

Linguagem C

Professor Alexandre Moreno

Comandos úteis

A biblioteca **conio.c** é a modificação da biblioteca conio.h para poder disponibilizar algumas funcionalidades no programa [Dev C/C++](#)

Com ela é possível usar os comandos `textcolor()`, `backgroundcolor()`, `gotoxy()`, `clrscr()`, etc, e deixar a saída dos dados com um layout mais agradável para o usuário.

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

int main () {

    system("pause");
    return 0;
}
```

Observação:

É necessário instalar a biblioteca em seu compilador.

O compilador disponibilizado pelo professor Moreno, já possui a biblioteca inserida.

O comando **textcolor(x)** possibilita modificar a cor da letra do texto.

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

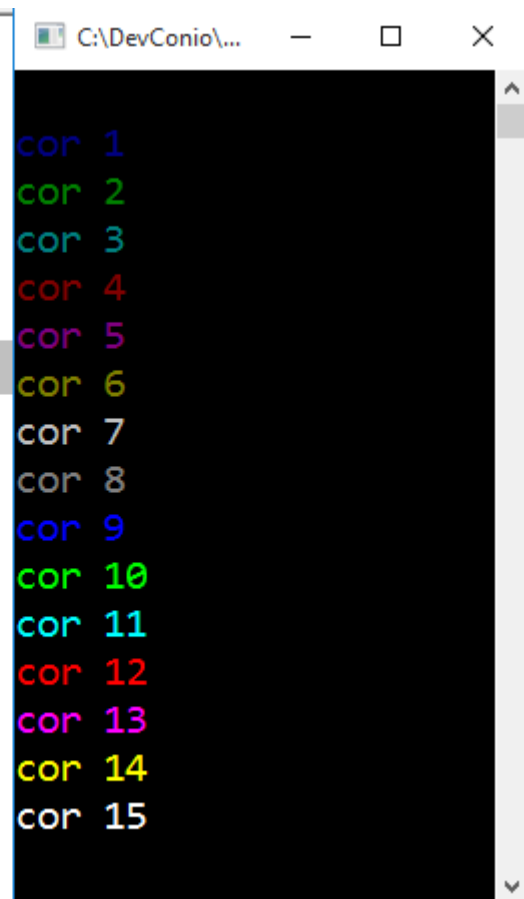
int main () {
    char c;
    textcolor(14);
    printf("escrevendo em AMARELO \n\n");
    textcolor(10);
    printf("escrevendo em VERDE \n\n");
    c=getch();
    return 0;
}
```



As cores do textcolor variam de 0 a 15. A cor 0 (zero) é preto

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

int main () {
    char c;
    int x;
    for (x=0; x<=15; x++){
        textcolor(x);
        printf("cor %d \n",x);
    }
    c=getch();
    return 0;
}
```

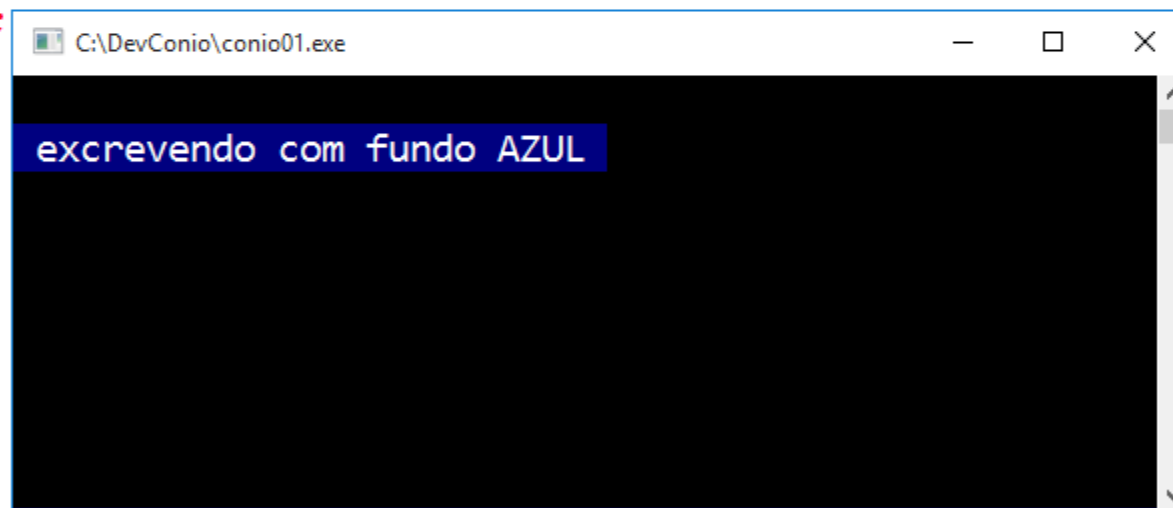


```
cor 1
cor 2
cor 3
cor 4
cor 5
cor 6
cor 7
cor 8
cor 9
cor 10
cor 11
cor 12
cor 13
cor 14
cor 15
```

O comando **textbackground(x)** possibilita modificar a cor do fundo do texto.

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

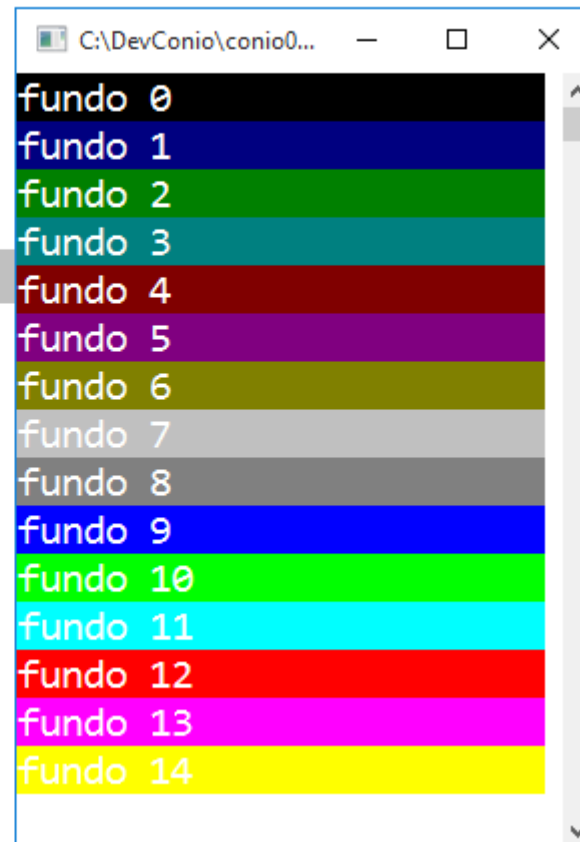
int main () {
    char c;
    int x;
    textcolor(15);    //letra branca
    textbackground(1); //fundo azul
    printf("\n excrevendo com fundo AZUL ");
    c=getch();
    return 0;
}
```



O comando **textbackground(x)** possibilita modificar a cor do fundo do texto.

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

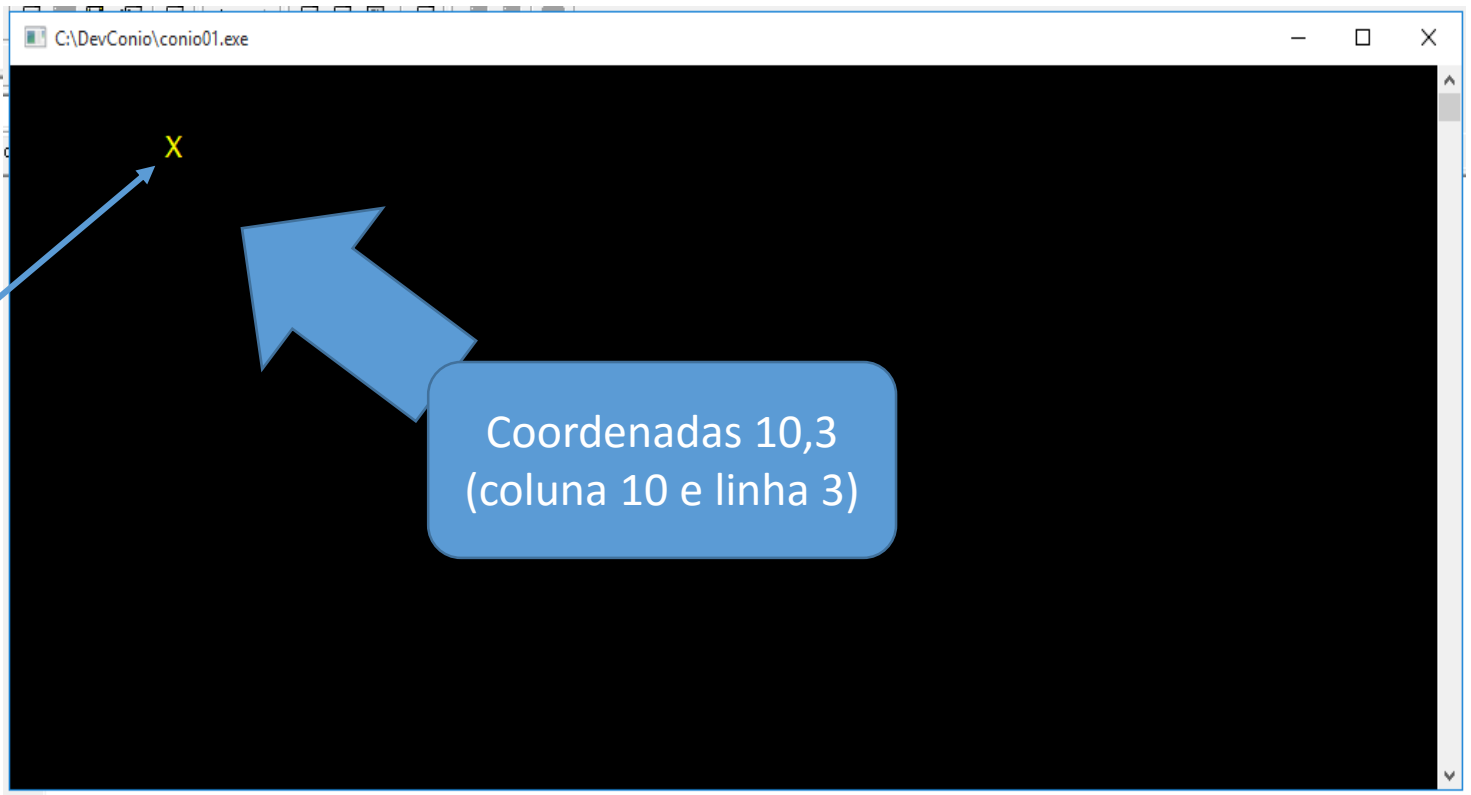
int main () {
    char c;
    int x;
    textcolor(15); //letra branca
    for (x=0; x<=15; x++) {
        textbackground(x);
        printf("fundo %d \n", x);
    }
    c=getch();
    return 0;
}
```



O comando **gotoxy(x,y)** possibilita posicionar o cursor em uma determinada localização da tela antes de uma entrada ou saída. Os parâmetros são COLUNA, LINHA. Exemplo: `gotoxy(10,3);`

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.c>

int main () {
    char c;
    gotoxy(10,3);
    printf("X \n\n");
    c=getch();
    return 0;
}
```



Geralmente, a tela (terminal CMD) possui o tamanho 80,25 (80 colunas por 25 linhas). Caso queira redimensionar sua tela de terminal, basta utilizar o comando `system("mode 80,25");`

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

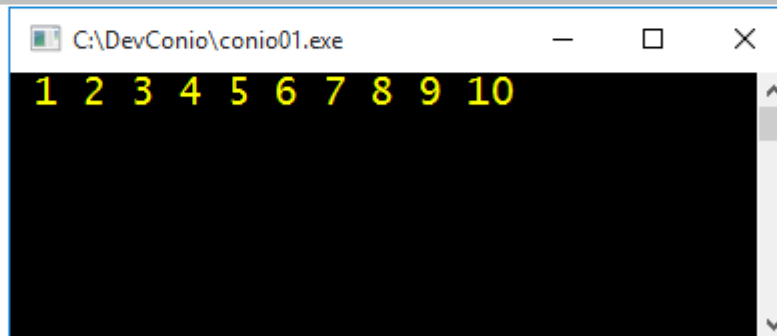
int main () {
    char c;
    system("mode 80,25");
    c=getch();
    return 0;
}
```

O comando Sleep efetua uma pausa (definida por parâmetro em milissegundos). Exemplo: Sleep (300); (pausa de 300 milissegundos)

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.c>

int main () {
    char c;
    for (int x=1; x<=10; x++){
        Sleep(300);
        printf(" %d", x);
    }
    c=getch();
    return 0;
}
```

Sleep (com o
S maiúsculo)



O comando Beep emite um som de frequência x por um tempo de y milissegundos (dois parâmetros). Exemplo: **Beep** (300);

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.c>
```

```
int main () {
    char c;
```

```
    Beep(150,400);
```

```
    c=getch();
    return 0;
}
```



Beep (com o
B maiúsculo)

O comando **kbhit** permite que seja recebida o valor de uma tecla e o programa continue em funcionamento (sem aguardar a tecla)

```
#include <conio.c>
int main () {
    char tecla;
    int x=0;
    do{
        gotoxy(10,10); printf("%7d",x++);
        if (kbhit()) {
            tecla=getch();
            if (tecla<0) //pressionado tecla especial
                tecla=getch();
            gotoxy(10,11);
            switch (tecla){
                case 72: //cima
                    printf("pressionou para cima "); break;
                case 80: //baixo
                    printf("pressionou para baixo "); break;
                case 75: //esq
                    printf("pressionou para esquerda"); break;
                case 77: //dir
                    printf("pressionou para direita "); break; }
        }
    } while (tecla!=27);
}
```