**Pontificia Universidad Católica Madre y Maestra**

****

**ARQUITECTURA COMPUTACIONAL**

ST-ISC-359-T-001

**Presentado por:**

Cesar Darinel Ortiz 2008-1175

**Presentado al profesor:**

ÁLVARO A. REYES P.

**Tema:**

Programa trabajar con archivos

**Fecha de entrega:**

Jueves, 25 de Octubre del 2018

; Autor: Cesar Darinel Ortiz

; Tarea: 5 laboratorio

; Fecha Entrega: 25/10/2018

.model small

.stack 256

.data

;========================Variables declaradas aqui===========================

letrero DB 'Digite el archivo a leer -->: $'

uncaracter db 1 dup (' ')

error1 DB 'El archivo no pudo ser abierto o no existe : $'

; texto a solicita la cadena

nombreArchivo db 51 dup (0) ; buffer de lectura de cadena

punteroArchivo dw ?

;============================================================================

; macro printf

printf macro texto              ; espera en dl el caracter a mostrar.

    mov ah,09 ; Para mostrar en pantalla una cadena

    mov dx, offset texto ; posición de la cadena a montar

    int 21h ; llamó al sistema

endm

; tratando de hacer una funcion

.code

main:

    mov ax,@data

    mov ds,ax

    mov es, ax ;set segment register

and sp, not 3 ;align stack to avoid AC fault

;====================================Código==================================

    printf letrero

    call ciclodelectura

;====================================ciclo\_principal mantener codigo ==================================

ciclo\_principal:

    mov ah,01 ; para lectura de teclado.

    int 21h ; llamada al SO

    cmp al,13 ; verificar si se pulsa el Enter.

    je fin ; saltamos a solicitar el caracter si presiona enter

    ;ciclo principal

    call abrir\_archivo

    call imprimir\_archivo

    call cerrar\_archivo

    jmp ciclo\_principal

;====================================leer el archivo a buscar==================================

ciclodelectura:

    mov ah,01 ; para lectura de teclado.

    int 21h ; llamada al SO

    cmp al,13 ; verificar si se pulsa el Enter.

    je ciclodelecturafin ; saltamos a solicitar el caracter si presiona enter

    mov nombreArchivo[bx],al ; copiar el carácter tomado en el buffer.

    cmp bx,50 ; verificó si debo de salir.

    je ciclodelecturafin ; si escribimos más de 50 caracteres dejo de leer

    inc bx ; apuntó a la sgte. posición del buffer.

    jmp ciclodelectura ; continuó leyendo

    ciclodelecturafin:

    ret

;====================================Abrir el archivo==================================

abrir\_archivo:

    mov ah, 3dh

    mov cx, 0

    mov al, 2

    mov dx, offset nombreArchivo

    int 21h

    jc error

    mov punteroArchivo, ax

    jmp imprimir\_archivo

    error:

    printf error1 ; posición de la cadena a montar

    jmp fin

    fin\_abrir\_archivo:

    ret

;====================================imprimo el archivo pantalla==================================

imprimir\_archivo:

    mov ax,0b800h

    mov es,ax

    mov bx, 0

    call limpia

    mov bx, 0                   ;punteroArchivo para video

    mov di, 0                   ;punteroArchivo letrero

    imprimo\_un\_caracter:

    call leo\_archivo ;mov al,nombreArchivo[di]

    cmp ax,0                    ;verifico si terminó la candena

    je fin\_imprimir\_archivo

    mov al,uncaracter

    cmp al,13                  ;verifico si terminó la candena

    je enter\_

    mov es:[bx], al             ;coloco en pantalla

    inc bx

    inc bx                  ;apunto a próximo caracter

    jmp imprimo\_un\_caracter

    fin\_imprimir\_archivo:

    ret

leo\_archivo:

    push bx cx dx        ;

    mov ah,3fh              ;para mostrar caracter

    mov bx,punteroArchivo

    mov dx,offset uncaracter

    mov cx,1

    int 21h                     ;

    pop dx cx bx            ;

    ret                     ;

enter\_:

    push ax cx dx

    inc bx

    pop dx cx ax

    ret

limpia: ;

    push ax bx cx dx

    mov ah, 0x06

    mov al, 0

    int 10h

    pop dx bx cx ax

    ret

cerrar\_archivo: ;

    push ax bx cx dx

    mov ah, 3eh

    mov bx , punteroArchivo

    int 21h

    pop dx bx cx ax

    ret

fin:

    call cerrar\_archivo

;============================================================================

.exit

;================================Funciones aqui==============================

;============================================================================

end main