

Trabalhando com components gráfico em tela com linha de comando: biblioteca CONIO2.h

**Usando biblioteca externa
Compilando e Linkando**

PRO - Fundamentos de programação

Biblioteca conio2.h

Modo gráfico em linha de comando

- Permite manipular componentes gráficos pela interface de linha de comando
- Algumas funções disponíveis nesta biblioteca:
 - `gotoxy(1,1);`
 - `textcolor(15);`
 - `textbackground(1);`

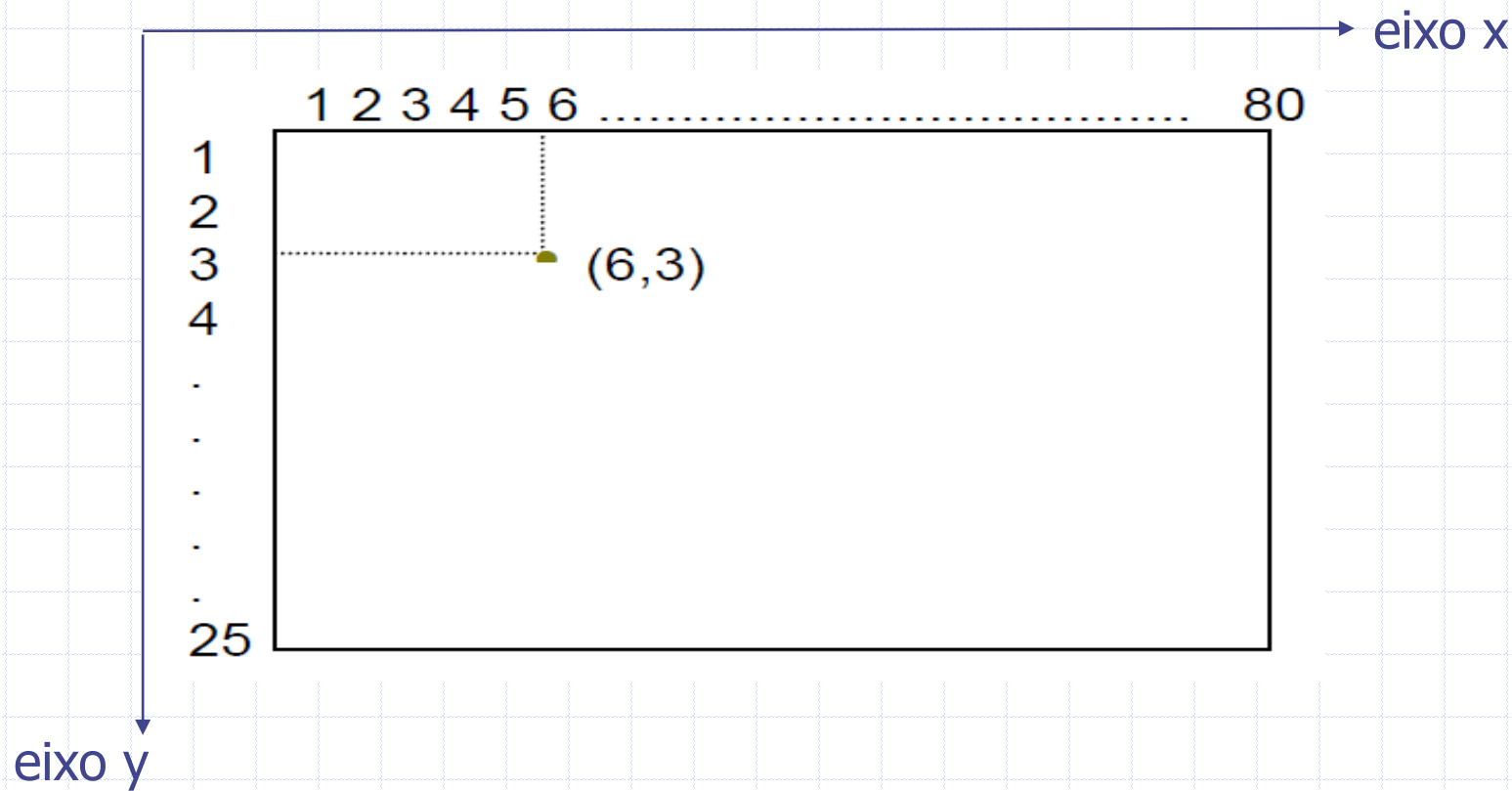
Cores

`textcolor(15);` // Altera a cor do texto

`textbackground(1);` // Altera a cor do fundo

Cor	Nº	Cor	Nº	Cor	Nº	Cor	Nº
Preto	0	Vermelho	4	Cinza escuro	8	Vermelho claro	12
Azul	1	Magenta	5	Azul claro	9	Magenta claro	13
Verde	2	Marrom	6	Verde claro	10	Amarelo	14
Ciano	3	Cinza claro	7	Ciano claro	11	Branco	15

A tela em modo texto



- A tela em modo texto contém **25 linhas** (numeradas de 1 a 25) sendo que cada linha contém **80 colunas** (numeradas de 1 a 80), formando assim uma matriz de **80 colunas** por **25 linhas**.

gotoxy(coluna,linha)

```
gotoxy(1,1); printf(".-----.");
```

//posiciona para ser impresso na coluna 1 e linha 1

```
gotoxy(1,2); printf("| Exemplo de tela |");
```

//posiciona para ser impresso na coluna 1 e linha 2

```
gotoxy(1,3); printf("| Linguagem C |");
```

//posiciona para ser impresso na coluna 1 e linha 3

```
gotoxy(1,4); printf("'-----'");
```

//posiciona para ser impresso na coluna 1 e linha 4

```
int main()
```

```
{
```

```
    textcolor(15);
```

```
    textbackground(1);
```

```
    gotoxy(1,1); printf("-----.");
```

```
    gotoxy(1,2); printf("| Exemplo de tela |");
```

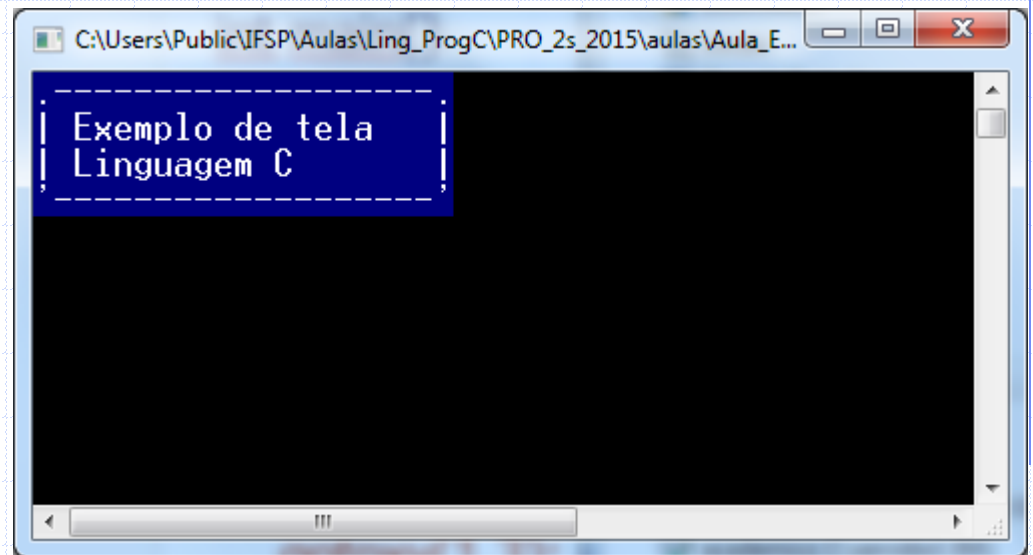
```
    gotoxy(1,3); printf("| Linguagem C |");
```

```
    gotoxy(1,4); printf("-----.");
```

```
    printf("\n\n");
```

```
    getch();
```

```
}
```



```
int main()
{
    int num;
    textcolor(15);
    textbackground(1);

    gotoxy(1,1); printf("-----.");
    gotoxy(1,2); printf("| Exemplo de tela      |");
    gotoxy(1,3); printf("| Linguagem C          |");
    gotoxy(1,4); printf("| Digite um numero:    | ");
    gotoxy(1,5); printf("-----.");
    gotoxy(20,4); scanf("%d", &num);

    clrscr();      // Serve para Limpar a tela.
    gotoxy(20,12); printf("Voce digitou o numero: %d", num);
    printf("\n\n");
    getch();
}
```

```
int main()
```

```
{
```

```
    int num;
```

```
    textcolor(15);
```

```
    textbackground(1);
```

```
    gotoxy(1,1); printf(".------. ");
```

```
    gotoxy(1,2); printf("| Exemplo de tela |");
```

```
    gotoxy(1,3); printf("| Linguagem C |");
```

```
    gotoxy(1,4); printf("| Digite um numero: | ");
```

```
    gotoxy(1,5); printf(".------.");
```

```
    gotoxy(20,4); scanf("%d", &num);
```

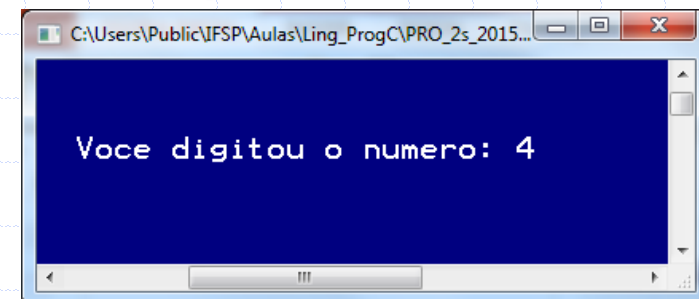
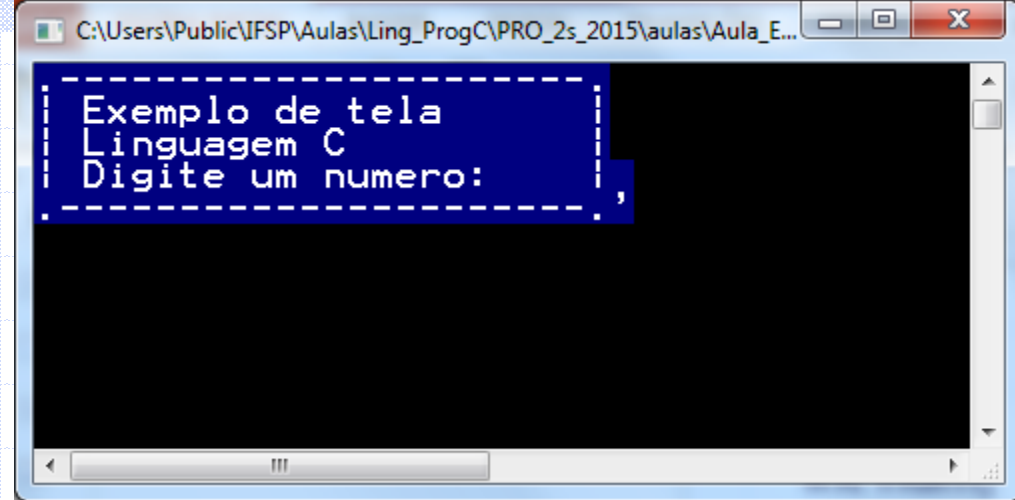
```
    clrscr();           // Serve para Limpar a tela.
```

```
    gotoxy(20,12); printf("Voce digitou o numero: %d", num);
```

```
    printf("\n\n");
```

```
    getch();
```

```
}
```



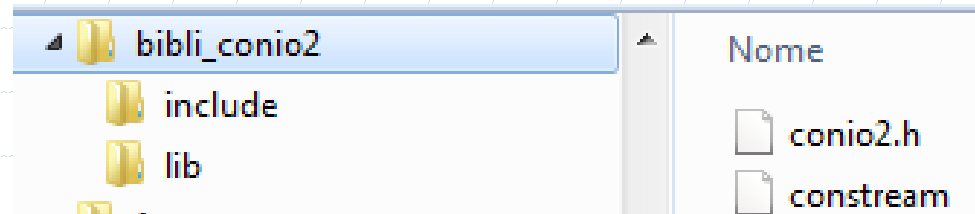
Utilizando conio2.h

GCC: linha de comando

- Compilar:
 - adicionar a biblioteca .h externa ao gcc (protótipos) colocando em "" o caminho em que foi adicionado a biblioteca.

```
#include "C:\Temp\bibli_conio2\include\conio2.h"  
gcc exemplo1.c -c
```

- No exemplo a biblioteca foi instalada em C:\Temp\biblio_conio2

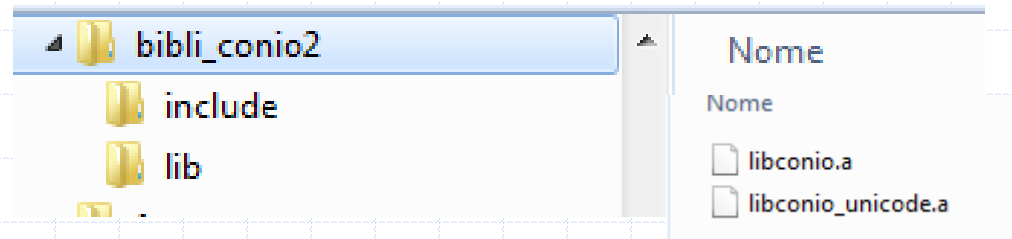


Utilizando conio2.h

GCC: linha de comando

- Linkagem:
- No exemplo a biblioteca foi instalada em C:\Temp\biblio_conio2\lib

```
C:\Temp> gcc -LC:\Temp\biblio_conio2\lib -o exemplo1.exe xemplo1.exe -l conio
```



Outras funções conio2

- **void cclear (void) :**
 - Limpa o resto da linha da posicao que o curso se encontra até o final sem mover o cursor
- **void clrscr (void):**
 - Limpa toda a tela
- **void delline (void):**
 - Remove a linha corrente (que está o cursor) e move todas as linhas abaixo para cima.
- **void insline (void):**
 - Insere uma nova linha na posição que se encontra o cursor. O conteúdo é movido para linha abaixo
- **int wherex (void);**
- **int wherey (void);**
- **void puttext (int left, int top, int right, int bottom, struct char_info * buf);**
- **void movetext (int left, int top, int right, int bottom, int destleft, int desttop);**
- **void cputsxy (int x, int y, char * str):** posiciona string na posicao x,y
- **void putchxy (int x, int y, char ch):** posiciona o char na posição x,y
- **void highvideo (void); void lowvideo (void);**

Utilizando conio2.h

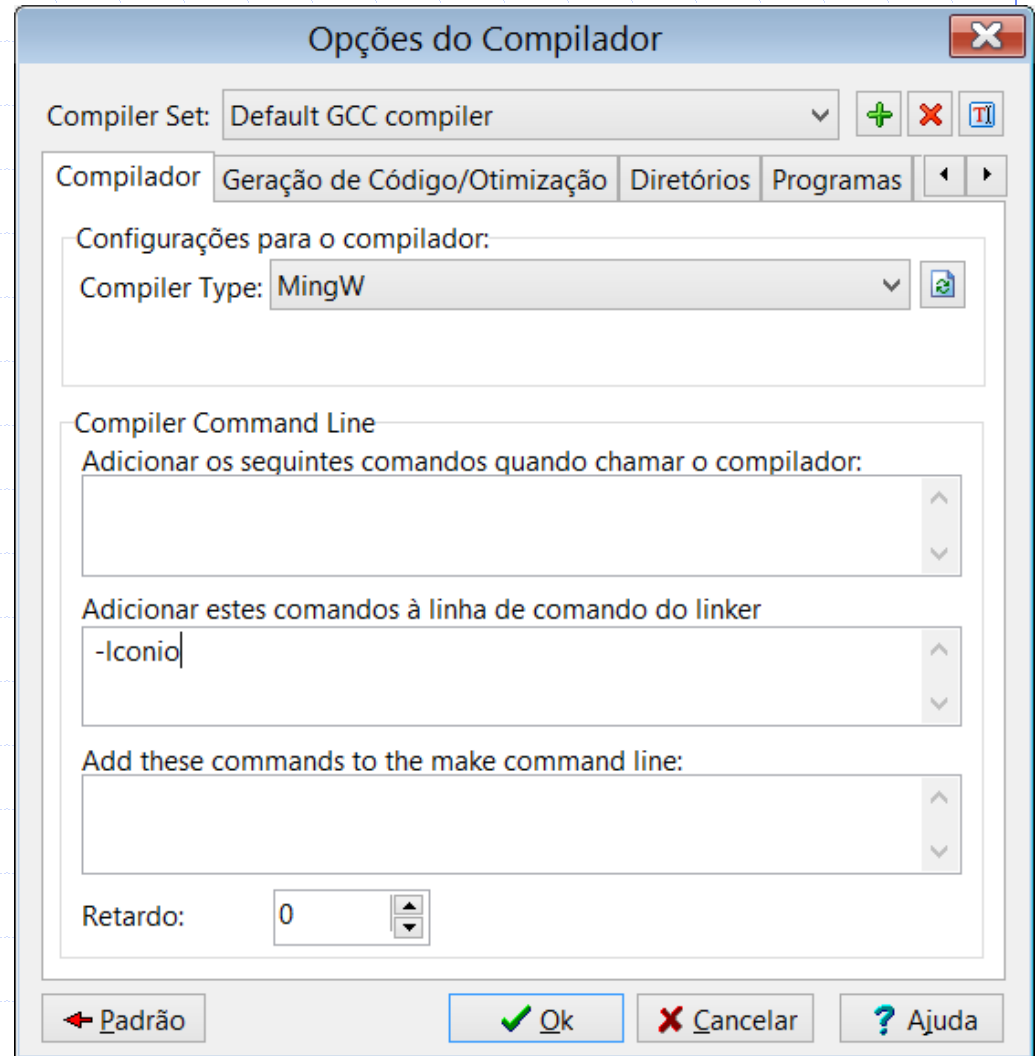
DEV C++

- Algumas funções que vamos utilizar estão na biblioteca **conio2.h** que não estão presentes na instalação padrão do DEV C++.
- Copie os arquivos **conio2.h** e **constream** para a pasta **MinGW32\include** do DEV C++.
- Copie os arquivos **libconio.a** e **libconio_unicode.a** para a pasta **MinGW32\lib** do DEV C++.

Configurando o DEV C++

No DEV C++:

- Clique no menu **Ferramentas**
- Clique em **Opções do compilador**
- Digite o código **-lconio** no campo: **Adicionar estes comandos à linha de comando do linker.**
- Clique em **OK** para confirmar.



Cores

`textcolor(15);` // Altera a cor do texto

`textbackground(1);` // Altera a cor do fundo

Cor	Nº	Cor	Nº	Cor	Nº	Cor	Nº
Preto	0	Vermelho	4	Cinza escuro	8	Vermelho claro	12
Azul	1	Magenta	5	Azul claro	9	Magenta claro	13
Verde	2	Marrom	6	Verde claro	10	Amarelo	14
Ciano	3	Cinza claro	7	Ciano claro	11	Branco	15

enum COLORS

```
{ BLACK, BLUE , GREEN ,CYAN,  
  RED,MAGENTA, BROWN, LIGHTGRAY,  
  DARKGRAY,LIGHTBLUE,LIGHTGREEN,LIGHTCYAN,  
  LIGHTRED,LIGHTMAGENTA,YELLOW,WHIT };
```