

Trabalho em Grupo Sobre Placa Gráfica no Modo Texto

Gerado por Doxygen 1.12.0

1 Documentação do Projecto Trabalho 1 em Grupo	1
1.1 Introdução	1
1.2 Autores	1
1.3 Dificuldades Encontradas Durante o Desenvolvimento	5
1.3.1 Incompatibilidade de Nomes de Ficheiro no Ambiente DOSBox	5
1.3.2 Configuração da Geração de Documentação em PDF	5
2 Índice de Tópicos	7
2.1 Tópicos	7
3 Índice dos ficheiros	9
3.1 Lista de ficheiros	9
4 Tópico	11
4.1 UserTypes	11
4.1.1 Descrição detalhada	11
4.1.2 Documentação dos valores da enumeração	11
4.1.2.1 Bool	11
5 Documentação do ficheiro	13
5.1 LC_VID.h	13
5.2 utypes.h	13

Capítulo 1

Documentação do Projecto Trabalho 1 em Grupo

1.1 Introdução

Este é o documento gerado automaticamente para o projecto Trabalho 1 em Grupo. Ele descreve a estrutura do código, funções, tipos de dados e mais.

1.2 Autores

Este projecto foi desenvolvido por:

- António Frazão
- Ekoko Clesh

Versão

1.0

Data

2025-09-01

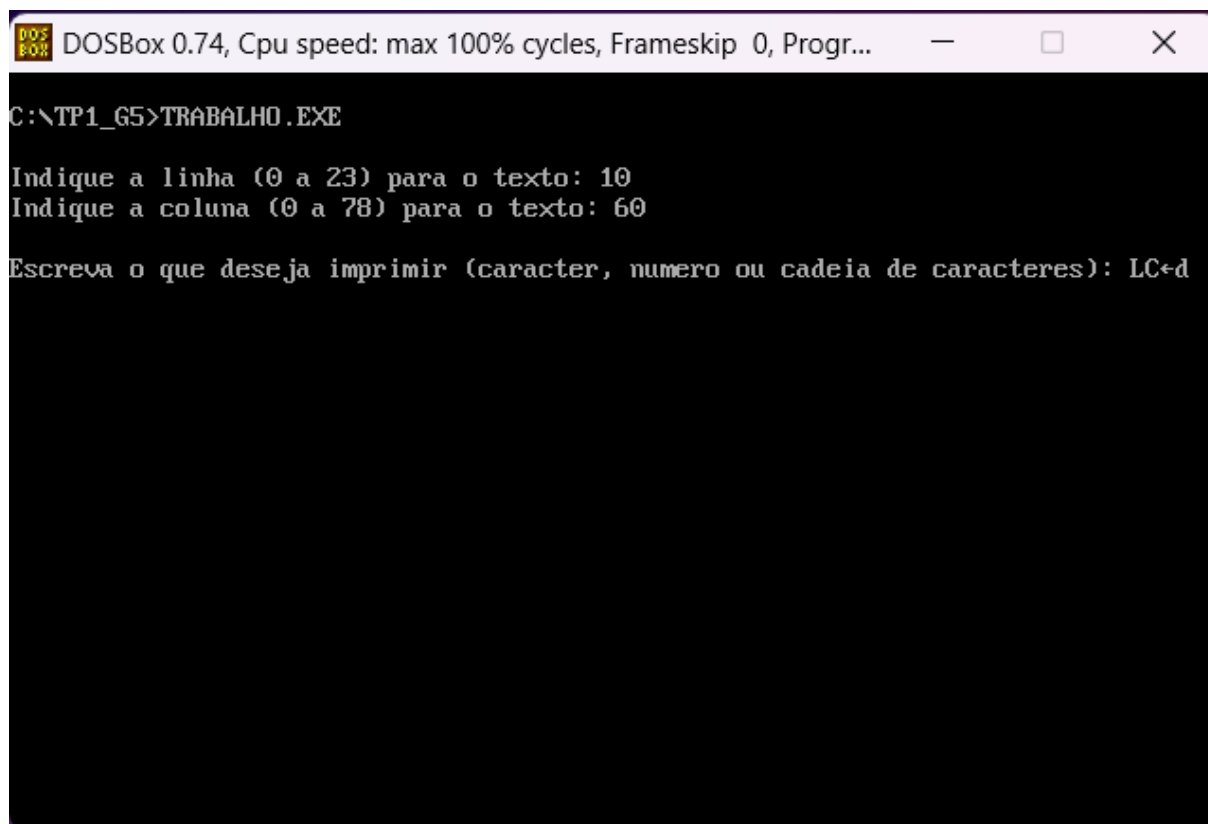


Figura 1.1 Figura 1: Execução 1

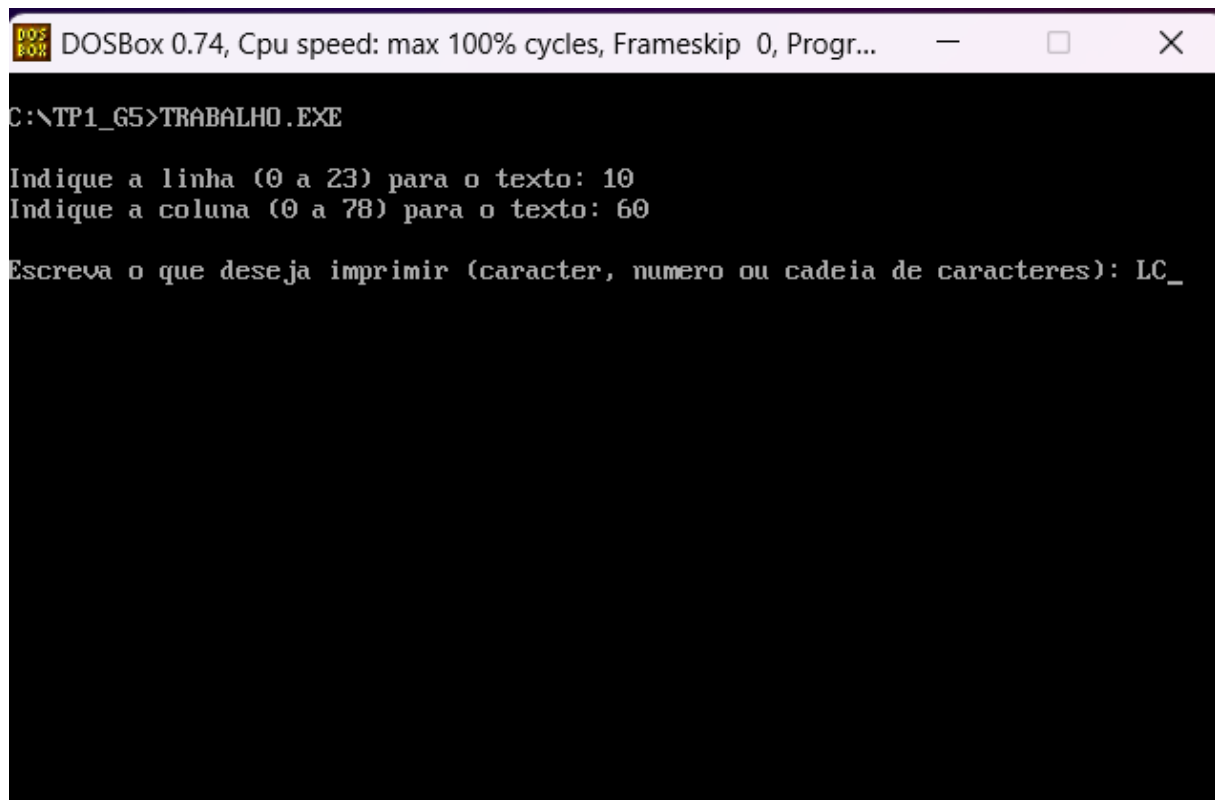


Figura 1.2 Figura 2: Execução 2

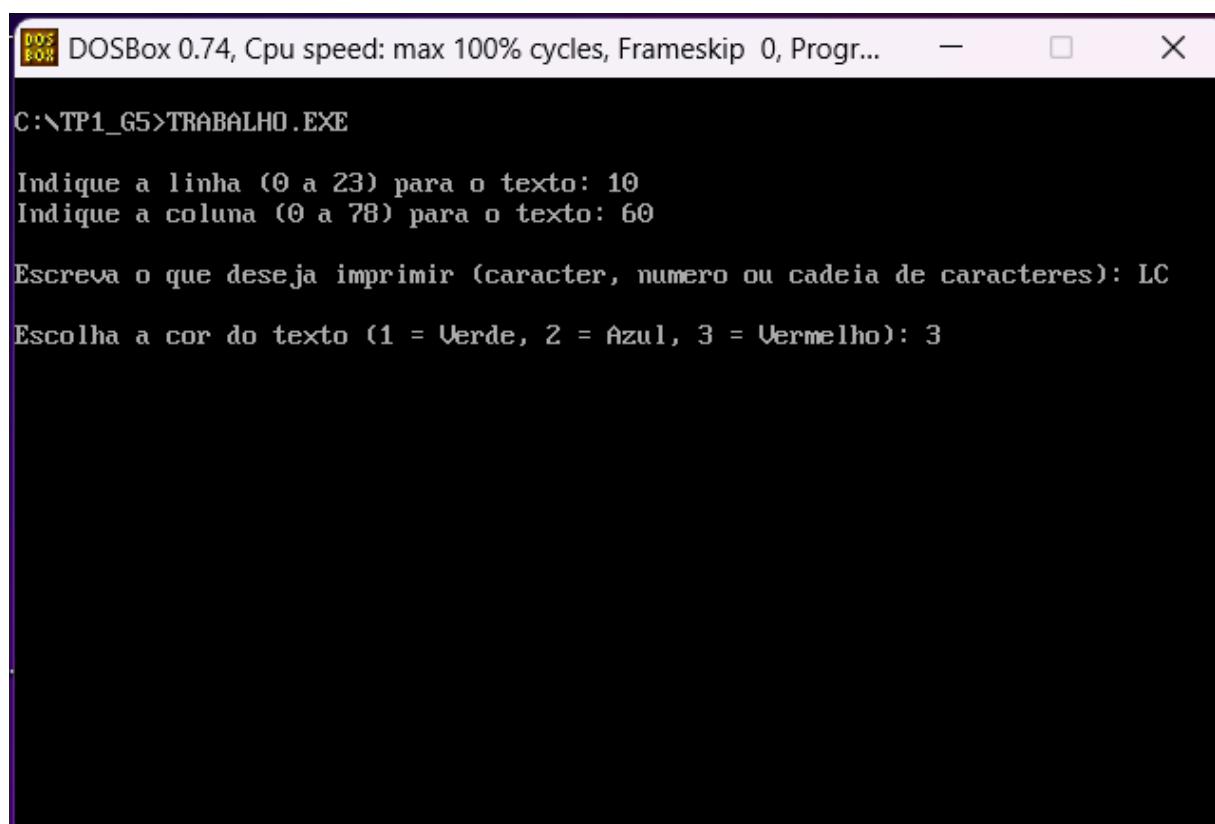


Figura 1.3 Figura 3: Execução 3

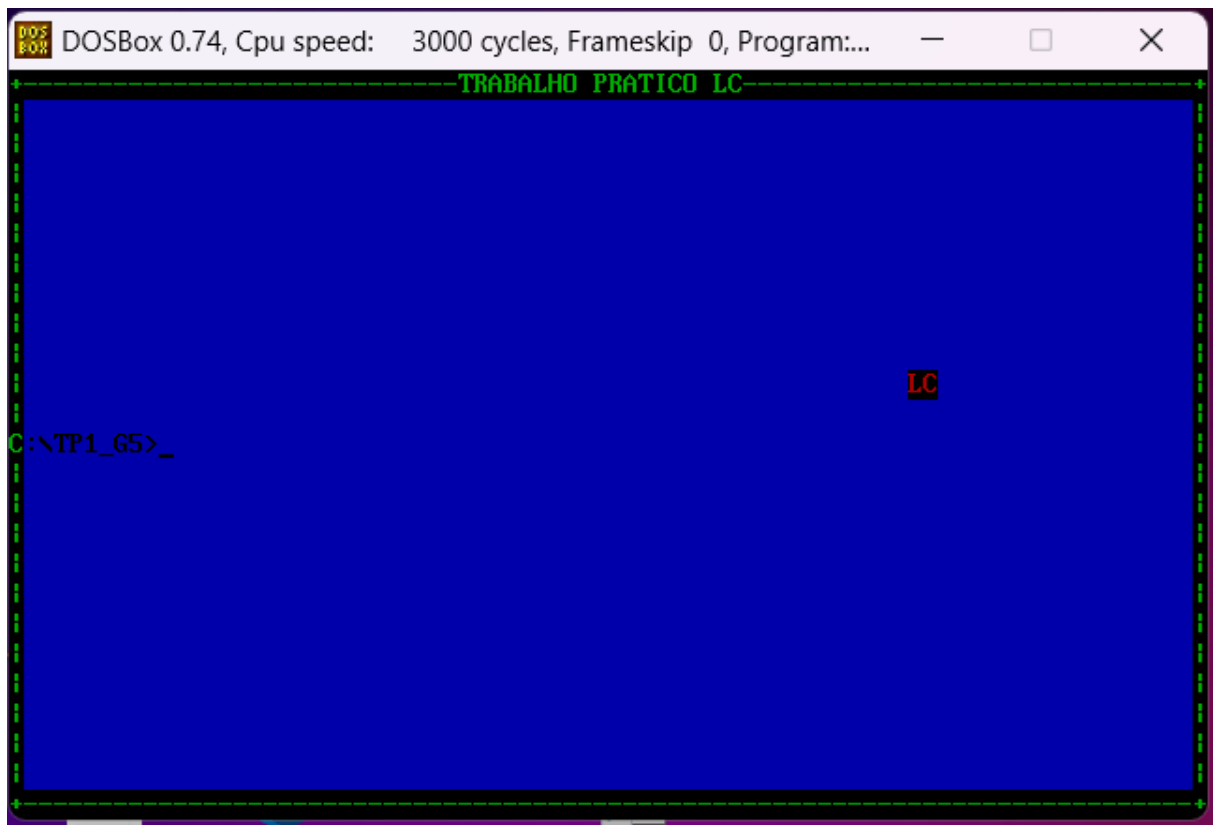


Figura 1.4 Figura 4: Execução 4

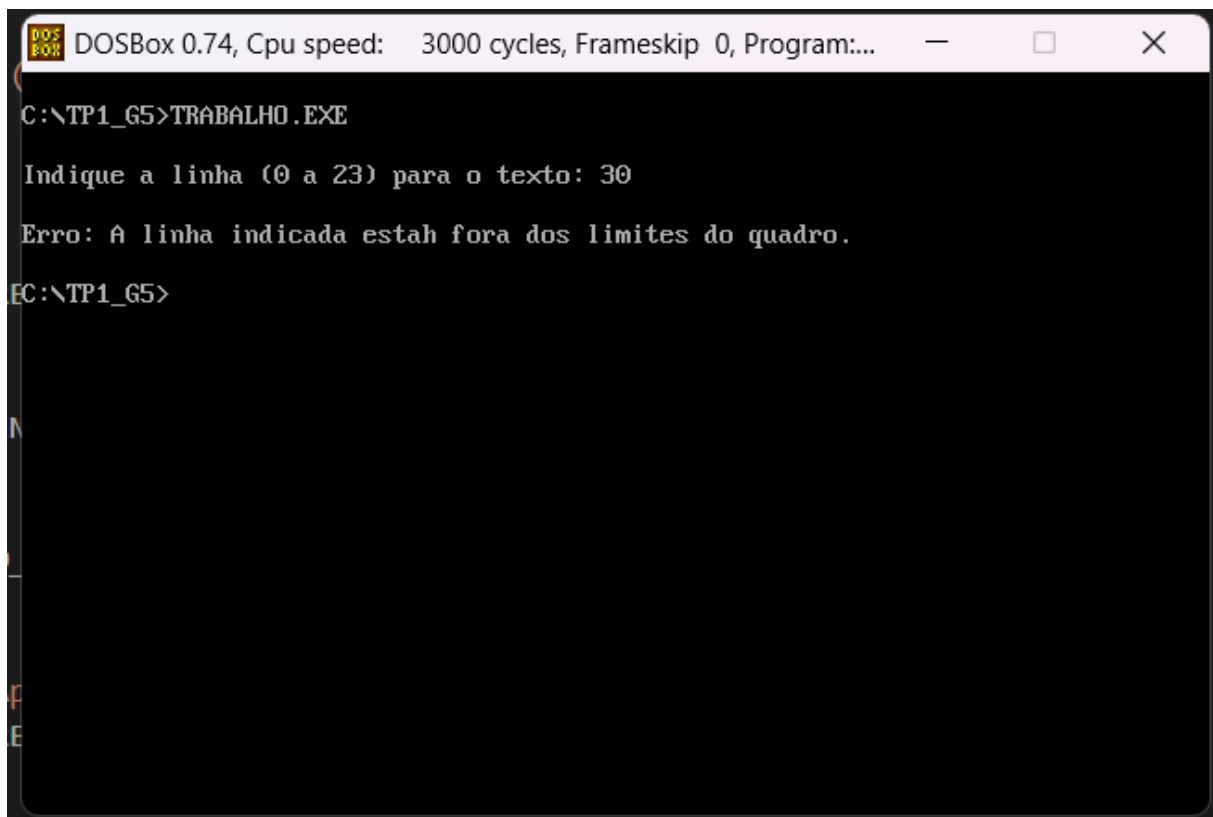


Figura 1.5 Figura 4: Execução 5

/**

1.3 Dificuldades Encontradas Durante o Desenvolvimento

A execução deste projecto, levou-nos a enfrentar alguns obstáculos técnicos que exigiram uma análise cuidada e soluções específicas da nossa parte. Estes desafios, embora tivessem atrasado momentaneamente o progresso, acabaram por se revelar oportunidades de aprendizagem valiosas sobre as ferramentas que tínhamos à nossa disposição.

1.3.1 Incompatibilidade de Nomes de Ficheiro no Ambiente DOSBox

O primeiro grande desafio com que nos deparámos surgiu durante a compilação do código-fonte. O projecto recorreu ao emulador DOSBox para garantir a compatibilidade com compiladores mais antigos, mas esta decisão trouxe consigo uma limitação inesperada. O processo de compilação, automatizado através de um `Makefile`, falhava consistentemente sempre que os ficheiros de código tinham nomes mais longos, como `LC_video_↵text.c`.

A nossa análise do problema revelou que a causa era a estrita convenção de nomes de ficheiro **8.3** (oito caracteres para o nome, três para a extensão), imposta pelo sistema de ficheiros FAT16 que o DOSBox emula. O emulador renomeava automaticamente os ficheiros longos para um formato abreviado (ex: `LC_VID~1.C`), mas o `Makefile` que tínhamos construído continuava a referenciar o nome original. Esta discrepância resultava em erros de "Ficheiro não encontrado", bloqueando a compilação.

A solução que encontrámos passou por uma refactorização: todos os nomes de ficheiro do projecto foram por nós encurtados para se conformarem ao padrão 8.3, como por exemplo `LC_vid.c`. Esta adaptação, embora simples, foi crucial para assegurarmos a estabilidade do ambiente de compilação e demonstrou na prática como sistemas legados podem impor restrições ao desenvolvimento moderno.

1.3.2 Configuração da Geração de Documentação em PDF

Um segundo obstáculo, não menos complexo, surgiu na fase final, quando tentámos gerar a documentação do projecto em formato PDF. A cadeia de ferramentas, composta pelo Doxygen e pelo sistema de composição de texto MiKTeX (LaTeX), apresentou-nos uma série de problemas de configuração.

O processo falhava durante a compilação dos ficheiros `.tex` intermediários, com erros que indicavam a falta de "pacotes" LaTeX necessários para processar os comandos específicos do Doxygen. Além disso, as definições de segurança do sistema operativo Windows impediam que o MiKTeX descarregasse e instalasse automaticamente estes pacotes em falta, e bloqueavam a execução do compilador `pdflatex` quando este era invocado a partir de um terminal com privilégios de administrador.

A resolução deste impasse exigiu da nossa equipa uma intervenção a dois níveis. Primeiro, foi necessário configurar manualmente o MiKTeX (em modo de administrador) para que este passasse a instalar as dependências de forma automática. Segundo, tivemos de garantir que todo o processo de compilação da documentação fosse executado a partir de um terminal com permissões de utilizador padrão. Este processo reforçou a importância de compreendermos não só as ferramentas em si, mas também a forma como estas interagem com o sistema operativo.

Capítulo 2

Índice de Tópicos

2.1 Tópicos

Esta é uma lista de todos os tópicos e suas descrições:

UserTypes 11

Capítulo 3

Índice dos ficheiros

3.1 Lista de ficheiros

Lista de todos os ficheiros documentados com uma breve descrição:

LC_VID.h	13
utypes.h	13

Capítulo 4

Tópico

4.1 UserTypes

Definições de tipos

- typedef unsigned char **uchar**
uchar é mais breve/curto que unsigned char
- typedef unsigned short **ushort**
ushort é um inteiro curto sem sinal
- typedef unsigned int **uint**
uint é um inteiro sem sinal
- typedef unsigned long **ulong**
ulong é um inteiro longo sem sinal
- typedef unsigned char **Byte**
conjunto de 8 bits para máquinas i386
- typedef unsigned short **Word**
conjunto de 16 bits para máquinas i386

Enumerações

- enum Bool { FALSO = 0 , VERDADE = 1 }

4.1.1 Descrição detalhada

Definições de tipos para Programador

4.1.2 Documentação dos valores da enumeração

4.1.2.1 Bool

```
enum Bool
```

Definição de Tipo booleano

Exemplo de uso:

```
Bool feito = falso;  
enquanto (feito == falso){...}
```

Valores de enumerações

	FALSO	Valor Falso.
	VERDADE	Valor Verdadeiro.

Capítulo 5

Documentação do ficheiro

5.1 LC_VID.h

```
00001 #ifndef _LC_VIDEO_TEXT_H_
00002 #define _LC_VIDEO_TEXT_H_
00003
00004 #include "utypes.h" // Inclui o tipo Bool e outros tipos definidos pelo utilizador
00005
00014 #define ENDERECO_VIDEO 0xB8000
00015 #define LARGURA 80
00016 #define ALTURA 25
00022 #define AZUL_FRENTE (1 << 0)
00023 #define VERDE_FRENTE (1 << 1)
00024 #define VERMELHO_FRENTE (1 << 2)
00025
00026 #define INTENSO (1 << 3)
00027
00028 #define AZUL_FUNDO (1 << 4)
00029 #define VERDE_FUNDO (1 << 5)
00030 #define VERMELHO_FUNDO (1 << 6)
00031
00032 #define NORMAL (VERMELHO_FRENTE | VERDE_FRENTE | AZUL_FRENTE)
00044 Bool printCharAt(char ch, int x, int y, char atributos);
00045
00054 Bool printStringAt(const char *str, int x, int y, char atributos);
00055
00067 Bool drwaFrame(const char *titulo, char atributos, int x, int y, int largura, int altura);
00068
00078 Bool printCharRepeated(char ch, int count, int x, int y, char atributos);
00079
00089 Bool clearScreen(int x, int y, int largura, int altura, char atributos);
00090
00101 Bool scrollRegion(int x, int y, int largura, int altura, int linhas, char atributos);
00104 #endif // LC_VIDEO_TEXT_H_
```

5.2 utypes.h

```
00001 #ifndef _UTYPES_H_
00002 #define _UTYPES_H_
00003
00010 typedef unsigned char uchar;
00011 typedef unsigned short ushort;
00012 typedef unsigned int uint;
00013 typedef unsigned long ulong;
00014
00015 typedef unsigned char Byte;
00016 typedef unsigned short Word;
00017
00026 typedef enum {
00027     FALSO = 0,
00028     VERDADE = 1
00029 } Bool;
00031 #endif
00032
```

