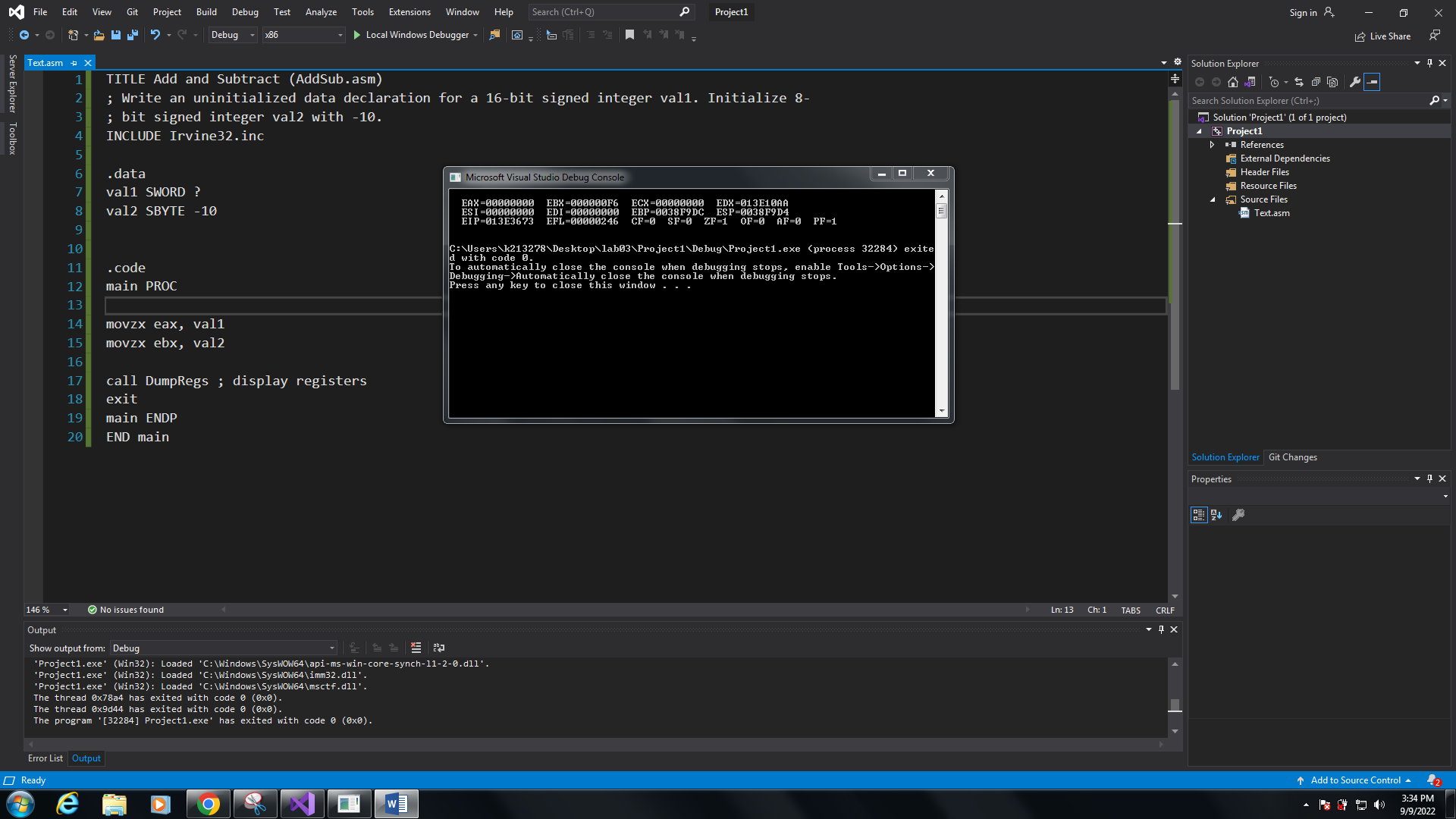
Q1.



TITLE Add and Subtract (AddSub.asm)

; Write an uninitialized data declaration for a 16-bit signed integer val1. Initialize 8-

; bit signed integer val2 with -10.

INCLUDE Irvine32.inc

.data

val1 SWORD ?

val2 SBYTE -10

.code

main PROC

movzx eax, val1

movzx ebx, val2

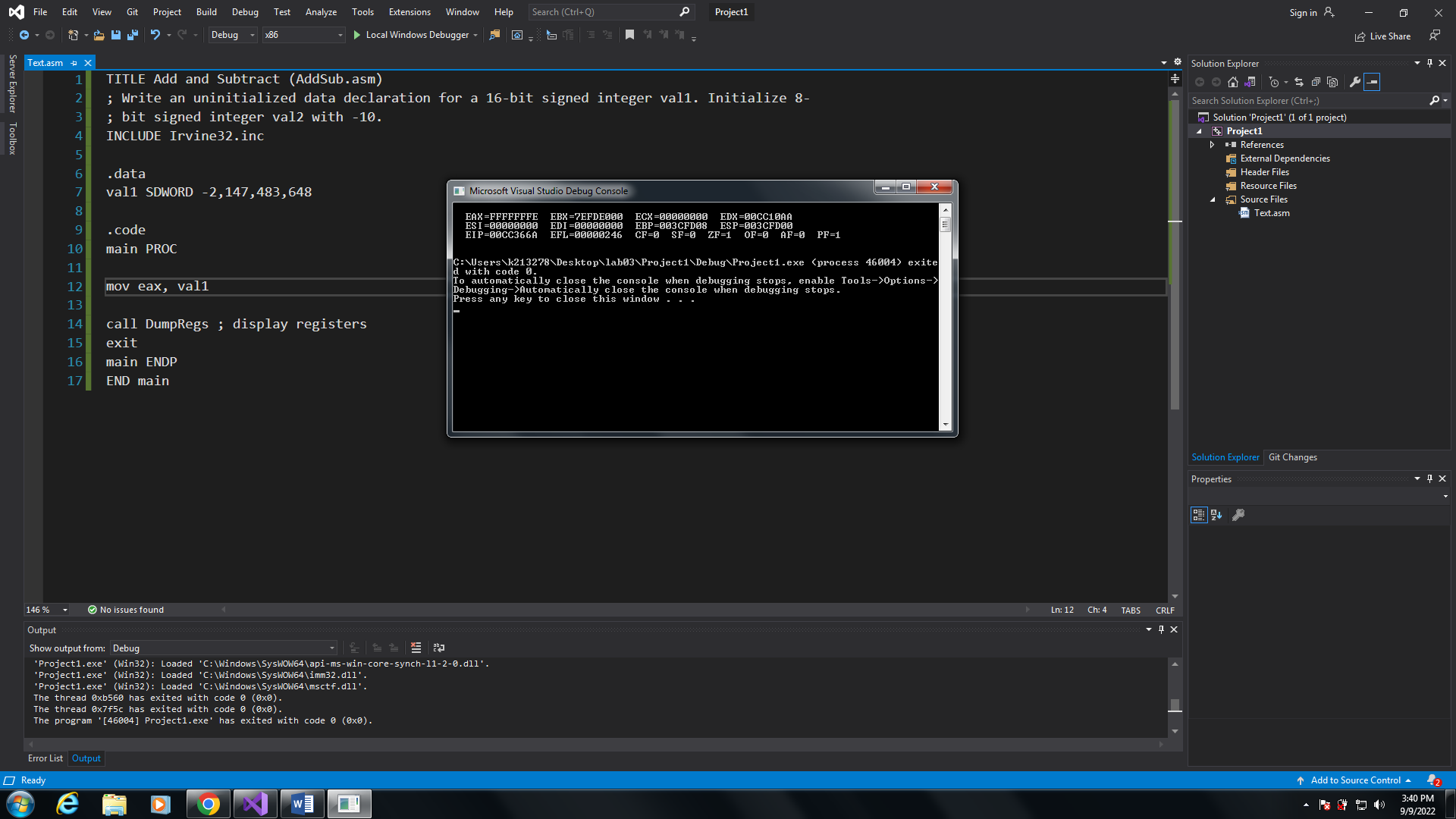
call DumpRegs ; display registers

exit

main ENDP

END main

Q2.



TITLE Add and Subtract (AddSub.asm)

; Write an uninitialized data declaration for a 16-bit signed integer val1. Initialize 8-

; bit signed integer val2 with -10.

INCLUDE Irvine32.inc

.data

val1 SDWORD -2,147,483,648

.code

main PROC

mov eax, val1

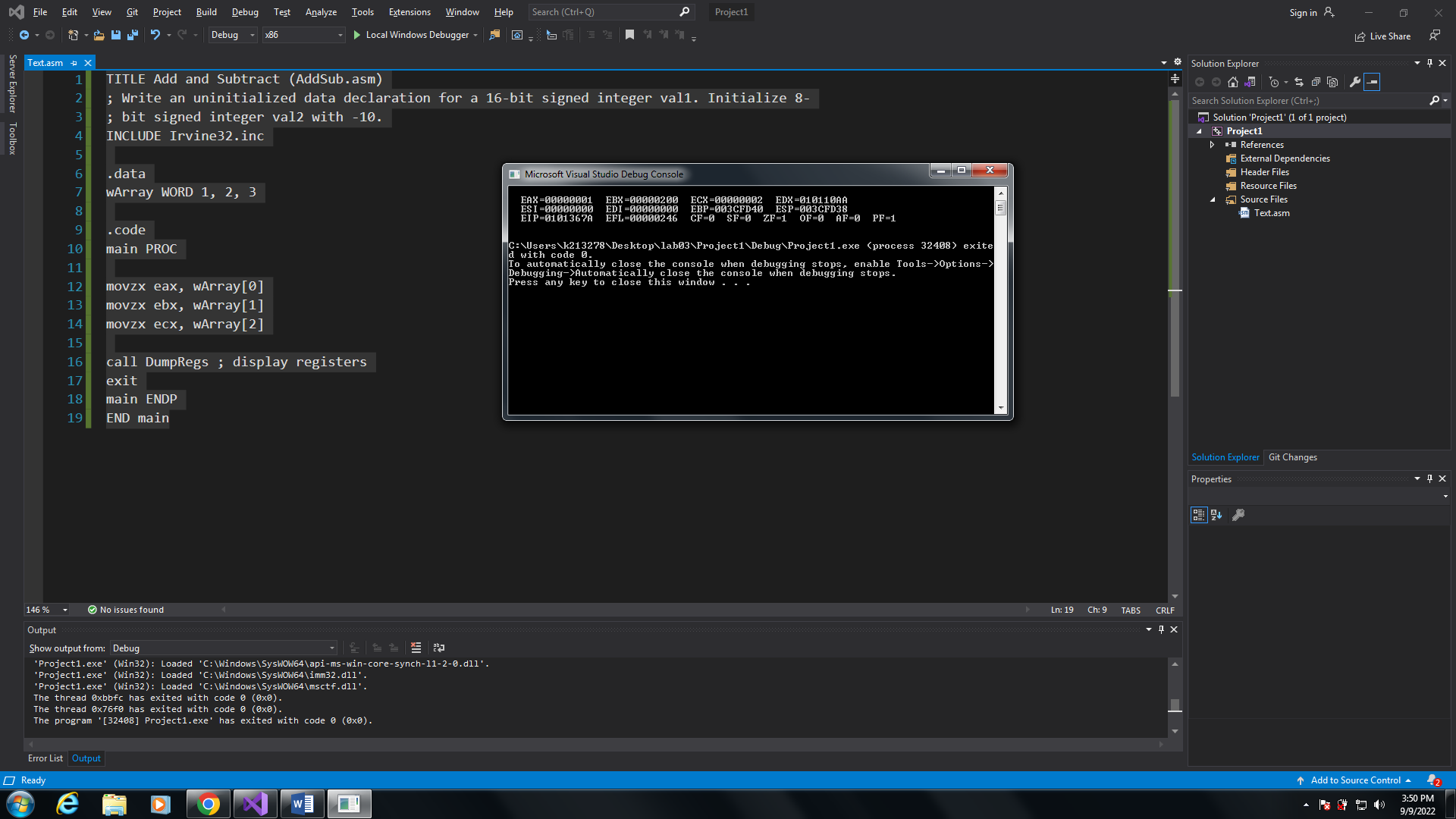
call DumpRegs ; display registers

exit

main ENDP

END main

Q3.



TITLE Add and Subtract (AddSub.asm)

; Write an uninitialized data declaration for a 16-bit signed integer val1. Initialize 8-

; bit signed integer val2 with -10.

INCLUDE Irvine32.inc

.data

wArray WORD 1, 2, 3

.code

main PROC

movzx eax, wArray[0]

movzx ebx, wArray[1]

movzx ecx, wArray[2]

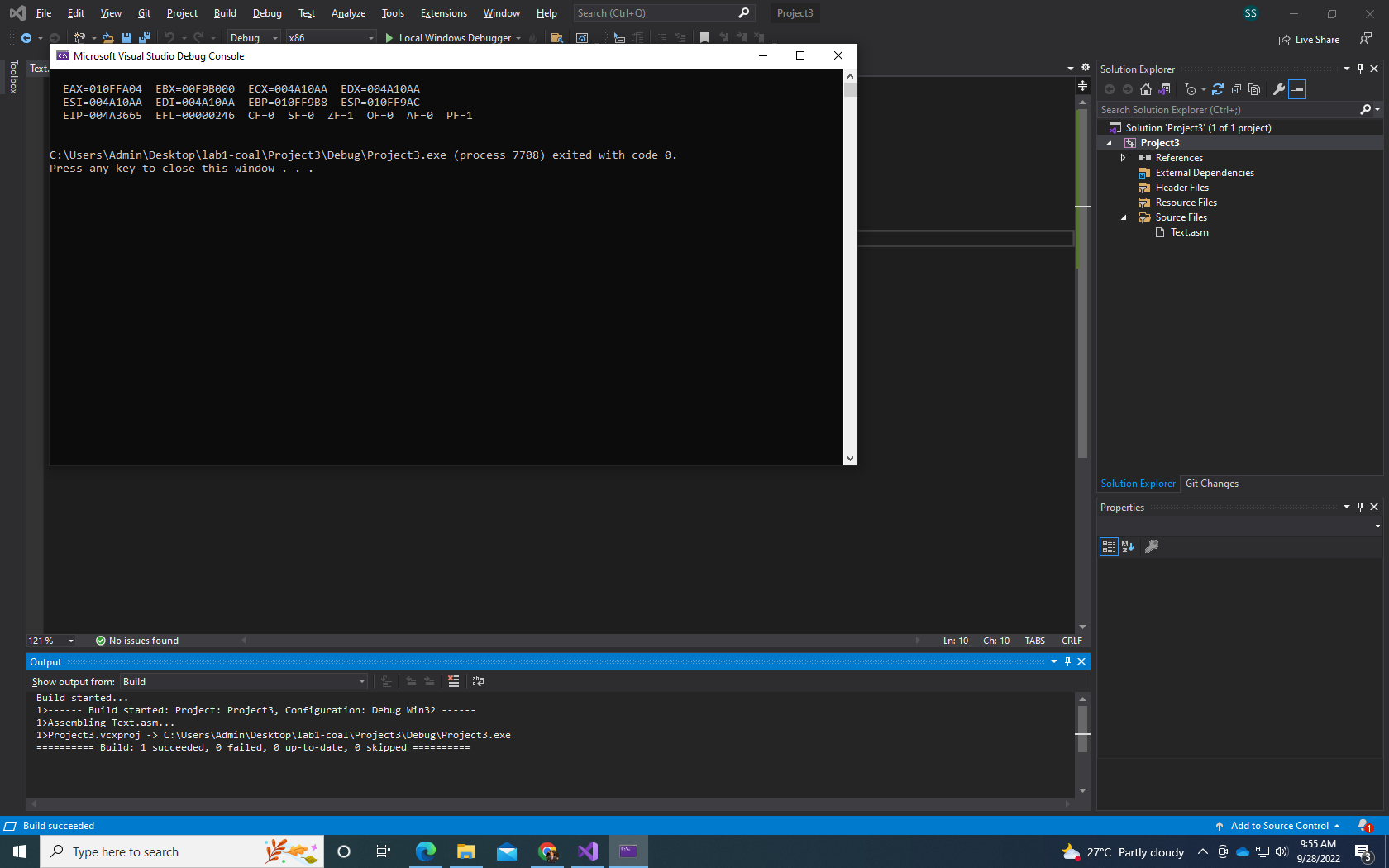
call DumpRegs ; display registers

exit

main ENDP

END main

Q4.



Include Irvine32.inc

.data

Color\_str\_var BYTE "BLUE", 0

var1 WORD 12

var2 WORD 2

var3 WORD 13

var4 WORD 8

var5 WORD 14

.code

main PROC

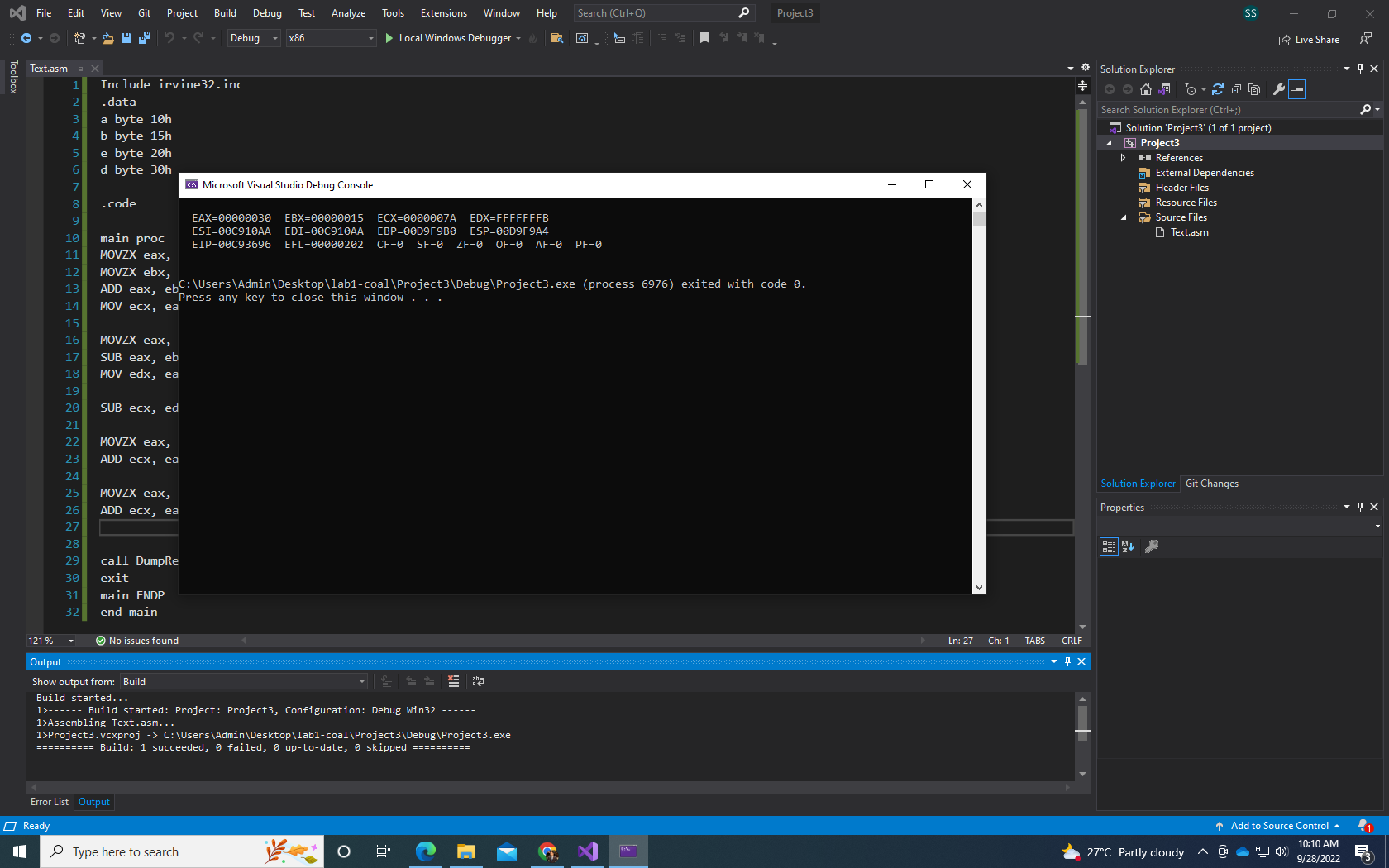
call DumpRegs

exit

main ENDP

end main

Q5.



Include irvine32.inc

.data

a byte 10h

b byte 15h

e byte 20h

d byte 30h

.code

main proc

MOVZX eax, a

MOVZX ebx, b

ADD eax, ebx

MOV ecx, eax ; ecx = a+b

MOVZX eax, a

SUB eax, ebx

MOV edx, eax ; edx = a-b

SUB ecx, edx; ; ecx = (a+b) - (a-b)

MOVZX eax, e

ADD ecx, eax

MOVZX eax, d

ADD ecx, eax ; final ans in ecx

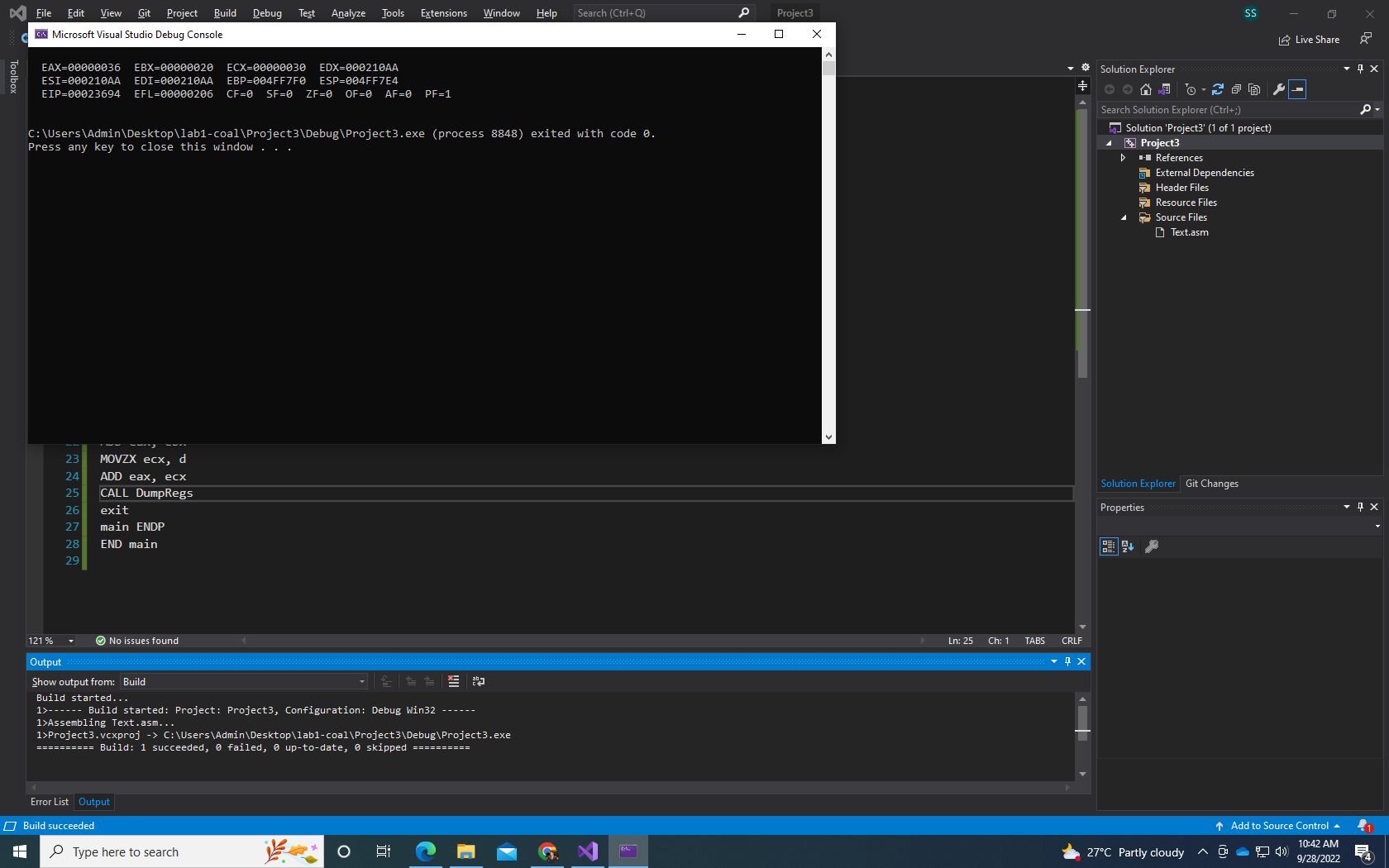
call DumpRegs

exit

main ENDP

end main

Q6.



Include irvine32.inc

.data

a byte 00010000b

b byte 00010101b

e byte 00100000b

d byte 00110000b

.code

main proc

MOVZX eax, a

MOVZX ebx, b

ADD eax, ebx

MOV ecx, eax

MOVZX eax, a

SUB eax, ebx

SUB eax, ebx

MOVZX ebx, e

ADD eax, ebx

MOVZX ecx, d

ADD eax, ecx

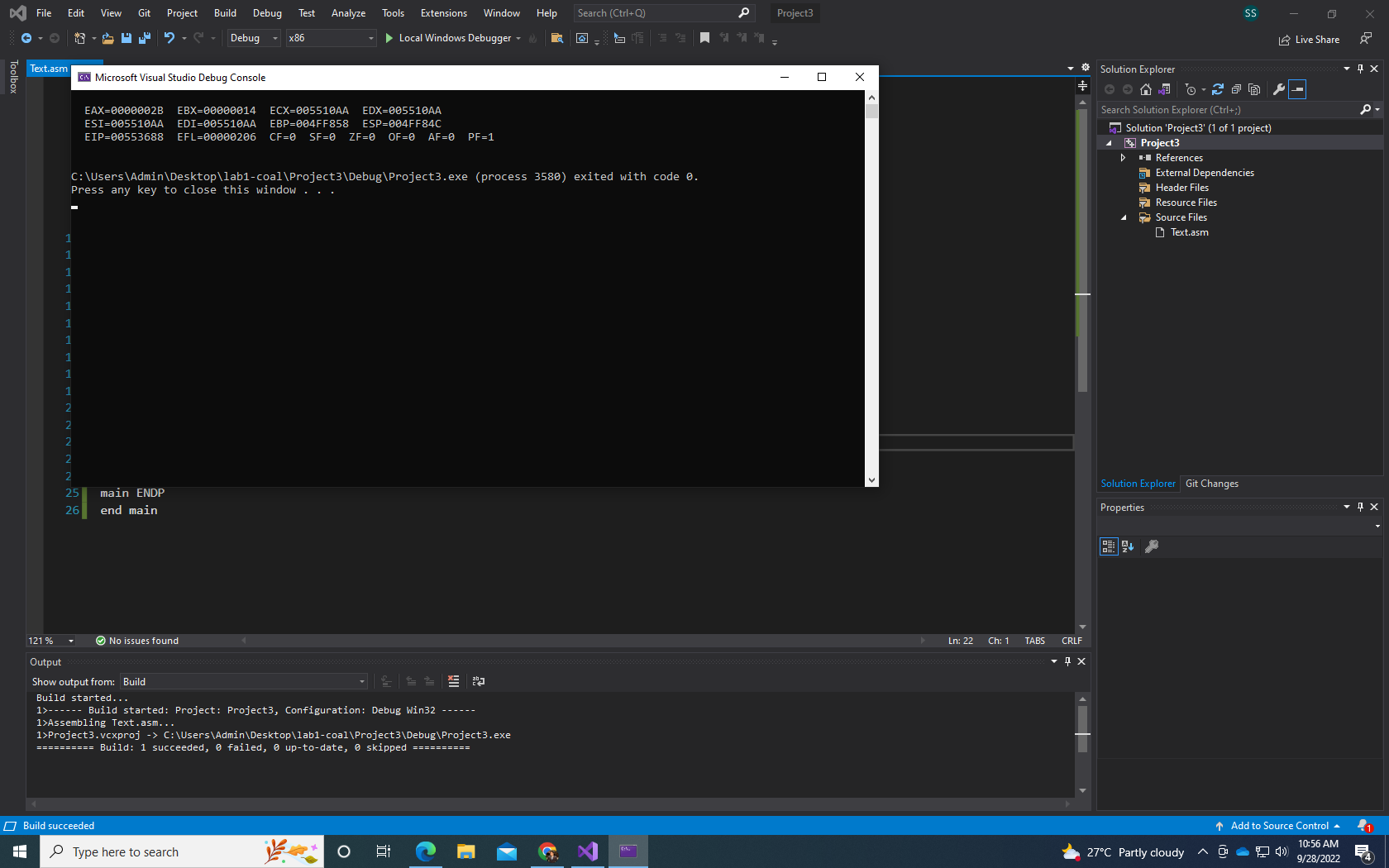
CALL DumpRegs

exit

main ENDP

END main

Q7.



Include Irvine32.inc

.data

Imm8 = 20

Data1 WORD 8

Data2 WORD 15

Data3 WORD 20

.code

main PROC

MOV eax, Imm8

MOVZX ebx, Data1

ADD eax, ebx

ADD eax, Imm8

MOVZX ebx, Data2

ADD eax, ebx

MOVZX ebx, Data3

SUB eax, ebx

CALL DumpRegs

exit

main ENDP

end main