

La biblioteca "conio" es la encargada de manipular directamente tanto la salida a consola como la entrada desde el teclado. Se debe incluir el siguiente fichero de cabecera:

```
#include <conio.h>
```

Posee los siguientes tipos y funciones:

```
struct text_info {
    unsigned char winleft;
    unsigned char wintop;
    unsigned char winright;
    unsigned char winbottom;
    unsigned char attribute;
    unsigned char normattr;
    unsigned char currmode;
    unsigned char screenheight; // Numero de lineas de la pantalla
    unsigned char screenwidth; // Numero de columnas de la pantalla
    unsigned char curx; // Columna donde se
encuentra el cursor
    unsigned char cury; // Fila donde se encuentra
el cursor
};
```

La esquina superior izquierda se corresponde con las coordenadas [1,1].

- Salida y entrada de texto

```
int cprintf(const char fmt[], ...);
    // envia texto formateado a la pantalla
int cputs(const char _str[]);
    // envia una cadena de caracteres a pantalla
int putch(int _c);
    // envia un caracter simple a pantalla
int cscanf(const char fmt[], ...);
    // entrada formateada de datos de consola
char *cgets(char _str[]);
    // entrada de una cadena de caracteres de consola
int getche(void);
    // lee un caracter (con eco)
int getch(void);
    // lee un caracter (sin eco)
int kbhit(void);
    // comprueba si hay pulsaciones de teclado
int ungetch(int);
    // devuelve el caracter al buffer de teclado
```
- Manipulacion de texto (y cursor) en pantalla

```
void clrscr(void);
    // borra e inicializa la pantalla
void clreol(void);
    // borra la linea desde el cursor al final
void delline(void);
    // elimina la linea donde se encuentra el cursor
void gotoxy(int x, int y);
    // posiciona el cursor
```

```

void insline(void);
    // inserta una linea en blanco
int movetext(int _left, int _top, int _right, int _bottom,
             int _destleft, int _desttop);
    // copia texto de una zona de pantalla a otra

- Movimientos de bloques
    int gettext(int _left, int _top, int _right, int _bottom,
void* _destin);
    // copia texto de pantalla a memoria
    int puttext(int _left, int _top, int _right, int _bottom,
void* _source);
    // copia texto de memoria a pantalla

- Control de ventanas
    void textmode(int _mode);
        // pone el modo texto
    void window(int _left, int _top, int _right, int _bottom);
        // define una ventana de texto
    void _set_screen_lines(int nlines);
    void _setcursortype(int _type);
        // _NOCURSOR, _SOLIDCURSOR, _NORMALCURSOR

- Ajuste de color del texto y del fondo
    void textcolor(int _color);
        // actualiza el color de texto
    void textbackground(int _color);
        // actualiza el color de fondo
    void textattr(int _attr);
        // actualiza el color de texto y el de fondo al mismo tiempo

- Control de intensidad
    void intensevideo(void);
    void highvideo(void);
        // alta intensidad
    void lowvideo(void);
        // baja intensidad
    void normvideo(void);
        // intensidad normal
    void blinkvideo(void);
        // parpadeo

- Informacion
    void gettextinfo(struct text_info *_r);
        // informacion sobre la ventana actual
    int wherex(void);
        // valor de la coordenada x del cursor
    int wherey(void);
        // valor de la coordenada y del cursor

```