**LAB# 08**

**Task 1:**

INCLUDE irvine32.inc

INCLUDE macros.inc

.data

Val1 SDWORD ?

Val2 SDWORD ?

Val3 SDWORD ?

ans1 SDWORD ?

ans2 SDWORD ?

ans3 SDWORD ?

.code

MAIN PROC

mov eax,0

mWrite "Enter value 1: "

call ReadInt

mov Val1,eax

mWrite "Enter value 2: "

call ReadInt

mov Val2,eax

mWrite "Enter value 3: "

call ReadInt

mov Val3,eax

mov eax,Val2

CDQ

mov ebx,Val3

idiv ebx ; Val2 / Val3

mov ans1,eax

mov eax,val1

CDQ

mov ebx,val2

idiv ebx ; Val1 / Val2

mov ans2,eax

mov eax,ans1

mov ebx,ans2

imul ebx ;ans1 \* ans2

mWrite "Answer: "

jo giveComplete

call WriteInt

jmp endProgram

giveComplete:

mov ans3,eax

mov eax,edx

call WriteInt

mWrite " "

mov eax,ans3

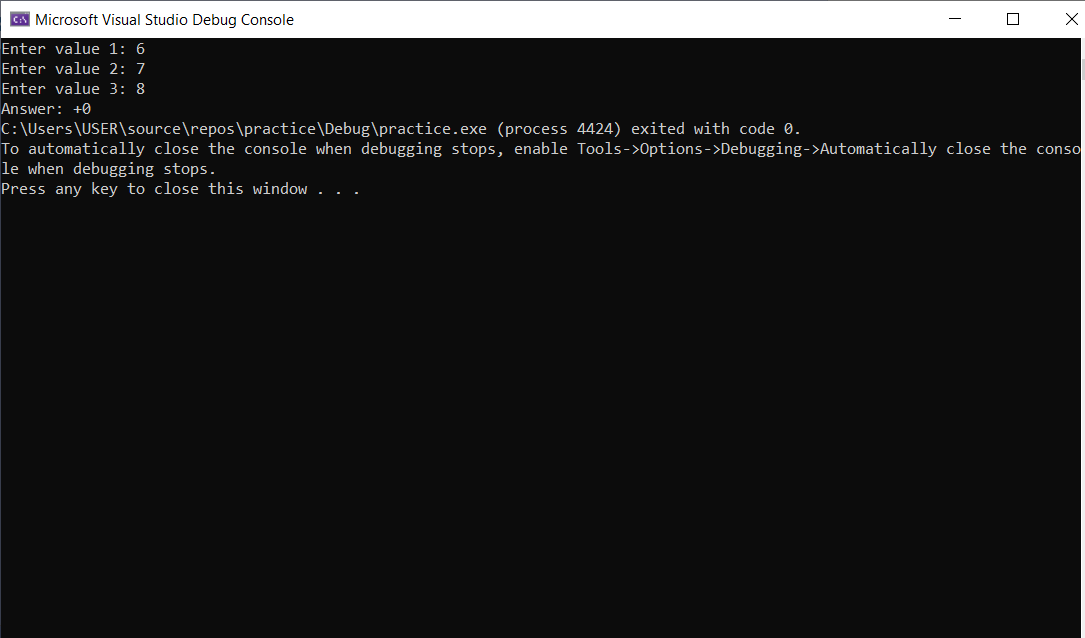
call WriteInt

endProgram:

exit

MAIN ENDP

END MAIN



**Task 2:**

INCLUDE irvine32.inc

INCLUDE macros.inc

.data

dum DWORD ?

origin DWORD ?

.code

MAIN PROC

mWrite "Please Enter the no: "

call ReadInt

mov dum,eax

mov origin,eax

mov ebx,dum

mov ecx,dum

;21 = 2^4 + 2^2 + 2^0

shl eax,4

shl ebx,2

shl ecx,0

mov dum,eax

add dum,ebx

add dum,ecx

mov eax,origin

call WriteDec

mWrite " X "

mov eax,21

call WriteDec

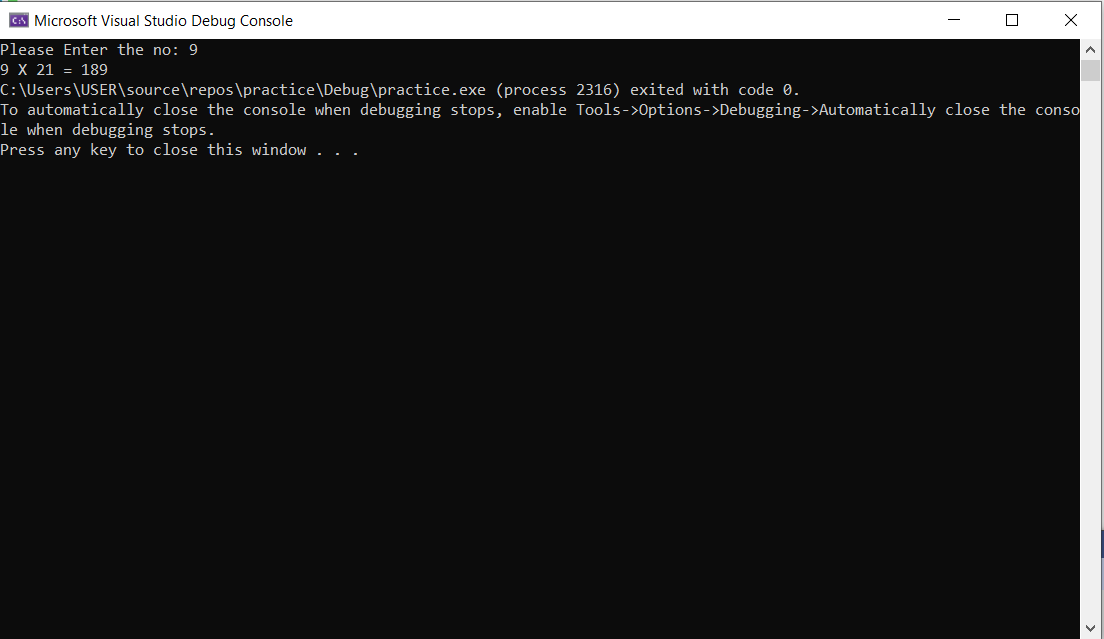
mWrite " = "

mov eax,dum

call WriteDec

exit

MAIN ENDP

END MAIN  


**Task 3:**

INCLUDE irvine32.inc

INCLUDE macros.inc

.code

MAIN PROC

mov ax,-128

rol eax,16

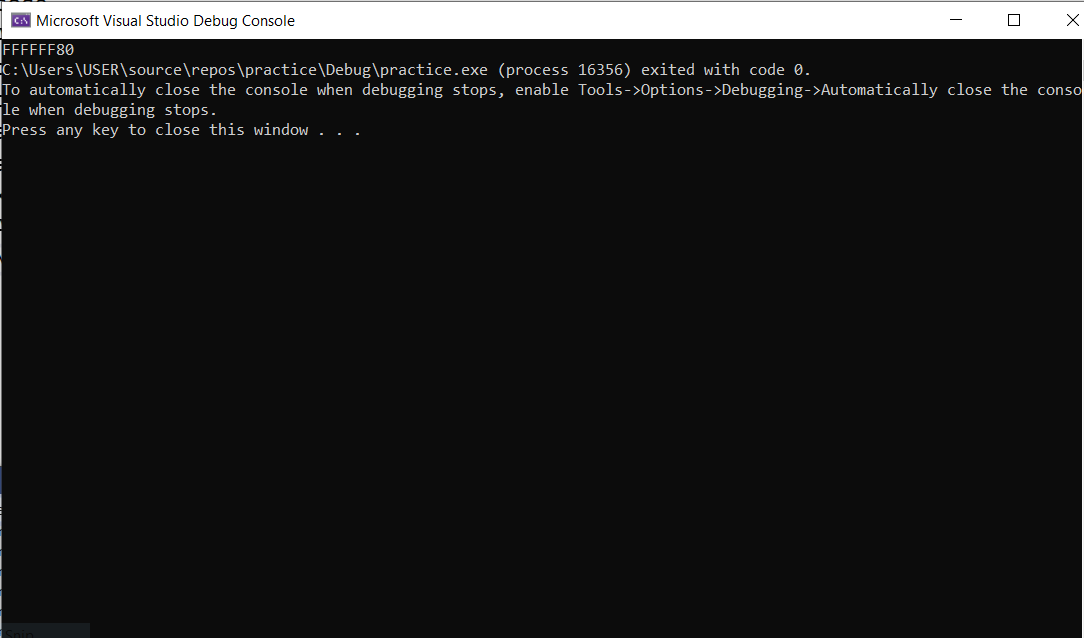
sar eax,16

call WriteHex

exit

MAIN ENDP

END MAIN



**Task 4:**

INCLUDE Irvine32.inc

.code

GCD PROC

push ebp

mov ebp, esp

; Save the x and y arguments on the stack

push ebx

push ecx

; Get the absolute value of x and y

mov eax, DWORD PTR [ebp+8] ; x

call abs

mov ebx, DWORD PTR [ebp+12] ; y

call abs

; Compute the GCD

jmp CHECK\_Y

DO\_LOOP:

mov eax, ebx ; n = x % y

mov edx, 0

div ebx

mov ebx, edx ; x = y, y = n

CHECK\_Y:

cmp ebx, 0 ; while (y > 0)

jg DO\_LOOP

mov eax, ebx ; return x (GCD)

pop ecx

pop ebx

mov esp, ebp

pop ebp

ret

GCD ENDP

abs PROC

push ebx

mov ebx, eax

sar ebx, 31

xor eax, ebx

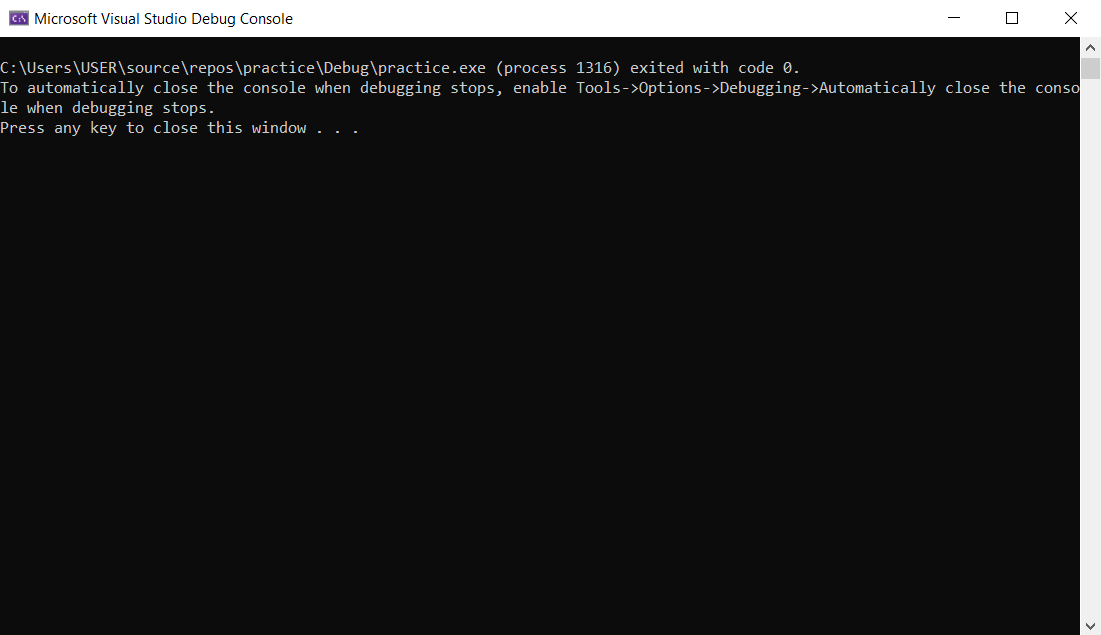
sub eax, ebx

pop ebx

ret

abs ENDP

End GCD



**Task 5:**

INCLUDE irvine32.inc

INCLUDE macros.inc

.data

Val1 QWORD 0123456789ABCDEF1h

Val2 QWORD 0123456789ABCDEF2h

ans DWORD 3 DUP(?)

.code

MAIN PROC

mov esi,OFFSET Val1

mov edi,OFFSET Val2

mov ebx,OFFSET ans

mov ecx,2

call Extended\_sub

mov eax,DWORD PTR ans + 8

call WriteHex

mov eax,DWORD PTR ans + 4

call WriteHex

mov eax,DWORD PTR ans

call WriteHex

exit

MAIN ENDP

Extended\_sub PROC

pushad

clc

LoopTime:

mov eax,[esi]

adc eax,[edi]

pushfd

mov [ebx],eax

sub esi,4

sub edi,4

sub ebx,4

popfd

loop LoopTime

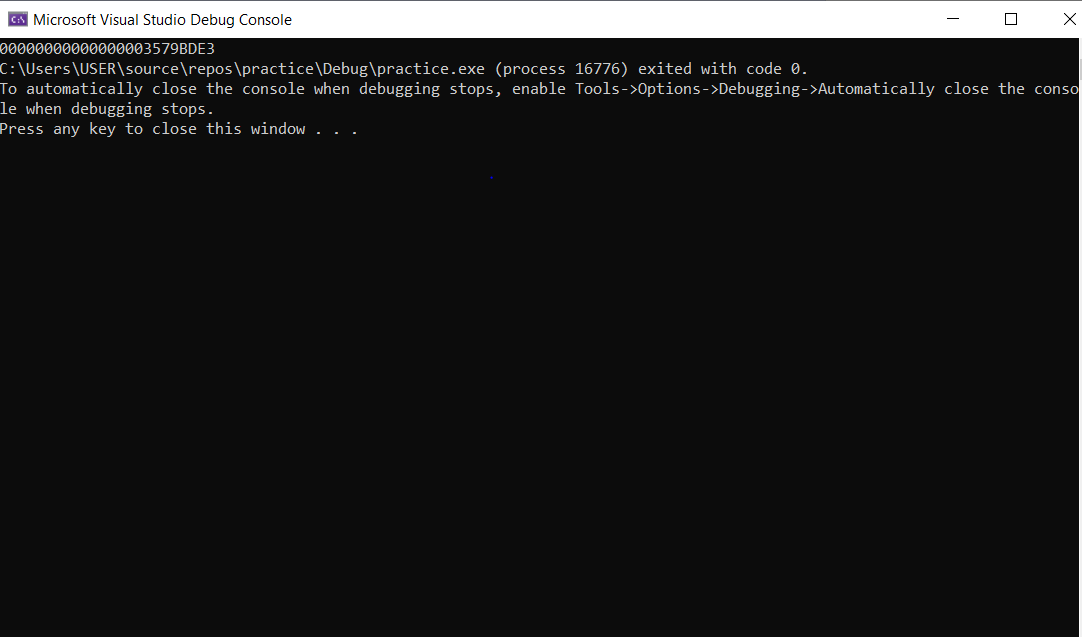
adc WORD PTR [ebx],0

popad

ret

Extended\_sub ENDP

END MAIN



**Task 6:**

INCLUDE irvine32.inc

INCLUDE macros.inc

.data

Val1 QWORD 0123456789ABCDEF1h

Val2 QWORD 0123456789ABCDEF2h

ans DWORD 3 DUP(?)

.code

MAIN PROC

mov esi,OFFSET Val1

mov edi,OFFSET Val2

mov ebx,OFFSET ans

mov ecx,2

call Extended\_Add

mov eax,DWORD PTR ans + 8

call WriteHex

mov eax,DWORD PTR ans + 4

call WriteHex

mov eax,DWORD PTR ans

call WriteHex

exit

MAIN ENDP

Extended\_Add PROC

pushad

clc

LoopTime:

mov eax,[esi]

adc eax,[edi]

pushfd

mov [ebx],eax

add esi,4

add edi,4

add ebx,4

popfd

loop LoopTime

adc WORD PTR [ebx],0

popad

ret

Extended\_Add ENDP

END MAIN

