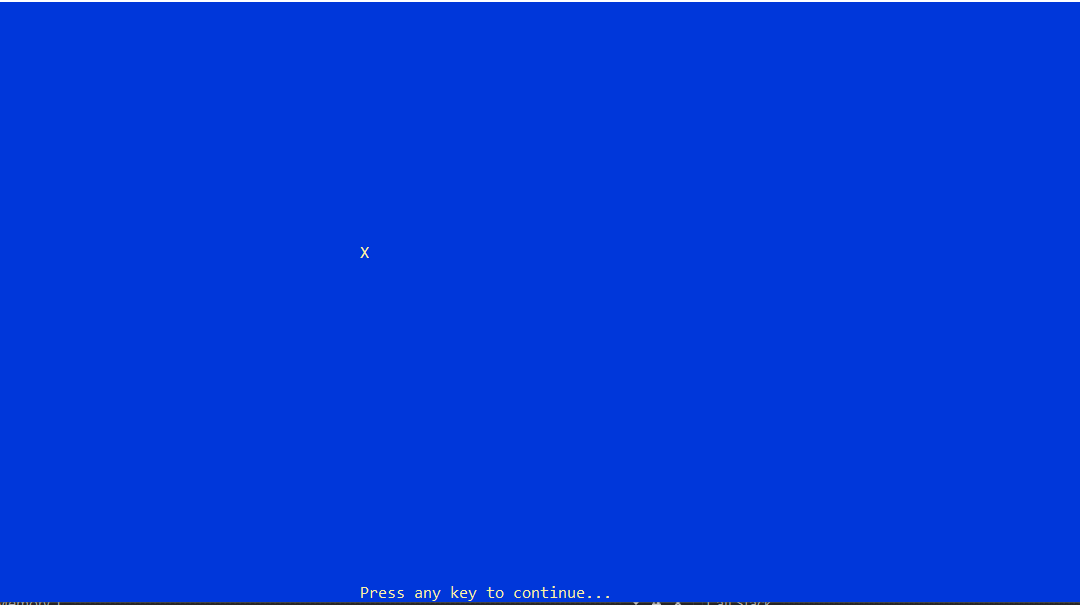
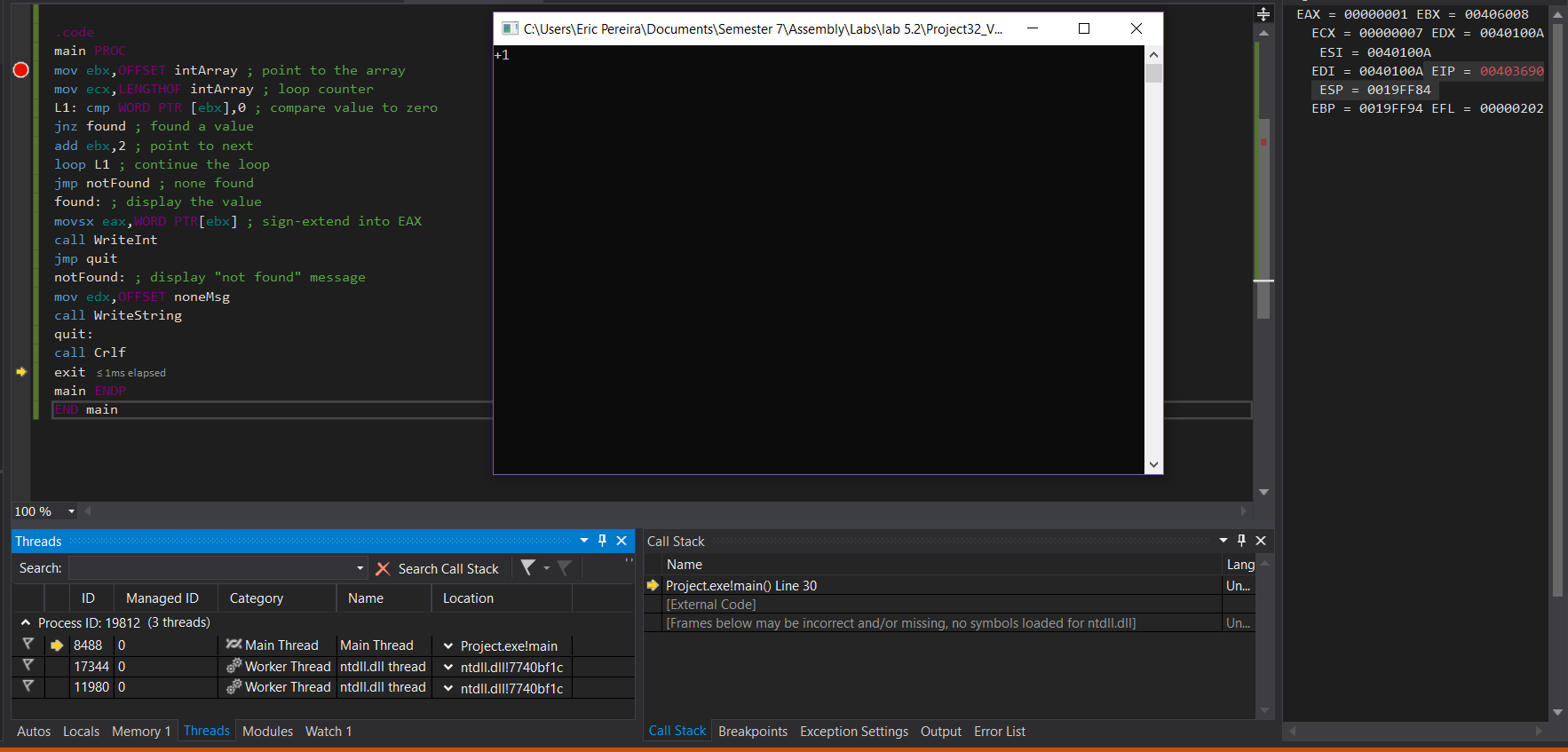


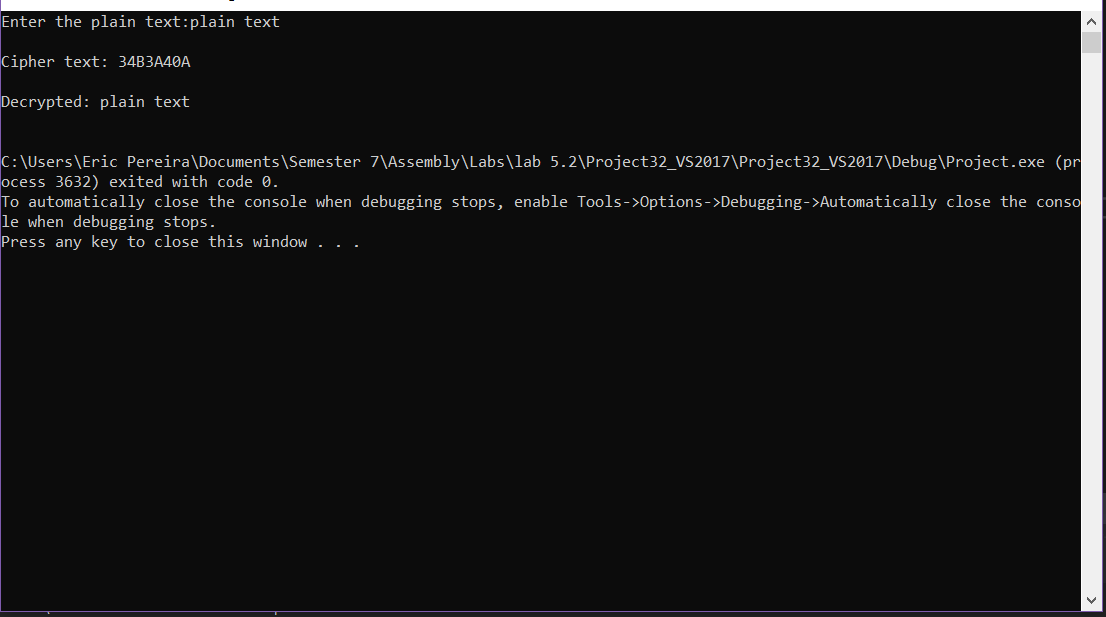
Task 1



Task2



Task3



Task4

centerX.asm

; Library Test #1: Integer I/O (InputLoop.asm)

; Tests the Clrscr, Crlf, DumpMem, ReadInt, SetTextColor,

; WaitMsg, WriteBin, WriteHex, and WriteString procedures.

include Irvine32.inc

.data

COUNT = 4

YellowTextOnGray = yellow + (blue \* 16)

DefaultColor = lightGray + (black \* 16)

arrayD SDWORD 12345678h,1A4B2000h,3434h,7AB9h

prompt BYTE "Press any key... ",0

.code

main PROC

mov eax,YellowTextOnGray

call SetTextColor

call Clrscr ; clear the screen

mov dh, 12

mov dl, 40

call Gotoxy

mov al, 'X'

call WriteChar

call GetMaxXY

sub dl, 1h

mov dh, dl

mov dl, 40

mov dh, 29

call Gotoxy

call WaitMsg

main ENDP

END main

Task4.asm

INCLUDE Irvine32.inc

KEY = 239 ; any value between 1-255

BUFMAX = 128 ; maximum buffer size

.data

sPrompt BYTE "Enter the plain text:",0

sEncrypt BYTE "Cipher text: ",0

sDecrypt BYTE "Decrypted: ",0

buffer BYTE BUFMAX+1 DUP(0)

bufSize DWORD ?

.code

main PROC

call InputTheString ; input the plain text

call TranslateBuffer ; encrypt the buffer

mov edx,OFFSET sEncrypt ; display encrypted message

call DisplayMessageEnc

call TranslateBuffer ; decrypt the buffer

mov edx,OFFSET sDecrypt ; display decrypted message

call DisplayMessage

exit

main ENDP

;-----------------------------------------------------

InputTheString PROC

;

; Prompts user for a plaintext string. Saves the string

; and its length.

; Receives: nothing

; Returns: nothing

;-----------------------------------------------------

pushad ; save 32-bit registers

mov edx,OFFSET sPrompt ; display a prompt

call WriteString

mov ecx,BUFMAX ; maximum character count

mov edx,OFFSET buffer ; point to the buffer

call ReadString ; input the string

mov bufSize,eax ; save the length

call Crlf

popad

ret

InputTheString ENDP

DisplayMessageEnc PROC

pushad

call WriteString

mov edx,OFFSET buffer ; display the buffer

call WriteHexB

call Crlf

call Crlf

popad

ret

DisplayMessageEnc ENDP

;-----------------------------------------------------

DisplayMessage PROC

;

; Displays the encrypted or decrypted message.

; Receives: EDX points to the message

; Returns: nothing

;-----------------------------------------------------

pushad

call WriteString

mov edx,OFFSET buffer ; display the buffer

call WriteString

call Crlf

call Crlf

popad

ret

DisplayMessage ENDP

;-----------------------------------------------------

TranslateBuffer PROC

;

; Translates the string by exclusive-ORing each

; byte with the encryption key byte.

; Receives: nothing

; Returns: nothing

;-----------------------------------------------------

pushad

mov ecx,bufSize ; loop counter

mov esi,0 ; index 0 in buffer

L1:

xor buffer[esi],KEY ; translate a byte

inc esi ; point to next byte

loop L1

popad

ret

TranslateBuffer ENDP

END main