

- **Ejercicio:** configurar DOSBox para que monte en la unidad C: el directorio donde se encuentra Borland C y asignar a la variable "PATH" el directorio "bc/bin" donde esta BC.EXE

Realizado en el seminario 1.

- **Ejercicio:** realizar un programa que llame a la función de cambio de modo de video y comprobar su funcionamiento.

```
pila segment stack 'stack'
    dw 100h dup (?)      ; tamaño de la pila
pila ends

datos segment 'data'
    cadena db 10,13,'Modo grafico (Pulsar tecla para salir)$'
datos ends

codigo segment 'code'
    assume cs:codigo, ds:datos, ss:pila
    main PROC
        mov ax,datos
        mov ds,ax

        mov al,4      ; MODO = 4 grafico 320x200 a 4 colores
        mov ah,0      ; función para poner modo texto o gráfico deseado
        int 10h

        mov dx,OFFSET cadena
        mov ah,9      ; guarda en ah el servicio 9 (imprimir por pantalla)
        int 21h      ; interrupcion 21 para el servicio 9

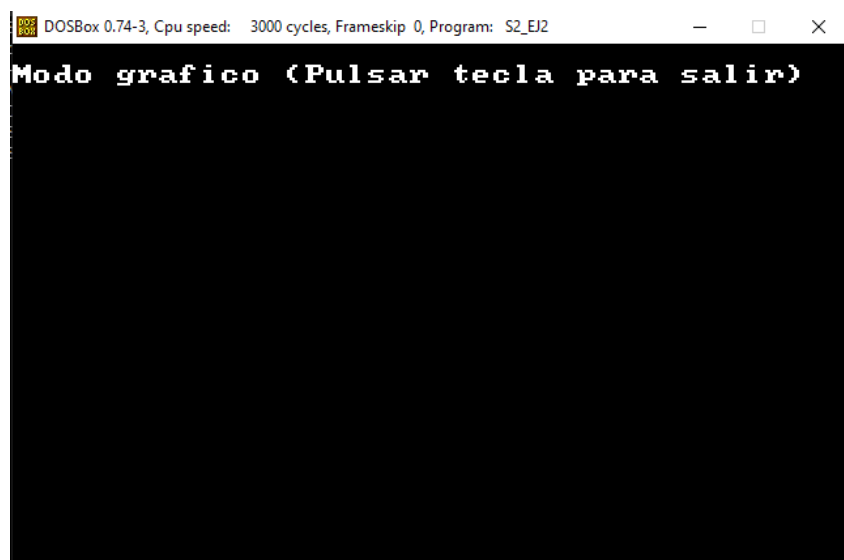
        mov ah,08h    ; función para leer una tecla
        int 21h      ; interrupción DOS para teclado (en AL devuelve el carácter tecleado)

        mov al,3      ; MODO = 3 para resolucio 80x25 color (DEFAULT)
        mov ah,0      ; función para poner modo texto o gráfico deseado
        int 10h

        mov ax,4C00h   ; guarda en ax el servicio 4C00h (return)
        int 21h      ; interrupcion 21 para el servicio 4C00h
    main ENDP
codigo ends

END main
```

Tras su compilación y enlazado con TASM/TLINK respectivamente y su ejecución el resultado por terminal sería el que se muestra a continuación:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: S2_EJ2
Modo grafico (Pulsar tecla para salir)
```

- **Ejercicio:** realizar un programa que llame a la función de lectura de caracteres. Leer varias pulsaciones de teclas y mostrarlas por pantalla.

```
pila segment stack 'stack'
    dw 100h dup (?)      ; tamaño de la pila
pila ends

datos segment 'data'
    character db ' ',10,13,'$'
datos ends

codigo segment 'code'
    assume cs:codigo, ds:datos, ss:pila
    main PROC
        mov ax,datos
        mov ds,ax

        lectura:
            mov ah,08h    ; función para leer una tecla
            int 21h        ; interrupción DOS para teclado (en AL devuelve el carácter tecleado)
            mov character,al

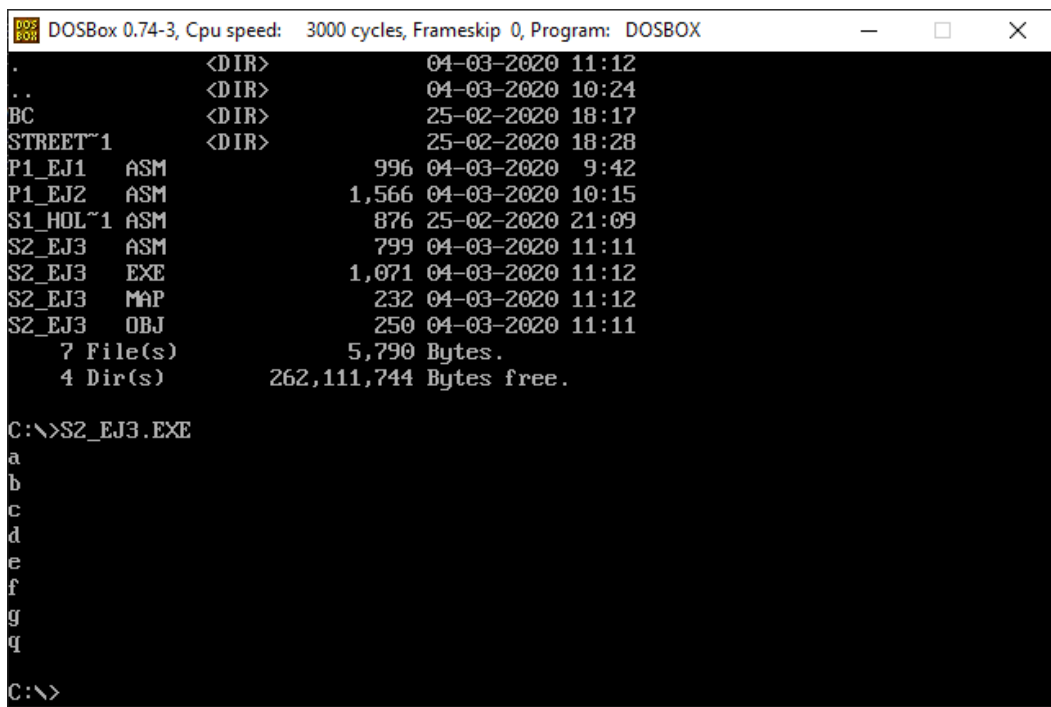
            mov dx,OFFSET character
            mov ah,9       ; guarda en ah el servicio 9 (imprimir por pantalla)
            int 21h        ; interrupción 21 para el servicio 9

            cmp character, 'q' ; carácter 'q' para salir
            jne lectura

            mov ax,4C00h    ; guarda en ax el servicio 4C00h (return)
            int 21h        ; interrupción 21 para el servicio 4C00h
        main ENDP
    codigo ends

END main
```

Al igual que antes tras su compilación y enlazado con TASM/TLINK respectivamente y su ejecución el resultado por terminal sería el que se muestra a continuación:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
.          <DIR>          04-03-2020 11:12
..         <DIR>          04-03-2020 10:24
BC         <DIR>          25-02-2020 18:17
STREET~1   <DIR>          25-02-2020 18:28
P1_EJ1     ASM            996 04-03-2020  9:42
P1_EJ2     ASM           1,566 04-03-2020 10:15
S1_HOL~1   ASM            876 25-02-2020 21:09
S2_EJ3     ASM            799 04-03-2020 11:11
S2_EJ3     EXE           1,071 04-03-2020 11:12
S2_EJ3     MAP            232 04-03-2020 11:12
S2_EJ3     OBJ            250 04-03-2020 11:11
  7 File(s)          5,790 Bytes.
  4 Dir(s)          262,111,744 Bytes free.

C:\>S2_EJ3.EXE
a
b
c
d
e
f
g
q
C:\>
```