;Q1

INCLUDE Irvine32.inc

.data

val1 SWORD ?

val2 SBYTE -10

.code

main PROC

mov eax,0

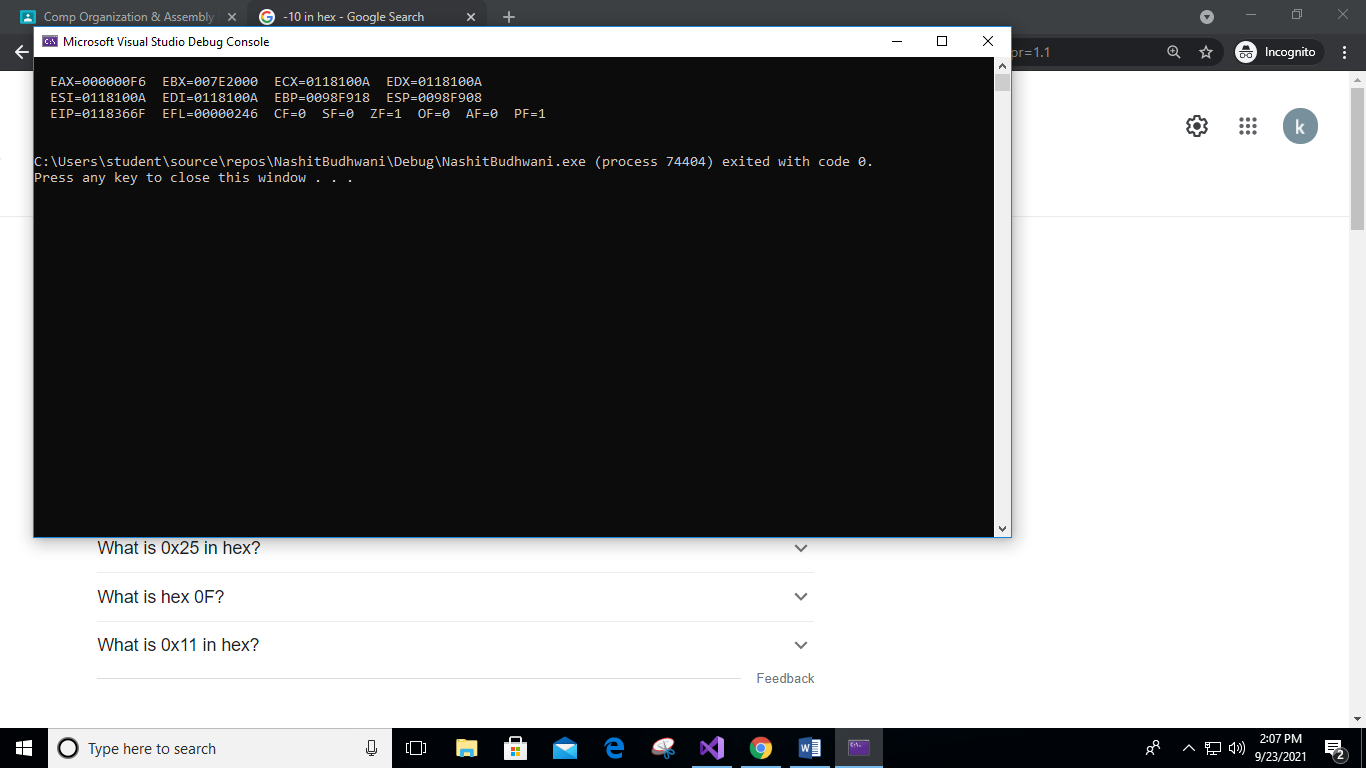
mov al,val2

call DumpRegs ; display registers

exit

main ENDP

END main



;Q2

INCLUDE Irvine32.inc

.data

val3 SDWORD -2147483648

.code

main PROC

mov eax,0

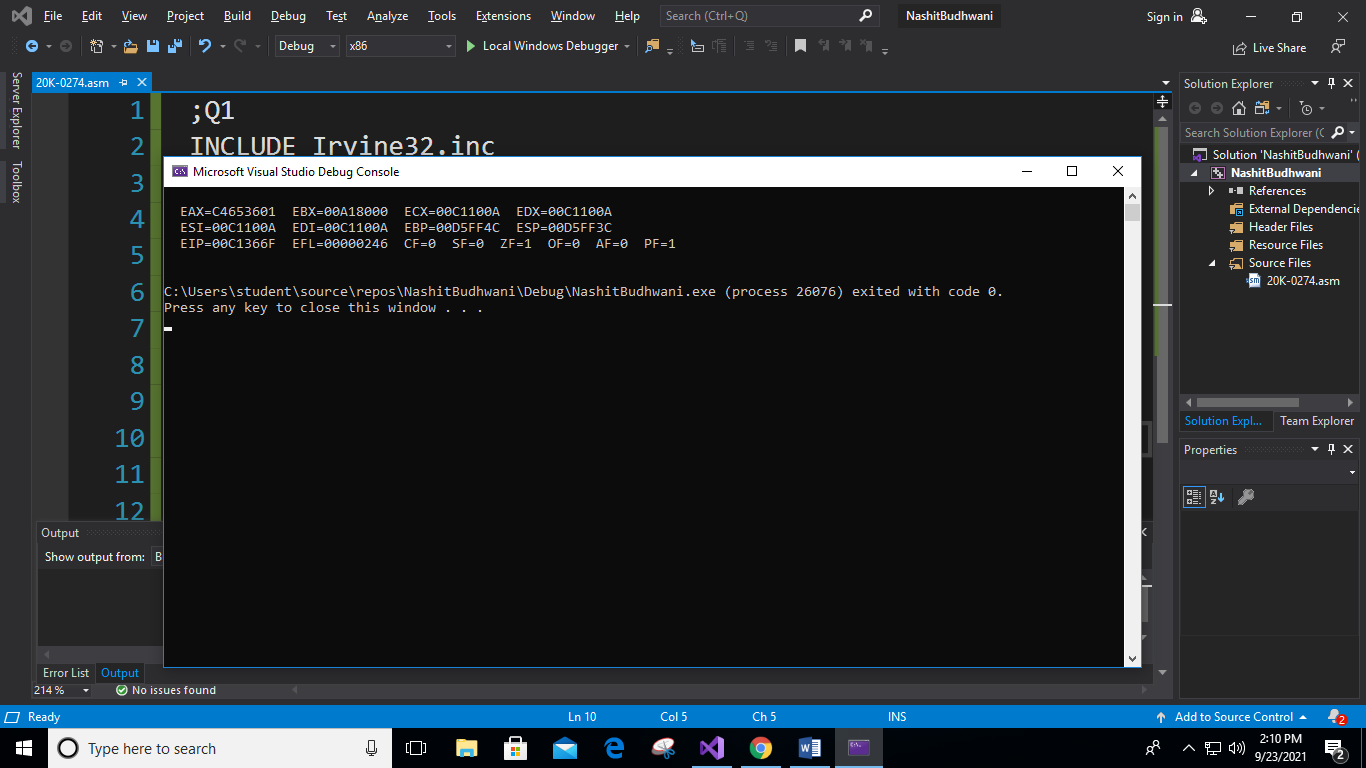
mov eax,val3

call DumpRegs ; display registers

exit

main ENDP

END main



;Q3

INCLUDE Irvine32.inc

.data

wArray WORD 0,1,2

.code

main PROC

mov eax,0

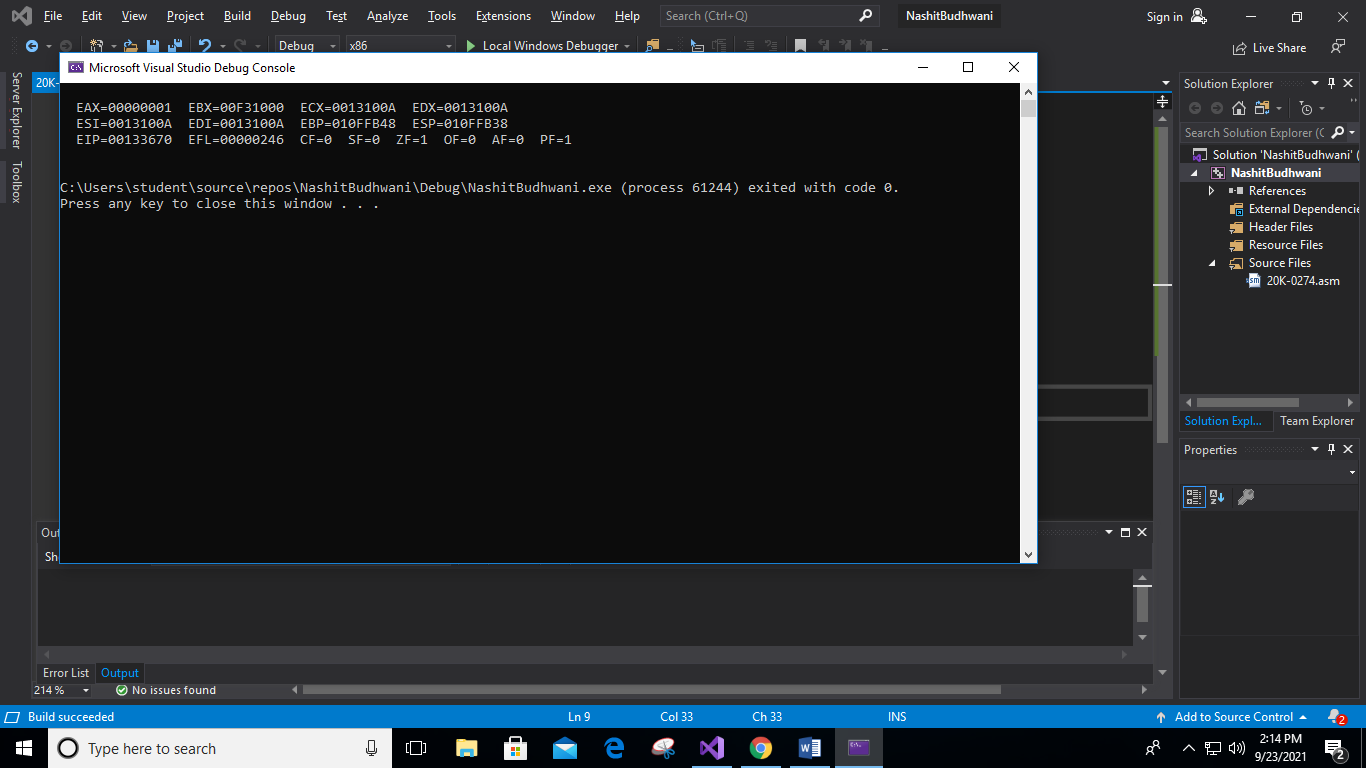
mov ax,wArray[2];displaying 1st index of array

call DumpRegs ; display registers

exit

main ENDP

END main



;Q4

INCLUDE Irvine32.inc

.data

string BYTE 'B','L','A','C','K','0'

.code

main PROC

mov eax,0

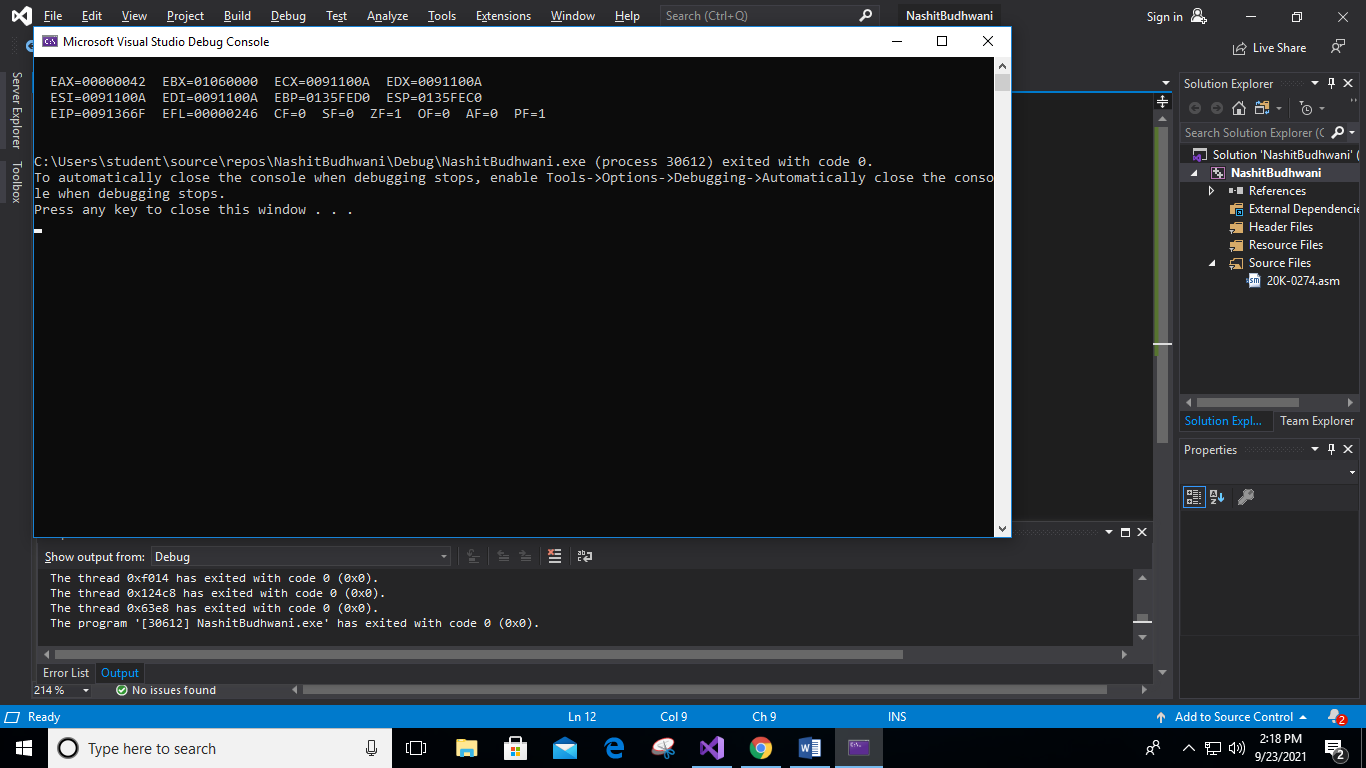
mov al,string

call DumpRegs ; display registers

exit

main ENDP

END main



;Q6

INCLUDE Irvine32.inc

.data

A WORD 12

B WORD 2

F WORD 13 ;instead of C intialized F

D WORD 8

E WORD 14

.code

main PROC

mov eax,0

mov ebx,0

mov ecx,0

mov edx,0

mov ax,A

mov bx,B

mov cx,F ;instead of C intialized F

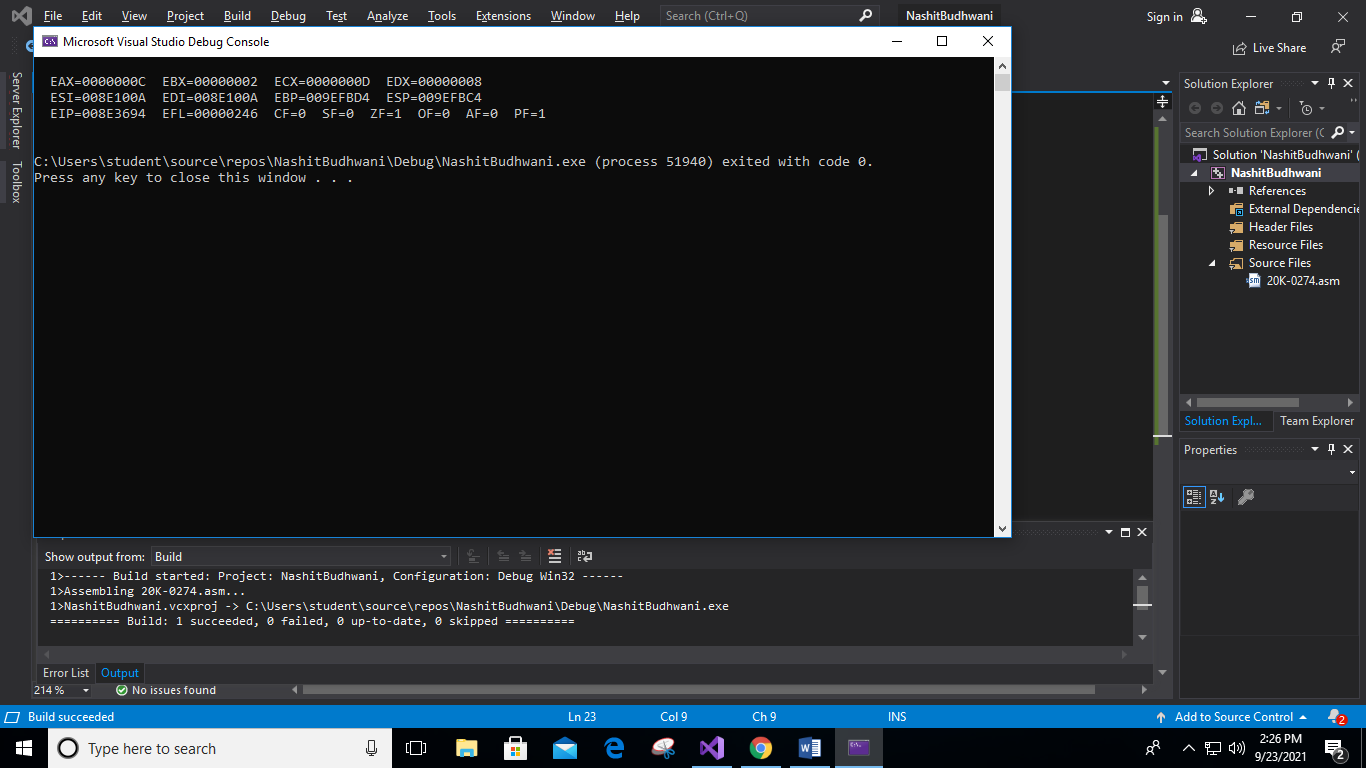
mov dx,D

call DumpRegs ; display registers

exit

main ENDP

END main



;Q5

INCLUDE Irvine32.inc

.data

x DWORD 10h

y DWORD 15h

z DWORD 30h

w DWORD 20h

.code

main PROC

mov ebx,0

mov ecx,0

mov ebx,x

add ebx,y

mov ecx,x

sub ecx,y

sub eax,ecx

add eax,w

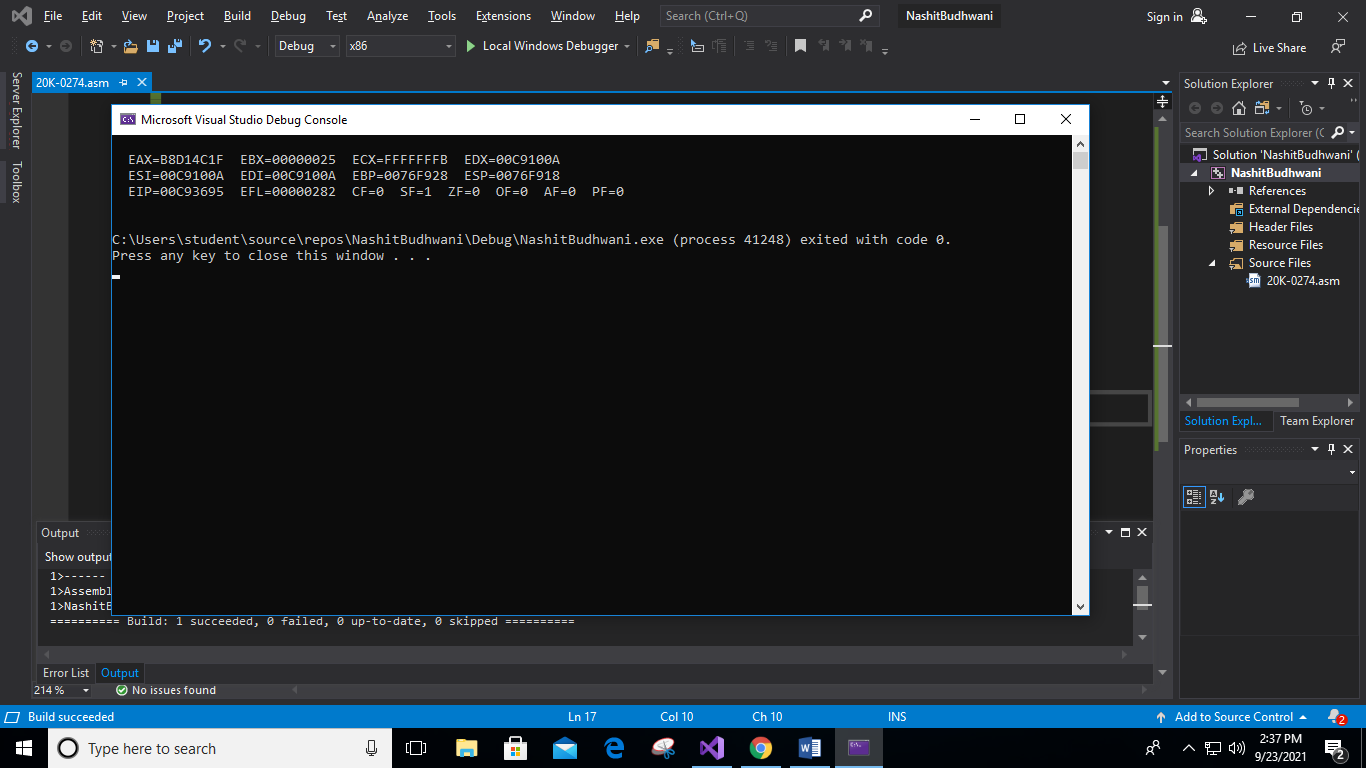
add eax,z

call DumpRegs ; display registers

exit

main ENDP

END main



;Q6

INCLUDE Irvine32.inc

.data

x DWORD 10000b

y DWORD 10101b

z DWORD 110000b

w DWORD 100000b

.code

main PROC

mov ebx,0

mov ecx,0

mov ebx,x

add ebx,y

mov ecx,x

sub ecx,y

sub eax,ecx

add eax,w

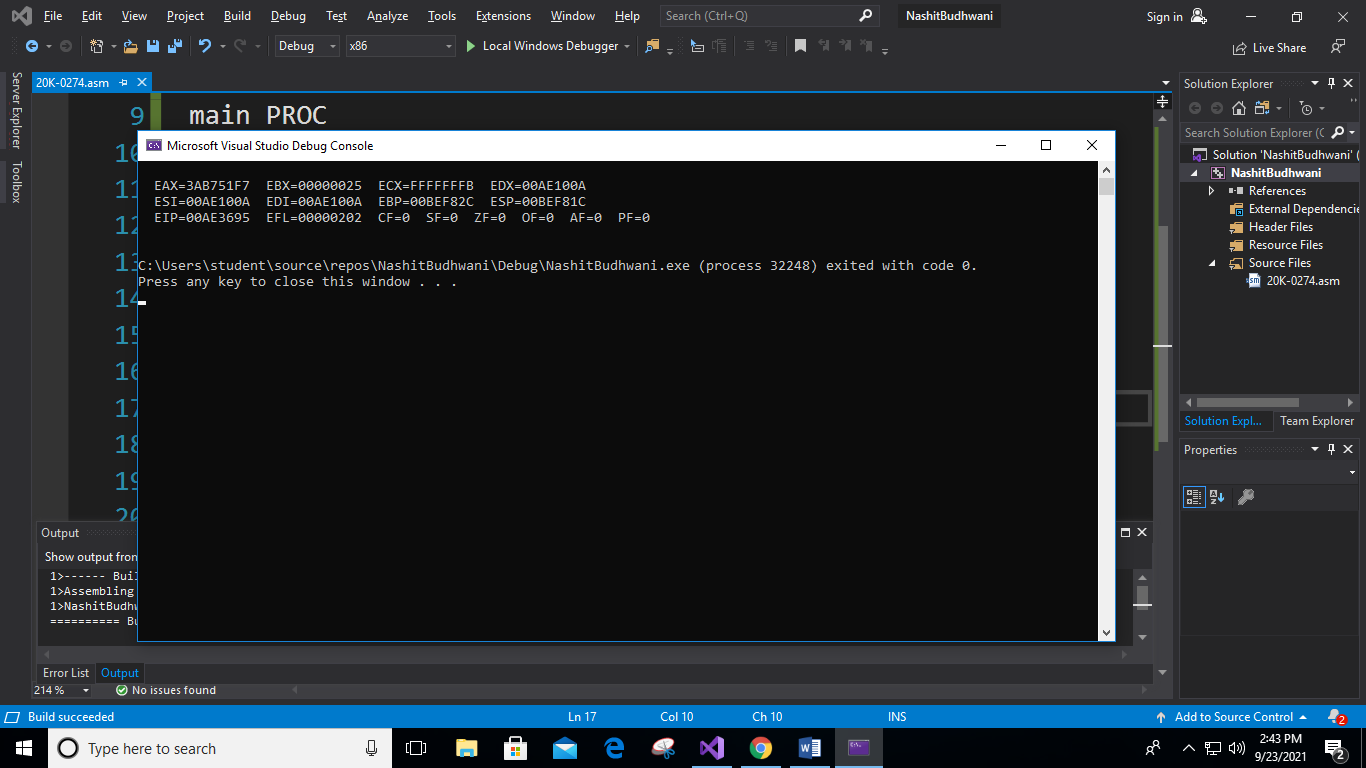
add eax,z

call DumpRegs ; display registers

exit

main ENDP

END main



;Q7

INCLUDE Irvine32.inc

.data

imm8 WORD 20

DATA1 WORD 8

DATA2 WORD 15

DATA3 WORD 20

.code

main PROC

mov eax,0

mov ax,imm8

add ax,DATA1

sub ax,DATA3

add ax,imm8

add ax,DATA2

call DumpRegs ; display registers

exit

main ENDP

END main

