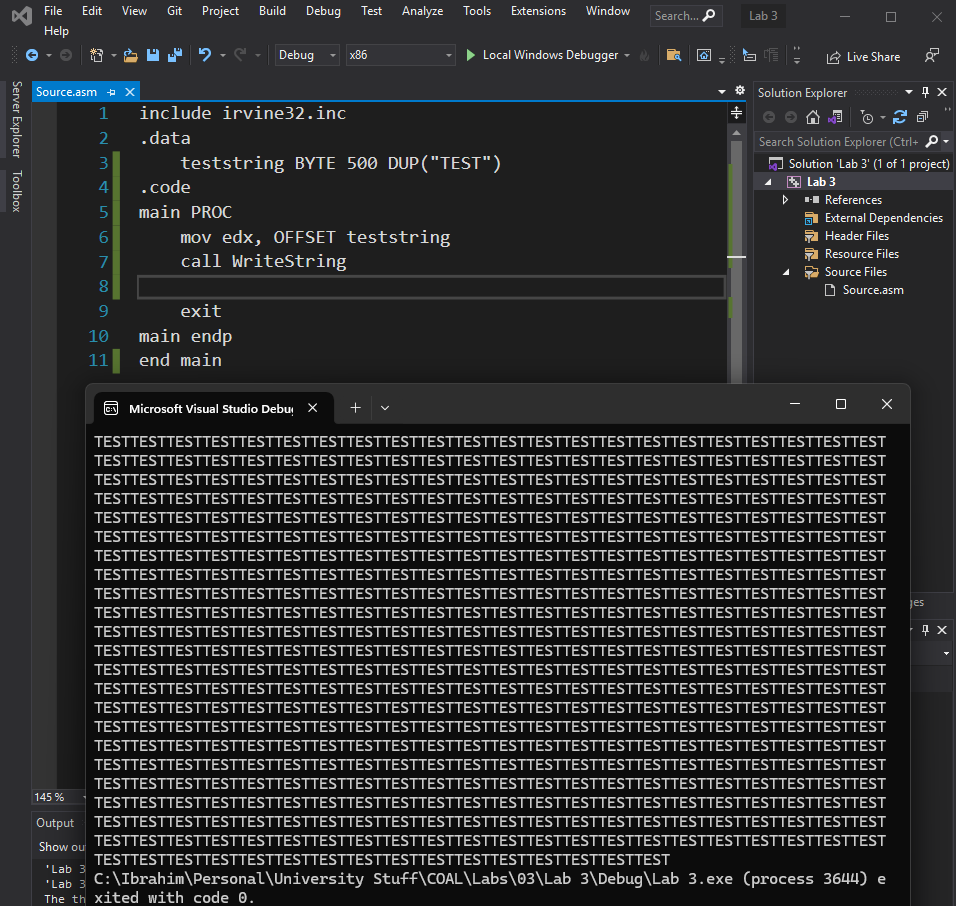
exit

main endp

end main

**Task 9:**

include irvine32.inc

.data 

teststring BYTE 500 DUP("TEST")

.code

main PROC

mov edx, OFFSET teststring

call WriteString

exit

main endp

end main

**Task 10:**

include irvine32.inc

.data

bArray BYTE 20 DUP(0)

.code

main PROC

call DumpRegs

exit

main endp

end main

**Activity 1:**

include irvine32.inc

.data

val BYTE 10

.code

main proc

mov eax,0

mov al,val

add al,40

call writeint

exit

main endp

end main