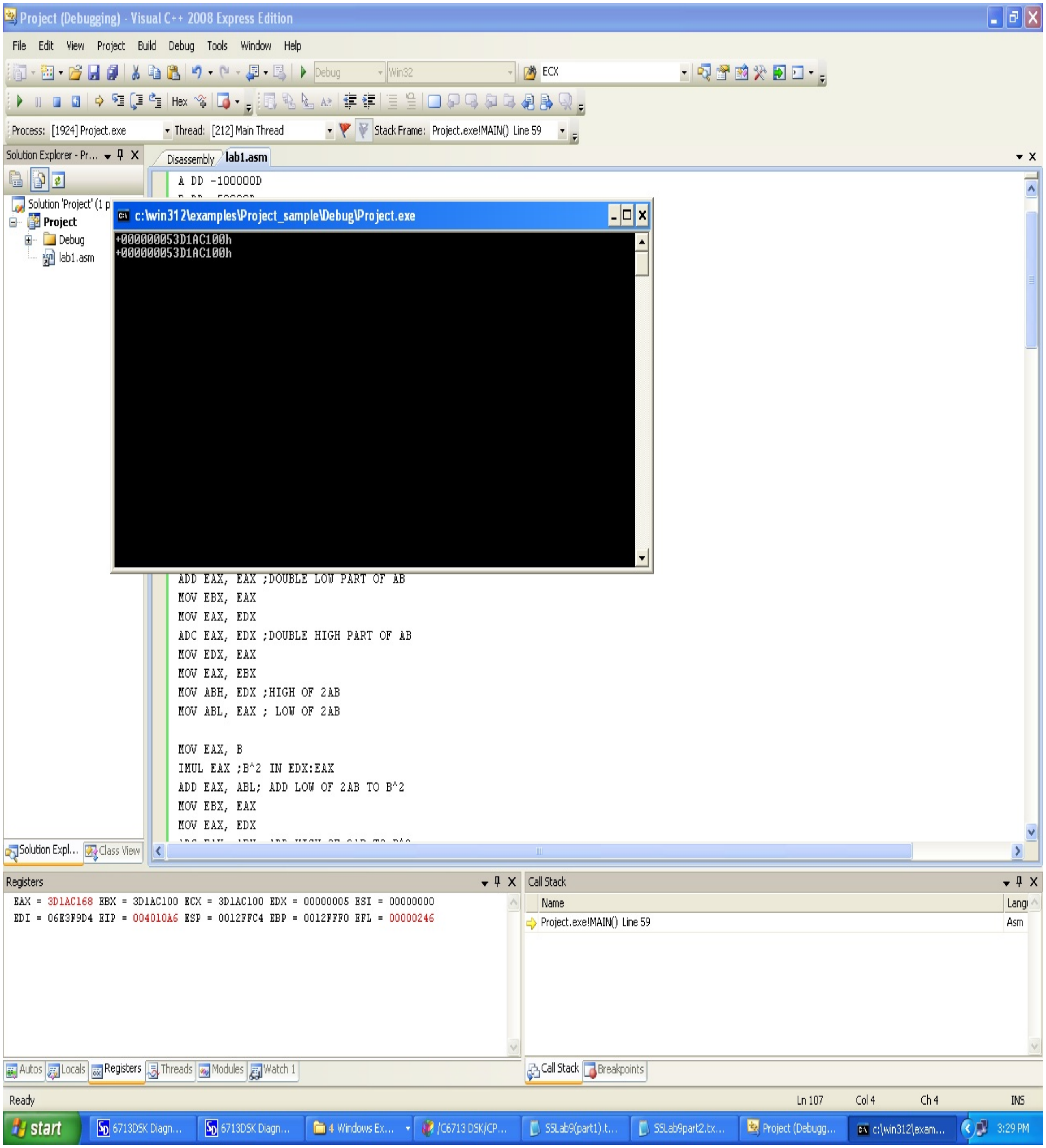
Andrew O’Neil-Smith
Lab 9

```
TITLE Lab 9
INCLUDE Irvine32.inc
.DATA
A DD -100000D
B DD -50000D
AAH DD 0000D
AAL DD 0000D
ABH DD 0000D
ABL DD 0000D
.STACK 100H
.CODE
MAIN PROC
CALL Clrscr
; left side = (A^2+2AB+B^2)
MOV EAX, A
IMUL EAX
MOV AAL, EAX ;LOW OF A^2
MOV AAH, EDX ;HIGH OF A^2
MOV EAX, A
MOV EBX, B
IMUL EBX ;AB IN EDX:EAX
ADD EAX, EAX ;DOUBLE LOW PART OF AB
MOV EBX, EAX
MOV EAX, EDX
ADC EAX, EDX ;DOUBLE HIGH PART OF AB
MOV EDX, EAX
MOV EAX, EBX
MOV ABH, EDX ;HIGH OF 2AB
MOV ABL, EAX ; LOW OF 2AB
MOV EAX, B
IMUL EAX ;B^2 IN EDX:EAX
ADD EAX, ABL; ADD LOW OF 2AB TO B^2
MOV EBX, EAX
MOV EAX, EDX
ADC EAX, ABH; ADD HIGH OF 2AB TO B^2
MOV EDX, EAX
MOV EAX, EBX
ADD EAX, AAL; ADD LOW OF A^2 TO 2AB+B^2
MOV EBX, EAX
MOV EAX, EDX
ADC EAX, AAH; ADD HIGH OF A^2 TO 2AB+B^2
MOV EDX, EAX
MOV EAX, EBX
CALL output
CALL Crlf
;right side = (A+B)^2
MOV EAX, A
ADD EAX, B
IMUL EAX
CALL output
CALL Crlf
EXIT
MAIN ENDP
output proc
Mov ECX, EAX
Mov EAX, EDX
Cmp EDX, 0ffffffffh
Jz neglower
Cmp eax, 0h
Jz poslower
Mov esi, edx
And esi, 80000000h
Jz pos
Mov al,'-'
Call WriteChar
Not ecx
Not edx
Add ecx, 1
Adc edx, 0
Mov eax,edx
Call WriteHex
Mov eax, ecx
Call WriteHex
Mov al,'h'
Call WriteChar
Ret
pos: mov al,'+'
Call WriteChar
Mov eax, edx
Call WriteHex
Mov eax, ecx
Call WriteHex
Mov al,'h'
Call WriteChar
Ret
neglower:
mov esi, ecx
And esi, 80000000h
JZ addnegsign
Mov eax, ecx
Call WriteInt
Ret
addnegsign:
```

```
mov al,'-'
Call WriteChar
Mov eax, ecx
Call WriteDec
Ret
poslower:       mov al, '+'
                call WriteChar
                        mov eax,ecx
                        call WriteDec
                        ret
output endp

end main
```



c:\win312\examples\Project_sample\Debug\Project.exe

```
+000000053D1AC100h
+000000053D1AC100h
```

```
A DD -100000D
```

```
ADD EAX, EAX ;DOUBLE LOW PART OF AB
MOV EBX, EAX
MOV EAX, EDX
ADC EAX, EDX ;DOUBLE HIGH PART OF AB
MOV EDX, EAX
MOV EAX, EBX
MOV ABH, EDX ;HIGH OF 2AB
MOV ABL, EAX ; LOW OF 2AB

MOV EAX, B
IMUL EAX ;B^2 IN EDX:EAX
ADD EAX, ABL; ADD LOW OF 2AB TO B^2
MOV EBX, EAX
MOV EAX, EDX
```

Registers

```
EAX = 3D1AC168 EBX = 3D1AC100 ECX = 3D1AC100 EDX = 00000005 ESI = 00000000
EDI = 06E3F9D4 EIP = 004010A6 ESP = 0012FFC4 EBP = 0012FFF0 EFL = 00000246
```

Call Stack

Name	Lang
Project.exe!MAIN() Line 59	Asm