**COAL LAB 8**

Questions:

Task#1

Write a program that takes four input integers from the user. Then compare and display a

message whether these integers are equal or not.

Task#2

Use cmp and jumps to find the first non-zero value in the given array:

intArr SWORD 0, 0, 0, 150, 120, 35, -12, 66, 4, 0

Task#3 Implement the following given code in Assembly and Consider var = 5 , edx =

var+1 and counter value from array initialized in task#2.

if ( var < ecx ) AND (ecx >= edx) then

x = 0

else

x = 1

Task#4 Implement the following given code in Assembly and consider var = 0.

while ( var <= 10)

if (var < 5)

Print “Hello”

else

Print “World”

var = var + 1

end while

Task#5

Write a program for sequential search. Take an input from the user and find if it occurs in

the following array:

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

Task#6

Write a program for bubble sort on this array.

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

Task#7

Write a program to print weekday based on given number.

Task#8

Write a program to check whether a character is alphabet or not.

Q1

Code:

INCLUDE Irvine32.inc

.data

arr DWORD 4 DUP(?)

str1 BYTE "Enter integer: " ,0

str2 BYTE "Intergers are equal" , 0

str3 BYTE "Intergers are not equal" , 0

.code

main PROC

mov eax , 0

mov ecx , 4

mov esi , offset arr

;taking input from the user

L1:

mov edx , offset str1

call Writestring

call Readint

mov [esi] , eax

add esi , 4

loop L1

;comparing values

mov eax , 0

mov esi , offset arr

mov eax , [esi]

mov ecx , 3

L2:

add esi , 4

cmp eax , [esi] ; if euqal then ZF = 1

jnz L3 ; if ZF = 0 jump t0 L3

loop L2

mov edx , offset str2

call Writestring

exit

L3:

mov edx , offset str3

call Writestring

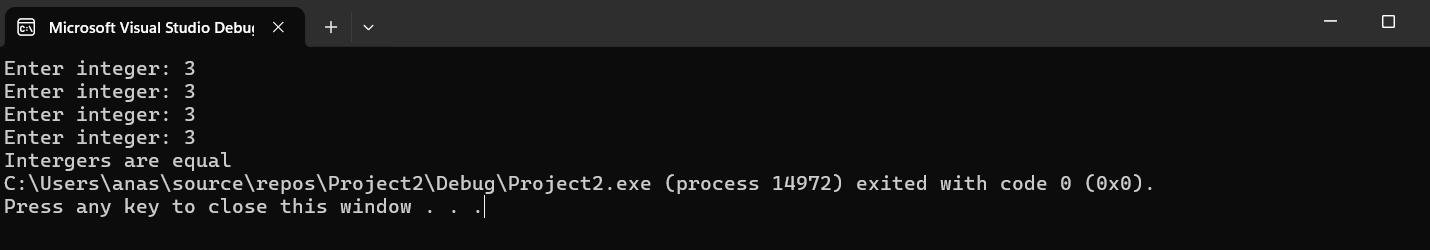
exit

call Dumpregs

exit

main ENDP

END main



Q2

Code:

INCLUDE Irvine32.inc

.data

intArr SWORD 0, 0, 0, 150, 120, 35, -12, 66, 4, 0

.code

main PROC

mov eax , 0

mov ecx , LENGTHOF intArr

mov esi , 0

mov ax , intArr[esi]

L1:

add esi , 2

cmp ax , intArr[esi] ;

jnz L2 ; jump to L3 if ZF = 0

loop L1

exit

L2:

mov ax , intArr[esi]

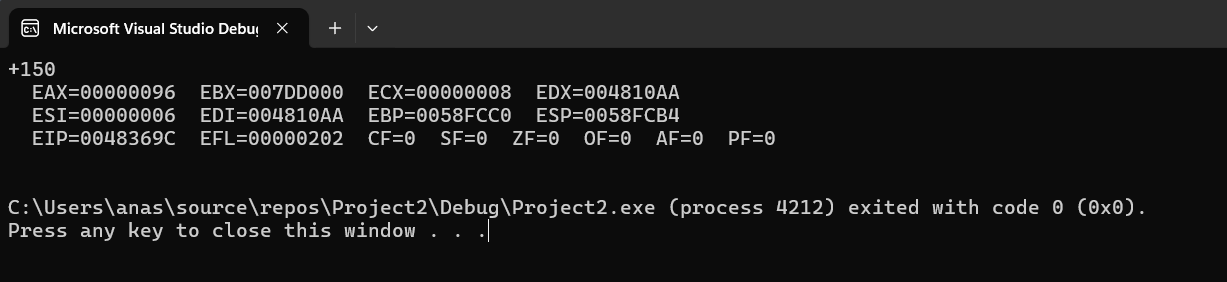
call Writeint

call Dumpregs

exit

main ENDP

END main



Q3

Code:

INCLUDE Irvine32.inc

.data

var DWORD 5

x DWORD ?

.code

main PROC

mov ecx , 10 ; from task 2

cmp var , ecx

jb L1

mov x , 1

mov eax , x

call Writeint

exit

L1:

inc var

mov edx , var

cmp ecx , edx

jae L2

mov x , 1

mov eax , x

call Writeint

exit

L2:

mov x , 0

mov eax , x

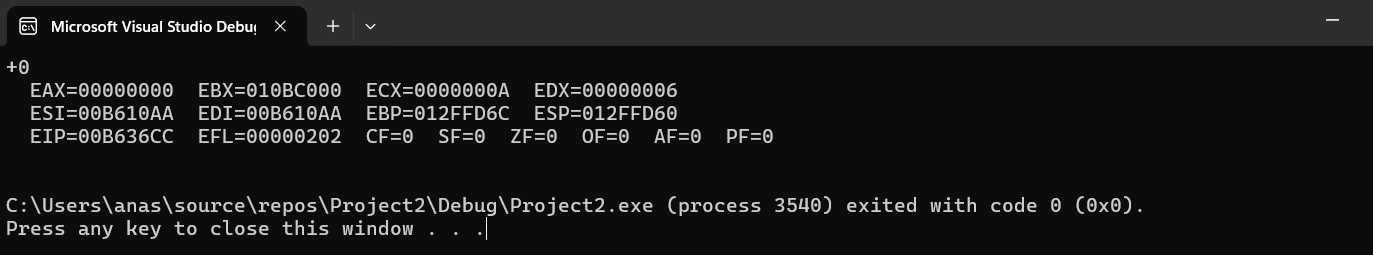
call Writeint

call Dumpregs

exit

main ENDP

END main



Q4

Code:

INCLUDE Irvine32.inc

.data

var DWORD 0

str1 BYTE "Hello" , 0

str2 BYTE "World" , 0

.code

main PROC

mov eax , 0

;mov ecx , 10 ; set the counter for while loop

L1:

cmp var , 10

jbe L2

exit

L2:

cmp var , 5

jb L3

mov edx , offset str2

call Writestring

call crlf

inc var

jmp L1

L3:

mov edx , offset str1

call Writestring

call crlf

inc var

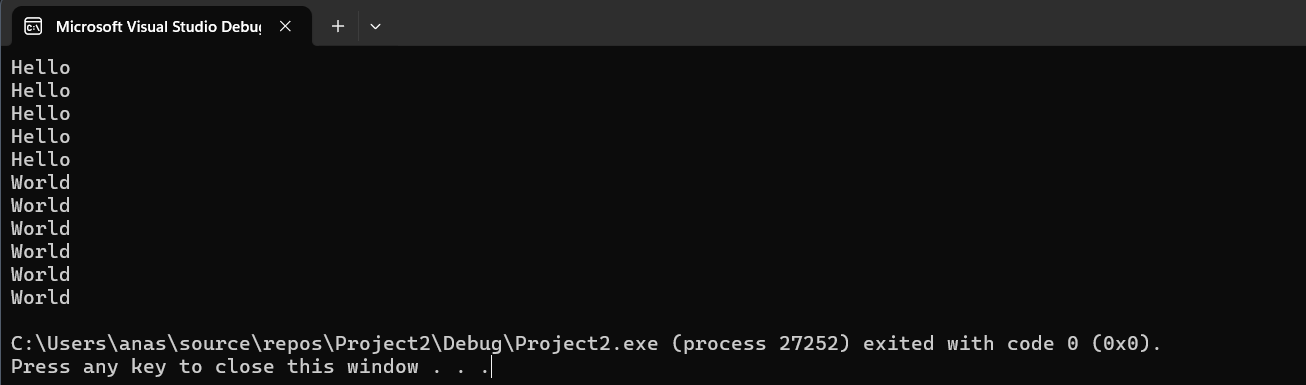
jmp L1

call Dumpregs

exit

main ENDP

END main



Q5

Code:

INCLUDE Irvine32.inc

.data

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

str1 BYTE "Enter the number you want to find in the array: " , 0

str2 BYTE "Value found" , 0

str3 BYTE "Value not found" , 0

.code

main PROC

mov eax , 0

mov edx , offset str1

call Writestring

call Readint

mov ecx , LENGTHOF arr

mov esi , 0

L1:

movzx ebx , arr[esi \* TYPE WORD]

cmp eax , ebx

je found

inc esi

loop L1

mov edx , offset str3

call Writestring

exit

found:

mov edx , offset str2

call Writestring

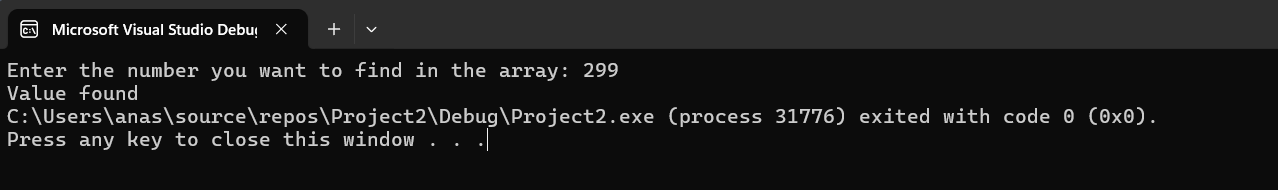
exit

call Dumpregs

exit

main ENDP

END main



Q6

INCLUDE Irvine32.inc

.data

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

arrSize = ($ - arr) / 2

.code

main PROC

mov eax , 0

mov ecx , LENGTHOF arr

mov esi , 0

outerloop:

dec ecx

jz endSort

mov esi , 0

innerloop:

mov ax , arr[esi]

mov bx , arr[esi+2]

cmp ax , bx

jle noswap

;swapping

mov arr[esi] , bx

mov arr[esi+2] , ax

noswap:

add esi , 2

cmp esi, (arrSize - 1) \* 2 ; to check kahi hm end pr tou nhi phooch gai

jl innerloop

jmp outerloop

mov ecx , arrSize

mov esi , 0

endSort:

mov ax , arr[esi]

call Writeint

call crlf

add esi , 2

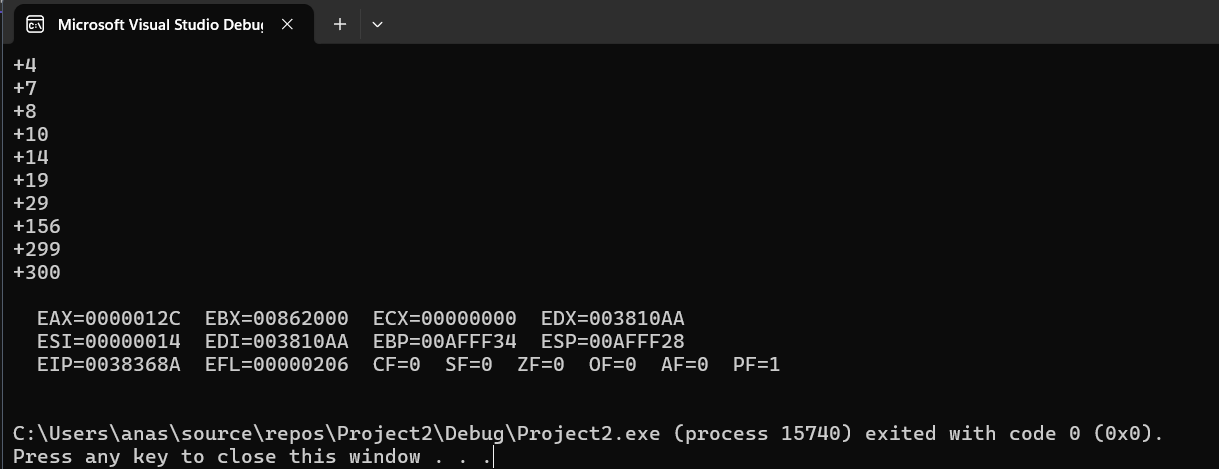
loop endSort

call Dumpregs

exit

main ENDP

END main



Q7

Code:

INCLUDE Irvine32.inc

.data

weekdays BYTE "Sunday", 0, "Monday", 0, "Tuesday", 0, "Wednesday", 0, "Thursday",

0, "Friday", 0, "Saturday", 0

str1 BYTE "Enter the number(1-7): " ,0

invalid BYTE "Invalid Input"

.code

main PROC

mov eax , 0

mov edx , offset str1 ;

call Writestring

call Readint

cmp eax , 1

jb InValid

cmp eax , 7

ja InValid

dec eax

mov ecx , 0

mov ebx , eax

findWeekday:

mov al , weekdays[ecx]

cmp al , 0

je CheckIndex

inc ecx

jmp findWeekday

checkIndex:

cmp ebx, 0

je printWeekday

dec ebx

inc ecx

jmp findWeekday

printWeekday:

mov edx, OFFSET weekdays[ecx]

call WriteString

call Crlf

exit

InValid:

mov edx, OFFSET invalid

call WriteString

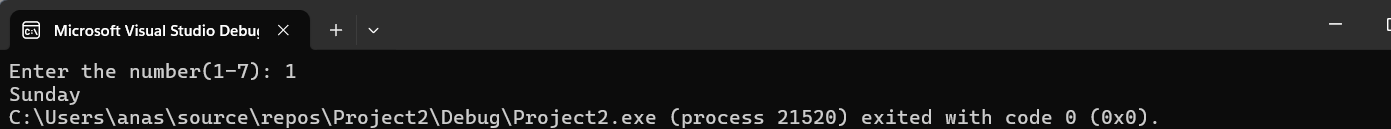
call Crlf

call Dumpregs

exit

main ENDP

END main



Q8

Code:

INCLUDE Irvine32.inc

.data

str1 BYTE "Enter the character: " , 0

str2 BYTE "This is not a character" , 0

str3 BYTE "This is a character" , 0

.code

main PROC

call clrscr

mov eax , 0

mov edx , offset str1

call Writestring

call Readchar ; stored in al

call Writechar

call crlf

mov al , al

cmp al , 'A'

jl isnotAlpha

cmp al , 'Z'

jle isAlpha

isAlpha:

mov edx , offset str3

call Writestring

exit

isnotAlpha:

mov edx , offset str2

call Writestring

call Dumpregs

exit

main ENDP

END main

