



**UNIVERSIDAD
DE GRANADA**

Seminario 2: PDIH

Autor: Juan José Martínez Águila



GRADO EN INGENIERÍA INFORMÁTICA (SISTEMAS DE LA INFORMACIÓN)

—
Curso 2022 - 2023

1. Ejercicio 1

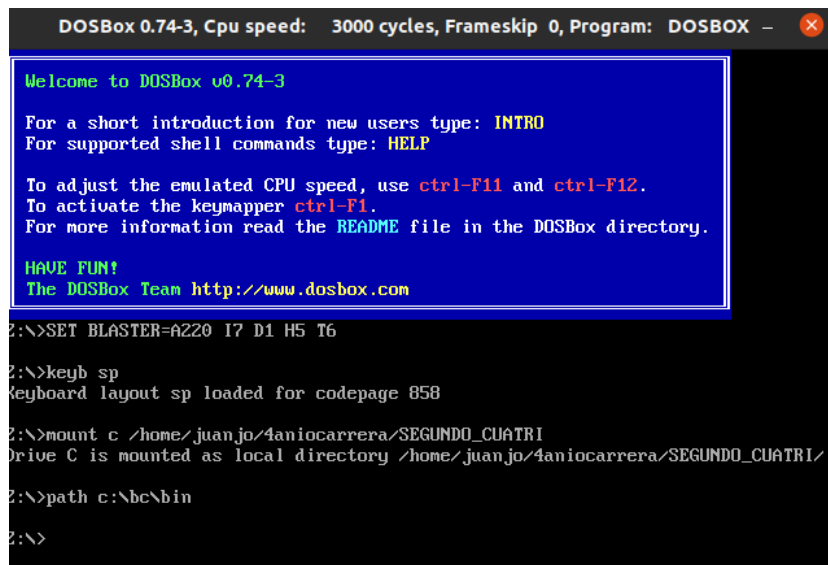
Realizar un programa en lenguaje C que llame a la función de cambio de modo de vídeo y comprobar su funcionamiento.

Para ello hemos creado una función setvideomode con el siguiente código:

```
void setvideomode(unsigned char modo){  
    union REGS inregs, outregs;  
    inregs.h.ah = 0x00;  
    inregs.h.al = modo;  
    int86(0x10, &inregs, &outregs);  
}
```

Figura 1: Código función setvideomode

Obtendremos los siguientes resultados:



The screenshot shows a DOSBox window titled "DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX". The main window has a blue background with white text. The text reads: "Welcome to DOSBox v0.74-3", "For a short introduction for new users type: INTRO", "For supported shell commands type: HELP", "To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.", "To activate the keymapper ctrl-F1.", "For more information read the README file in the DOSBox directory.", "HAVE FUN!", "The DOSBox Team http://www.dosbox.com". Below this, the command prompt shows the following commands and output: "Z:\>SET BLASTER=A220 I7 D1 H5 T6", "Z:\>keyb sp", "Keyboard layout sp loaded for codepage 858", "Z:\>mount c /home/juanjo/4aniocarrera/SEGUNDO_CUATRI", "Drive C is mounted as local directory /home/juanjo/4aniocarrera/SEGUNDO_CUATRI/", "Z:\>path c:\bc\bin", "Z:\>".

Figura 2: Modo de vídeo de texto



Figura 3: Modo de vídeo gráfico

2. Ejercicio 2

Realizar un programa en lenguaje C que llame a la función de interrupción correspondiente a la lectura de caracteres desde teclado. Ejecutarlo para leer pulsaciones de teclas y mostrarlas por pantalla usando otra función de interrupción.

El código de la función será:

```
int mi_getche(){  
    union REGS inregs, outregs;  
    int caracter;  
  
    inregs.h.ah = 1;  
    int86(0x21, &inregs, &outregs);  
  
    caracter = outregs.h.al;  
    return;  
}
```

Figura 4: Código de la función *mi_getche*

Y este es el funcionamiento:

```
Available memory 4164756  
C:\P1-EJE~1\FUNCIO~1  
A  
C:\P1-EJE~1>_
```

Figura 5: Salida de la función