COAL Lab 04 Assignment:

# Name: Owais Ali Khan

# Section: 3-F

# Roll no: 21K-3298

Question # 01:

TITLE Question 1

INCLUDE Irvine32.inc

.data

val1 SDWORD 8000h

.code

main PROC

mov eax, val1

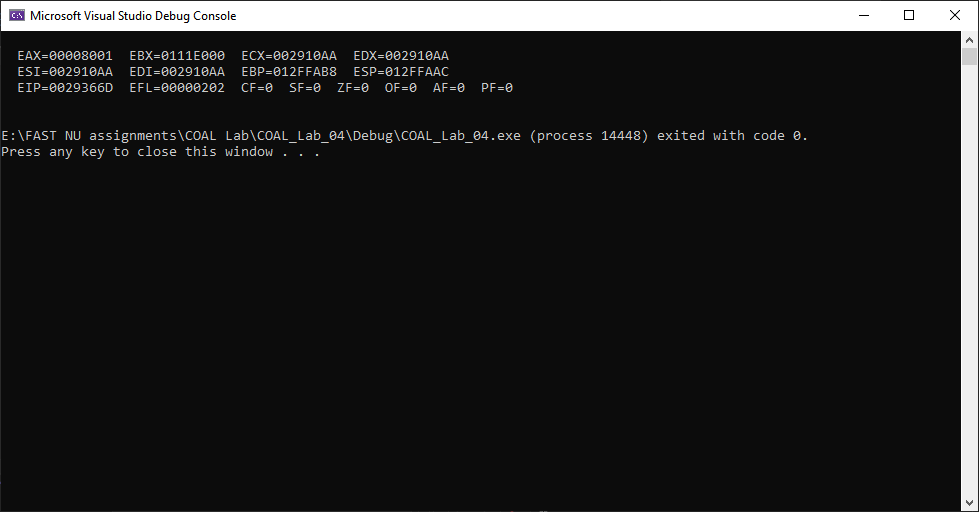
add eax, 1

call DumpRegs

exit

main ENDP

END main



Question # 02:

TITLE Question 2

INCLUDE Irvine32.inc

.data

.code

main PROC

mov eax, 7FF0h

add al, 10h

call DumpRegs

add ah, 1

call DumpRegs

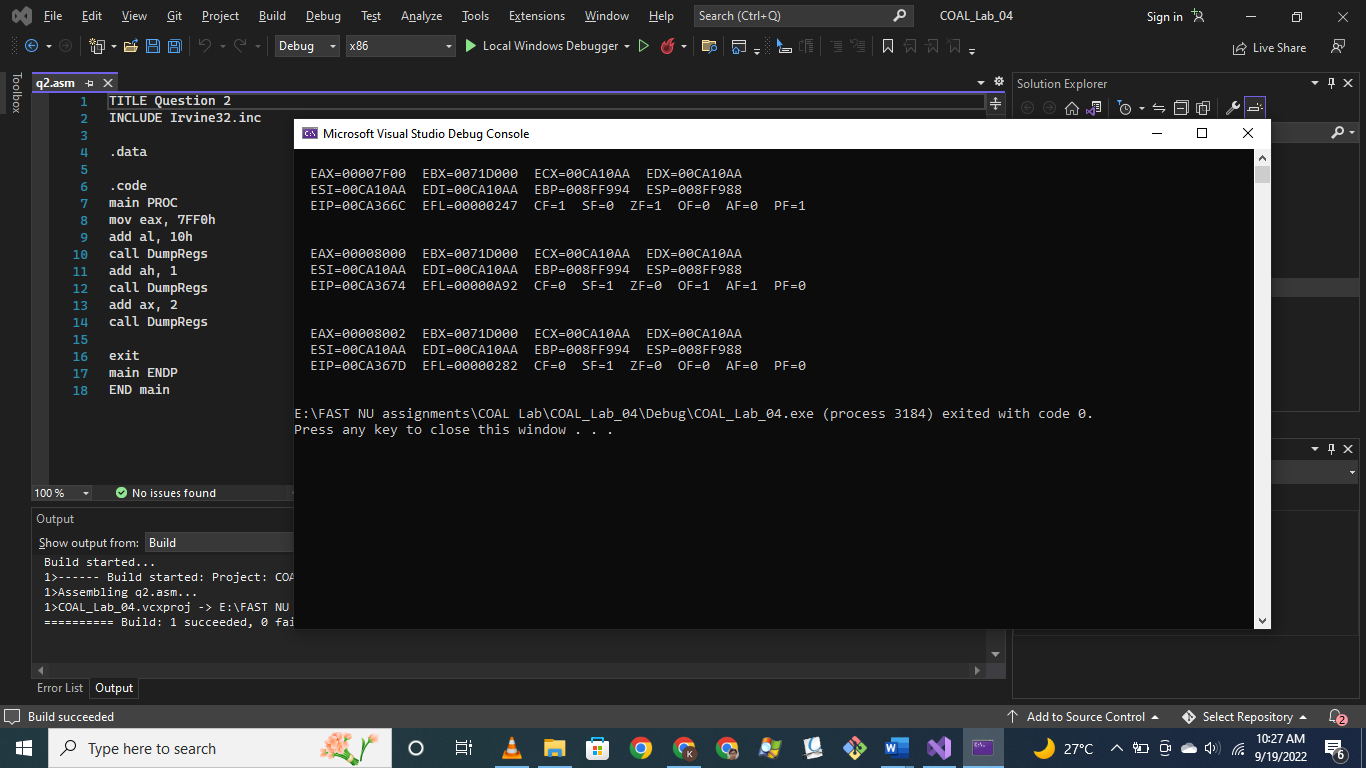
add ax, 2

call DumpRegs

exit

main ENDP

END main



Question # 03:

TITLE Question 3

INCLUDE Irvine32.inc

.data

array DWORD 8, 5, 1, 2, 6

.code

main PROC

mov eax, array[2 \* TYPE array]

xchg eax, array[0 \* TYPE array]

xchg eax, array[4 \* TYPE array]

xchg eax, array[3 \* TYPE array]

xchg eax, array[1 \* TYPE array]

xchg eax, array[2 \* TYPE array]

call DumpRegs

mov eax, [array]

call writeint

mov eax, [array + 4]

call writeint

mov eax, [array + 8]

call writeint

mov eax, [array + 12]

call writeint

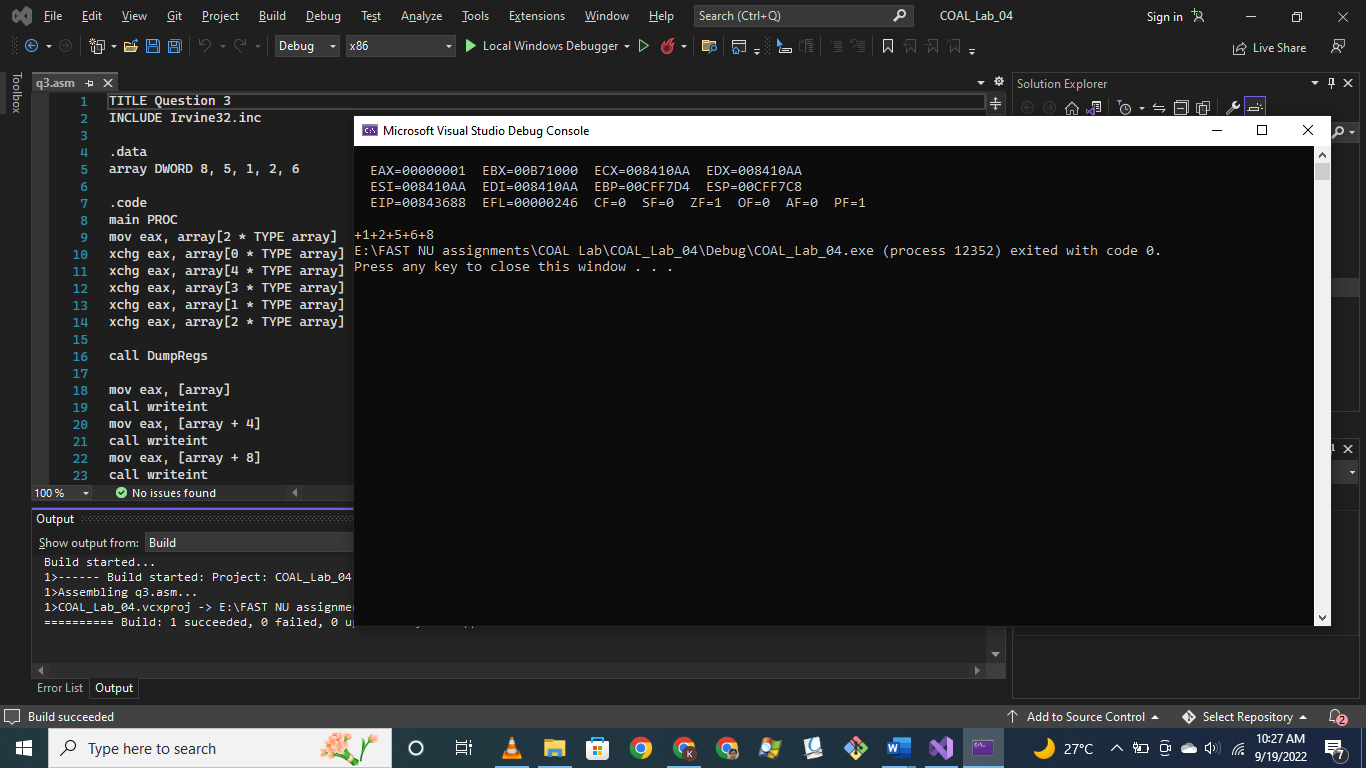
mov eax, [array + 16]

call writeint

exit

main ENDP

END main



Question # 04:

TITLE Question 4

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 ?

msg1 BYTE 13, 10, "SUM1 = ", 0

msg2 BYTE 13, 10, "SUM2 = ", 0

msg3 BYTE 13, 10, "SUM3 = ", 0

.code

main PROC

mov eax, 0

add al, [arrayB]

add ax, [arrayW]

add eax, [arrayD]

mov ebx, 0

add bl, [arrayB + 1]

add bx, [arrayW + 2]

add ebx, [arrayD + 4]

mov ecx, 0

add cl, [arrayB + 2]

add cx, [arrayW + 4]

add ecx, [arrayD + 8]

call DumpRegs

mov eax, eax

mov edx, OFFSET msg1

call writestring

call writeint

mov eax, ebx

mov edx, OFFSET msg2

call writestring

call writeint

mov eax, ecx

mov edx, OFFSET msg3

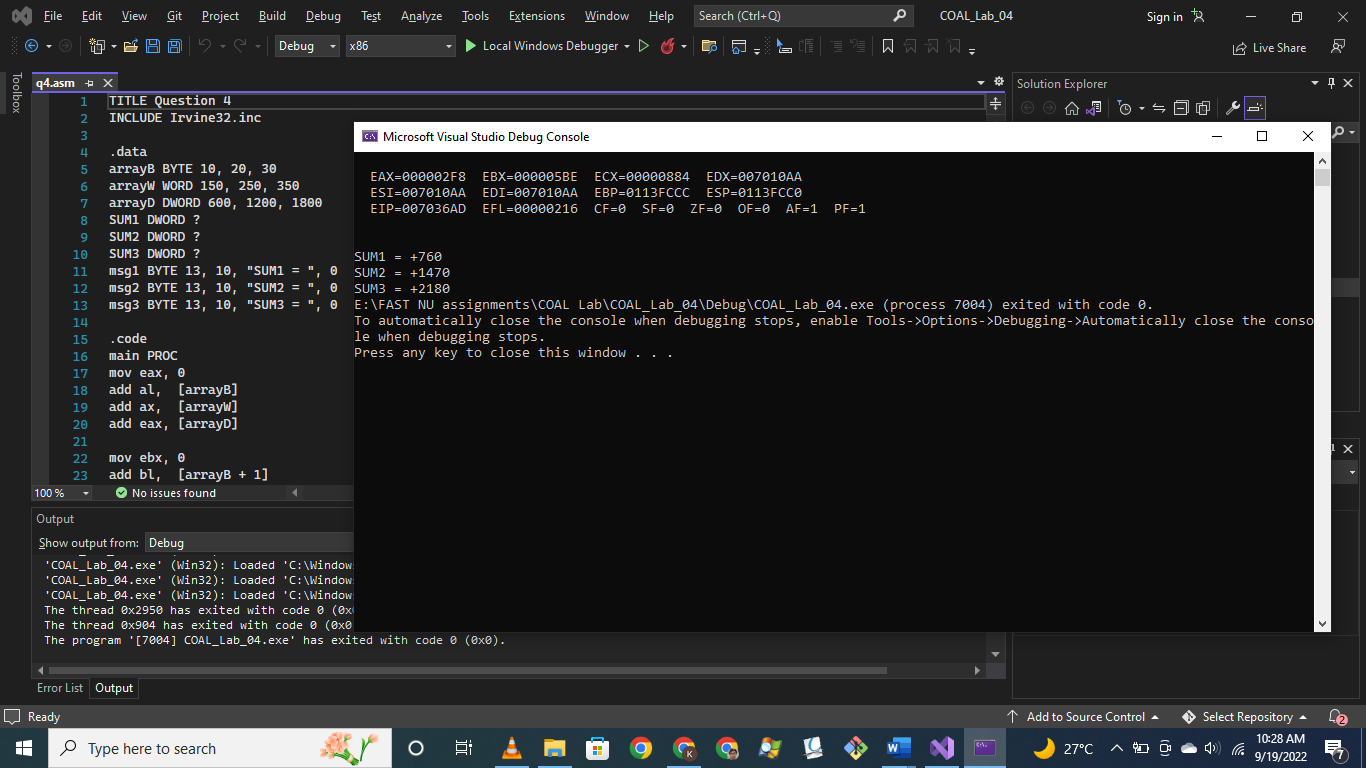
call writestring

call writeint

exit

main ENDP

END main



Question # 05:

TITLE Question 4

INCLUDE Irvine32.inc

.data

array1 BYTE 10, 20, 30, 40

array2 BYTE 4 DUP (?)

newline BYTE 13, 10, 0

.code

main PROC

mov eax, 0

;mov eax, DWORD PTR array1

;mov DWORD PTR array2, eax

mov esi, OFFSET array1

mov al, [esi + 3]

mov bl, [esi + 2]

mov cl, [esi + 1]

mov dl, [esi]

mov [array2], al

mov [array2 + 1], bl

mov [array2 + 2], cl

mov [array2 + 3], dl

call DumpRegs

mov eax, 0

mov al, [array1]

call writeint

mov al, [array1 + 1]

call writeint

mov al, [array1 + 2]

call writeint

mov al, [array1 + 3]

call writeint

mov edx, OFFSET newline

call writestring

mov eax, 0

mov al, [array2]

call writeint

mov al, [array2 + 1]

call writeint

mov al, [array2 + 2]

call writeint

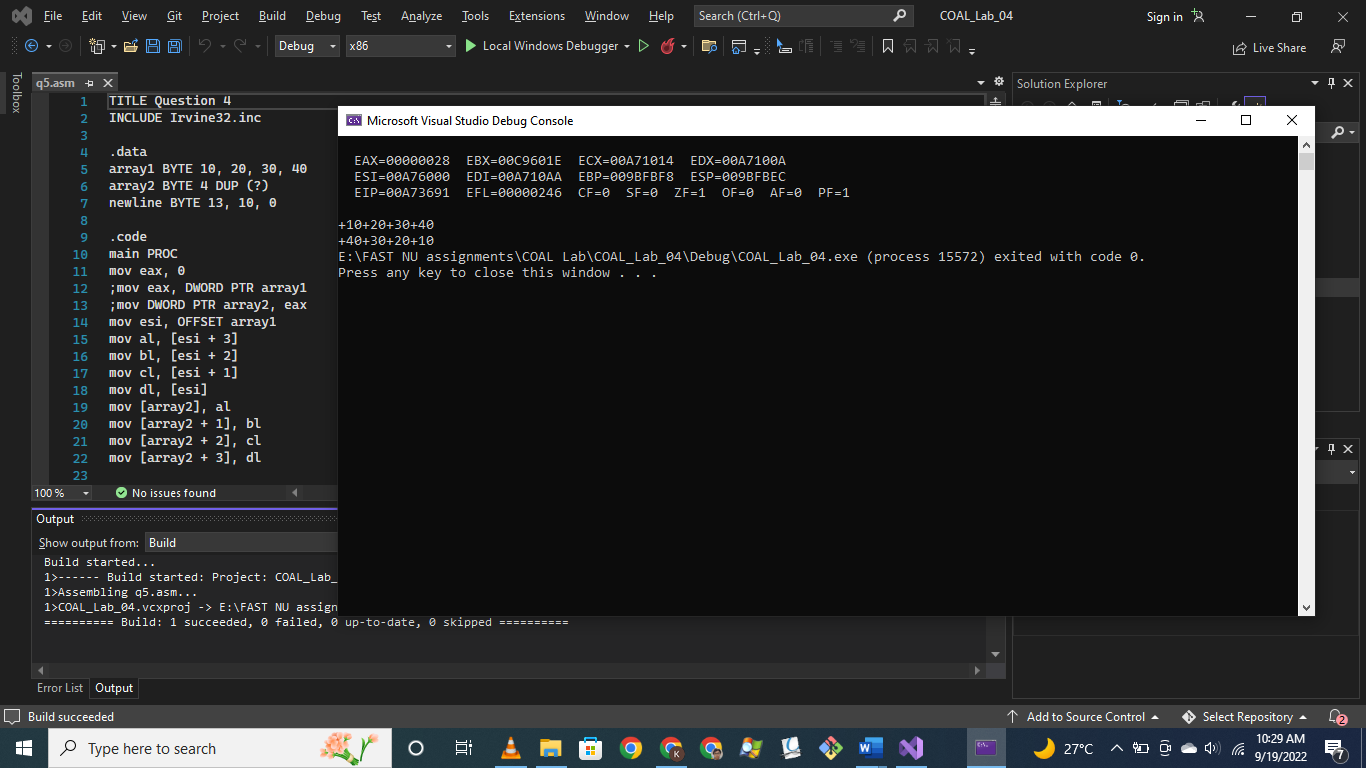
mov al, [array2 + 3]

call writeint

exit

main ENDP

END main



Question # 06:

TITLE Question 6

INCLUDE Irvine32.inc

.data

array DWORD 100, 200, 300, 400, 500

value DWORD 20

.code

main PROC

mov ecx, 5

mov esi, 0

Print\_before:

mov eax, [array + esi]

call writeint

add esi, TYPE DWORD

loop Print\_before

mov esi, 0

mov eax, value

sub [array + esi], eax

add esi, TYPE DWORD

sub [array + esi], eax

add esi, TYPE DWORD

sub [array + esi], eax

add esi, TYPE DWORD

sub [array + esi], eax

add esi, TYPE DWORD

sub [array + esi], eax

add esi, TYPE DWORD

call DumpRegs

mov ecx, 5

mov esi, 0

Print\_after:

mov eax, [array + esi]

call writeint

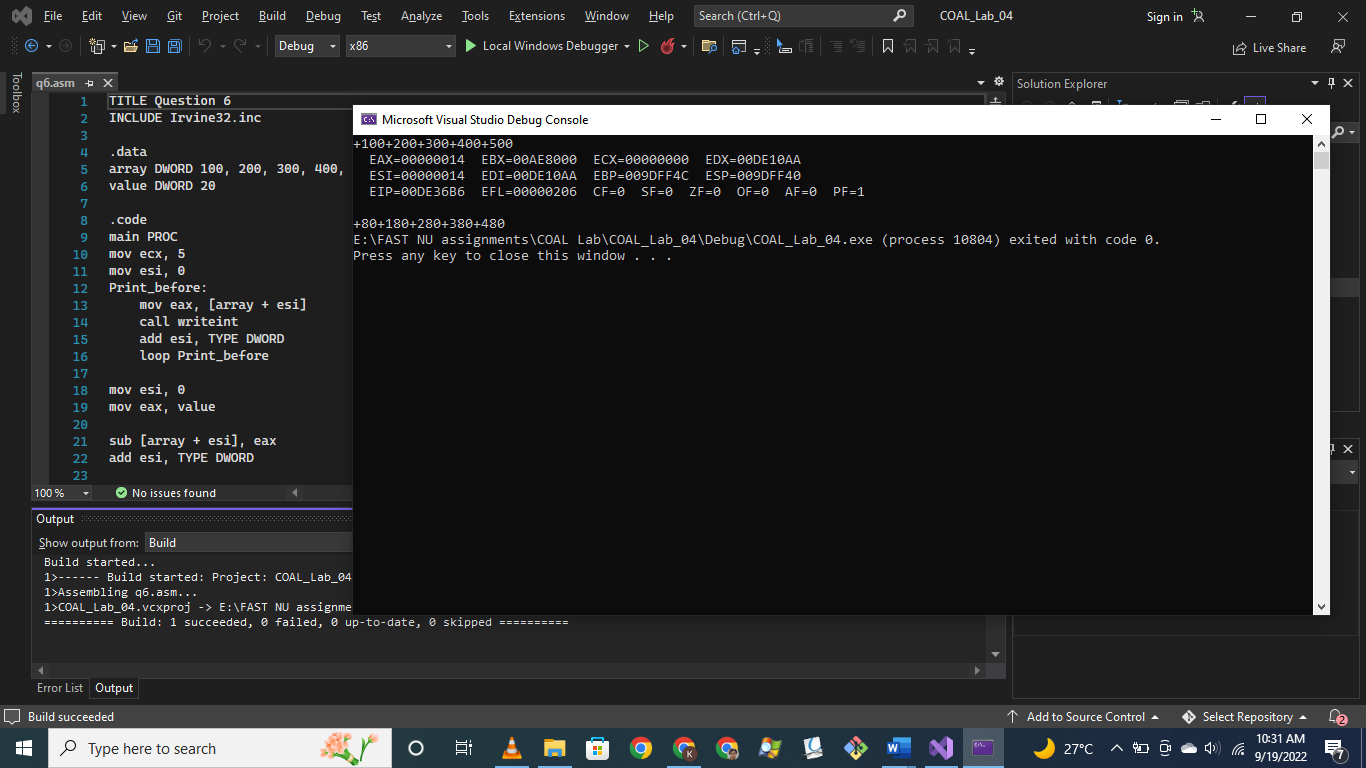
add esi, TYPE DWORD

loop Print\_after

exit

main ENDP

END main



Question # 07:

TITLE Question 7

INCLUDE Irvine32.inc

.data

arrayB BYTE 60, 70, 80

arrayW WORD 150, 250, 350

arrayD DWORD 600, 1200, 1800

.code

main PROC

movzx eax, arrayB

add al, arrayB[2 \* TYPE BYTE]

movzx ebx, arrayW

add bx, arrayW[2 \* TYPE WORD]

mov ecx, arrayD

add ecx, arrayD[2 \* TYPE DWORD]

call DumpRegs

exit

main ENDP

END main

