**LAB 11**

Questions:

1. Create a procedure named Scan\_String to find the index of the first occurrence of the character

‘#’ in the given string.

Str1 BYTE ‘127&j~3#^&\*#\*#45^’,0

2. Modify the above procedure to take offset of string1 and the character to be searched

as argument.

3. Create IsCompare procedure to compare two strings.

4. Create a Str\_Reverse procedure to reverse strings.

5. Create a procedure that Loads an array of integer by multiplying it with

Load(offset array, byte no)

Q1

INCLUDE Irvine32.inc

.data

Str1 BYTE "127&j~3#^&\*#\*#45^",0

str2 BYTE "Character found" , 0

str3 BYTE "Character not found" , 0

.code

main PROC

mov ebx , 0

Scan\_String PROC

mov edi , offset Str1

mov al , '#'

mov ecx , LENGTHOF Str1

cld ; forward direction

inc ebx

repne SCASB ; repeat while not equal

jnz quit

found:

mov edx , offset str2

call Writestring

call crlf

mov eax , LENGTHOF Str1

sub eax , ecx

dec eax

call Writeint

exit

quit:

mov edx , offset str3

call Writestring

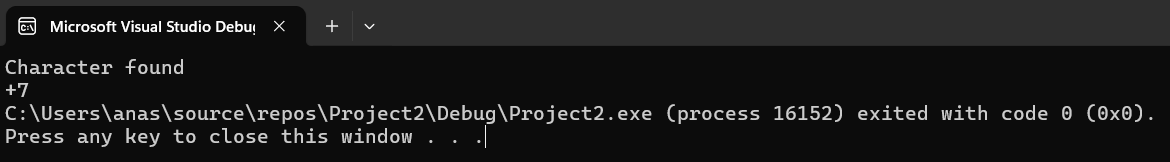
ret

Scan\_String ENDP

exit

main ENDP

END main



Q2

INCLUDE Irvine32.inc

.data

Str1 BYTE "127&j~3#^&\*#\*#45^",0

str2 BYTE "Character found" , 0

str3 BYTE "Character not found" , 0

.code

main PROC

Scan\_String PROTO, str4:PTR BYTE , charac:BYTE , len:DWORD

INVOKE Scan\_String , ADDR Str1 , '#' , LENGTHOF Str1

exit

main ENDP

Scan\_String PROC , string1: PTR BYTE , character:BYTE , Leng:DWORD

mov edi , string1

mov al , character

mov ecx , Leng

cld ; forward direction

repne SCASB ; repeat while not equal

jnz quit

found:

mov edx , offset str2

call Writestring

call crlf

mov eax , LENGTHOF Str1

sub eax , ecx

dec eax

call Writeint

exit

quit:

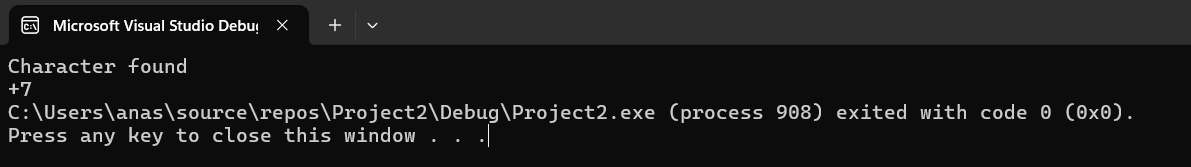
mov edx , offset str3

call Writestring

ret

Scan\_String ENDP

END main



Q3

INCLUDE Irvine32.inc

.data

str1 BYTE "abc",0

str2 BYTE "ab" , 0

greater BYTE "source > target",0

lessOrEqual BYTE "source < target",0

.code

main PROC

call IsCompare

exit

main ENDP

IsCompare PROC

mov esi , offset str1

mov edi , offset str2

CMPSB

ja L1

mov edx , offset lessOrequal

call Writestring

jmp quit

L1:

mov edx , offset greater

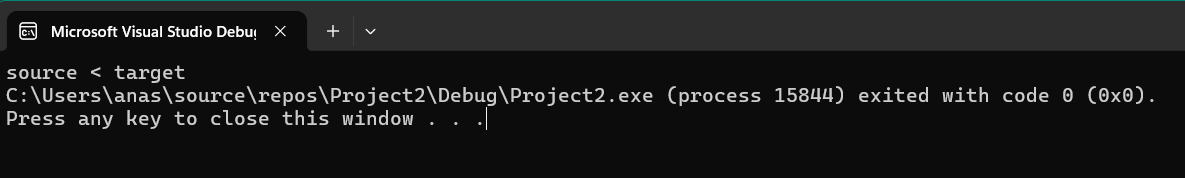
call Writestring

quit:

ret

IsCompare ENDP

END main



Q4

INCLUDE Irvine32.inc

.data

str1 BYTE "Hello",0

.code

main PROC

call Str\_reverse

exit

main ENDP

Str\_reverse PROC

mov esi , offset str1

mov edi , offset str1

add edi , LENGTHOF str1-2 ; mov edi to the end(-2 because of null terminator)

mov ecx , LENGTHOF str1/2 ; only need to process half of the string

L1:

mov al , [esi]

mov bl , [edi]

mov [esi] , bl

mov [edi] , al

inc esi

dec edi

loop L1

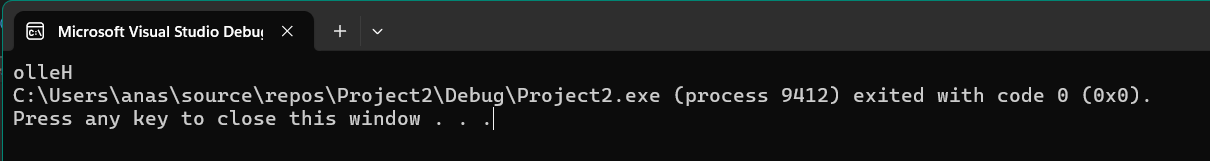
mov edx , offset str1

call Writestring

ret

Str\_reverse ENDP

END main



Q5

INCLUDE Irvine32.inc

.data

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

.code

main PROC

call arrMul

;print array

mov ecx , LENGTHOF array

mov esi , 0

L1:

mov eax , array[esi \*TYPE array]

call Writeint

call crlf

inc esi

loop L1

exit

main ENDP

arrMul PROC

mov esi , offset array

mov edi , esi

mov ecx , LENGTHOF array

mov ebx , (SIZEOF array)/10

L1:

lodsd ; [esi] stored in eax

mul ebx

stosd ; eax stored back into array pointed by edi this time

loop L1

ret

arrMul ENDP

END main

