23I0544  
MUHAMMAD HAMMAD

BCS-2D

INCLUDE Irvine32.inc

.data

.code

main PROC

mov ax, 3d

; mov 23, ax this is incorrect because this can't be done the correct syntax will bea s followed

mov ax, 23

;mov cx, ch ; you can not save a 8 bit register into its own 16 bit register

mov ax, 1h ; the correct command is mov not move

;add 2, cx ; destination is supposed to be before the data

add cx, 2 ; this is the correct synatax

;add 3, 6 ; this needs a destination in place of 3 and any number thats supposed to be added will be after ,

;inc ax, 2 ; this is also incorrect inc counter increases +1 automatically no need to assign any number with it

inc ax

inc ax ; this is correcct way to add 2

call WriteInt

call Crlf

exit

main ENDP

END main

Task 02:

INCLUDE Irvine32.inc

.data

nameBytes WORD 72, 65, 77 ; hammad

.code

main PROC

mov ax, 0

mov ax, nameBytes[0]

call Writechar

call Crlf

mov ax, 0

mov ax, nameBytes[2]

call Writechar

call Crlf

mov ax, 0

mov ax, nameBytes[4]

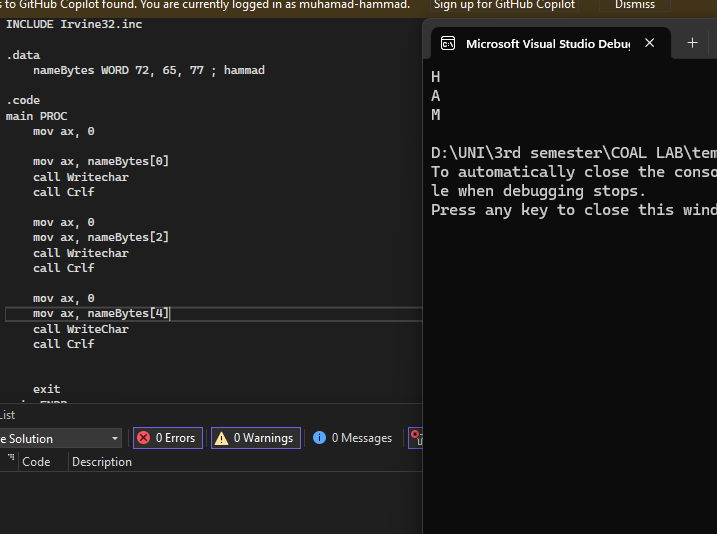
call WriteChar

call Crlf

exit

main ENDP

END main



TASK 03:  
INCLUDE Irvine32.inc

.data

varB BYTE +10d

varW WORD -150d

varD DWORD 600d

.code

main PROC

movsx eax, varB

call WriteInt

call Crlf

movsx ebx, varW

mov eax, ebx

call WriteInt

call Crlf

mov ecx, varD

mov eax, ecx

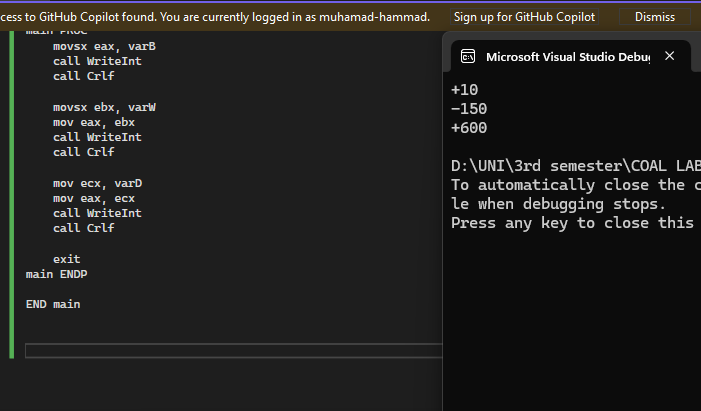
call WriteInt

call Crlf

exit

main ENDP

END main



Task 04:  
;1. EAX = 89 + 75Fh - 46o - 28 +1101b

;2. EAX = Val1 + Val2 - 654h + Val3

INCLUDE Irvine32.inc

.data

Val1 DWORD 25h

Val2 BYTE 36o

Val3 WORD 20d

.code

main PROC

mov eax, 89

add eax, 75Fh

sub eax, 46o

sub eax, 28

add eax, 1101b

call Writeint

call Crlf

mov eax, Val1

movzx ebx, Val2

add eax, ebx

mov ebx, -654h

sub eax, ebx

movzx ebx, Val3

add eax, ebx

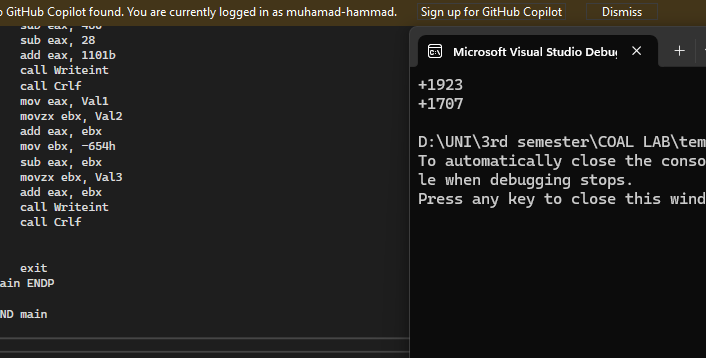
call Writeint

call Crlf

exit

main ENDP

END main



TASK 06:

INCLUDE Irvine32.inc

.data

SecondsInDay = 60 \* 60 \* 24

.code

main PROC

mov eax, SecondsInDay

call Writeint

call Crlf

exit

main ENDP

END main

TASK 07:  
INCLUDE Irvine32.inc

.data

A DWORD 0FF10h

B DWORD 0E108h

.code

main PROC

mov eax, A

mov ebx, B

xchg eax, ebx

mov A, ebx

mov B, eax

call writehex

call crlf

mov eax, A

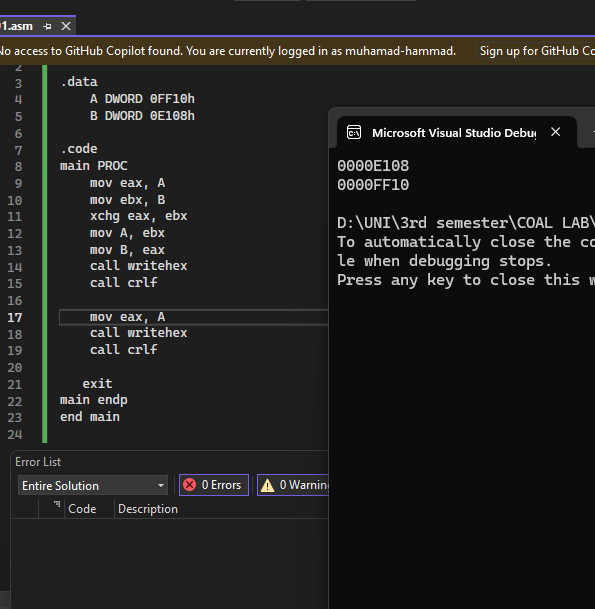
call writehex

call crlf

exit

main endp

end main



TAS 08:  
include irvine32.inc

.data

val1 BYTE 10h

val2 WORD 8000h

val3 DWORD 0FFFFh

val4 WORD 7FFFh

.code

main proc

mov eax, 0

inc val2

movzx eax, val2

call writehex

call crlf

movzx eax, val1

sub eax, val3

call writehex

call crlf

movzx eax, val2

movzx ebx, val4

sub eax, ebx

call writehex

call crlf

exit

main endp

end main

