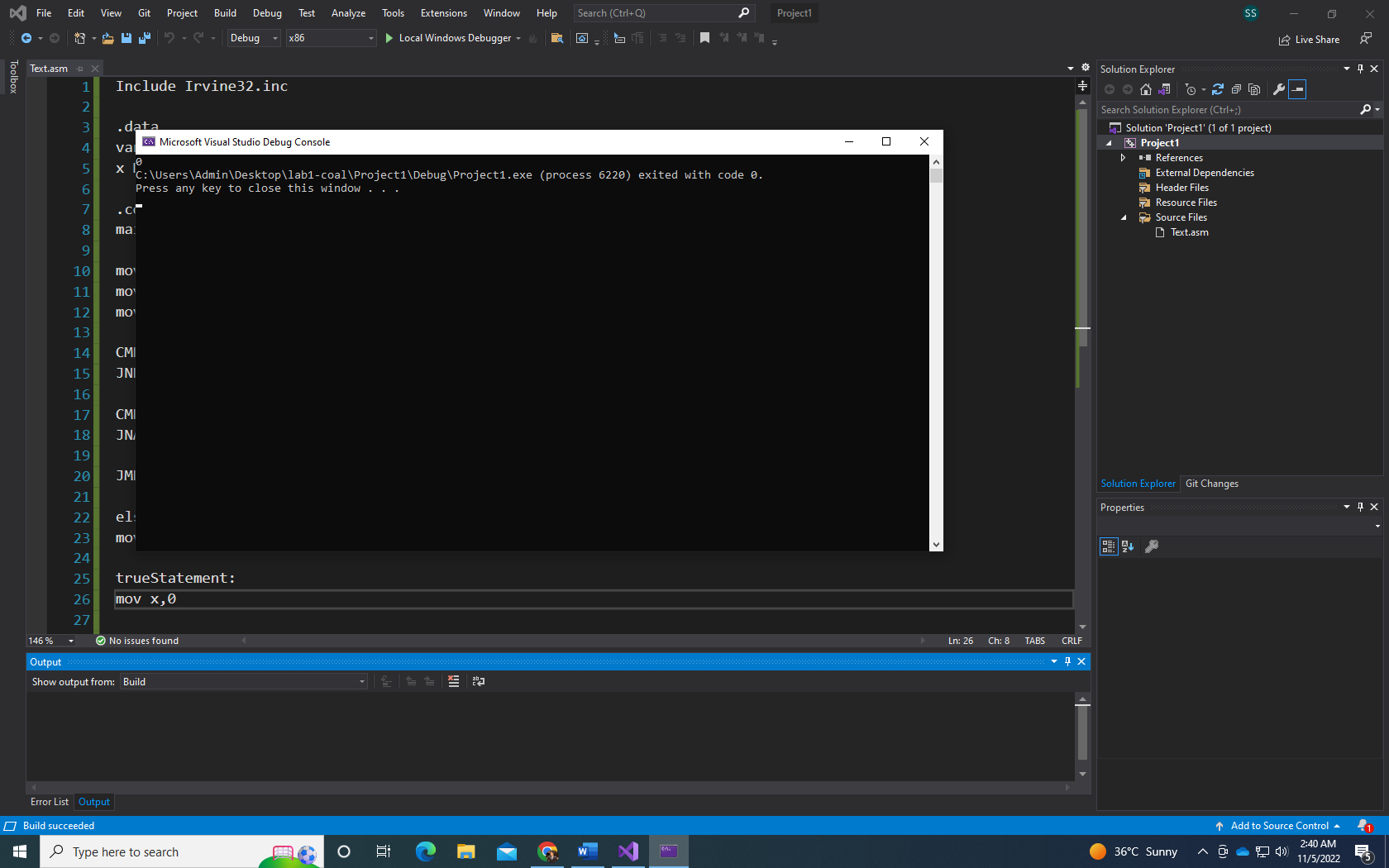
**LAB 07**

**21K-3278-D**

Q1.



Include Irvine32.inc

.data

var DWORD 5

x DWORD ?

.code

main proc

mov eax,var

mov ecx,6

mov edx,2

CMP eax,ecx

JNB elseStatement

CMP eax,edx

JNAE elseStatement

JMP trueStatement

elseStatement:

mov x,1

trueStatement:

mov x,0

mov eax,x

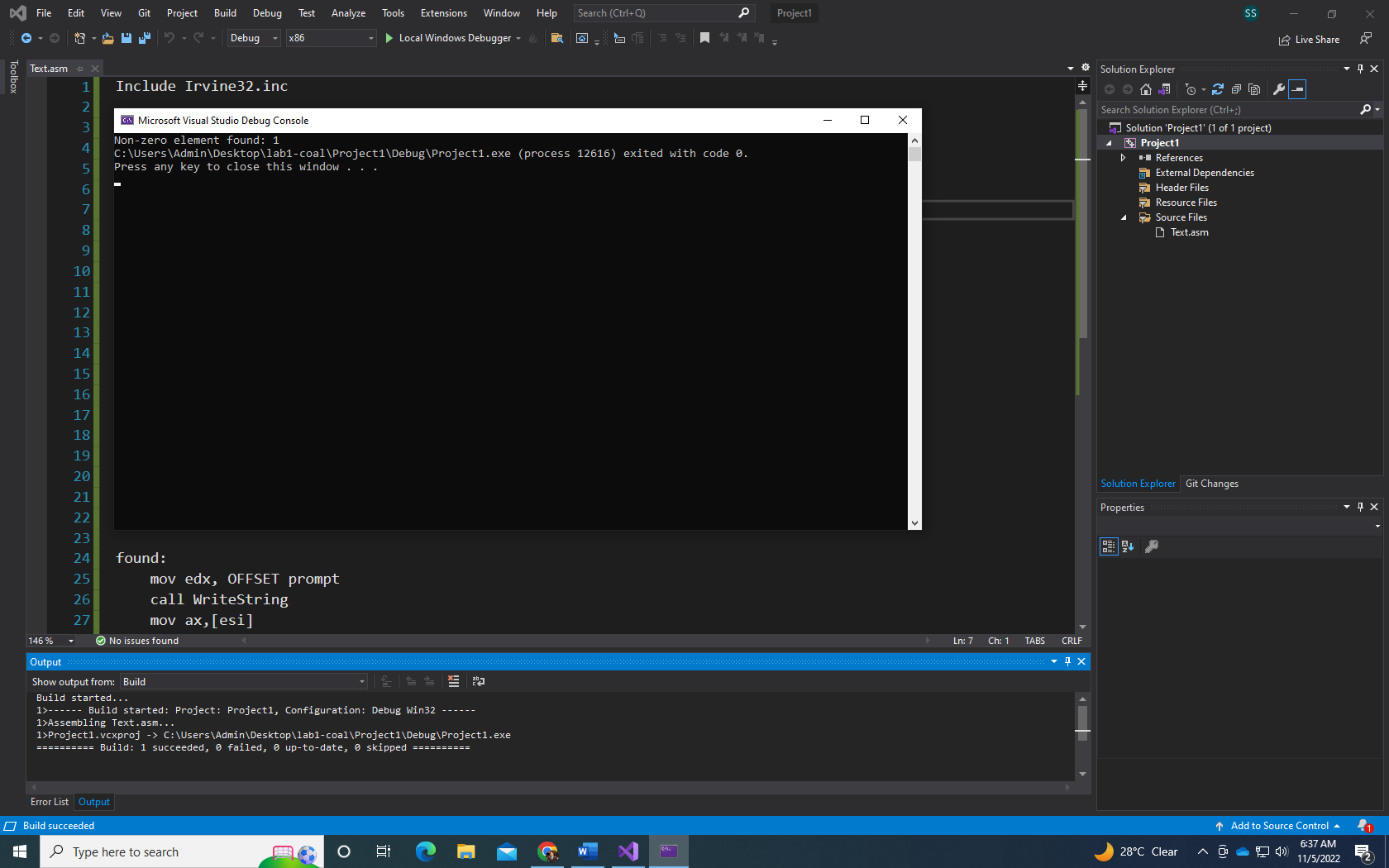
call WriteDec

exit

main ENDP

END main

Q2.



Include Irvine32.inc

.data

intArr SWORD 0,0,0,0,1,20,35,-12,66,4,0

prompt BYTE "Non-zero element found: ",0

required SWORD 0

.code

main proc

mov eax,0

mov esi,OFFSET intArr

mov ecx, LENGTHOF intArr

l1:

mov ax,[esi]

CMP ax,required

JNZ found

add esi, TYPE intArr

loop l1

jmp notFound

found:

mov edx, OFFSET prompt

call WriteString

mov ax,[esi]

call WriteDec

exit

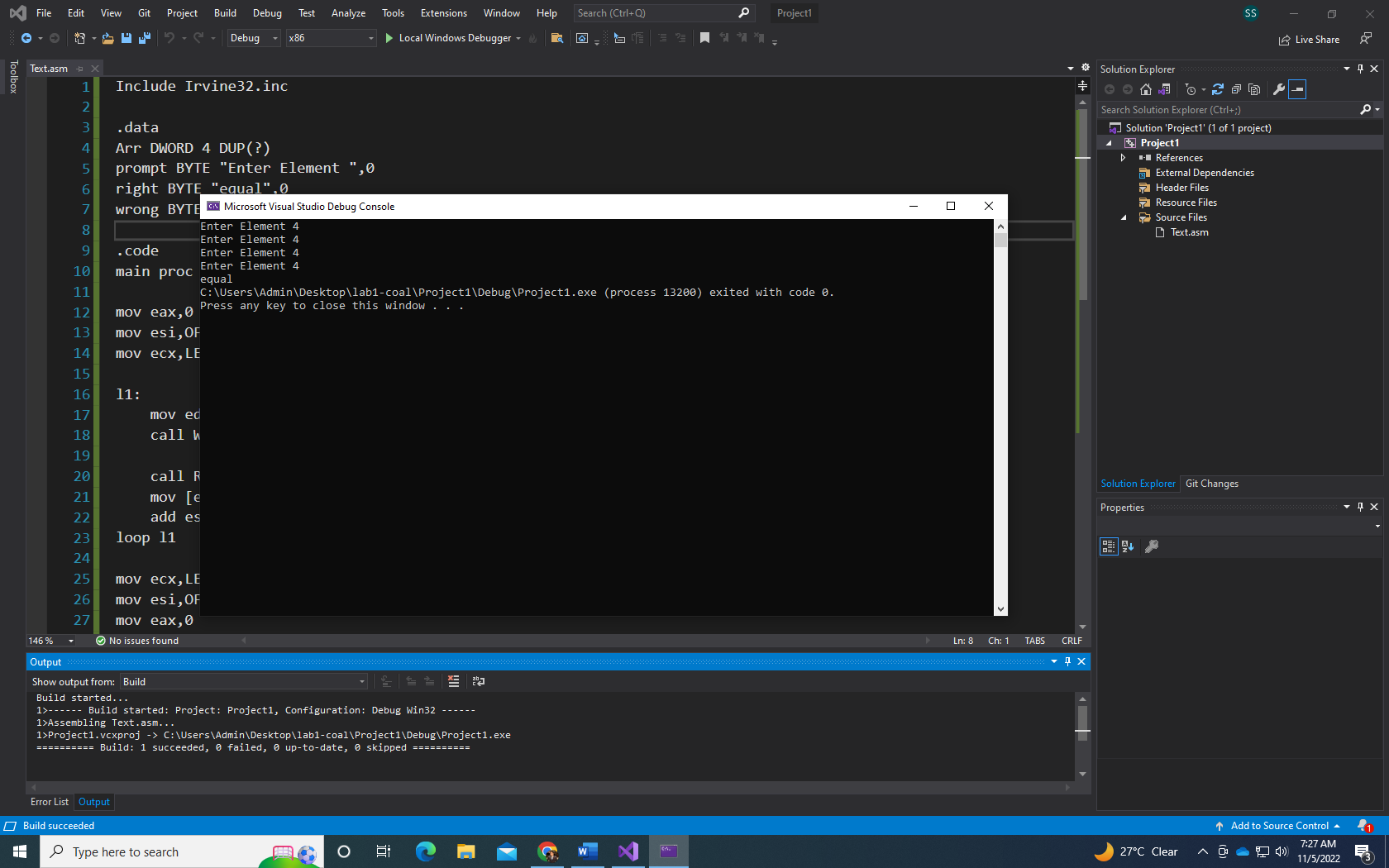
notFound:

exit

main ENDP

END main

Q3.



Include Irvine32.inc

.data

Arr DWORD 4 DUP(?)

prompt BYTE "Enter Element ",0

right BYTE "equal",0

wrong BYTE "unequal",0

.code

main proc

mov eax,0

mov esi,OFFSET Arr

mov ecx,LENGTHOF Arr

l1:

mov edx,OFFSET prompt

call WriteString

call ReadInt

mov [esi],eax

add esi, TYPE Arr

loop l1

mov ecx,LENGTHOF Arr-1

mov esi,OFFSET Arr

mov eax,0

l2:

mov eax,[esi]

CMP eax,[esi + TYPE Arr]

JNE unEqual

add esi,TYPE Arr

loop l2

Equal:

mov edx, OFFSET right

call WriteString

exit

unEqual:

mov edx,OFFSET wrong

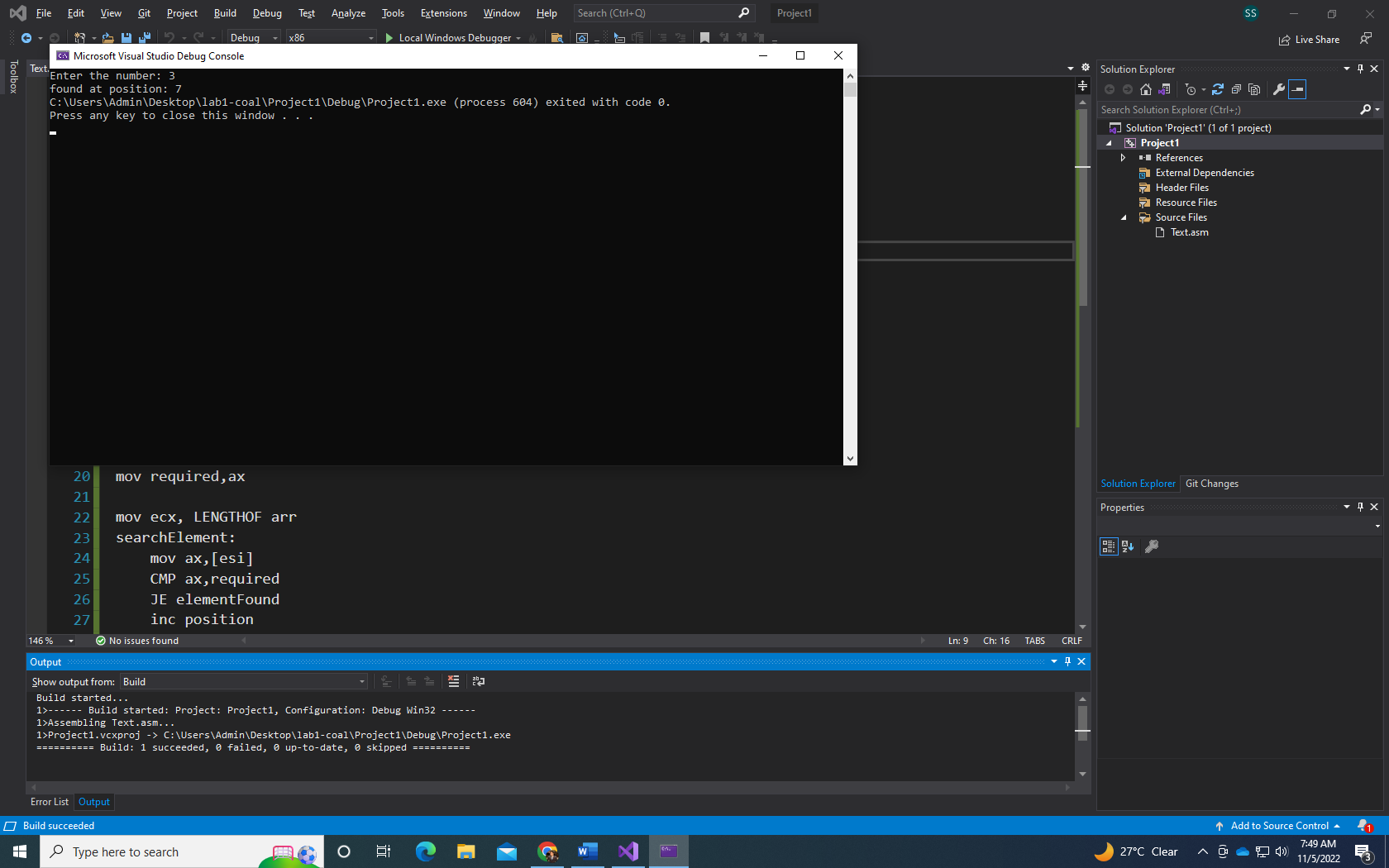
call WriteString

exit

main ENDP

END main

Q4.



Include Irvine32.inc

.data

arr WORD 10,4,7,14,299,156,3,19,29,300,20

question BYTE "Enter the number: ",0

prompt BYTE "found at position: ",0

prompt1 BYTE "Not found...",0

position WORD 1

required WORD ?

.code

main proc

mov eax,0

mov esi,OFFSET arr

mov edx,OFFSET question

call WriteString

call ReadInt

mov required,ax

mov ecx, LENGTHOF arr

searchElement:

mov ax,[esi]

CMP ax,required

JE elementFound

inc position

add esi, TYPE arr

loop searchElement

notFound:

mov edx,OFFSET prompt1

call WriteString

exit

elementFound:

mov edx,OFFSET prompt

call WriteString

mov ax,position

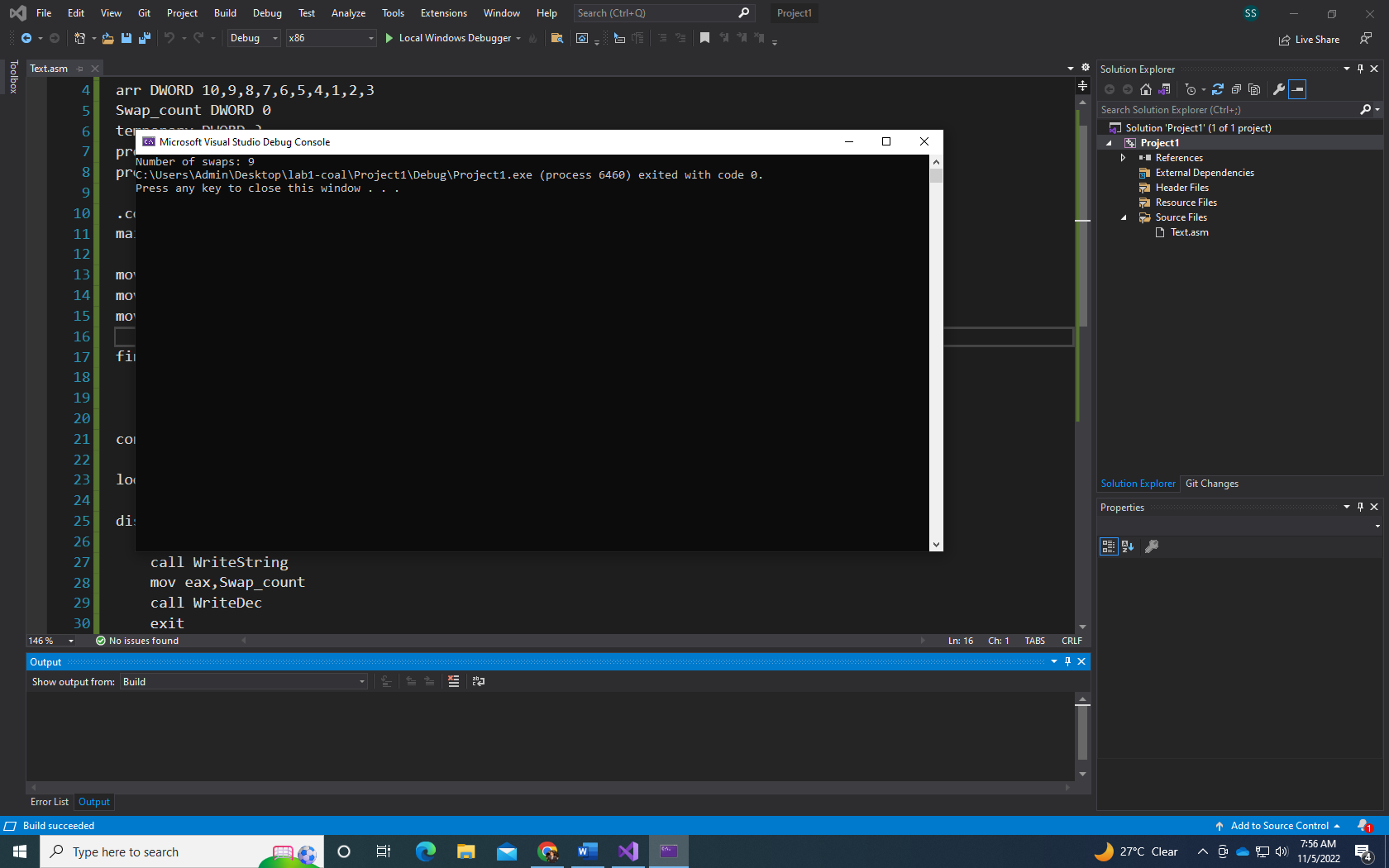
call WriteDec

exit

main ENDP

END main

Q5.



Include Irvine32.inc

.data

arr DWORD 10,9,8,7,6,5,4,1,2,3

Swap\_count DWORD 0

temporary DWORD ?

prompt BYTE "Number of swaps: ",0

prompt1 BYTE "Last value: ",0

.code

main proc

mov eax,0

mov esi,OFFSET arr

mov ecx,LENGTHOF arr-1

findSwaps:

mov eax,[esi]

CMP eax,[esi + 4]

JA swap

continue:

add esi,4

loop findSwaps

display:

mov edx,OFFSET prompt

call WriteString

mov eax,Swap\_count

call WriteDec

exit

swap:

inc Swap\_count

mov temporary,eax

mov ebx,[esi+4]

mov [esi],ebx

mov [esi+4],eax

jmp continue

exit

main ENDP

END main