

Trabalhando com o modo gráfico do Turbo C

Sempre que você quiser utilizar o modo gráfico no *Borland Turbo C*, você deve colocar o código que está em negrito no exemplo seguinte (em outros ambientes de programação o processo pode ser diferente):

```
#include <graphics.h> // biblioteca do modo gráfico
#include <stdio.h>
#include <conio.h>

void main()
{
    // variáveis necessárias ao modo gráfico:
    int driverVideo = DETECT,
        modoVideo, codigoErro;

    // Função que passa para o modo gráfico:
    initgraph(&driverVideo, &modoVideo, "c:\\tclite\\bgi");

    // Verifica se houve algum erro e, se não,
    // continua o programa:
    codigoErro = graphresult();
    if(codigoErro == grOk)
    {
        // só entra neste IF se o modo
        // gráfico funcionar, logo, seu programa
        // deve vir aqui dentro!

        // Segue um pequeno exemplo:
        line(100, 100, 400, 400); // desenha uma linha
        getch(); // espera pressionar uma tecla

        // Antes de terminar o programa, você deve
        // voltar para o modo texto. Para fazer
        // isso, use closegraph:
        closegraph();
    }
    else
    {
        // só entra aqui se ocorreu um erro,
        // ou seja, se o modo gráfico
        // não estava disponível:
        printf("Erro ao passar para o modo gráfico:");
        printf("%s\n", grapherrormsg(codigoErro));
        printf("Pressione qualquer tecla para terminar");
        getch();
    }
}
```

É (somente) nesta região que você utiliza as funções de modo gráfico!

Algumas funções do modo gráfico

Depois de estar no modo gráfico você pode utilizar uma série de funções de desenho:

- **setcolor(cor);**
Muda a cor de desenho. As cores possíveis são:
WHITE - branco BLACK - Preto
GREEN - verde YELLOW - Amarelo
LIGHTGRAY - cinza claro DARKGRAY - cinza escuro
BLUE - azul LIGHTBLUE - azul claro
GREEN - verde LIGHTGREEN - verde claro
CYAN - marinho LIGHTCYAN - marinho claro
RED - vermelho LIGHTRED - vermelho claro
MAGENTA - laranja LIGHTMAGENTA - Laranja claro
BROWN - marrom
- **setbkcolor(cor);**
Muda a cor do fundo da tela (inteira).
As cores possíveis são as mesmas da função anterior.
- **line(colInic, linInic, colFinal, linFinal);**
Desenha uma linha.
- **rectangle(colInic, linInic, colFinal, linFinal);**
Desenha um retângulo (seu fundo não é preenchido).
- **circle(col, lin, raio);**
Desenha um círculo (seu fundo não é preenchido).
- **setfillstyle(padrão, cor);**
Muda o padrão e a cor de preenchimento que são utilizados nas funções *floodfill*, *bar* e *bar3d*.
Alguns padrões possíveis são os seguintes:
SOLID_FILL - cor sólida
LINE_FILL - tracejado horizontal (---)
LTSLASH_FILL - tracejado inclinado grosso (///)
BKSLASH_FILL - tracejado inclinado invertido (\\ \\ \\)
HATCH_FILL - em forma de cruz
WIDE_DOT_FILL - em forma de pontos espaçados
As cores possíveis são as mesmas da função *setcolor*.
- **floodfill(col, lin, corBorda);**
Preenche (pinta) a partir das coordenadas passadas como parâmetro, utilizando o padrão e a cor especificados pela função *setfillstyle*. O preenchimento só pára quando a cor da borda for encontrada.
- **settextstyle(fonte, direção, tamanho);**
Muda a fonte, a direção e o tamanho de um texto a ser escrito na tela. As fontes possíveis são:
DEFAULT_FONT (padrão), SMALL_FONT (pequena),
TRIPLEX_FONT, SANS_SERIF_FONT e GOTHIC_FONT.
As direções possíveis são: HORIZ_DIR e VERT_DIR
Os tamanhos são: 1(8x8), 2(16x16)... 10(80x80).
- **outtextxy(col, lin, "texto");**
Escreve um texto na tela, utilizando as configurações de *settextstyle*.

OBS1: A variável *driverVideo* é inicializada como *DETECT*, indicando para o TurboC que ele deve detectar o melhor driver e o melhor modo de vídeo para o dispositivo atual.

OBS2: O terceiro parâmetro da função *initgraph* indica o diretório onde os *drivers* (controladores) de vídeo e os arquivos de fontes de texto do modo gráfico encontram-se.