**COAL LAB#12 TASKS**

Name: Saman Khan

ID: 19K-0354

Section: H

**TASK#01:**

TITLE My First Program (Test.asm)

INCLUDE Irvine32.inc

.data

count BYTE 1

Str1 BYTE '127&amp;j~3#^&amp;\*#\*#45^',0

msg1 BYTE "Index is : ",0

msg2 BYTE "# NOT FOUND!",0

.code

main PROC

enter 0,0

call Scan\_String

leave

ret

exit

main ENDP

Scan\_String PROC

enter 0,1

mov al, '#'

mov edi, OFFSET Str1

mov ecx, LENGTHOF Str1

cld

L1:

scasb

je print

inc count

LOOP L1

mov edx, offset msg2

call writestring

jmp last

print:

mov edx, offset msg1

call writestring

mov al, count

movzx eax, al

call writedec

last:

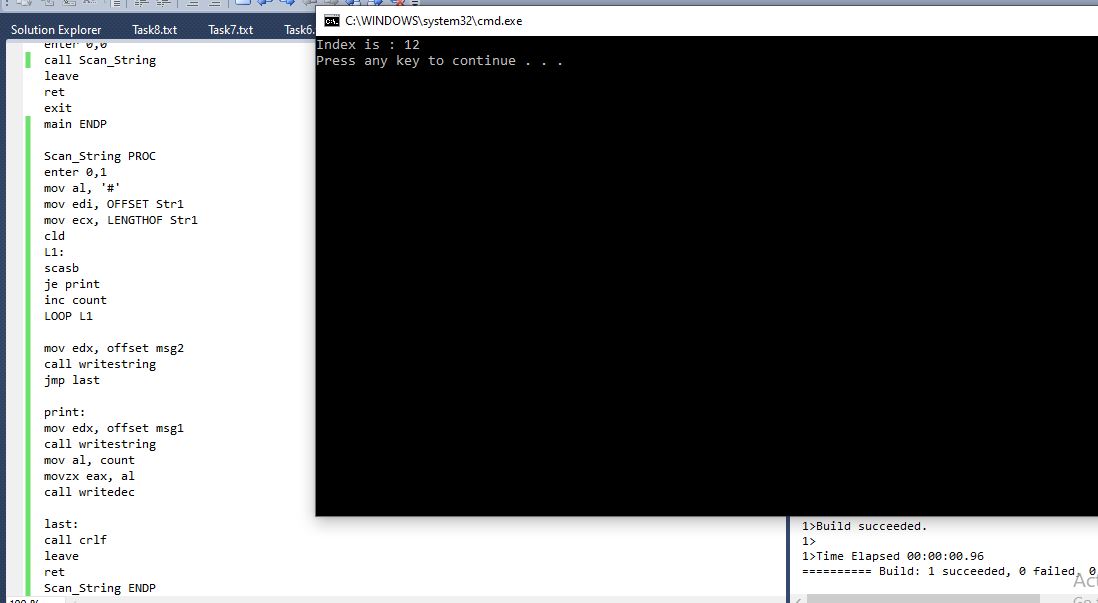
call crlf

leave

ret

Scan\_String ENDP

end main



**PS: The output (in task 1 and task2) is 12 because I‘m assuming that my index is starting from 1, otherwise it would’ve been zero I my count starts from 0.**

**TASK#02:**

TITLE My First Program (Test.asm)

INCLUDE Irvine32.inc

.data

count BYTE 1

Str1 BYTE '127&amp;j~3#^&amp;\*#\*#45^',0

msg1 BYTE "Index is : ",0

msg2 BYTE "# NOT FOUND!",0

Scan\_String PROTO

.code

main PROC

enter 0,0

push offset Str1

push '#'

INVOKE Scan\_String

leave

ret

exit

main ENDP

Scan\_String PROC

enter 0,1

mov al, [ebp+8]

mov edi, [ebp+12]

mov ecx, LENGTHOF Str1

cld

L1:

scasb

je print

inc count

LOOP L1

mov edx, offset msg2

call writestring

jmp last

print:

mov edx, offset msg1

call writestring

mov al, count

movzx eax, al

call writedec

last:

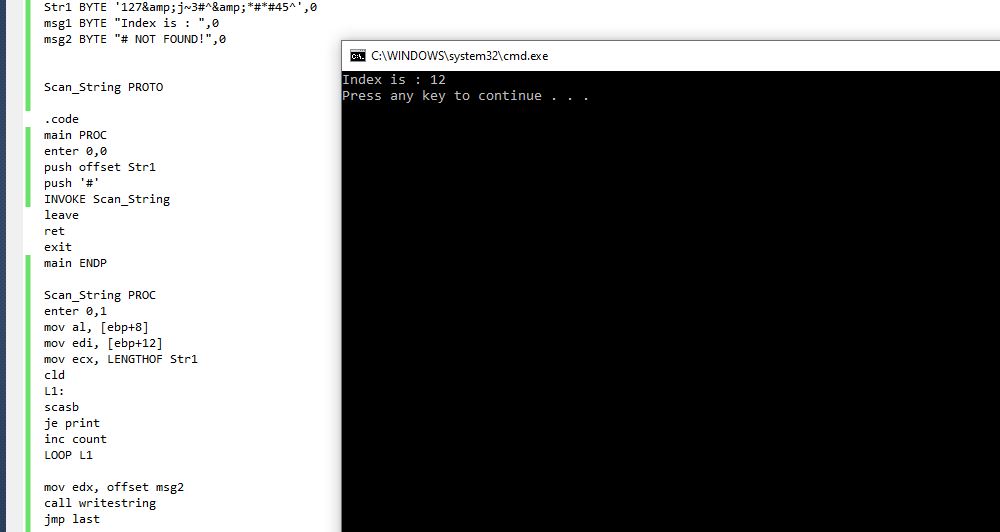
call crlf

leave

ret

Scan\_String ENDP

end main



**TASK#03:**

TITLE My First Program (Test.asm)

INCLUDE Irvine32.inc

.data

Str1 BYTE "Saman Khan",0

Str2 BYTE "Saman",0

msg1 BYTE "Stings are equal ",0

msg2 BYTE "Stings are not equal ",0

.code

main PROC

enter 0,0

push offset Str1

push offset Str2

call IsCompare

leave

ret

exit

main ENDP

IsCompare PROC

enter 0,1

mov edi, [ebp+8]

mov esi, [ebp+12]

mov ecx, LENGTHOF Str1

cld

repe cmpsb

jz print

mov edx, offset msg2

call writestring

jmp last

print:

mov edx, offset msg1

call writestring

last:

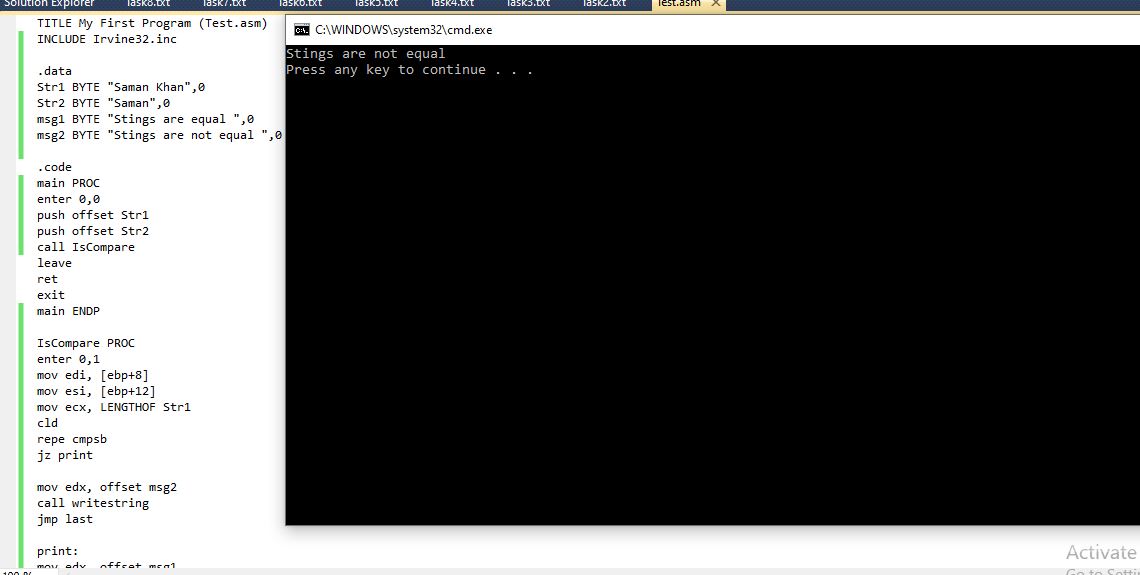
call crlf

leave

ret

IsCompare ENDP

end main

****

**TASK#04:**

TITLE My First Program (Test.asm)

INCLUDE Irvine32.inc

.data

Str1 BYTE "Saman Khan",0

Str2 BYTE 20 dup(?)

msg1 BYTE "Sting 2 is : ",0

.code

main PROC

enter 0,0

push offset Str1

push offset Str2

call Move\_String

leave

ret

exit

main ENDP

Move\_String PROC

enter 0,1

mov edi, [ebp+8]

mov esi, [ebp+12]

mov ecx, LENGTHOF Str1

cld

rep movsb

mov edx, offset msg1

call writestring

mov edx, offset Str2

call writestring

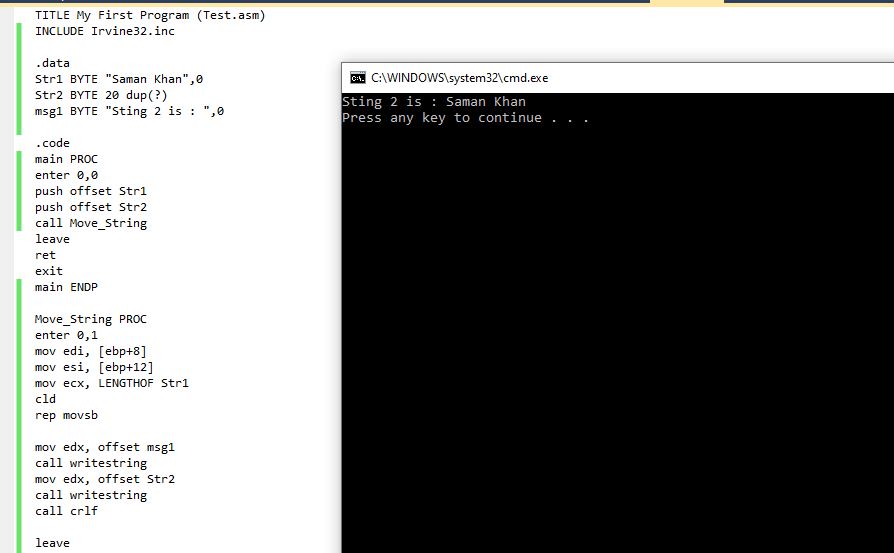
call crlf

leave

ret

Move\_String ENDP

end main



**TASK#05:**

TITLE My First Program (Test.asm)

INCLUDE Irvine32.inc

.data

Str1 BYTE "Hi how are you?",0

Str2 BYTE 20 dup(?)

msg1 BYTE "Original String : ",0

msg2 BYTE "New String : ",0

.code

main PROC

enter 0,0

mov edx, offset msg1

call writestring

mov edx, offset Str1

call writestring

call crlf

push offset Str1

push LENGTHOF Str1

call Str\_Reverse

pop eax

pop eax

pop eax

leave

ret

exit

main ENDP

Str\_Reverse PROC

enter 0,1

mov esi, [ebp+12]

mov ecx, [ebp+8]

dec ecx

L1:

push [esi]

inc esi

LOOP L1

mov esi, [ebp+12]

mov ecx, [ebp+8]

dec ecx

L2:

pop [esi]

inc esi

LOOP L2

mov edx, offset msg2

call writestring

mov edx, offset Str1

call writestring

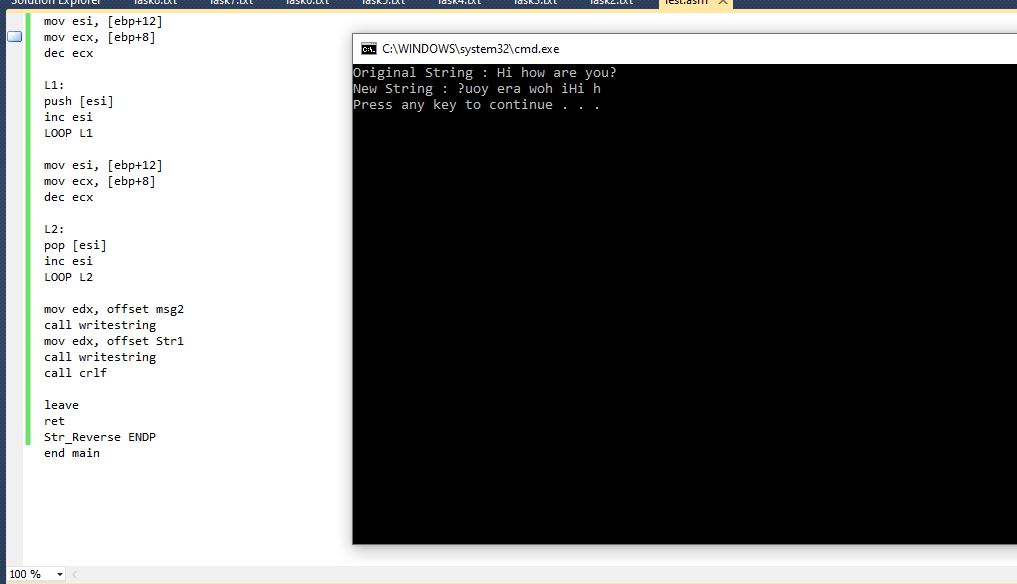
call crlf

leave

ret

Str\_Reverse ENDP

end main

****

**TASK#06:**

TITLE My First Program (Test.asm)

INCLUDE Irvine32.inc

.data

arr SDWORD 6, -7 ,9 ,-4 , 1, 0, 2, -5

num SDWORD ?

msg BYTE " ",0

msg1 BYTE "Original Array : ",0

msg2 BYTE "New Array : ",0

msg3 BYTE "Enter the number you want multiply the array with : ",0

.code

main PROC

enter 0,0

mov edx, offset msg1

call writestring

mov ecx, lengthof arr

mov esi, offset arr

L1:

lodsd

cmp eax, 0

jge next

call writeint

jmp skip

next:

call writedec

skip:

mov edx, offset msg

call writestring

LOOP L1

call crlf

mov edx, offset msg3

call writestring

call readint

mov num, eax

push lengthof arr

push offset arr

push eax

call LOAD

pop eax

pop eax

pop eax

leave

ret

exit

main ENDP

LOAD PROC

enter 0,1

mov ecx, [ebp+16]

mov esi, [ebp+12]

mov ebx, [ebp+8]

mov edi, esi

L1:

lodsd

imul ebx

stosd

LOOP L1

mov ecx, [ebp+16]

mov esi, [ebp+12]

mov edx, offset msg2

call writestring

L2:

lodsd

cmp eax, 0

jge next

call writeint

jmp skip

next:

call writedec

skip:

mov edx, offset msg

call writestring

LOOP L2

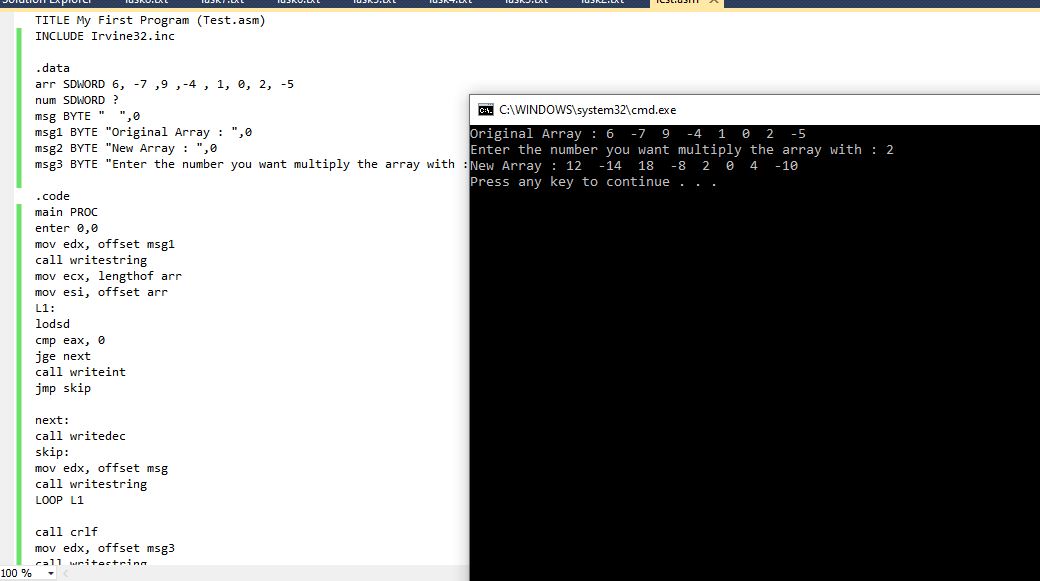
call crlf

leave

ret

LOAD ENDP

end main

****

**TASK#07:**

TITLE My First Program (Test.asm)

INCLUDE Irvine32.inc

.data

arr BYTE 0, 2, 5, 10, 12, 15, 20

num SDWORD ?

var DWORD 2

msg BYTE " ",0

msg1 BYTE "Array : ",0

msg2 BYTE "Number found at index : ",0

msg3 BYTE "The number you want to search does not exist in the array!",0

msg4 BYTE "Enter the number you want to search : ",0

.code

main PROC

enter 0,0

mov edx, offset msg1

call writestring

mov ecx, lengthof arr

mov esi, offset arr

L1:

lodsb

movsx eax, al

cmp eax, 0

jge next

call writeint

jmp skip

next:

call writedec

skip:

mov edx, offset msg

call writestring

LOOP L1

call crlf

mov edx, offset msg4

call writestring

call readint

mov num, eax

push lengthof arr

push eax

call BinarySearch

pop eax

pop eax

leave

ret

exit

main ENDP

BinarySearch PROC

enter 0,1

mov ecx, [ebp+12] ;ecx will be used as right index

dec ecx

mov ebx, [ebp+8]

mov edx, 0 ;edx will be used as left index

;index starts from 0

L1:

push edx

add edx, ecx

mov eax, edx ;eax is being used as the middle index

mov edx, 0

div var

pop edx

push eax

mov esi, eax

mov al, arr[esi]

movsx eax, al

cmp eax, ebx

pop eax

je print ;if equal than print the index

ja right

inc eax ;setting left index

mov edx, eax

jmp check

right: ;setting left index

dec eax

mov ecx, eax

check:

cmp edx, ecx

jbe L1

mov edx, offset msg3

call writestring

jmp final

print:

mov edx, offset msg2

call writestring

call writedec

final:

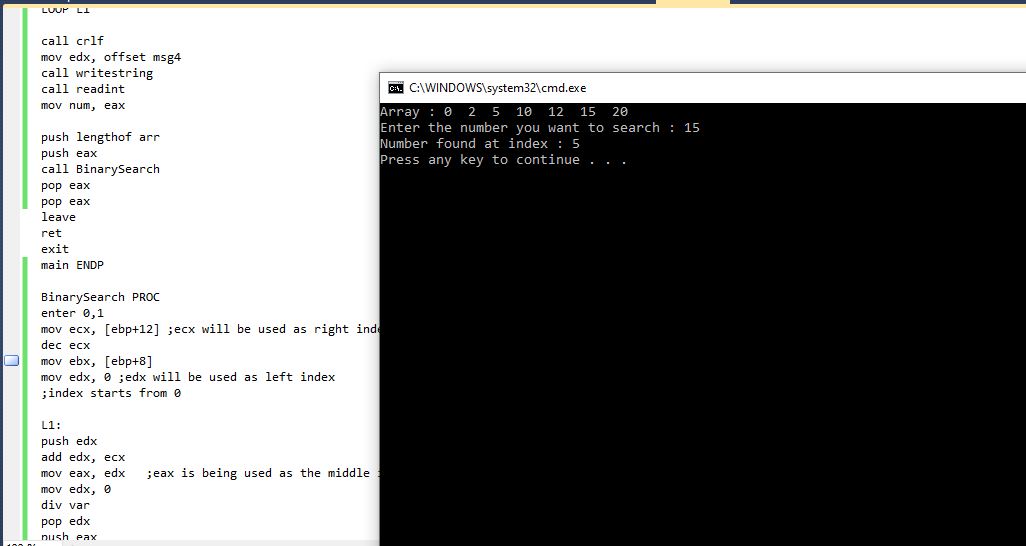
call crlf

leave

ret

BinarySearch ENDP

end main



**PS: The index in this program starts from 0.**

**TASK#08:**

TITLE My First Program (Test.asm)

INCLUDE Irvine32.inc

.data

count DWORD 0

target BYTE "AAEBDCFBBC",0

freqTable DWORD 10 DUP(0)

msg1 BYTE "Frequency of characters : ",0

msg2 BYTE " ",0

Get\_frequencies PROTO, X: PTR BYTE, Y: PTR DWORD

.code

main PROC

enter 0,0

mov ecx, lengthof target

mov esi,0

dec ecx

L1:

push ecx

mov ecx, lengthof target

sub ecx, esi

mov edi, 0

add edi, esi

L2:

mov al, target[esi]

mov dl, target[edi]

cmp al, dl

jbe next

mov target[esi], dl

mov target[edi], al

next:

inc edi

LOOP L2

new:

inc esi

pop ecx

LOOP L1

mov ecx, lengthof target

mov esi, offset target

L3:

lodsb

movzx eax, al

call writechar

LOOP L3

call crlf

INVOKE Get\_frequencies, addr target, addr freqTable

leave

ret

exit

main ENDP

Get\_frequencies PROC, X: PTR BYTE, Y: PTR DWORD

enter 0,1

mov edi, Y

mov esi, X

mov ecx, lengthof target

inc esi

L1:

mov al, [esi]

mov ebx, 0

top:

inc esi

inc ebx

push ebx

mov ebx, 1

add [edi], ebx

mov dl, [esi]

cmp al, dl

pop ebx

je top

inc esi

;mov eax, count

;mov [edi], eax

add edi, 4

;mov count, 0

sub ecx, ebx

LOOP L1

mov esi, Y

mov ecx, lengthof target

mov edx, offset msg1

call writestring

L2:

lodsd

call writedec

mov edx, offset msg2

call writestring

loop L2

call crlf

leave

ret

Get\_frequencies ENDP

end main