

Práctica 1 - Documentación

En este documento se demostrará el funcionamiento de las distintas funciones que se piden en la práctica 1. Se mostrarán imágenes del código de la función, de su resultado y de el la función *main* correspondiente.

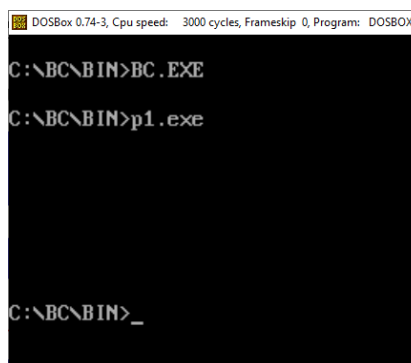
gotoxy()

Esta función permite cambiar la posición del cursor.

```
void gotoxy(int column, int fila){  
    union REGS inregs, outregs;  
    inregs.h.ah = 2;  
    inregs.h.dl = column;  
    inregs.h.dh = fila;  
    inregs.h.bh = 0;  
    int86(0x10, &inregs, &outregs);  
}
```

```
void main(){  
    gotoxy(10, 10);_  
}
```

En este caso, como Dosbox funciona en modo texto, solo mueve deslaza la línea en vertical.



The screenshot shows a DOSBox window with a black background and white text. The text shows the command prompt 'C:\BC\BIN>' followed by 'BC.EXE' and 'p1.exe'. The cursor is visible at the end of the line 'C:\BC\BIN>_'.

setCursorType()

Permite cambiar el tipo de cursor en pantalla. En este caso se ha programado para que pueda ser invisible, normal o grueso.

```

void setCursorType(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;
            break;
    }
    int86(0x10, &inregs, &outregs);
}

```

```

void main(){
    setCursorType(2);
}

```

Se puede observar claramente cómo el cursor pasa a tener mayor tamaño.

```

C:\BC\BIN>BC.EXE

C:\BC\BIN>p1.exe

C:\BC\BIN>

```

setVideoMode()

Aquí se cambia el modo en el que se muestra información en pantalla en acorde a una lista de códigos provista de antemano.

```

void setVideoMode(int modo){
    union REGS inregs, outregs;
    inregs.h.al = modo;
    inregs.h.ah = 0x00;
    int86(0x10, &inregs, &outregs);
}

```

```

void main(){
    setVideoMode(4);
}

```

En el caso de DosBox se puede ver como el modo 4 aumenta considerablemente el tamaño de las letras.

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

```

C:\BC\BIN>

```

getVideoMode()

Esta función simplemente devuelve el modo en que se muestra la información por pantalla.

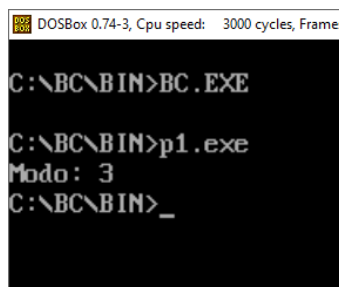
```
int getVideoMode(void){
    union REGS inregs, outregs;
    int modo_video;

    inregs.h.ah = 0x0F;
    int86(0x10, &inregs, &outregs);

    modo_video = outregs.h.al;
    return modo_video;
}
```

```
void main(){
    int modo;

    modo = getVideoMode();
    printf("Modo: %d", modo);
}
```



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frame

C:\BC\BIN>BC.EXE

C:\BC\BIN>p1.exe
Modo: 3
C:\BC\BIN>_
```

textColor()

```
void textColor(int unColor){
    color = unColor;
}
```

textBackground()

```
void textBackground(int unColor){
    back_color = unColor;
}
```

cputchar()

Se ocupa de escribir en pantalla un carácter que viene dado como argumento a la función. Utilizando las dos funciones anteriores se puede establecer el color de fondo y del propio carácter. Ambos datos se escriben en 4 bits del registro bl.

```
void cputchar(char letra){
    union REGS inregs, outregs;

    inregs.h.ah = 0x09;
    inregs.h.al = letra;
    inregs.h.bl = (color << 4) | back_color;
    inregs.h.bh = 0x00;
    inregs.x.cx = 1;

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

```
void main(){
    textColor(4);
    textBackground(7);
    cputchar('a');
}
```

```
C:\BC\BIN>p1.exe
a
C:\BC\BIN>
```

getche()

Esta función lee un carácter introducido a través del teclado y lo pasa por argumento a cputchar, haciendo así que dicho carácter se escribe en pantalla.

```
void getChe(void){
    union REGS inregs, outregs;
    char letra;

    inregs.h.ah = 0x00;
    int86(0x16, &inregs, &outregs);
    letra = outregs.h.al;
    cputchar(letra);
}
```

```
void main(){
    textColor(6);
    textBackground(3);
    getChe();_
}
```

```
C:\BC\BIN>p1.exe
d
C:\BC\BIN>_
```

clrscr()

Por último la función clrscr() debe limpiar la pantalla de todo texto. Se ha intentado conseguir a partir de la interrupción de movimiento vertical pero no daba buen resultado así que se ha optado por restablecer el modo visual para conseguir el efecto deseado.

```
void clrscr(void){
    setVideoMode(3);
}
```