**PONTIFICIA UNIVERSIDAD CATOLICA MADRE Y MAESTRA**

****

**Nombre:**

Félix Alejandro Guzmán           2014 - 0565

**Materia:**

Arquitectura Computacional

**Profesor:**

Alvaro A. Reyes P.

**Practica sobre:**

Práctica #1

**Fecha de Entrega:**

viernes, 13 de octubre de 2017

# Código

.model small

.stack 256

.data

;========================Variables declaradas aquí===========================

mensaje db 'Digite 20 caracteres:','$'  ;string

caracter db 20 dup (0)                  ; buffer para 20 bytes de caracteres horizontales

caracterv db 30 dup(0)                  ; buffer para 30 bytes de caracteres verticales

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

.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==================================

call clrscr

call texto

mov ah,02                               ;para imprimir en pantalla un carácter.

mov dl, 0ah                             ; cambio de línea.

int 21h                                 ; llamada al sistema operativo.

mov bx, 0                               ; índice de lectura de caracteres horizontal en cero.

mov si, 0                               ;índice de lectura de caracteres vertical en cero.

ok:

mov ah,1                                ;capturar de pantalla.

int 21h                                 ;llamada al sistema operativo.

mov caracter[bx],al                     ;guardo el valor capturado por el teclado en buffer de horizontales.

mov caracterv[si],al                    ;guardo el valor capturado por el teclado en buffer de verticales.

inc si                                  ;muevo el índice a la siguiente posición

mov caracterv[si],0ah                   ;agrego un tab horizontal

inc si                                  ;muevo el índice de verticales a la siguiente posición

inc bx                                  ; apunto a la siguiente posición del buffer.

cmp bx,20                               ;reviso si he tomado los 20 caracteres.

jne ok                                  ;sigo tomando caracteres hasta tener 20.

mov ah, 09

mov ah, 0ah

mov ah, 09

call verticales

;mov ah, 09

;mov ah, 09

call horizontales

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

.exit

;================================Funciones aquí==============================

horizontales:

mov caracter[bx],'$'

mov ah, 09                          ;para imprimir en pantalla un string

mov dx, offset caracter             ;ubicación de caracteres horizontales

int 21h

ret

verticales:

mov caracterv[si],'$'

mov ah, 09                          ;para imprimir en pantalla un string

mov dx, offset caracterv            ;ubicación de caracteres verticales

int 21h

ret

texto:

mov ah,09                           ;imprimir un string.

mov dx, offset mensaje              ; paso al sistema la ubicación del letrero.

int 21h                             ;llamada al sistema operativo.

ret

clrscr proc                         ;limpiar pantalla

mov ax,0003h

int 10h

ret

endp

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

end main