**Haseeb Tariq**

**232506**

**Computer Organization an Assembly Language**

**Question 1**

**Code:**INCLUDE Irvine32.Inc

.data

readings BYTE 1, 10, 18, 24, 55, 66, 67, 68, 69, 73

threshold BYTE 70

maximumValue BYTE ?

warnmsg BYTE "caution!!: Sensor reading has exceeded threshold. Increased Value is: ", 0

.code

main PROC

mov esi, OFFSET readings

mov ecx, LENGTHOF readings

xor eax,eax

findMaximum:

mov bl, [esi]

cmp bl, al

jbe skip

mov al, bl

skip:

inc esi

loop findMaximum

mov maximumValue, al

cmp al, threshold

jbe done

mov edx, OFFSET warnmsg

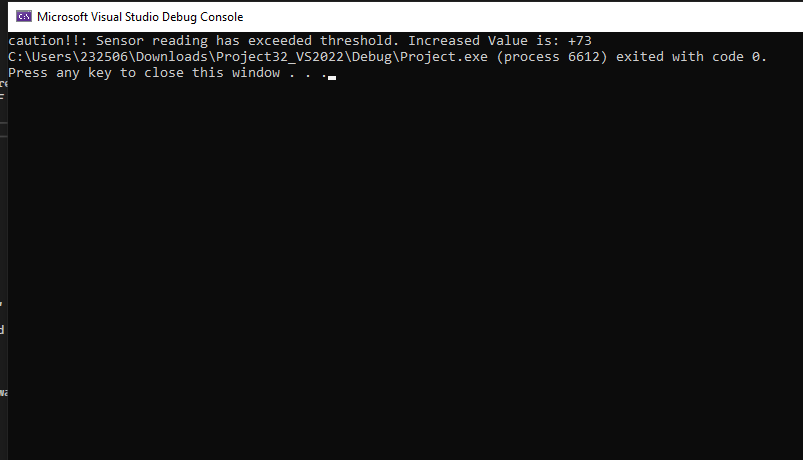
call WriteString

call WriteInt

done:

exit

main ENDP

END main  
  
  
**Output:**  


**Question 2:**Code:  
INCLUDE Irvine32.Inc

.data

productSerial BYTE 1, 2, 3, 4, 5

defectiveProductlist BYTE 2, 4, 5

totalProducts DWORD 5

totaldefectivesProducts DWORD 3

defectivemsg BYTE"PRoduct is defective: " ,0

productOkmsg BYTE "Product is OK: ",0

.code

main PROC

mov ecx, totalProducts

mov esi, 0 ; index for productSerial

product\_loop:

movzx eax, productSerial[esi]

push eax

call checkProductDefective

pop eax

cmp bl, 1

je isProductDefective

mov edx, OFFSET ProductOKmsg

call WriteString

movzx eax, productSerial[esi]

call WriteInt

call Crlf

jmp next\_item

isProductDefective:

mov edx, OFFSET defectivemsg

call WriteString

movzx eax, productSerial[esi]

call WriteInt

call Crlf

next\_item:

inc esi

loop product\_loop

exit

main ENDP

checkProductDefective PROC

push ecx

push esi

mov bl, 0

mov ecx, totaldefectivesProducts

mov esi, 0

check\_loop:

movzx edx, defectiveProductlist[esi]

cmp al, dl

je foundDefectiveProducts

inc esi

loop check\_loop

jmp done

foundDefectiveProducts:

mov bl, 1

done:

pop esi

pop ecx

ret

checkProductDefective ENDP

END main

OUTPUT:  
