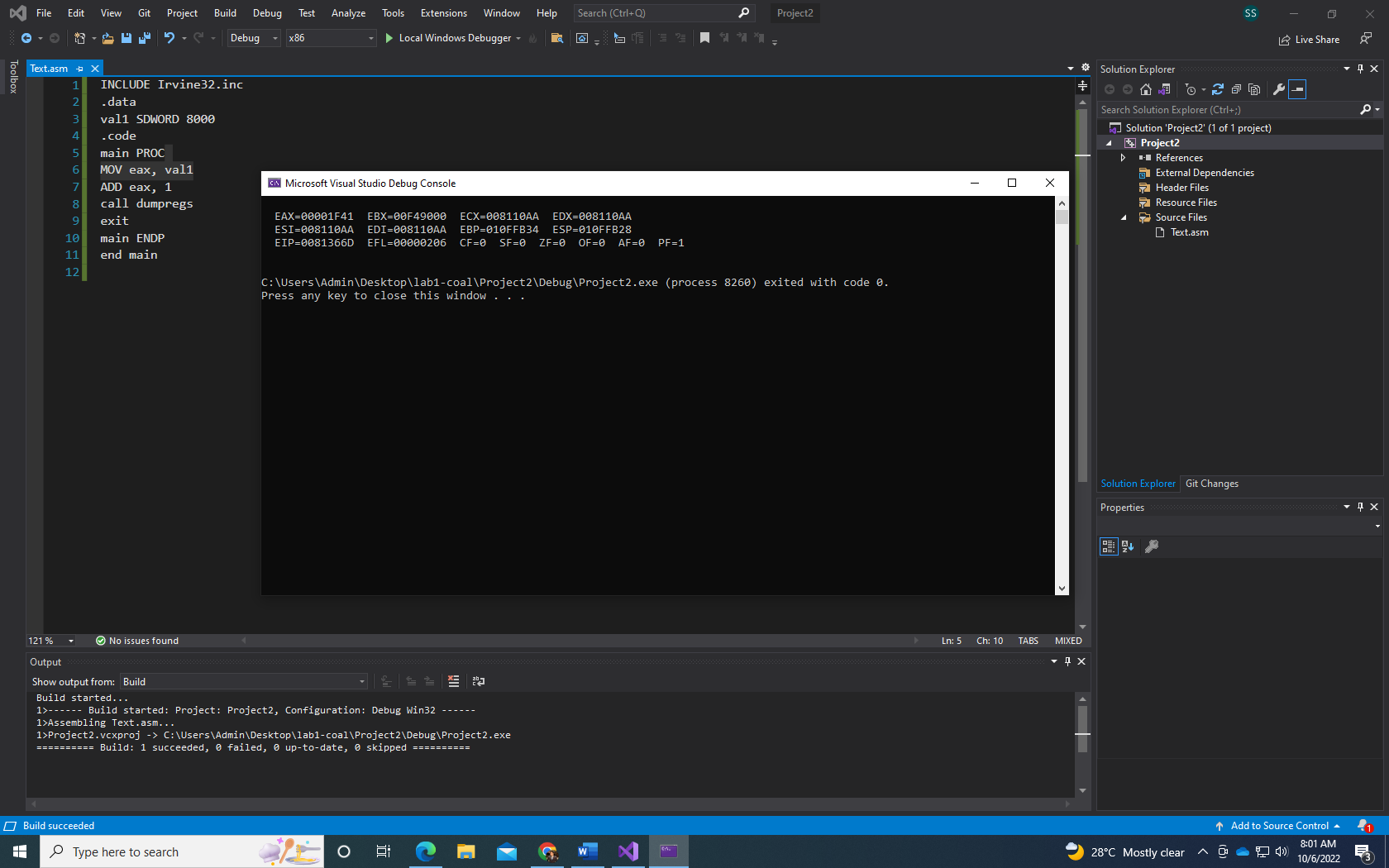
**LAB 04**

**Q1.**



INCLUDE irvine32.inc

.data

val1 SDWORD 8000

.code

main PROC

MOV eax, val1

ADD eax, 1

call dumpRegs

exit

main ENDP

end main

**Q2.**

a) CF = 1 SF = 0 ZF = 1 OF = 0

b) CF = 0 SF = 1 ZF = 0 OF = 1

c) CF = 0 SF = 1 ZF = 0 OF = 0

**Q3.**

include irvine32.inc

.data

arr DWORD 8, 5, 1, 2, 6

.code

Main PROC

mov esi, OFFSET arr

mov eax, DWORD ptr 0 ; initializing with 0

mov edx, DWORD ptr 0

mov ebx, DWORD ptr 0

mov eax, [esi]

mov ebx, [esi+16]

mov [esi+16], eax ; 8 in 5th pos

mov eax, [esi+8]

mov [esi], eax ; 1 in 1st pos

mov eax, [esi+4]

mov edx, [esi+12]

mov [esi+4], edx ; 2 in 2nd pos

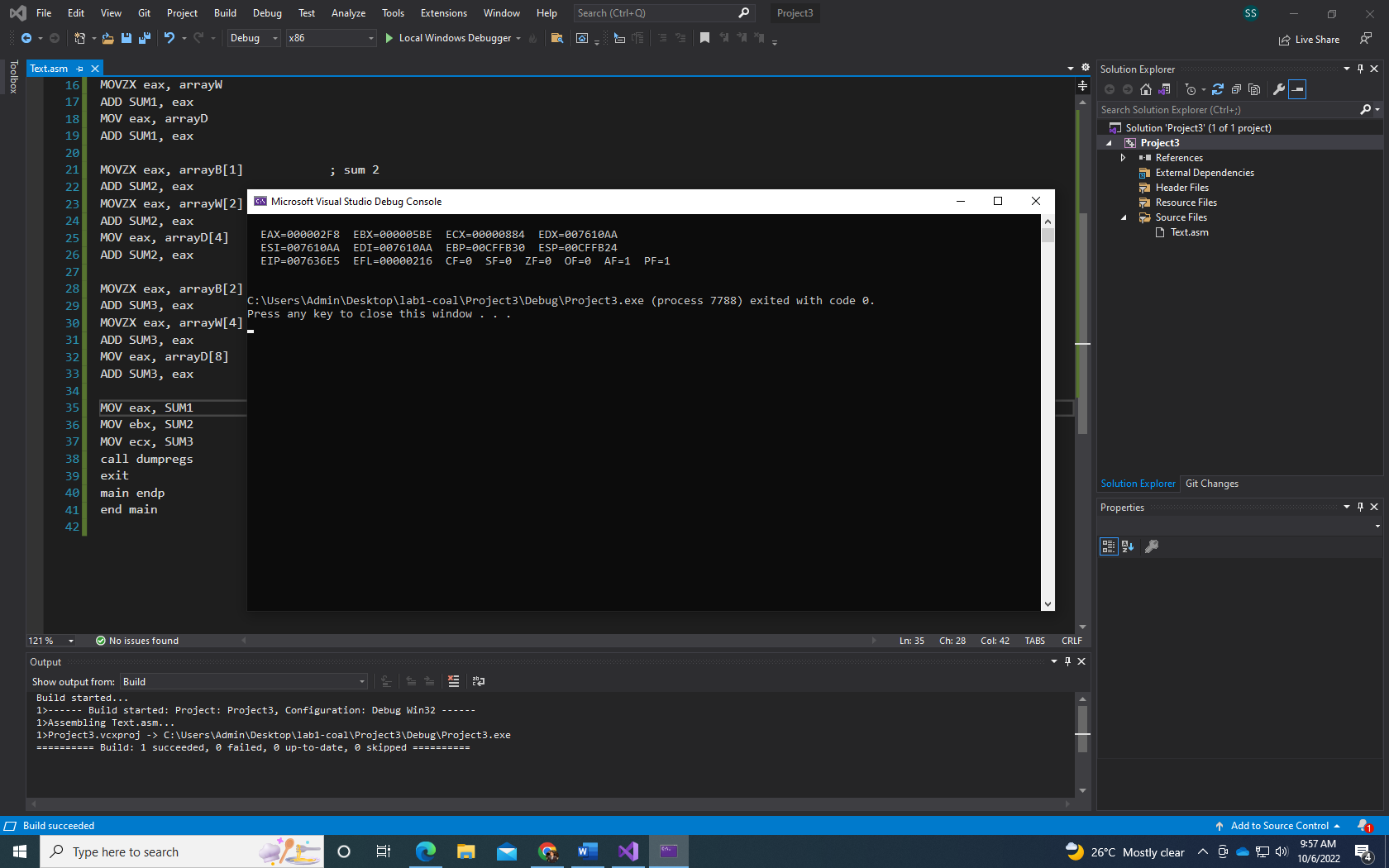
mov [esi+8], eax ; 5 in 3rd pos

mov [esi+12], ebx ; 6 in 4th pos

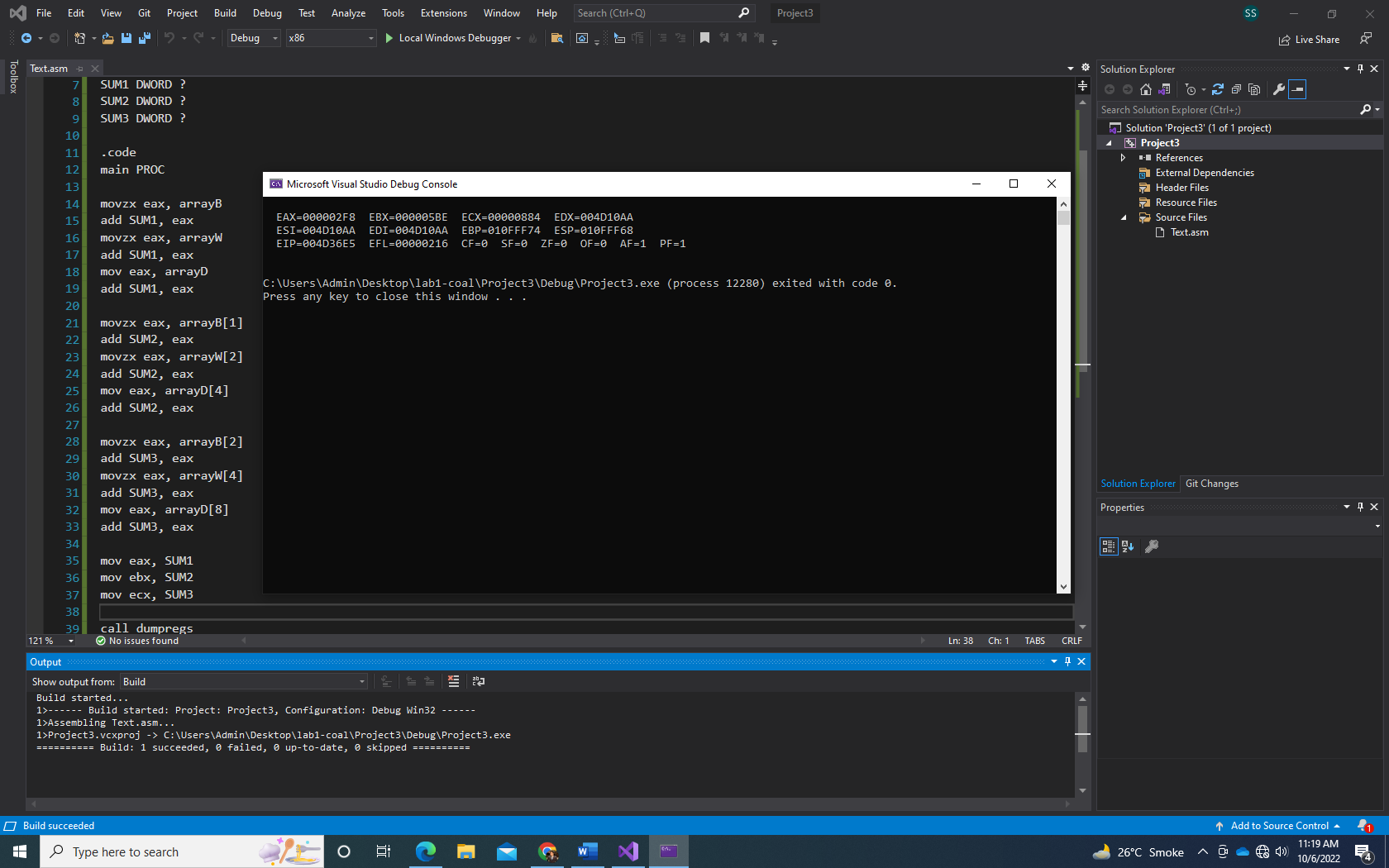
call dumpRegs

exit

main ENDP

end main

Q4.



include irvine32.inc

.data

arrayB BYTE 10, 20, 30

arrayW WORD 150, 250, 350

arrayD DWORD 600, 1200, 1800

SUM1 DWORD ?

SUM2 DWORD ?

SUM3 DWORD ?

.code

main PROC

movzx eax, arrayB

add SUM1, eax

movzx eax, arrayW

add SUM1, eax

mov eax, arrayD

add SUM1, eax

movzx eax, arrayB[1]

add SUM2, eax

movzx eax, arrayW[2]

add SUM2, eax

mov eax, arrayD[4]

add SUM2, eax

movzx eax, arrayB[2]

add SUM3, eax

movzx eax, arrayW[4]

add SUM3, eax

mov eax, arrayD[8]

add SUM3, eax

mov eax, SUM1

mov ebx, SUM2

mov ecx, SUM3

call dumpregs

exit

main endp

end main

Q5.

include irvine32.inc

.data

array1 BYTE 10, 20, 30, 40

array2 BYTE 4 dup(?)

.code

main PROC

MOV eax, DWORD ptr 0

MOV al, array1[3]

MOV array2, al

MOV al, array1[2]

MOV array2[1], al

MOV al, array1[1]

MOV array2[2], al

MOV al, array1

MOV array2[3], al

MOVZX eax, array2

MOVZX ebx, array2[1]

MOVZX ecx, array2[2]

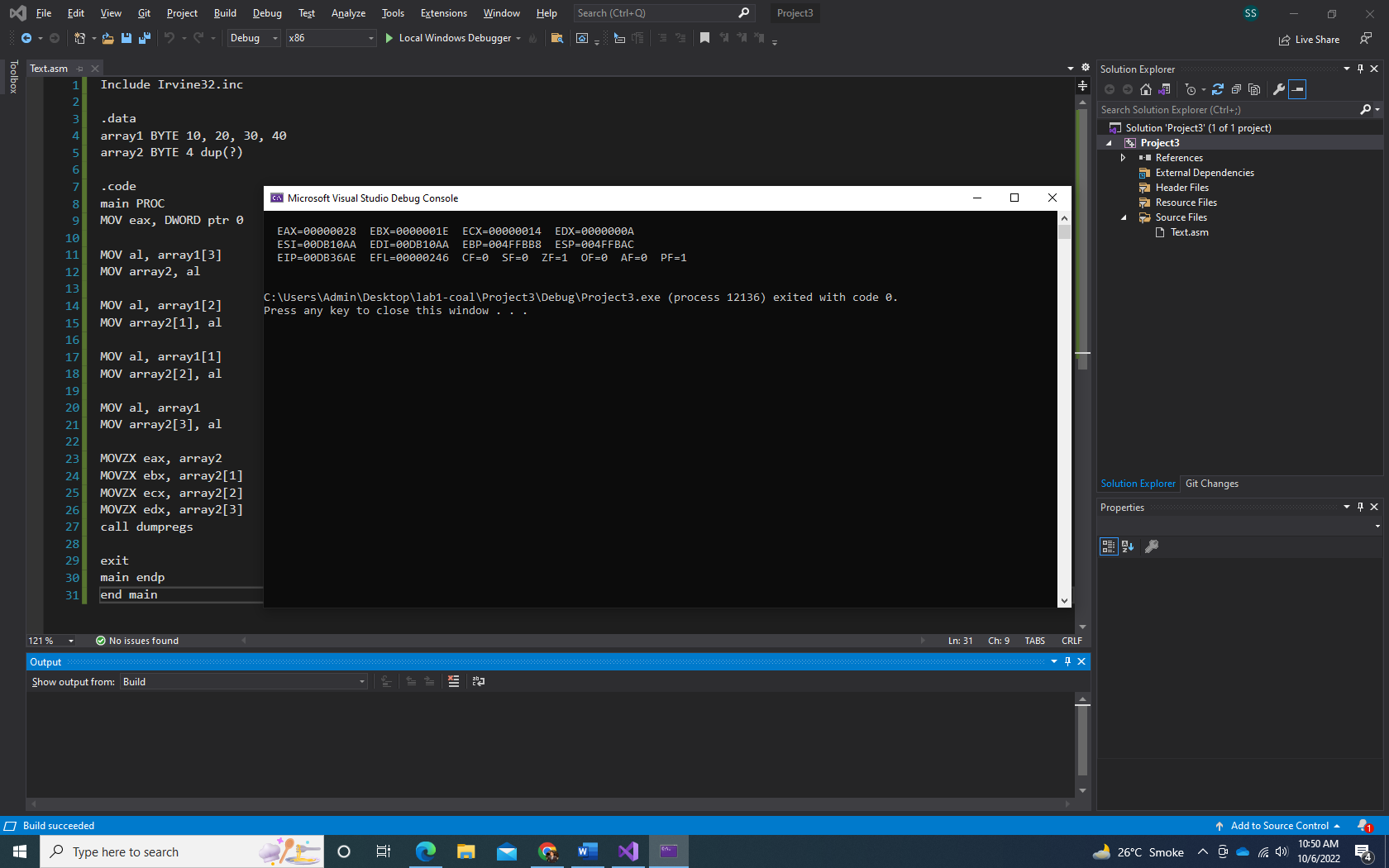
MOVZX edx, array2[3]

call dumpregs

exit

main endp

end main



Q6.

include irvine32.inc

.data

array1 DWORD 5, 6, 7, 8, 9

array2 DWORD 10, 12, 14, 15, 19

.code

main PROC

mov esi, array1

mov edi, array2

mov eax, [esi]

sub [edi], eax

add esi, 4

add edi, 4

mov eax, [esi]

sub [edi], eax

add esi, 4

add edi, 4

mov eax, [esi]

sub [edi], eax

add esi, 4

add edi, 4

mov eax, [esi]

sub [edi], eax

add esi, 4

add edi, 4

mov eax, [esi]

sub [edi], eax

;MOV eax, [array2]

;mov ebx, array2[1]

;mov ecx, array2[2]

;mov edx, array2[3]

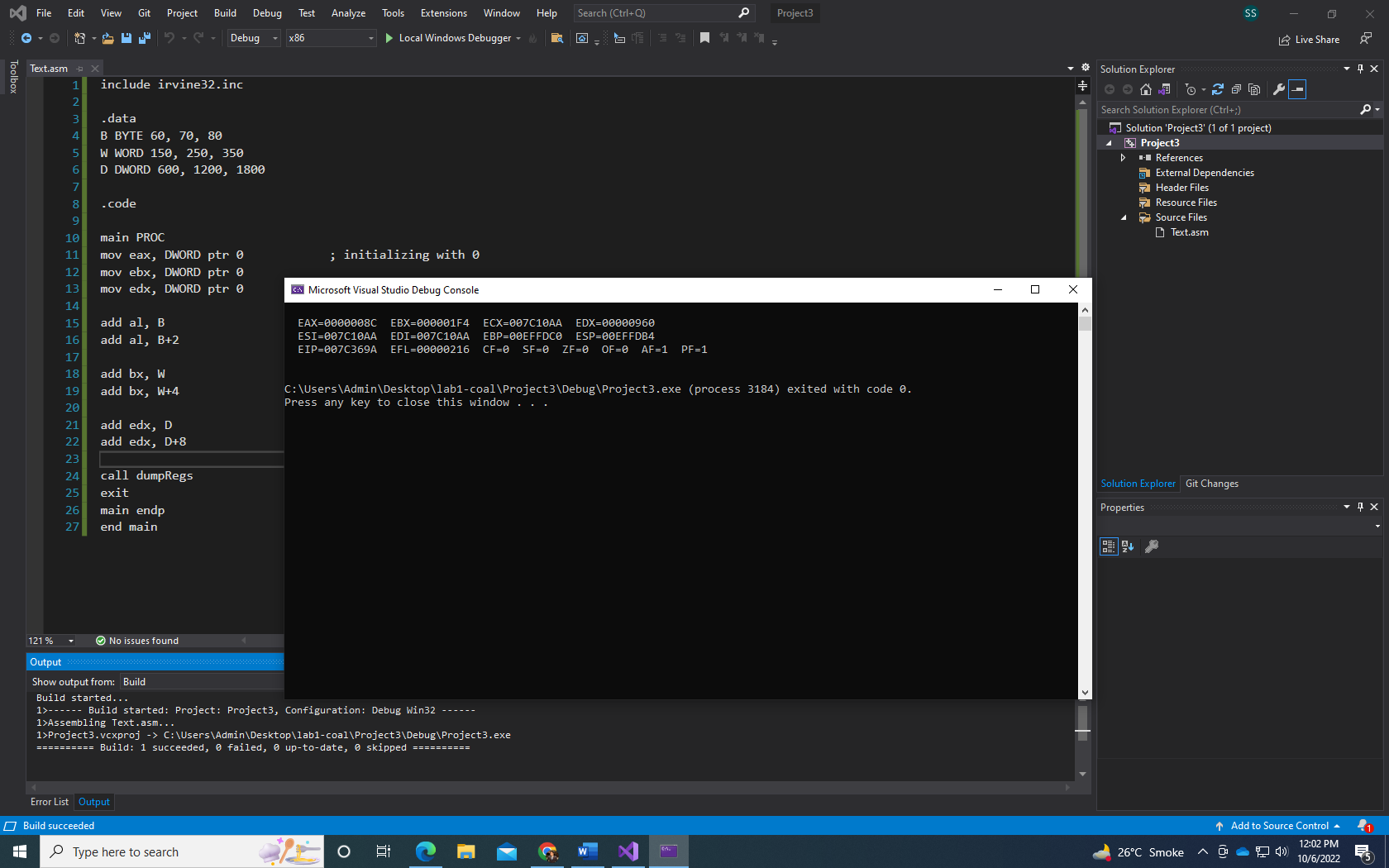
;call dumpregs

exit

main endp

end main

Q7.



include irvine32.inc

.data

B BYTE 60, 70, 80

W WORD 150, 250, 350

D DWORD 600, 1200, 1800

.code

main PROC

mov eax, DWORD ptr 0 ; initializing with 0

mov ebx, DWORD ptr 0

mov edx, DWORD ptr 0

add al, B

add al, B+2

add bx, W

add bx, W+4

add edx, D

add edx, D+8

call dumpRegs

exit

main endp

end main