



**PDIH**

## **PRÁCTICA 1**

**Autor**

**Pablo Fernández Gallardo**



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

Granada, 02/03/2024

<b>Código fuente del programa</b>	<b>2</b>
<b>Salidas del programa</b>	<b>6</b>

## Código fuente del programa

```
C/C++
#include <dos.h>
#include <stdio.h> // Agregado para printf
#define BYTE unsigned char

// Definición de constantes para modos de texto y video
BYTE TEXTO_MODE = 3;
BYTE VIDEO_MODE = 4;

// Variables globales para el color de fondo y texto
unsigned char back_color = 0;
unsigned char text_color = 1;

// Función para cambiar la posición del cursor en la pantalla
void cambiarPosicionCursor(int x, int y) {
    union REGS inregs, outregs;
    inregs.h.ah = 2;
    inregs.h.dh = y;
    inregs.h.dl = x;
    inregs.h.bh = 0x00;
    int86(0x10, &inregs, &outregs);
}

// Función para pausar el programa hasta que se presione una tecla
void pausar() {
    union REGS inregs, outregs;
    inregs.h.ah = 0x00;
    int86(0x16, &inregs, &outregs);
}

// Función para cambiar el tipo de cursor en la pantalla
void cambiarTipoCursor(int tipo_cursor) {
    union REGS inregs, outregs;
    inregs.h.ah = 0x01;
    switch(tipo_cursor){
        case 0:
            inregs.h.ch = 010;
            inregs.h.cl = 000;
            break;
        case 1:
            inregs.h.ch = 010;
            inregs.h.cl = 010;
            break;
        case 2:
            inregs.h.ch = 000;
            inregs.h.cl = 010;
    }
}
```



```

    inregs.h.bh = 0x00;
    inregs.x.cx = 1;

    int86(0x10, &inregs, &outregs);
}

// Función para leer un carácter ingresado por el usuario
int leerCaracter() {
    union REGS inregs, outregs;
    int caracter;
    inregs.h.ah = 1;
    int86(0x21, &inregs, &outregs);
    caracter = outregs.h.al;

    return caracter;
}

// Función para dibujar un rectángulo relleno en la pantalla
void dibujarRectanguloRelleno(int x_ini, int y_ini, int lado, unsigned char
color_texto, unsigned char color_fondo) {
    int i, j;

    limpiarPantalla();
    printf("\nEjercicio extra: Pintar rectangulo relleno");

    cambiarColorTexto(color_texto);
    cambiarColorFondo(color_fondo);

    for (i = y_ini; i < y_ini + lado; ++i) {
        for (j = x_ini; j < x_ini + lado; ++j) {
            cambiarPosicionCursor(j, i);
            imprimirCaracter(' ');
        }
    }
}

int main() {
    int aux;
    int video_mode;

    limpiarPantalla();

    // Ejemplo de uso de las funciones definidas arriba
    printf("\nEjercicio 1: Pulsa una tecla (GOTOXY 20,20): ");
    cambiarPosicionCursor(20,20);
    pausar();

    printf("\n");
}

```

```

    printf("\nEjercicio 2: Pulsa una tecla (Cursor invisible): ");
    cambiarTipoCursor(0);
    pausar();
    printf("\nEjercicio 2: Pulsa una tecla (Cursor grueso): ");
    cambiarTipoCursor(2);
    pausar();
    printf("\nEjercicio 2: Pulsa una tecla (Cursor normal): ");
    cambiarTipoCursor(1);
    pausar();
    printf("\n");

    printf("\nEjercicio 3 y 4: Pulsa una tecla (setVideo): ");
    cambiarModoVideo(VIDEO_MODE);
    video_mode = obtenerModoVideo();
    printf("\nEjercicio 3 y 4: Pulsa una tecla (MOD0 GRAFICO 4): %i",
video_mode);
    pausar();
    printf("\nEjercicio 3 y 4: Pulsa una tecla (MOD0 TEXTO 3): ");
    cambiarModoVideo(TEXT0_MODE);
    video_mode = obtenerModoVideo();
    printf("\nEjercicio 3 y 4: (MOD0 TEXTO 3) video mode: %i", video_mode);
    pausar();
    printf("\n");

    printf("\nEjercicios 5, 6, 8 y 9: Introduce una tecla: (GETCHE)");
    aux = leerCaracter();
    pausar();
    printf("\nEjercicios 5, 6, 8 y 9: Pulsa una tecla: ( TEXTCOLOR 4 (rojo)
y background 2 (verde) )\n");
    pausar();
    cambiarColorTexto(4); // Rojo
    cambiarColorFondo(2); // Verde
    imprimirCaracter((char)aux);
    pausar();
    printf("\n");

    printf("\nTEXT0");
    printf("\nEjercicios 7: Pulsa una tecla: (Limpiar Pantalla)\n");
    printf("\nSe ha limpiado la pantalla.");
    pausar();

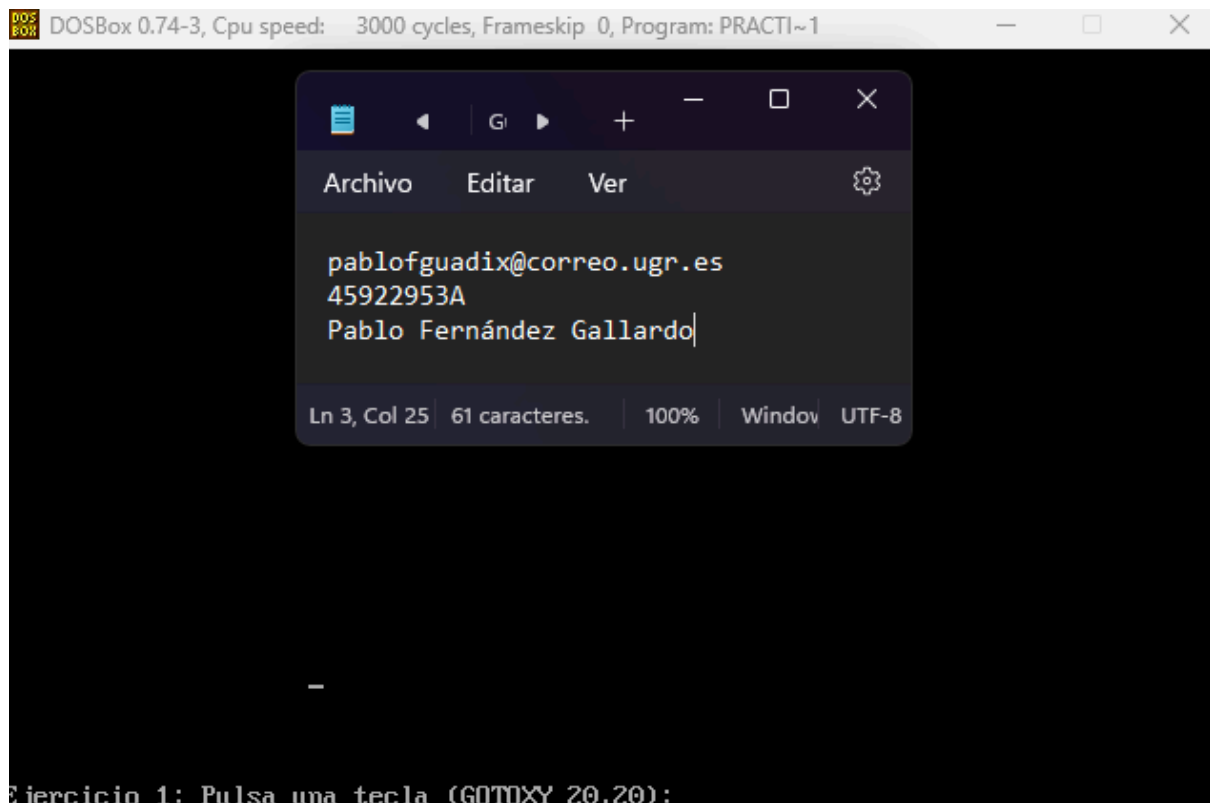
    pausar();
    dibujarRectanguloRelleno(1,1,5,4,2); // Rojo y verde

    return 0;
}

```

Salidas del programa

gotoxy(): coloca el cursor en una posición determinada



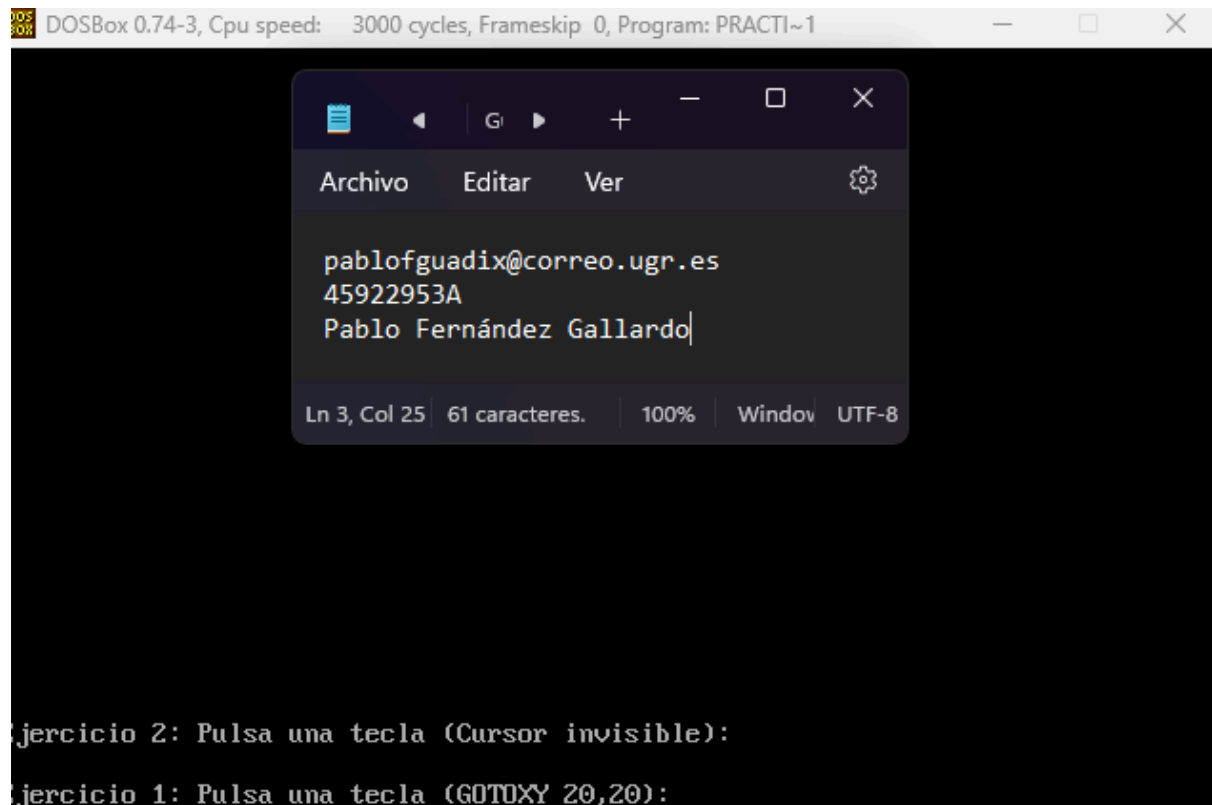
The screenshot shows a DOSBox window titled "DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: PRACTI~1". Inside the window, a text editor is open with a dark background and light-colored text. The editor has a menu bar with "Archivo", "Editar", and "Ver", and a settings icon. The text content is:

```
pablofguadix@correo.ugr.es  
45922953A  
Pablo Fernández Gallardo|
```

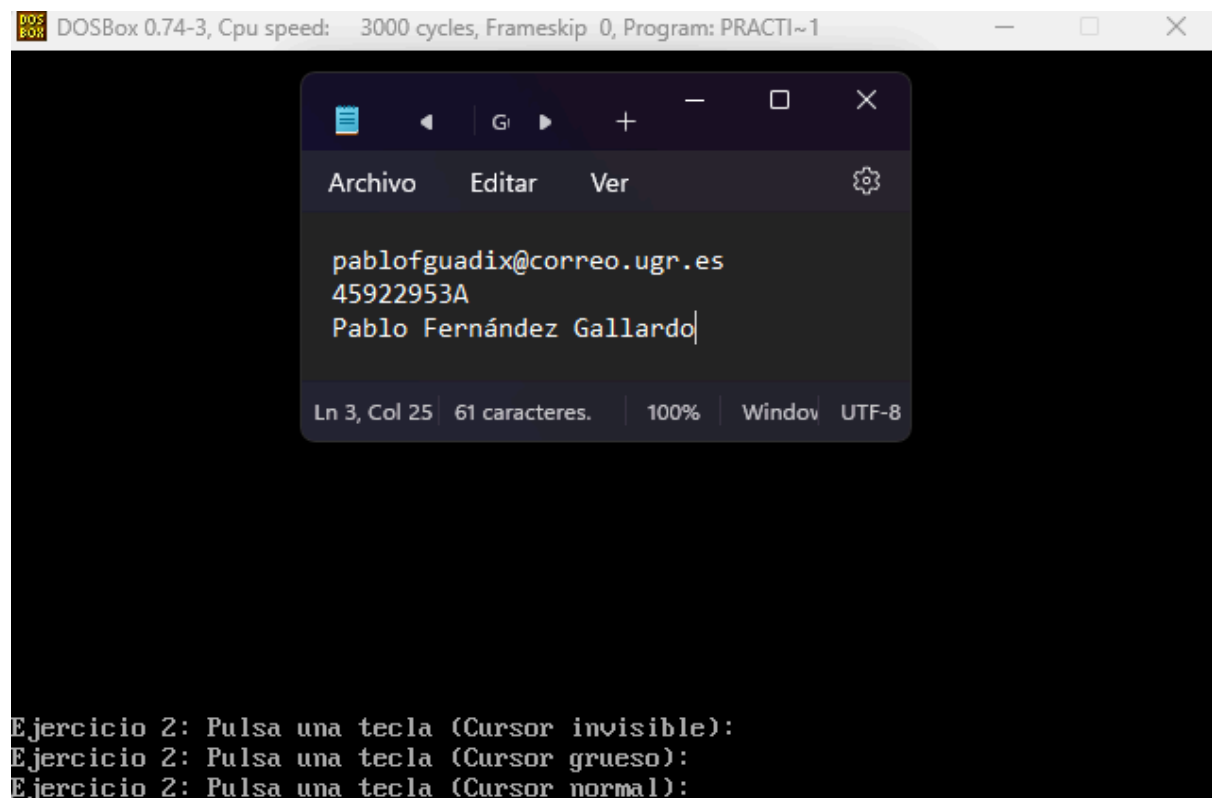
The status bar at the bottom of the editor shows "Ln 3, Col 25", "61 caracteres.", "100%", "Window", and "UTF-8".

setcursortype(): fijar el aspecto del cursor, debe admitir tres valores: INVISIBLE, NORMAL y GRUESO.

Invisible

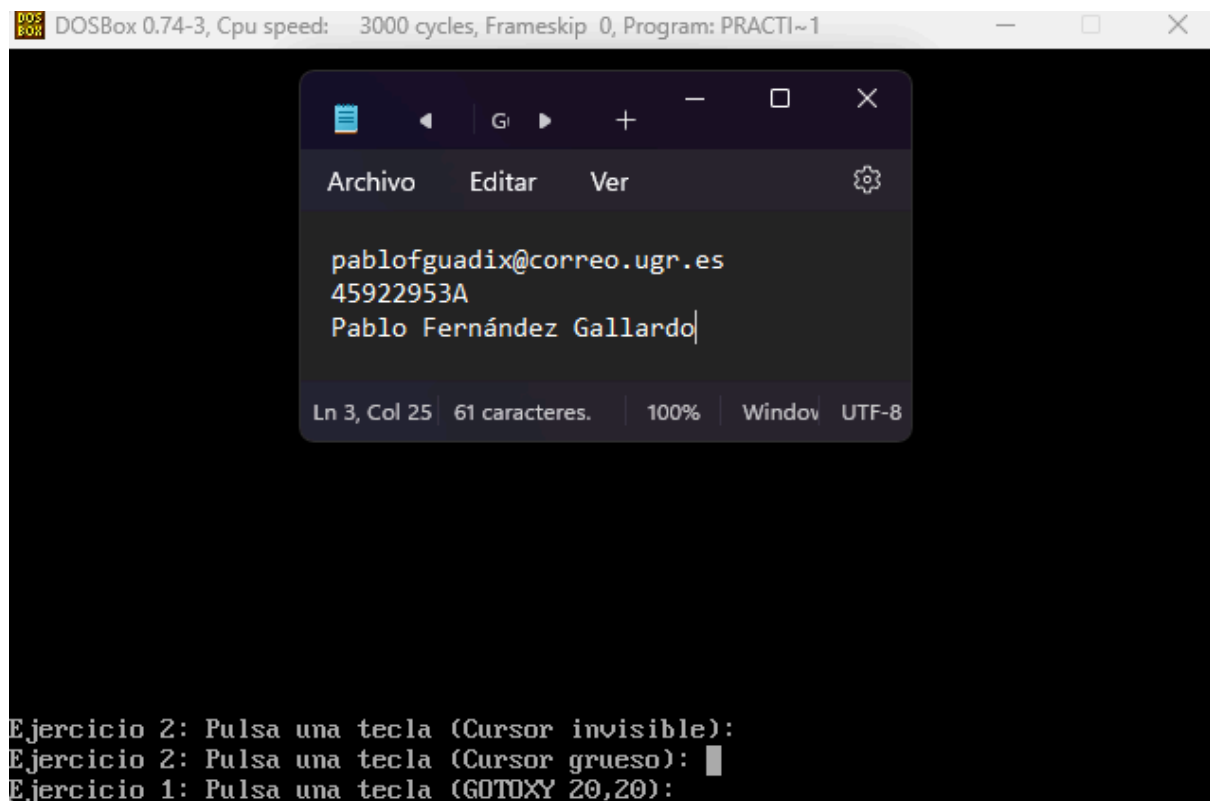


Normal (He intentado hacerle una captura pero no está estático, pero funciona perfectamente).

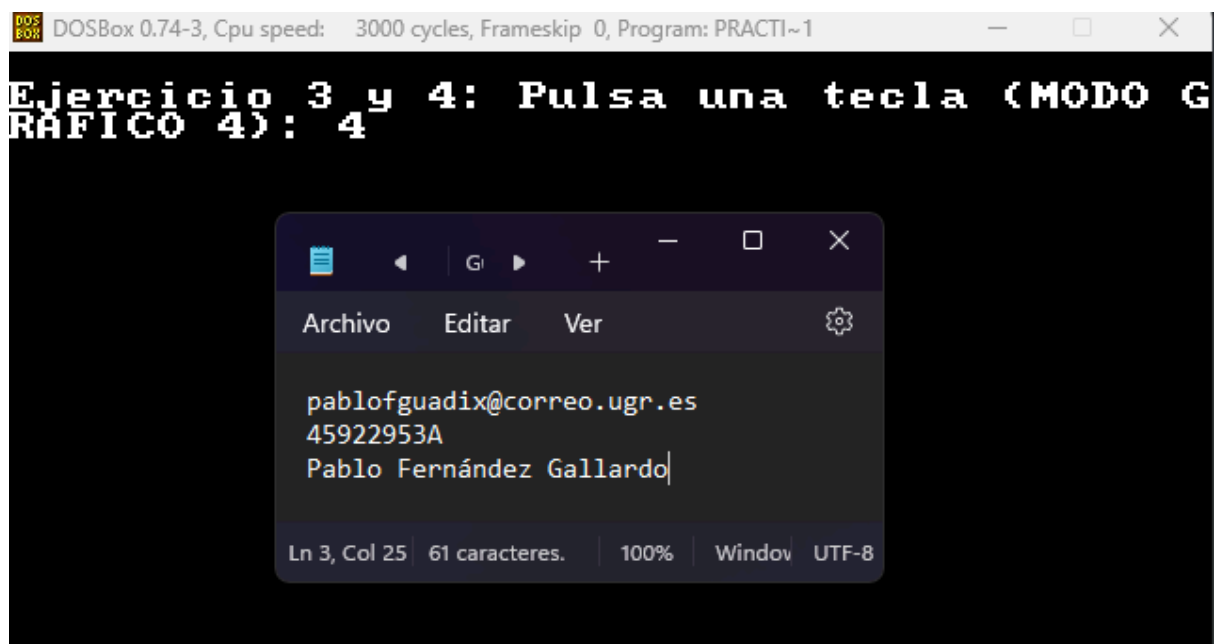


Grueso

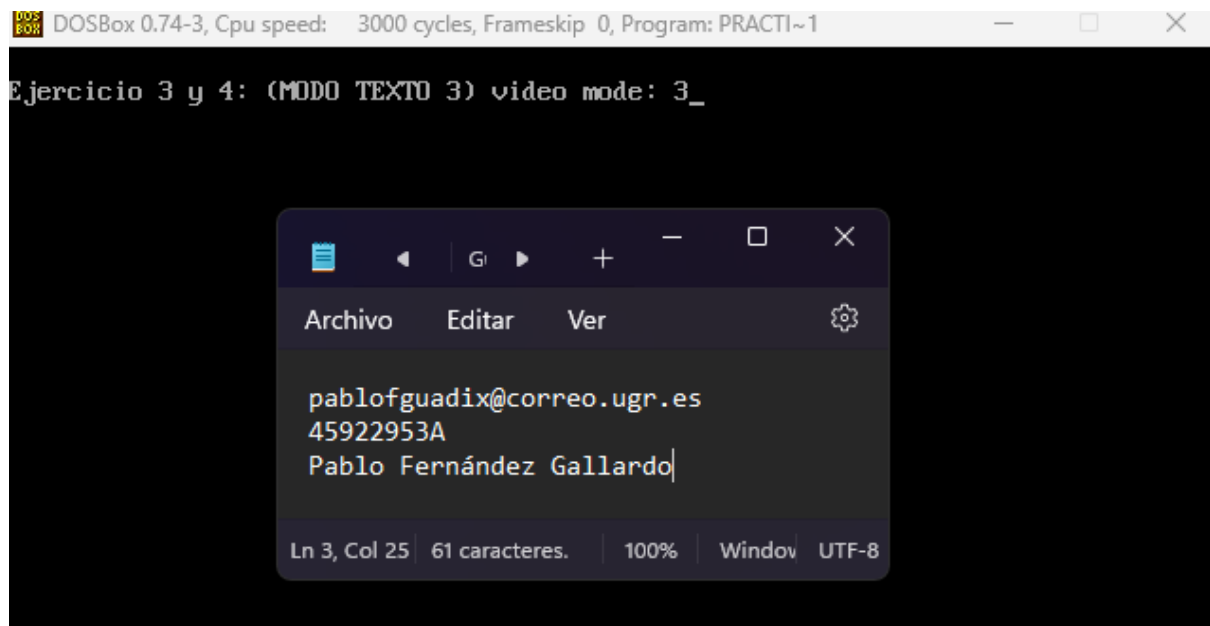




setvideomode(): fija el modo de video deseado



getvideomode(): obtiene el modo de video actual

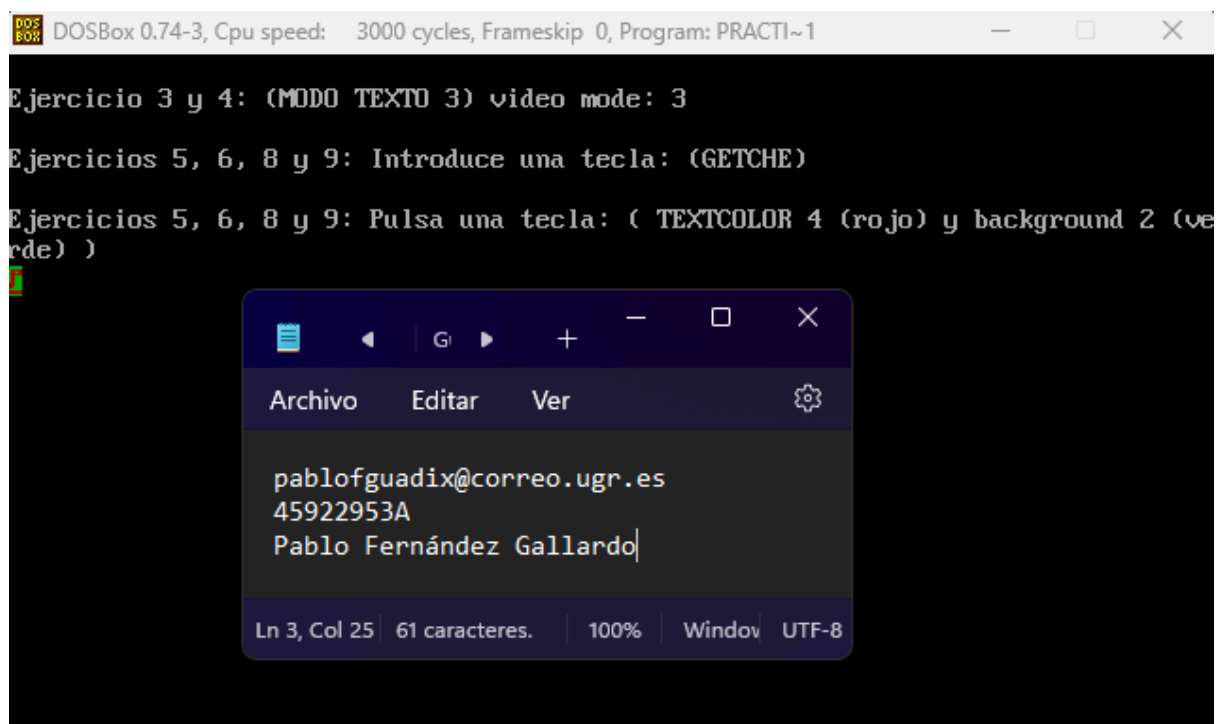


textcolor(): modifica el color de primer plano con que se mostrarán los caracteres

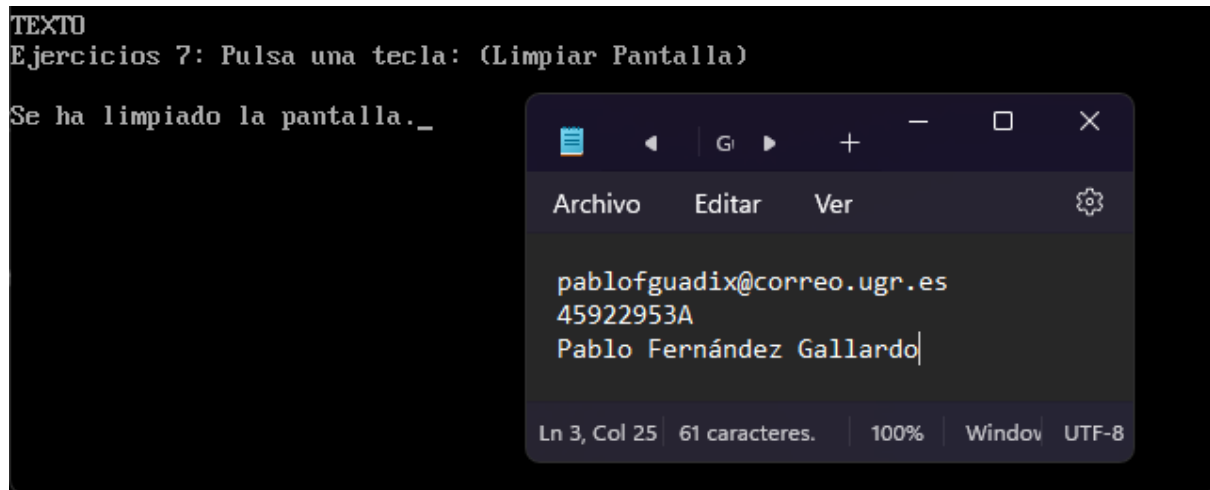
textbackground(): modifica el color de fondo con que se mostrarán los caracteres

cputchar(): escribe un carácter en pantalla con el color indicado actualmente

getche(): obtiene un carácter de teclado y lo muestra en pantalla



clrscr(): borra toda la pantalla



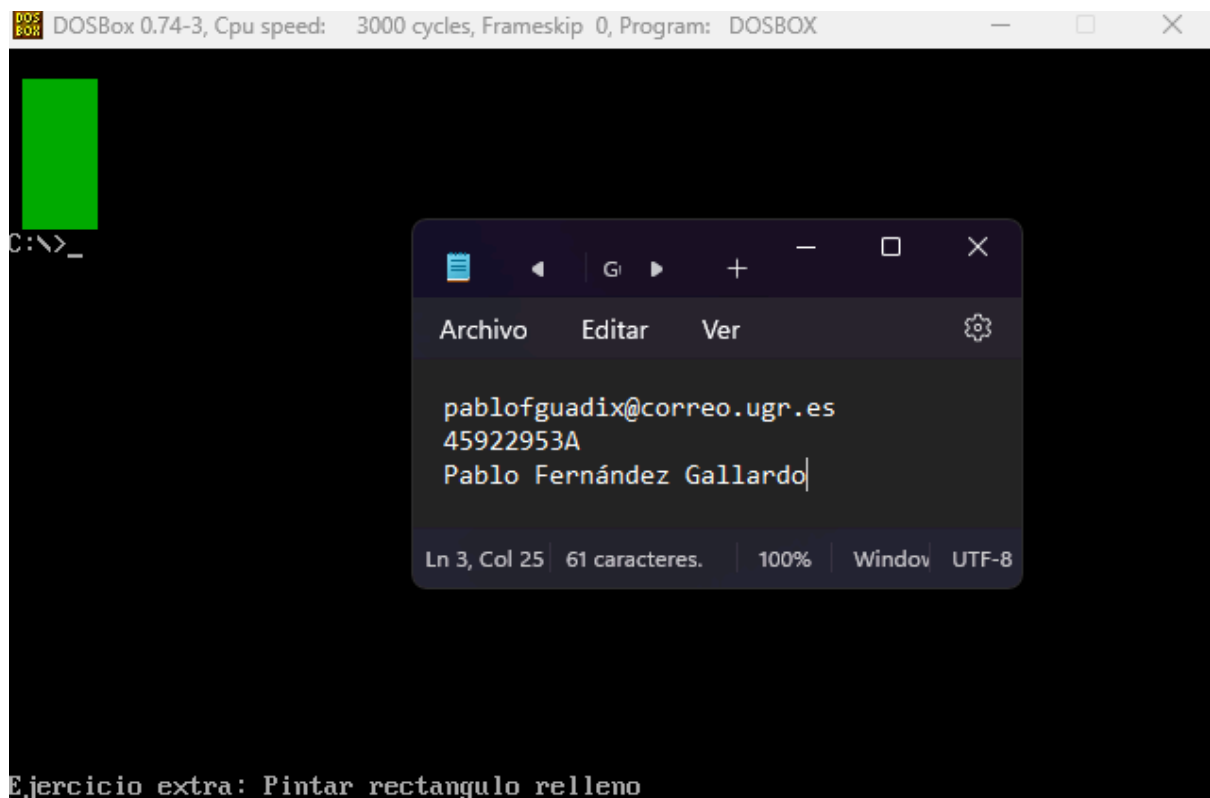
TEXT0  
Ejercicios 7: Pulsa una tecla: (Limpiar Pantalla)  
Se ha limpiado la pantalla.\_

Archivo Editar Ver

pablofguadix@correo.ugr.es  
45922953A  
Pablo Fernández Gallardo

Ln 3, Col 25 61 caracteres. 100% Window UTF-8

Dibujo de rectángulo relleno



DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

C:\>\_

Archivo Editar Ver

pablofguadix@correo.ugr.es  
45922953A  
Pablo Fernández Gallardo

Ln 3, Col 25 61 caracteres. 100% Window UTF-8

Ejercicio extra: Pintar rectangulo relleno