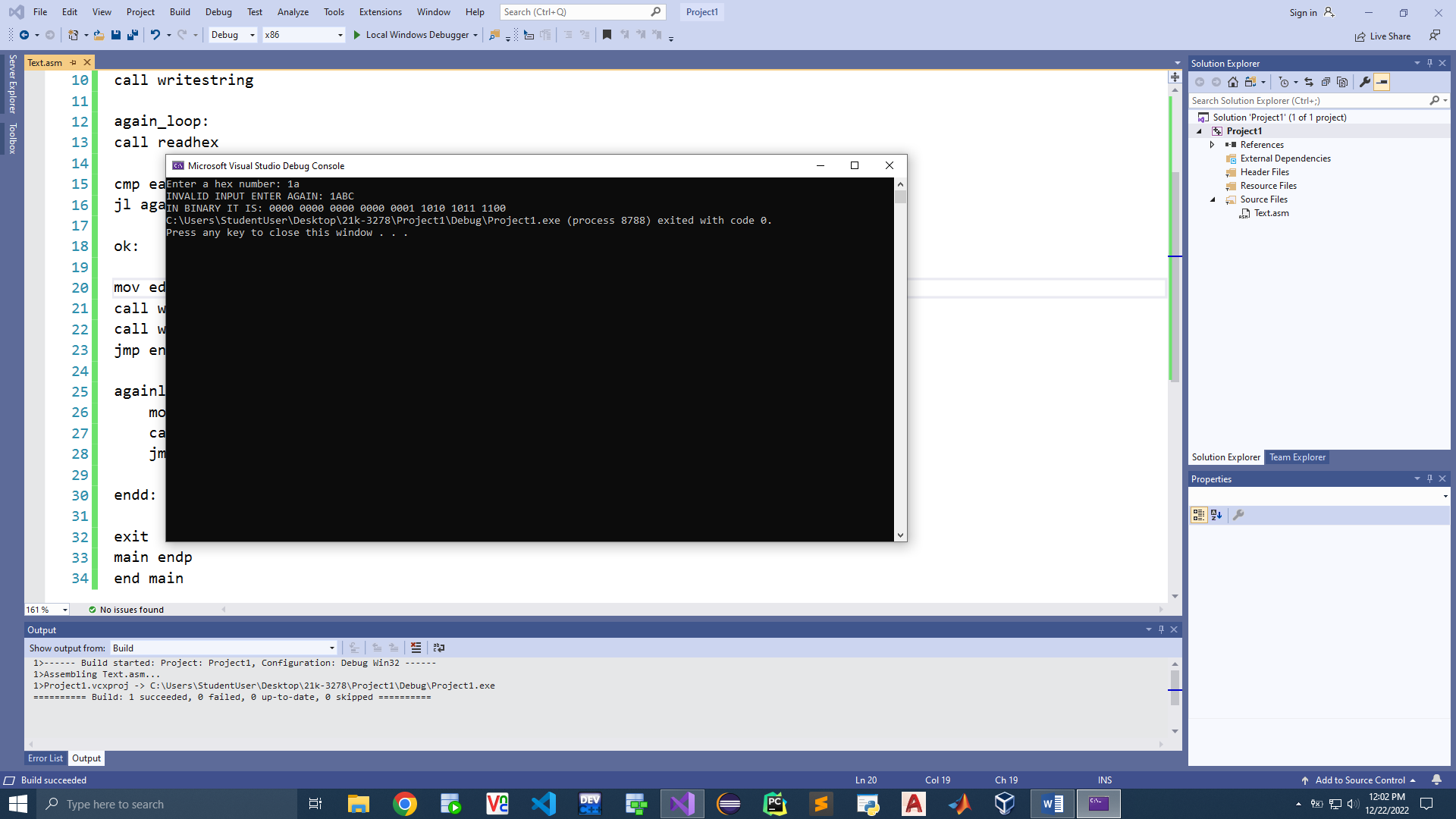
**21K-3278-D**

**Sohaib Sarosh Shamsi**

**Q1.**



Include Irvine32.inc

.data

p1 byte "Enter a hex number: ", 0

p2 byte "INVALID INPUT ENTER AGAIN: ", 0

p3 byte "IN BINARY IT IS: ", 0

.code

main proc

mov edx, offset p1

call writestring

again\_loop:

call readhex

cmp eax, 1111

jl againloop\_prompt

ok:

mov edx, offset p3

call writestring

call writebin

jmp endd

againloop\_prompt:

mov edx, offset p2

call writestring

jmp again\_loop

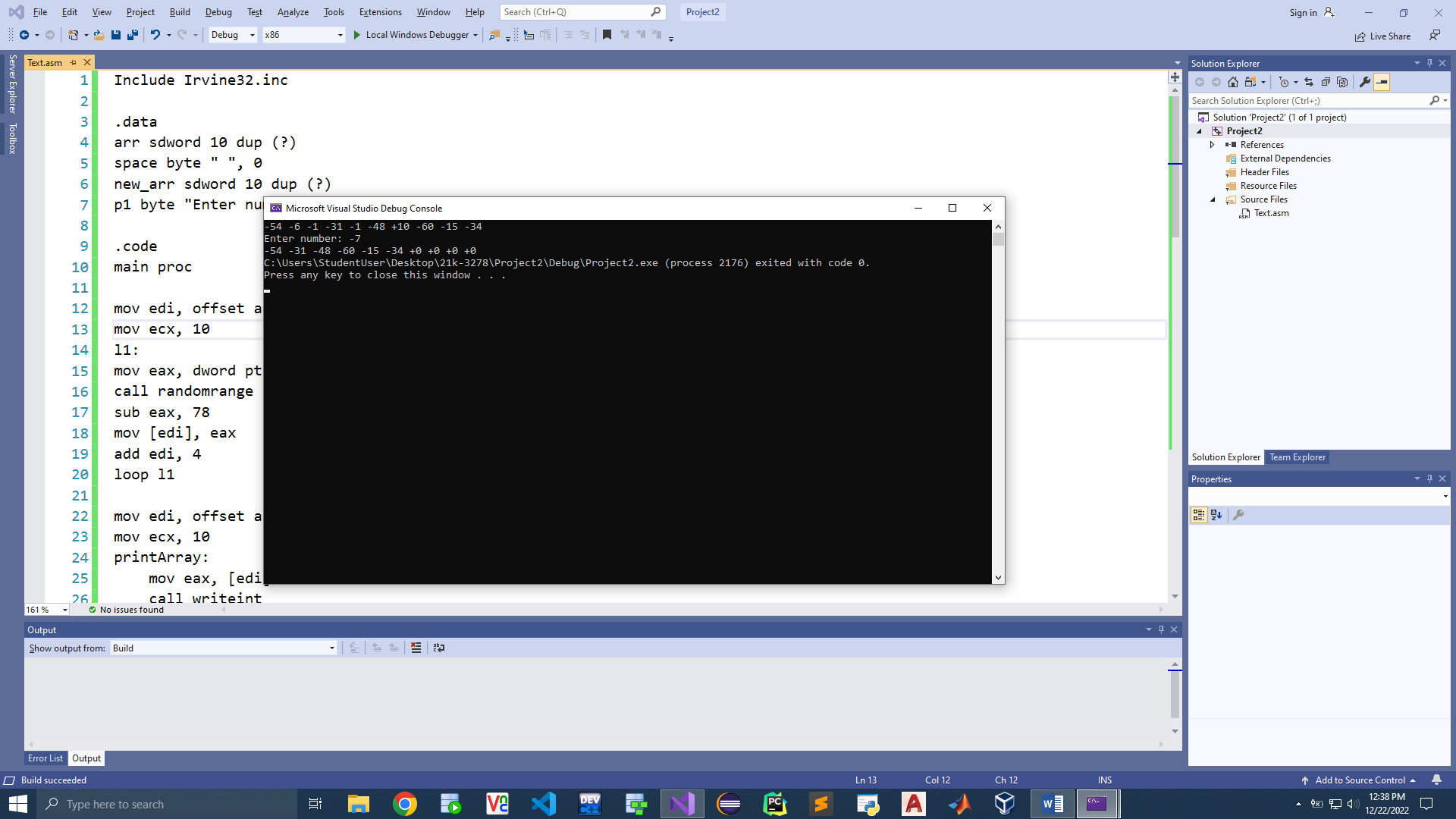
endd:

exit

main endp

end main

Q2.



Include Irvine32.inc

.data

arr sdword 10 dup (?)

space byte " ", 0

new\_arr sdword 10 dup (?)

p1 byte "Enter number: ", 0

.code

main proc

mov edi, offset arr

mov ecx, 10

l1:

mov eax, dword ptr 90 ; seed

call randomrange

sub eax, 78

mov [edi], eax

add edi, 4

loop l1

mov edi, offset arr

mov ecx, 10

printArray:

mov eax, [edi]

call writeint

mov edx, offset space

call writestring

add edi, 4

loop printArray

call crlf

mov edx, offset p1

call writestring

call readint

mov ebx, eax

mov edi, offset new\_arr

mov esi, offset arr

mov ecx, 10

CheckArray:

mov eax, [esi]

cmp ebx, eax

jl forwardd

mov [edi], eax

add edi, 4

forwardd:

add esi, 4

loop CheckArray

mov edi, offset new\_arr

mov ecx, 10

printArray1:

mov eax, [edi]

call writeint

mov edx, offset space

call writestring

add edi, 4

loop printArray1

exit

main endp

end main

**Q3.**

Include Irvine32.inc

.data

arr1 sdword 10 dup (?)

arr2 sdword 10 dup (?)

x dword ?

countt dword ?

.code

main proc

mov eax, lengthof arr1

mov x, eax

push eax ; len

push offset arr1

push offset arr2

call countmatches

; returns ebx with count

mov eax, ebx

call writeint

exit

main endp

countmatches proc

push ebp

mov ebp, esp

mov ecx, [ebp+16]

mov esi, [ebp+12]

mov edi, [ebp+8]

mov ebx, 0

l1:

mov eax, [esi]

cmp esi, [edi]

jne outt

add ebx, 1 ; count

outt:

add esi, 4

add edi, 4

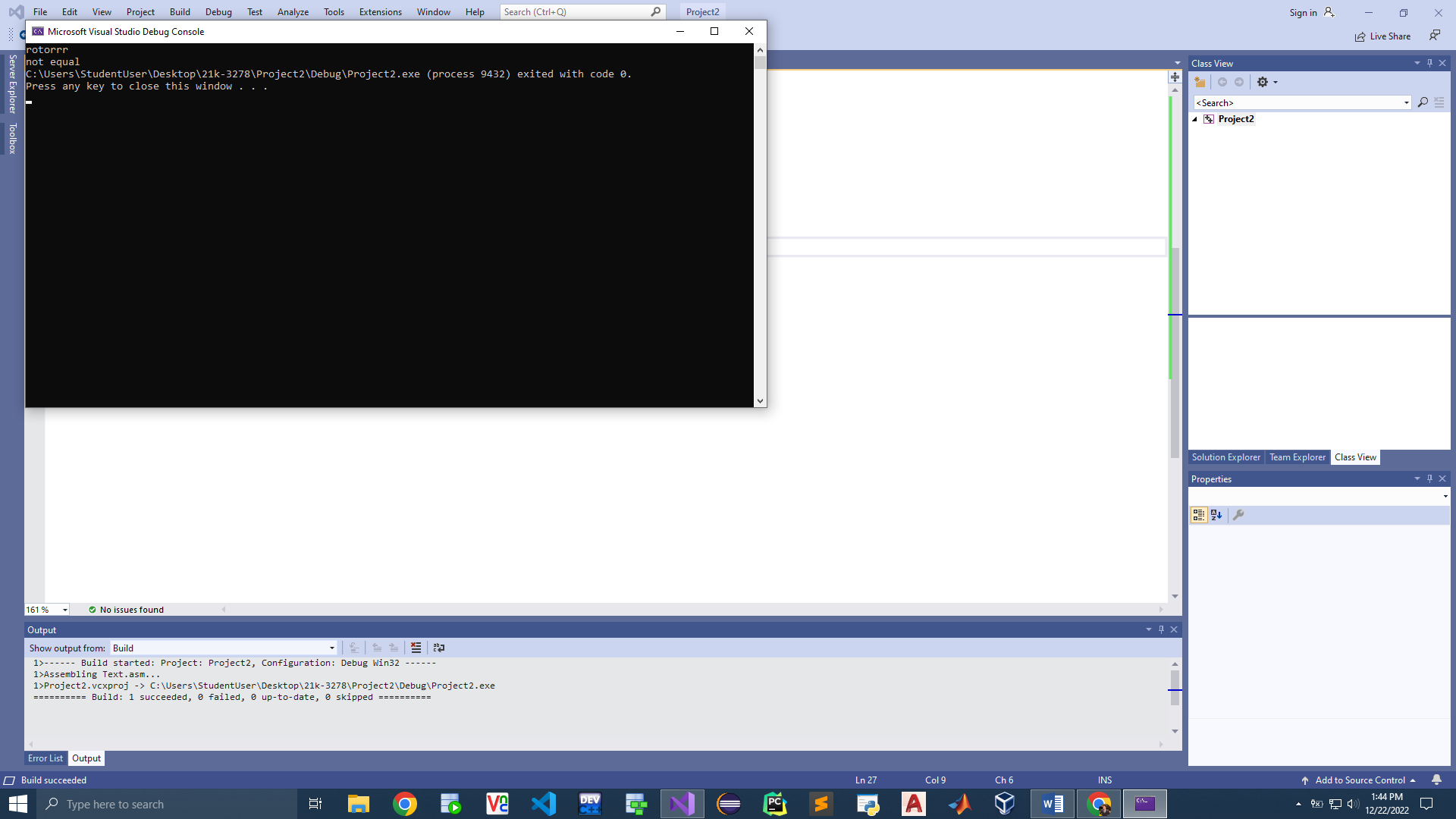
loop l1

ret

countmatches endp

end main

**Q4.**



Include Irvine32.inc

.data

p1 byte "rotorrr", 0

p2 byte "not equal", 0

sizee dword 7

.code

main proc

mov edx, offset p1

call writestring

mov esi, offset p1

mov edi, offset p2

mov ecx, sizee

l1:

mov eax, [esi]

mov ebx, [edi]

cmp eax, ebx

jne exitt

loop l1

exitt:

call crlf

mov edx, offset p2

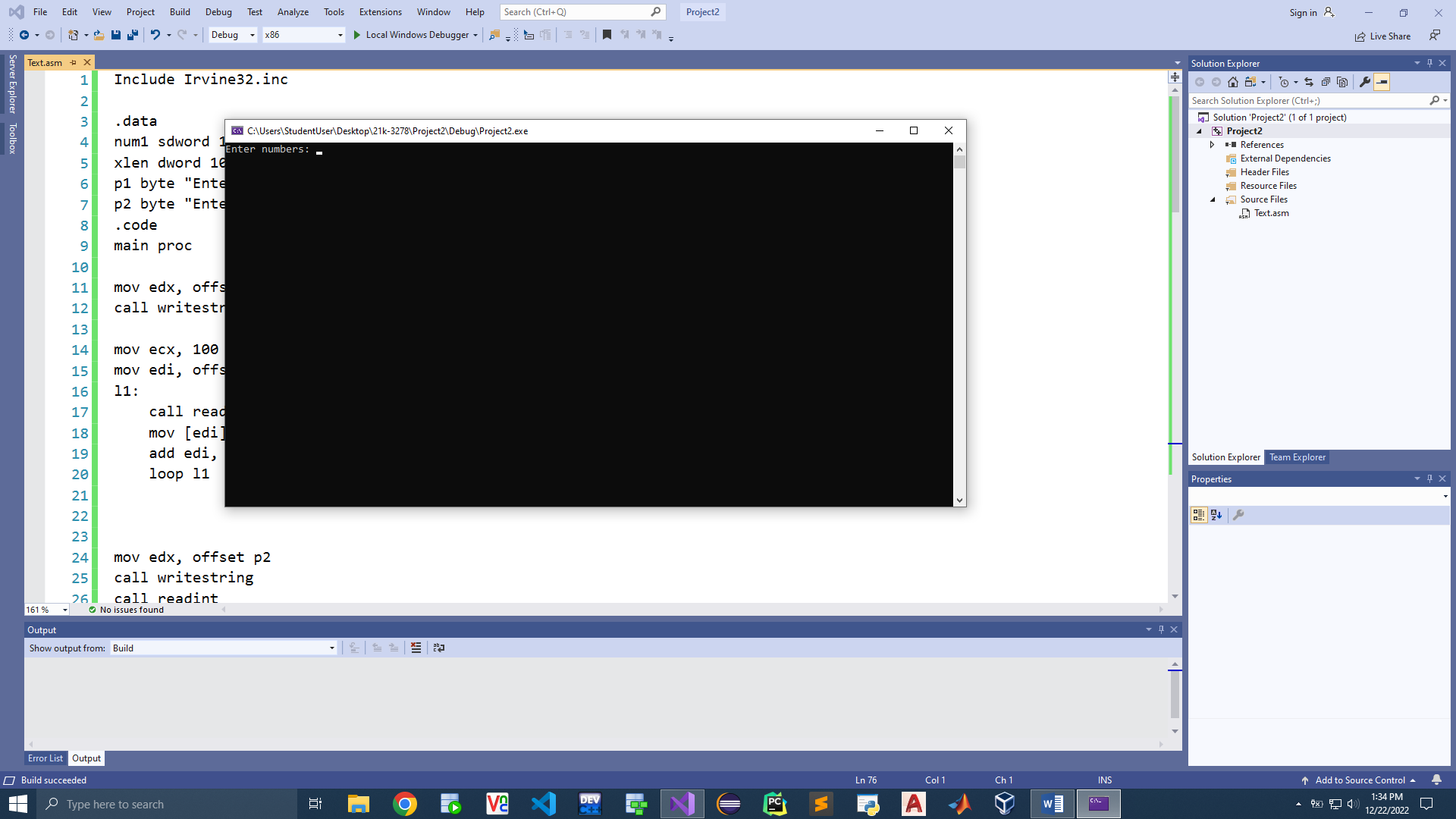
call writestring

exit

main endp

end main

**Q5.**



Include Irvine32.inc

.data

num1 sdword 100 dup (?)

xlen dword 100

p1 byte "Enter numbers: ", 0

p2 byte "Enter target: ", 0

.code

main proc

mov edx, offset p1

call writestring

mov ecx, 100

mov edi, offset num1

l1:

call readint

mov [edi], eax

add edi, 4

loop l1

mov edx, offset p2

call writestring

call readint

push eax ; target

push xlen

push offset num1

call binarySearch1

exit

main endp

binarysearch1 proc

push ebp

mov ebp, esp

mov eax, [ebp+16]

mov ecx, [ebp+12]

mov edi, [ebp+8]

mov ebx, 0

sub eax, 1

add ebx, eax

mov edx, 2

div edx

l1:

sub ebx, eax

cmp ebx, eax

jge endd

cmp [edi], eax

call swap

mov ebx, eax

jge l1

cmp [edi], eax

call swap

mov eax, ebx

jle l1

loop l1

endd:

ret

binarysearch1 endp

swap proc uses ebx

mov ebx, eax

mov eax, [edi]

mov [edi], ebx

swap endp

end main