**LAB # 03**

**NAME :MUHAMMAD HASSAM KHAN**

**S.ID :11141**

**Q1:**

1. **(a + b) – (a - b)**

**CODE:**

Include Irvine32.inc

.data

Number1 BYTE "FIRST NUMBER", 0dh, 0ah,0

Number2 BYTE "SECOND NUMBER", 0dh, 0ah,0

.code

main proc

mov edx, offset number1

call writestring

call readint

mov edx,eax

mov edx, offset number2

call writestring

call readint

mov ecx,eax

mov eax,ebx

add ebx,ecx

sub eax,ecx

sub ebx,eax

mov eax,ebx

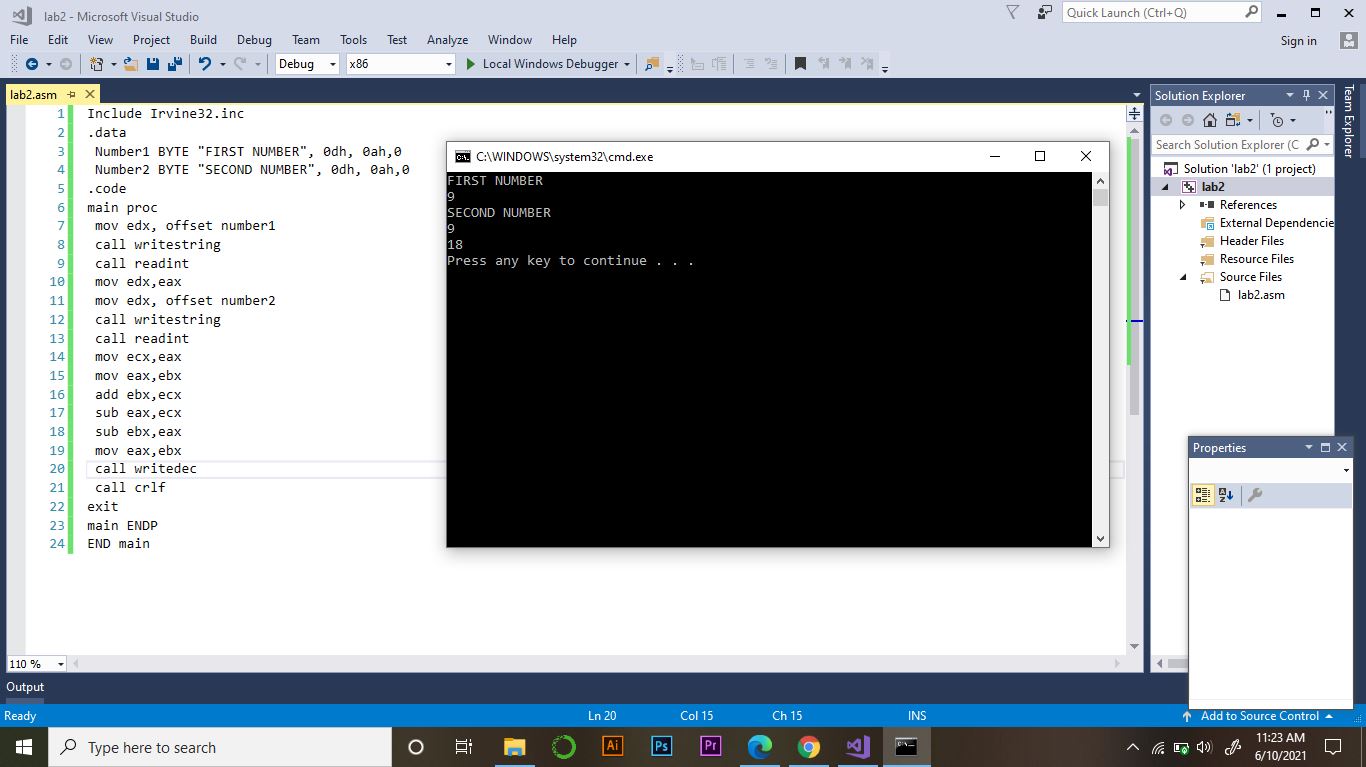
call writedec

call crlf

exitmain endp

end main

**SCREENSHORTS:**



1. **a2**

**CODE:**

Include Irvine32.inc

.data

msg1 BYTE "square value number", 0dh, 0ah,0

msg2 BYTE "your square value:"

.code

main proc

mov edx, offset msg1

call writestring

call readint

mov ebx,eax

mul ebx

mov edx, offset msg2

call writestring

call writedec

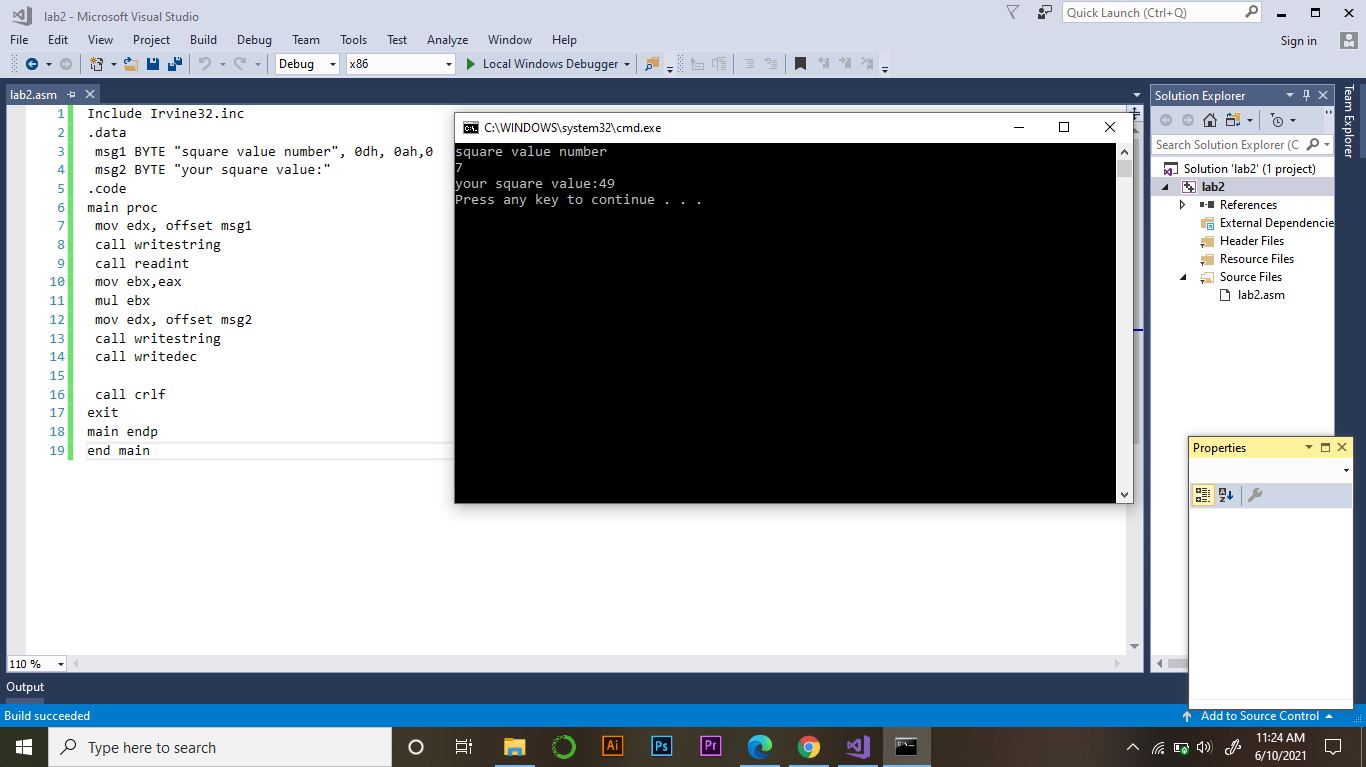
call crlf

exit

main endp

end main

**SCREENSHORTS:**



1. **b3**

**code:**

Include Irvine32.inc

.data

msg1 BYTE "cube value number", 0dh, 0ah,0

msg2 BYTE "your cube value:"

.code

main proc

mov edx, offset msg1

call writestring

call readint

mov ebx,eax

mul ebx

mul ebx

mov edx, offset msg2 call writestring

call writedec

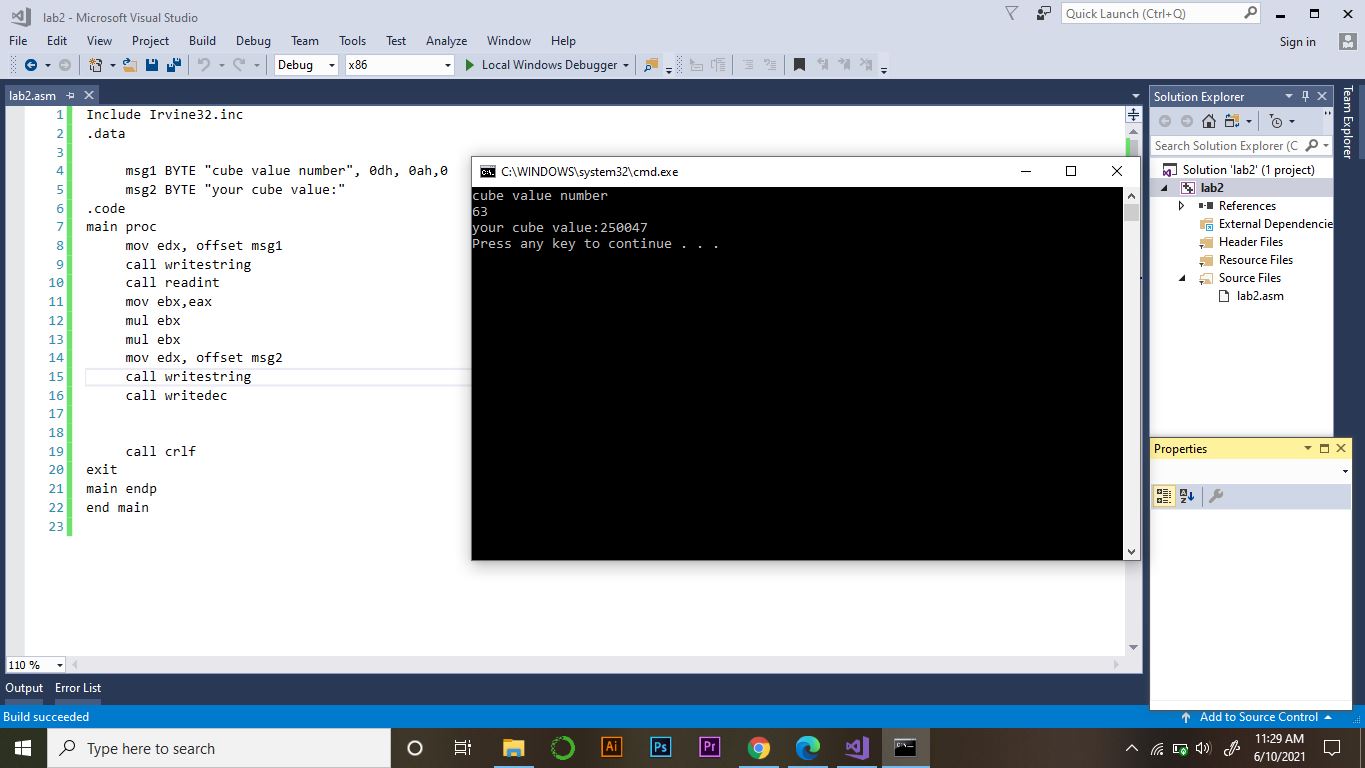
call crlf

exit

main endp

end main

**SCREENSHORTS:**



1. **a2+b2**

**CODE:**

Include Irvine32.inc

.data

msg1 BYTE " enter squre value number", 0dh, 0ah,0

msg2 BYTE " enter squre value number", 0dh, 0ah,0

msg3 BYTE "your squre value:"

.code

main proc

mov edx, offset msg1

call writestring

call readint

mov ebx,eax

mov edx, offset msg2 call writestring

call readint

mov ecx,eax

imul ebx,edx

imul ecx,ecx

add ebx,ecx

mov eax,ebx

mov edx, offset msg3

call writestring

call writedec

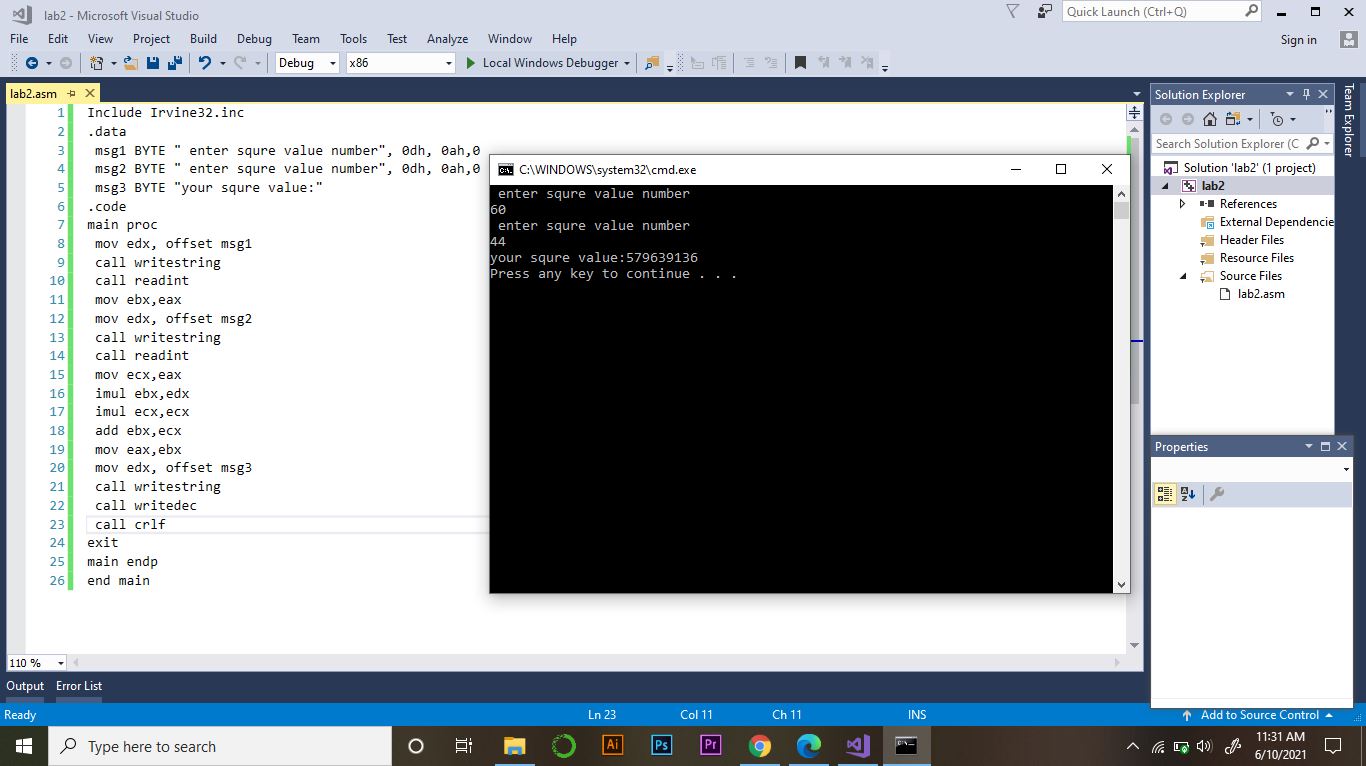
call crlf

exit

main endp

end main

**SCREENSHORTS:**



1. **A3+B3**

**CODE:**

Include Irvine32.inc

.data

msg1 BYTE " enter cube value number", 0dh, 0ah,0

msg2 BYTE " enter cube value number", 0dh, 0ah,0

msg3 BYTE "your sum cude value:"

.codemain proc

mov edx, offset msg1

call writestring

call readint

mov ebx,eax

mov edx, offset msg2

call writestring

call readint

mov ecx,eax

mov eax,ebx

imul ebx,edx

mul ebx

mov ebx,ecx

mov eax,ecx

imul ecx,ecx

mul ecx

mov ecx,eax

add ebx,ecx

mov eax,ebx

mov edx, offset msg3

call writestring

call writedec

call crlf

exit

main endp

end main

**SCREENSHORTS:**

