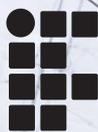


The background of the cover is a complex, multi-layered wireframe structure composed of interconnected lines forming various geometric shapes, primarily hexagons and pentagons, creating a crystalline or molecular-like appearance. The lines are thin and light gray, set against a plain white background.

L^AT_EX

Noções Básicas



INSTITUTO FEDERAL
Rio Grande do Sul
Campus Caxias do Sul

ISBN 978-65-86734-00-3

Munique dos Santos Lima
Luís Henrique Ribeiro da Silva
Érick Scopel

L^AT_EX

Noções Básicas

Autores

Munique dos Santos Lima
Luís Henrique Ribeiro da Silva
Érick Scopel

Edição

Munique dos Santos Lima
Luís Henrique Ribeiro da Silva

Edição Visual

Luís Henrique Ribeiro da Silva

Colaboração

Greice da Silva Lorenzetti Andreis
Lucas Pinto Dutra



Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul,
Campus Caxias do Sul

51 L7321	<p>Lima, Munique dos Santos LaTeX : noções básicas / Munique dos Santos Lima, Luís Henrique Ribeiro da Silva, Érick Scopel; colaboração, Greice da Silva Lorenzetti Andreis, Lucas Pinto Dutra -- Caxias do Sul, RS, 2019.</p> <p>81 p.: il. Inclui referências ISBN 978-65-86734-00-3</p> <p>1. Licenciatura em matemática. 2. LaTeX - noções básicas. 3. Tipografia matemática. I. Andreis, Greice da Silva Lorenzetti, colab. II. Dutra, Lucas Pinto, colab. III. Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul. IV. Título.</p> <p style="text-align: right;">CDU 51</p>
-------------	---

Ficha catalográfica elaborado pela bibliotecária Jaçanã Egges Pando - CRB 10/1936

PREFÁCIO

Ao ingressar no Instituto Federal do Rio Grande do Sul como docente, perguntei-me o que poderia fazer para oportunizar novas experiências aos professores em formação do Curso de Licenciatura em Matemática do Câmpus Caxias do Sul. Desde aluno à professor do IF, tinha conhecimento sobre a estrutura curricular do curso e, assim, comecei a pensar em como poderia agregar valor e ampliar essa formação propiciando aparatos teórico-práticos relevantes ao licenciando.

Refletindo sobre a formação continuada, cheguei a aspectos que me fizeram falta em um curso Pós-graduação em Matemática. Um deles, foi a ausência de contato com o LaTeX na graduação. Principalmente, porque o LaTeX é uma ferramenta utilizada mundo a fora para criar textos científicos configurando-se como o editor padrão dentro da área de matemática. Logo, surgiu a ideia de criar um projeto de ensino com o objetivo de oportunizar a familiarização com o LaTeX a partir do reconhecendo de comandos e da estrutura básica. E, o mais importante, propiciar ao licenciando em matemática um recurso que o auxilie em sua prática docente, desde a elaboração de uma prova até a produção de um artigo científico.

A partir de tais necessidades e anseios, em 2017 o projeto “LaTeX: Noções básicas” teve seu início através do edital de Ensino do Câmpus Caxias do Sul, o qual contemplou o projeto com uma bolsa. Desse modo, o grupo contou com dois professores em formação, Munique dos Santos Lima (bolsista) e Luís Henrique Ribeiro da Silva (voluntário), os quais realizaram as atividades propostas sob minha orientação.

Ao longo do desenvolvimento do projeto, conhecimentos foram adquiridos e compartilhados, exemplo disso são os minicursos de LaTeX oferecidos no IF e esta apostila.

Os minicursos instanciaram a prática docente, desde seu planejamento a sua execução. Em quatro edições, os minicursos contaram com cerca de 55 participantes dentre eles licenciandos do curso de Licenciatura em Matemática, professores e servidores do IF. Durante essas edições, surgiram questionamentos que levaram à modificações do material inicial até chegarmos nessa apostila, a qual se diferencia das demais por sua estrutura didática e seus exemplos em cada comando apresentado. Tudo pensado para o público alvo, professores de matemática.

Aos leitores que objetivam conhecer o LaTeX e/ou ter um a fonte rápida de consulta, tenham certeza que aqui alcançarão!

Caxias do Sul, 10 de junho de 2019.

Érick Scopel.

Sumário

1	Introdução ao Software e Primeiros Passos	9
1.1	O que é o \LaTeX ?	9
1.2	Termos Importantes	10
1.3	Utilizando o \LaTeX com o \TeXmaker	11
1.3.1	Instalando os <i>softwares</i> necessários	11
1.3.2	Abrindo o <i>software</i> pela primeira vez	11
1.3.3	Configurações Importantes do \TeXmaker	13
1.3.4	Dicionário	15
1.4	<i>Overleaf</i> , uma alternativa <i>online</i>	16
2	Configuração do Preâmbulo	17
2.1	Classes de Documentos	17
2.2	Pacotes	18
2.3	O ambiente <code>document</code>	19
3	Formatação do Texto	20
3.1	Caracteres Reservados	20
3.2	Unidades de Medida	21
3.3	Trabalhando em Capítulos e Seções	21
3.4	Utilização das Aspas	22
3.5	Alinhamento do Texto	22
3.5.1	Os comandos <code>flushleft</code> , <code>flushright</code> e <code>center</code>	22
3.5.2	Os comandos <code>hspace</code> e <code>vspace</code>	23
3.6	Tabulação	23
3.7	Quebras de Linha	23
3.8	Quebras de Página	24
3.9	Espaçamento entre Linhas	24
3.10	Estilos de Fonte	25
3.11	Cores	25
3.11.1	Texto em Destaque	27
3.11.2	Personalizando Cores	27

3.12	Listas	28
3.12.1	Listas aninhadas	29
3.13	Ambiente Verbatim	31
3.14	Ambiente Minipage	32
3.15	Ambiente <code>framed</code>	33
3.16	Criação de Títulos	33
3.16.1	Título com <code>maketitle</code>	34
4	Estrutura do Documento	36
4.1	Layout da Página	36
4.2	Sumário	36
4.3	Notas de Rodapé	37
4.4	Referências e Citações	37
4.4.1	O ambiente <code>thebibliography</code>	37
4.4.2	Citações	38
4.4.3	Referência Cruzada	38
4.5	Inserindo hyperlinks	39
4.5.1	Páginas da web	39
4.6	Documento por Módulos	40
5	Inserindo Imagens	41
5.1	Alinhando figuras	41
5.2	Ambiente <code>Figure</code>	43
5.3	Inserindo figuras lado a lado	43
5.4	Refletindo figuras	45
5.5	Rotacionando figuras	45
5.6	Figura ao lado de texto	45
5.7	Referenciando figuras	46
6	Matemática no \LaTeX	47
6.1	Expressões matemáticas	47
6.1.1	Ambiente <code>math</code>	47
6.1.2	Ambiente <code>displaymath</code>	47
6.2	Assistente para inserção de fórmulas (<code>TeXmaker</code>)	48
6.2.1	Ambiente <code>equation</code>	48
6.3	Fontes no modo matemático	49
6.4	Expoentes e índices	50
6.5	Frações	51
6.6	Raízes	51
6.7	Limites	52

6.8	Somatórios e Produtórios	53
6.9	Derivadas	53
6.10	Integrais	54
6.11	Linhas e chaves	55
6.12	Delimitadores	56
6.13	Estruturas	56
6.14	Matrizes	58
6.15	Símbolos	58
7	Tabelas	63
7.1	Ambiente <code>tabbing</code>	63
7.2	Ambiente <code>tabular</code>	64
7.2.1	Os comandos <code>cline</code> e <code>multicolumn</code>	65
7.3	Ambiente <code>table</code>	67
7.4	Ambiente <code>array</code>	68
7.4.1	Matrizes	68
7.4.2	Sistemas de equações e funções definidas por mais de uma sentença . . .	69
7.5	Ambiente <code>eqnarray</code>	70
7.5.1	Outros apontamentos	71
8	Caixas personalizadas com o pacote <code>mdframed</code>	73
8.1	O ambiente <code>mdframed</code>	73
8.2	Personalizando as caixas	73
8.2.1	Modificando as margens	74
8.2.2	Alterando a cor da fonte e de fundo	75
8.2.3	Criando um título e subtítulo	76
8.3	O ambiente <code>newmdenv</code>	79
	Referências	81

Capítulo 1

Introdução ao Software e Primeiros Passos

1.1 O que é o \LaTeX ?

O \LaTeX é um conjunto de macros (comandos) baseado no sistema de tipografia \TeX . O \TeX foi lançado em 1978, criado por Donald Knuth, com o objetivo de ser uma ferramenta prática e universal para a produção de textos com alta qualidade tipográfica. Por se tratar de um *software* livre, o seu uso é gratuito. Então, na década seguinte, surge uma adaptação do \TeX , o \LaTeX , desenvolvido por Leslie Lamport. O \LaTeX , em comparação com o seu antecessor, possui uma linguagem de mais fácil compreensão, podendo ser utilizada por leigos.

Retomando um termo utilizado anteriormente, tipografia é uma palavra de origem grega, *typos* (forma) e *graphein* (escrita). Desse modo, a tipografia é, basicamente, a forma como escrevemos o texto. Por exemplo, as fontes utilizadas em *softwares* de edição de texto têm tipografias distintas, como as fontes *Arial* e *Times New Roman*.

Esta apostila foi construída com um viés didático e visa instruir e facilitar a escrita de documentos no \LaTeX , com uma tipografia matemática. Ou seja, serão disponibilizadas ferramentas para a construção de expressões matemáticas, frequentemente utilizadas por estudantes de Licenciatura em Matemática e professores de Matemática, mas não descartando que esse *software* também poderá ser utilizado para a criação de documentos de outras áreas do conhecimento.

No campo da Matemática, o \LaTeX é predominantemente utilizado para criar listas de exercícios, avaliações e artigos científicos. Na área da Química, ele pode ser utilizado para escrever equações de balanceamento e fórmulas de composto químicos orgânicos e inorgânicos com qualidade singular. Além disso, na área da música, é possível digitar partituras musicais, novamente, com uma ótima qualidade tipográfica. Em resumo, no \LaTeX é possível digitar de documentos simples até livros completos.

Existem outros *softwares* para edição de texto, como o *Microsoft Office Word* e o *LibreOffice Writer*. Então, por que devo utilizar a linguagem \LaTeX ? Esses dois *softwares* são os mais conhecidos do mercado. Eles são práticos para digitar textos comuns, pois se tratam de *softwares*

do tipo WYSIWYG¹. Dessa forma, vemos o texto pronto à medida que digitamos. O \LaTeX não funciona assim. Aqui, digitamos o texto com base em uma linguagem de programação, e só vemos o documento depois que compilamos² ele.

São várias as vantagens de utilizar uma linguagem de programação. A primeira delas é que como trabalhamos com conceitos lógicos, é mais difícil que ocorra uma distração por conta da formatação visual do texto. Outra vantagem é que o controle sobre equações e tabelas é muito maior. Ao digitar uma equação em um editor de texto qualquer é comum ter problemas para centralizá-la de forma correta ou que ocorram alterações indesejadas. No \LaTeX , a equação é digitada manualmente com comandos, o que eleva o controle sobre ela.

1.2 Termos Importantes

Para dar continuidade com o estudo da linguagem \LaTeX , é necessário apresentar a definição de três termos muito utilizados na informática: código-fonte, compilação e *case-sensitive*, que serão utilizados com frequência neste material.

Código-fonte: Quando inicia-se um documento novo, deve-se escrever comandos de acordo com a sintaxe da linguagem \LaTeX . A sintaxe trabalha com a linguagem no nível lógico. É possível tomar um exemplo na Língua Portuguesa: o modo que posicionamos a pontuação em uma frase pode mudar o sentido da mensagem que queremos passar.

Compilação: No âmbito computacional, é o acontecimento que “executa” uma tarefa ou um conjunto de tarefas. Por exemplo, o ato de pressionar uma tecla no computador é um processo de compilação, pois ele converte a informação “R”, por exemplo, para um código binário “01010010”. A Figura 1.1 apresenta uma ilustração simplificada de um compilador.

Figura 1.1 – O Compilador.



Case-sensitive: é uma palavra com origem na Língua Inglesa, que significa sensível a letras maiúsculas e minúsculas. O \LaTeX diferencia caracteres maiúsculos e minúsculos, assim como em senhas. Os comandos `LARGE`, `Large` e `large`, utilizados para tamanhos de fontes, geram resultados distintos.

¹Do inglês *what you see is what you get*, significa que o texto que digitamos é o texto que vemos.

²Por enquanto entenderemos compilar como executar.

1.3 Utilizando o L^AT_EX com o T_EXmaker

Para utilizar a linguagem L^AT_EX em um computador local é necessário um conjunto de *softwares*. Um deles, considerado o principal, tem a função de interpretar e executar os comandos da linguagem. Esta seção aborda a instalação de todos os programas necessários, bem como uma visão geral da interface do *TeXmaker*.

1.3.1 Instalando os *softwares* necessários

Para a utilização do L^AT_EX é necessária a instalação de um conjunto de *softwares*. A ordem de instalação não deve ser alterada, a fim de que não ocorram problemas de compilação.

1. Leitor de arquivos em formato PDF³: Recomenda-se o *Adobe Acrobat Reader*. Não obstante, existem outros disponíveis como *Foxit PDF Reader* e *Sumatra PDF Reader*. Todos eles são gratuitos. Clique aqui para fazer o *download* do *Acrobat Reader*.
2. *Ghostscript*: É um *software* que interpreta arquivos nos formatos *postscript* e PDF. Clique aqui para fazer o *download* do *Ghostscript*.
3. *MiKTeX*: É uma espécie de tradutor. Por exemplo, para conversar com uma pessoa estrangeira, se não entendemos sua língua, precisamos de um tradutor. O *MiKTeX* transforma a linguagem de programação que digitamos em algo que o L^AT_EX entenda. Clique aqui para fazer o *download* do *MikTeX*.
4. *TeXmaker*: É o ambiente no qual o documento é criado e editado. Clique aqui para fazer o *download* do *TeXmaker*.

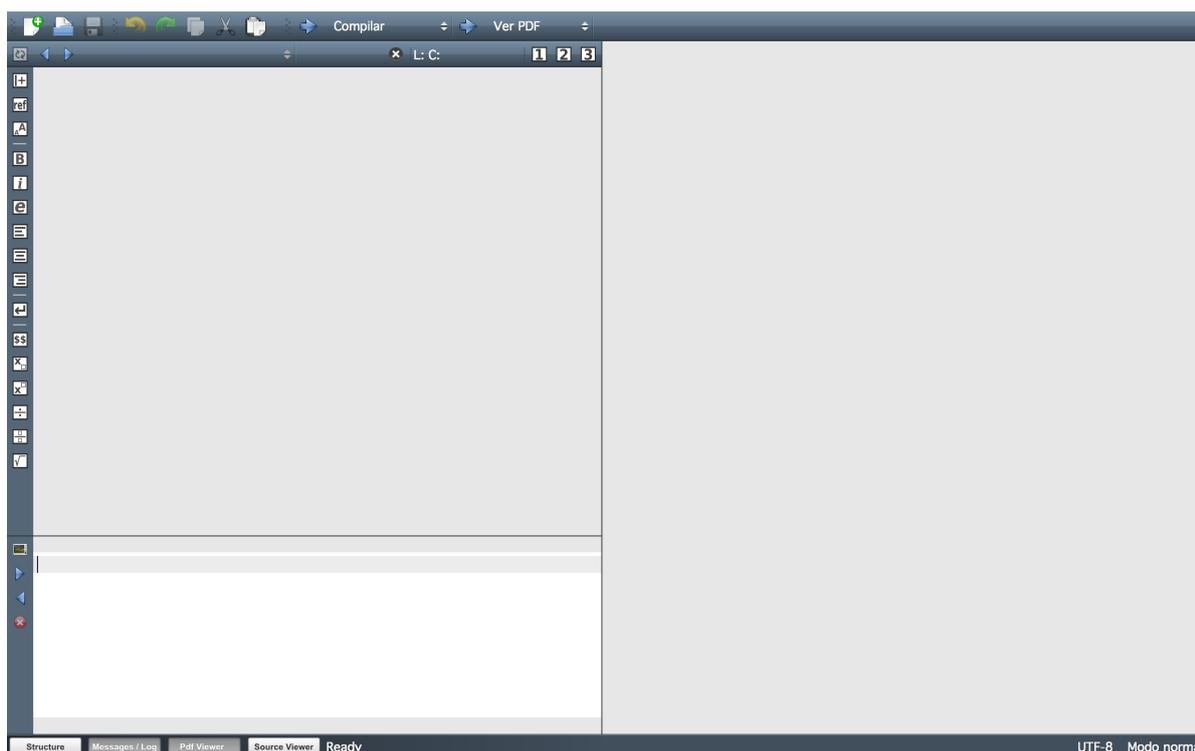
1.3.2 Abrindo o *software* pela primeira vez

A Figura 1.2 mostra a tela inicial do programa *TeXmaker*. As janelas podem ser configuradas de forma personalizada. Os botões no canto inferior esquerdo da tela permitem mostrar ou omitir as janelas existentes no L^AT_EX. São elas:

- *Structure*: Janela de estrutura do documento. Mostra a organização dos capítulos, seções e subseções;
- *Messages / Log*: Console de Mensagens. Mostra o andamento do processo de compilação, erros e logs;
- *PDF Viewer*: Permite a visualização do documento *PDF* ao lado da janela onde digitamos o código-fonte.

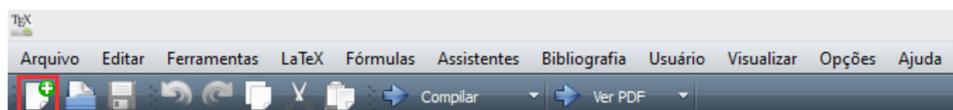
³PDF é extensão para o formato de arquivo *Portable Document Format*.

Figura 1.2 – Tela inicial do *TeXmaker*.



A Figura 1.3 mostra a barra de ferramentas do \LaTeX . O primeiro botão (em destaque) cria um novo documento. O segundo botão permite abrir um arquivo já salvo. O terceiro botão salva o documento atual.

Figura 1.3 – Barra de Ferramentas do *TeXmaker*.



O quadro abaixo apresenta o código-fonte mínimo para gerar um documento utilizando a linguagem \LaTeX .

```
\documentclass[a4paper, 12pt]{report}
\usepackage[utf8]{inputenc}

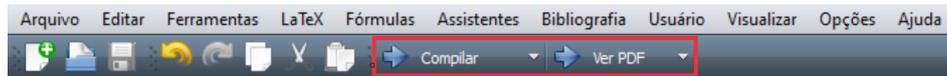
\begin{document}
Estrutura básica para gerar um documento em LaTeX.
\end{document}
```

O ambiente `document`, ou seja, os comandos `\begin{document}` e `\end{document}`, delimitam o conteúdo do documento. Tudo o que não for escrito entre estes comandos será ignorado pelo \LaTeX como conteúdo de texto e, por isso, não será exibido no arquivo PDF.

A Figura 1.4 apresenta os botões `Compilar` e `Ver PDF`. Para compilar um documento, basta

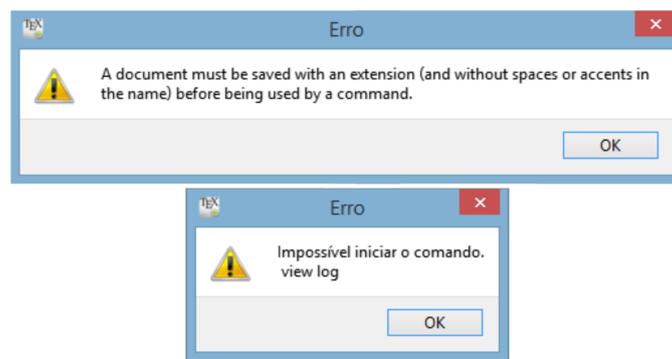
clique na seta azul, ao lado da escrita.

Figura 1.4 – Botões Compilar e Ver PDF.



Toda vez que um documento novo é criado, é necessário salvá-lo para poder executá-lo. Caso isso não seja feito, erros como o da Figura 1.5 poderão aparecer na tela do usuário. Além disso, é aconselhável que o usuário aloque o arquivo TEX dentro de uma pasta, pois o programa cria alguns arquivos auxiliares.

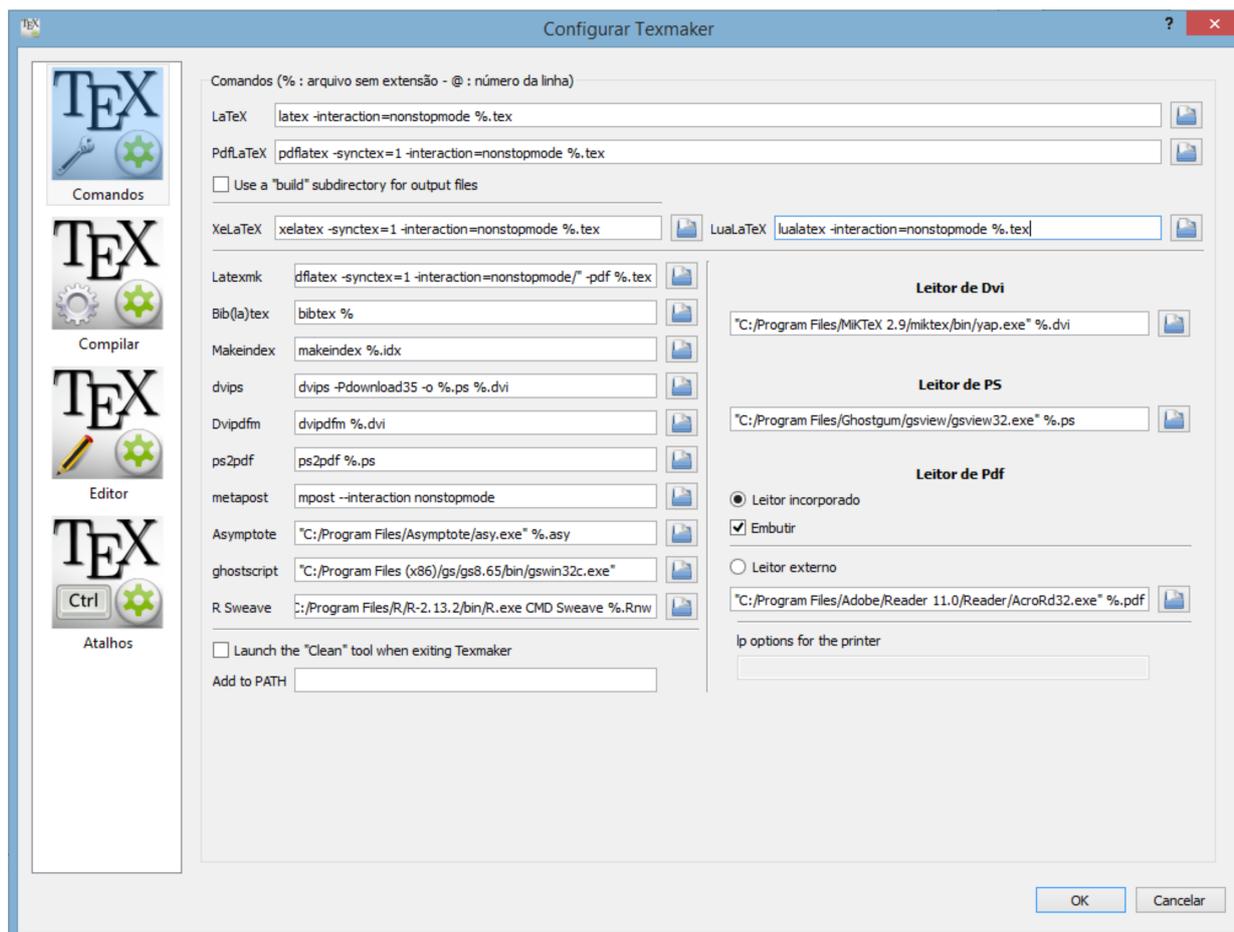
Figura 1.5 – Possíveis erros na execução de um documento.



1.3.3 Configurações Importantes do *TeXmaker*

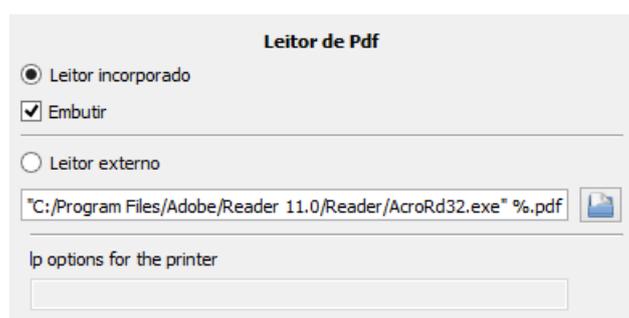
Nesta seção é possível encontrar algumas configurações importantes do *TeXmaker*. Para abrir o menu das configurações, clique em *Opções* → *Configurar o TeXmaker* (ver Figura 1.6). Nesta aba é possível definir o tamanho da fonte do editor $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, adicionar um dicionário e personalizar atalhos. Recomenda-se que o usuário aprenda os atalhos básicos a fim de obter um ganho de produtividade.

Figura 1.6 – Menu Configurações do *TeXmaker*.



A seção *Comandos* exibe configurações de como queremos utilizar o leitor PDF no *TeXmaker*. A Figura 1.7 apresenta as configurações possíveis para esse caso.

Figura 1.7 – Configuração do Leitor PDF.



Quando o item *Leitor Incorporado* estiver marcado, o documento PDF será apresentado pelo leitor do *TeXmaker* em uma janela externa. Ao marcar a opção *Embutir*, o documento será apresentado ao lado do editor (área à direita na Figura 1.2). A opção *Leitor Externo* permite que o documento seja visualizado em um leitor PDF externo (em outra janela que não seja a do *TeXmaker*, como por exemplo o *Adobe Acrobat Reader*).

A seção *Compilar* apresenta Configurações avançadas de compilação. Para o uso básico, este tipo de configuração não será importante.

A seção *Editor* apresenta configurações do editor de texto. As opções são:

- Editor de família da fonte: Muda o tipo de fonte (*Arial*, *Comic Sans...*);
- Editor de tamanho da fonte (10pt, 11pt, 12pt);
- Editor de codificação da fonte (UTF-8 é a codificação utilizada pelo alfabeto da Língua Portuguesa);
- Completar: completa os comandos digitados. Por exemplo: quando digita-se `\use`, o *software* sugere `\usepackage{•}`. Então, ao pressionar a tecla *Enter*, o comando é completado.
- Mostrar a numeração das linhas;
- Dicionário: Ver Seção 1.3.4.
- Comprimento da tabulação: Espaço de recuo quando apertamos a tecla *Tab*. Note que uma boa indentação⁴ ajuda na organização visual .
- Cores: Permite alterar o tema (tema *dark*, tema padrão), cores do texto e de fundo.

A seção *atalhos* permite alterar e adicionar atalhos. As configurações encontradas aqui são muito úteis para comandos utilizados com frequência, por exemplo, a tecla *F1*, que compila o documento.

1.3.4 Dicionário

O *TeXmaker* permite adicionar dicionários de vários idiomas. Para inserir um dicionário da Língua Portuguesa:

1. Baixar o VERO - Verificador Ortográfico do *LibreOffice*. Clique aqui para fazer o *download* do VERO;
2. Em *Opções* → *Configurar o TeXmaker* → *Editor* → *Dicionário*, clicar no ícone da pasta e navegar até encontrar o arquivo;
3. Salvar seu documento e reiniciar o *TeXmaker*.

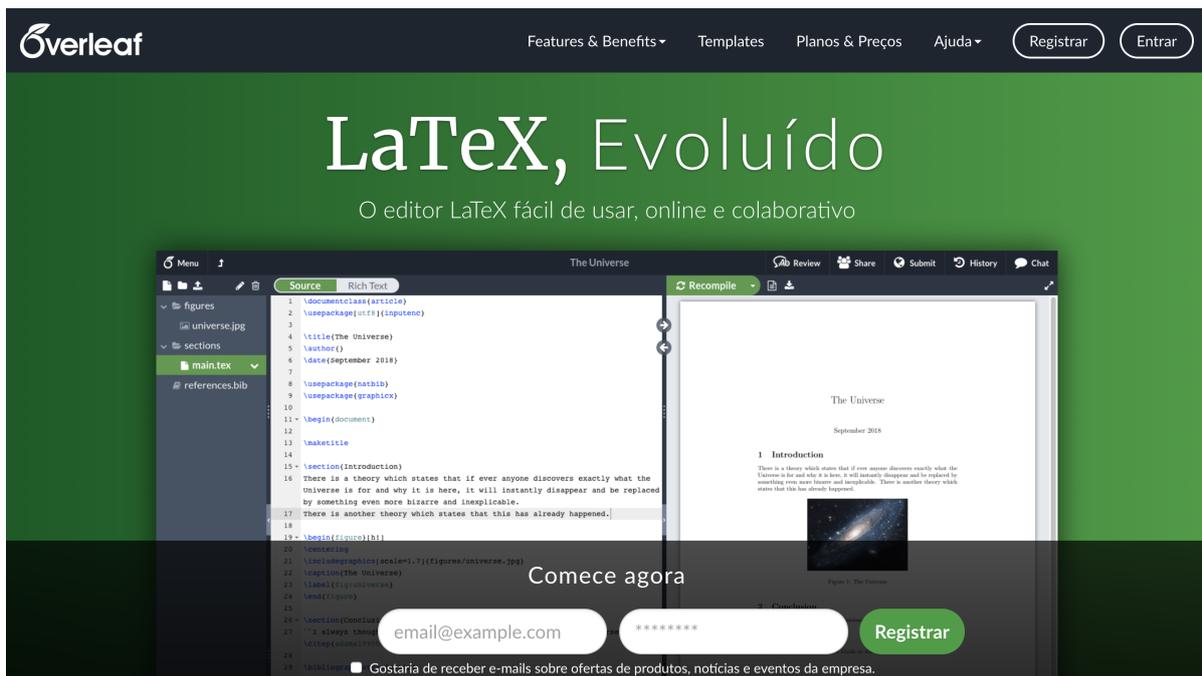
Observação 1. *Note que se a opção Em linha estiver marcada, o software verifica a ortografia em tempo real.*

⁴Entende-se por recuo.

1.4 *Overleaf*, uma alternativa *online*

O *Overleaf* é um editor de linguagem $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ *online*. Ele possui uma versão gratuita e uma versão paga. A versão gratuita possui algumas vantagens em relação ao *T_EXmaker*, como a utilização sem que haja a necessidade de instalar programas e a possibilidade de trabalhar com 1 colaborador em um mesmo documento. Para documentos pequenos e médios, a versão gratuita do *Overleaf* é mais do que suficiente. Para documentos suficientemente grandes, como um livro, recomenda-se a instalação do *T_EXmaker*, visto que o *Overleaf* tem um limite de tempo de 1 minuto para compilar um documento.

Figura 1.8 – Página inicial do *Overleaf*.



A Figura 1.8 apresenta a página inicial do *Overleaf*. Para utilizar esta plataforma é necessário criar uma conta. Para isso, basta clicar em *Registrar* no canto superior direito da tela, informar um e-mail e criar uma senha.

Capítulo 2

Configuração do Preâmbulo

Preâmbulo, segundo o dicionário do *Google*, significa “que caminha na frente”. Dessa forma, o Preâmbulo é composto pela configuração inicial do nosso documento. Por exemplo, se não sabemos como fazer um bolo, vamos atrás de uma receita, reunimos todos os ingredientes necessários em um local e iniciamos o processo de preparação.

No \LaTeX é a mesma coisa: reunimos todas as configurações necessárias em um só local, que é o Preâmbulo, para poder iniciar o processo de criação do documento. Este capítulo apresenta os “ingredientes” do \LaTeX , que são as classes de documentos e os pacotes.

2.1 Classes de Documentos

O primeiro comando do preâmbulo é a classe do documento. Ela define o tipo de documento que está sendo criado.

```
\documentclass[parâmetro opcional]{parâmetro obrigatório}
```

Este código possui dois parâmetros importantes: o envolvido em colchetes [], que é opcional, e o envolvido em chaves {}, que é obrigatório. Considera-se que este é o comando mais importante do \LaTeX , pois é ele que identifica a criação do documento. Dentre os parâmetros opcionais estão:

- Tamanho da fonte: `10pt`, `11pt`, `12pt`;
- Tipo de papel: `a5paper`, `a4paper`, `letter`;
- *Layout* da página: `oneside` (imprime em um lado da página), `twoside` (imprime nos lados opostos das páginas, como em um livro);
- Orientação: `landscape` (paisagem), `portrait` (retrato).

O parâmetro obrigatório da classe do documento tem uma característica única: ele define o tipo do documento que é criado. Dentre os parâmetros obrigatórios podemos escolher:

- `article`: artigo;
- `report`: relatório;
- `book`: livro;
- `beamer`: apresentação;
- `letter`: carta.

2.2 Pacotes

O comando `usepackage` implementa o uso de pacotes. Os pacotes possibilitam o uso de imagens, equações matemáticas, definições de margens, texto colorido, entre outros. O \LaTeX , por padrão, escreve o texto em cor preta. Para escrevermos em vermelho, por exemplo, devemos utilizar o pacote `xcolor`. A sintaxe deste comando é a apresentada no quadro abaixo.

```
\usepackage[opções]{pacote a ser utilizado}
```

O parâmetro `opções` mantém as configurações específicas do pacote. O segundo parâmetro define qual pacote será utilizado. A lista a seguir apresenta alguns pacotes importantes.

- `amsmath`, `amssymb`, `amsfonts`, `amsthm`, `exscale`: habilitam a utilização de fontes e símbolos matemáticos, enumeração automática de teoremas, entre outros.
- `array`: habilita a criação de matrizes e determinantes;
- `caption`: habilita a inserção de legendas em *floats* (objetos flutuantes), como tabelas e imagens;
- `color`: habilita a utilização de textos coloridos;
- `fancyhdr`: habilita a inserção e configuração de cabeçalho e rodapé;
- `float`: habilita um pacote que aprimora o posicionamento de *floats*;
- `framed`: habilita a criação de quadros utilizados para destaque de conteúdos;
- `geometry`: habilita a configuração das margens;
- `glossaries`: habilita a criação de glossários;
- `graphicx`: habilita a inserção de imagens;
- `hyperref`: habilita a inserção *hyperlinks*;

- `indentfirst`: habilita a tabulação automática de todos os parágrafos;
- `inputenc`: habilita a inserção de caracteres especiais (por exemplo, ç, à, á, é e ã);
- `lineno`: habilita a enumeração das linhas;
- `mdframed`: habilita a criação de quadros coloridos para destaque de conteúdo;
- `multicol`: habilita a implementação de múltiplas colunas em certas partes do documento;
- `setspace`: habilita comandos que permitem modificar o espaçamento entre as linhas;
- `wrapfig`: habilita a inserção de figuras ao lado de texto;
- `tikz`: habilita um pacote gráfico que permite criar desenhos.

2.3 O ambiente document

Após o preâmbulo utiliza-se o ambiente `document`. Insere-se dentro deste ambiente todo o conteúdo do documento, como textos, quadros, figuras e tabelas. A sintaxe dele é apresentada a seguir.

```
\begin{document}
<conteúdo do documento>
\end{document}
```

Na linguagem \LaTeX , utilizam-se dois elementos durante a criação do documento: ambientes e comandos. Os ambientes têm a estrutura apresentada no quadro anterior, iniciando-se com `begin` e finalizando-o `end`, juntamente com o nome do ambiente desejado. Já os comandos não possuem `begin` e `end` (por exemplo `usepackage`).

Capítulo 3

Formatação do Texto

Neste capítulo serão apresentados tópicos relacionados à formatação do texto. Para tal, é necessário saber que existem caracteres especiais, chamados de caracteres reservados. Dentre esses tópicos estão: organização hierárquica do documento, criação de lista de apêndices, documento por módulos, inserção de *hyperlinks*, notas de rodapé, referências e citações.

3.1 Caracteres Reservados

Antes de começar, devemos ter em mente que o \LaTeX possui alguns caracteres reservados. Esses caracteres possuem um uso especial dentro da linguagem \LaTeX . Ao utilizar algum desses caracteres, pode ocorrer um erro na hora de compilar, ou o *software* não irá reproduzi-lo na tela. São eles:

```
# ~ & _ { } ~ \ % $
```

A Tabela 3.1 apresenta os comandos necessários para utilizar os caracteres especiais como texto em um documento.

Tabela 3.1 – Caracteres especiais.

Código-fonte	Resultado
$\#$	#
$\$$	\$
$\%$	%
$\sim\{\}$	^
$\&$	&
$_$	–
$\{ \}$	{ }
\sim	~
\backslash	\

3.2 Unidades de Medida

No \LaTeX é possível utilizar várias unidades de medida. Essas unidades são muito utilizadas nas opções de alinhamento de texto com os comandos `hspace` e `vspace`, por exemplo. As unidades mais utilizadas encontram-se na Tabela 3.2.

Tabela 3.2 – Caracteres especiais.

Abreviação	Definição
<code>pt</code>	pontos (padrão)
<code>mm</code>	milímetro
<code>cm</code>	centímetro
<code>in</code>	polegada

3.3 Trabalhando em Capítulos e Seções

Para manter um documento ordenado em capítulos, seções e subseções, utilizam-se os comandos `chapter`, `section` e `subsection`. É importante ressaltar que o comando `chapter` pode ser utilizado apenas nas classes de documento `book` e `report`.

Exemplo 1. *Criando um documento estruturado.*

```
\chapter{Nome do capítulo}  
\section{Seção}  
\subsection{Subseção}
```

Capítulo 1

Nome do capítulo

1.1 Seção

1.1.1 Subseção

3.4 Utilização das Aspas

Para implementar o sinal gráfico de aspas (simples e duplas), utilizam-se os sinais de crase para abrir e aspas para fechar.

Exemplo 2. *Utilização de aspas simples e aspas duplas.*

Eu ``amo'' matemática. \\	Eu “amo” matemática.
Eu `amo' matemática.	Eu ‘amo’ matemática.

3.5 Alinhamento do Texto

Esta seção apresenta os comandos para alinhamento do texto à esquerda, à direita e ao centro. Além disso, são apresentados os comandos de espaçamento vertical e horizontal.

3.5.1 Os comandos `flushleft`, `flushright` e `center`

Para alinhar o texto horizontalmente utilizam-se três ambientes. O ambiente `flushleft` alinha o texto à esquerda, o `flushright` alinha à direita e o `center` alinha ao centro.

Exemplo 3. *Utilizando os ambientes de alinhamento horizontal do texto.*

```
\begin{flushright}
{\bf Alinhamento pela direita}: Com \texttt{flushright}
alinhamos pela direita. \\ Esta é a segunda linha, alinhada à direita.
\end{flushright}
\begin{center}
{\bf Alinhamento pelo centro}: Com \texttt{center}
alinhamos no centro. \\ Esta é a segunda linha, centralizada.
\end{center}
\begin{flushleft}
{\bf Alinhamento pela esquerda}: Com \texttt{flushleft}
alinhamos pela esquerda. \\ Esta é a segunda linha, alinhada à esquerda.
\end{flushleft}
```

Alinhamento pela direita: Com `flushright` alinhamos pela direita.
Esta é a segunda linha, alinhada à direita.

Alinhamento pelo centro: Com `center` alinhamos no centro.
Esta é a segunda linha, centralizada.

Alinhamento pela esquerda: Com `flushleft` alinhamos pela esquerda.

Esta é a segunda linha, alinhada à esquerda.

3.5.2 Os comandos `hspace` e `vspace`

Os comandos `hspace` e `vspace` permitem ao usuário alinhar o texto manualmente. Enquanto o comando `hspace` refere-se ao alinhamento horizontal, o `vspace` refere-se ao alinhamento vertical. Assim, utiliza-se uma medida junto ao comando. Essa medida pode ser expressa com as unidades apresentadas na Tabela 3.2.

Exemplo 4. *Utilização dos comandos `hspace` e `vspace`. O resultado foi omitido.*

```
\hspace{2cm}  
\vspace{-1cm}
```

3.6 Tabulação

Para ajustar automaticamente a tabulação de todos os parágrafos do documento utiliza-se o pacote `indentfirst`.

```
\usepackage{indentfirst}
```

Caso seja necessário que um dos parágrafos não receba a tabulação, utiliza-se o comando `noindent` logo no início da linha desejada.

```
\noindent
```

Caso seja necessário alterar o valor da tabulação do parágrafo, utiliza-se o pacote `parskip`. Além disso, utiliza-se o comando `setlength`. O Exemplo 5 apresenta uma configuração de 15 pontos de tabulação.

Exemplo 5. *Parágrafo configurado com 15 pontos de tabulação.*

```
\setlength{\parindent}{15pt}
```

3.7 Quebras de Linha

Existem quatro formas possíveis para aplicar uma quebra de linha no documento. Digitando `\\` ou `\newline`, o texto é “jogado” para a próxima linha, sem deixar a tabulação do parágrafo. Ao digitar `\par` ou deixando uma linha completamente em branco no código-fonte, o \LaTeX vai pular para uma nova linha, criando um parágrafo.

A aplicação de uma quebra de linha manual (`\\` ou `\newline`) de forma incorreta pode gerar o seguinte um erro:

```
! LaTeX Error: There's no line here to end.
```

3.8 Quebras de Página

Além da quebra de linha, é possível aplicar uma quebra de página. Para isso, utiliza-se o comando `newpage`.

```
\newpage
```

3.9 Espaçamento entre Linhas

Para definir o espaçamento entre linhas de todo o documento, utiliza-se o pacote `setspace`.

```
\usepackage{setspace}
```

Este pacote habilita um conjunto de comandos que permitem definir o espaçamento. Estes comandos podem ser inseridos no preâmbulo e são apresentados a seguir.

- `\singlespacing`: espaçamento simples;
- `\onehalfspacing`: espaçamento de um e meio;
- `\doublespacing`: espaçamento duplo;
- `\setstretch{xx}`: define um espaçamento personalizado. O parâmetro `xx` é expresso em linhas.

Caso seja necessário modificar o espaçamento de apenas uma parte do documento, é possível utilizar os ambientes `singlespace`, `onehalfspace` e `doublespace`.

Exemplo 6. *Utilizando o ambiente `doublespace`.*

```
\begin{doublespace}
Suspendisse pretium metus eu metus venenatis varius a at mi. Vestibulum
↪ malesuada pulvinar est, vitae molestie nisi sagittis sit amet. Integer
↪ luctus in sem in pharetra.
\end{doublespace}
```

Suspendisse pretium metus eu metus venenatis varius a at mi. Vestibulum malesuada pulvinar est, vitae molestie nisi sagittis sit amet. Integer luctus in sem in pharetra.

Exemplo 7. Utilizando o comando `setstretch` para definir um espaçamento de 0.5 entre as linhas.

```
\setstretch{0.5}
```

Suspendisse pretium metus eu metus venenatis varius a at mi. Vestibulum
→ malesuada pulvinar est, vitae molestie nisi sagittis sit amet. Integer
→ luctus in sem in pharetra.

Suspendisse pretium metus eu metus venenatis varius a at mi. Vestibulum malesuada pulvinar est, vitae molestie nisi sagittis sit amet. Integer luctus in sem in pharetra.

3.10 Estilos de Fonte

A Tabela 3.3 apresenta os comandos para aplicar os estilos negrito, itálico e sublinhado no texto do documento. Além disso, são apresentados os comandos `texttt` e `textsc` que, respectivamente, aplicam o estilo de máquina de escrever e versalete.

Tabela 3.3 – Estilos de fonte.

Estilo	Comando	Resultado
Negrito	<code>\textbf{texto}</code>	texto
Itálico	<code>\textit{texto}</code> ou <code>\emph{texto}</code>	<i>texto</i>
Sublinhado	<code>\underline{texto}</code>	<u>texto</u>
Máquina de escrever	<code>\texttt{texto}</code>	texto
Versalete	<code>\textsc{texto}</code>	TEXTO

3.11 Cores

O \LaTeX permite a inserção de textos coloridos e textos em destaque. Para isso, é necessário utilizar o pacote `xcolor`.

```
\usepackage{xcolor}
```

Este pacote habilita um conjunto de cores pré-definidas (Tabela 3.4). Também é possível criar uma cor.

Tabela 3.4 – Cores.

Parâmetro	Cor
black	preto
blue	azul
brown	marrom
cyan	ciano
darkgray	cinza escuro
gray	cinza
green	verde
lightgray	cinza claro
lime	lima
magenta	magenta
olive	verde-oliva
orange	laranja
pink	rosa
purple	roxo
red	vermelho
teal	verde-azulado
violet	violeta
white	branco
yellow	amarelo

Para aplicar uma cor, utiliza-se o comando `textcolor` juntamente com dois parâmetros.

```
\textcolor{<cor>}{<texto>}
```

Exemplo 8. *Utilização do comando `textcolor` para alterar a cor do texto.*

```
\textcolor{blue}{Este}  
\textcolor{green}{é}  
\textcolor{yellow}{um}  
\textcolor{cyan}{texto}  
\textcolor{red}{colorido.}
```

Este é um texto colorido.

3.11.1 Texto em Destaque

Para destacar um texto, ou seja, inserir um fundo colorido (como na utilização de uma caneta marca-texto), utiliza-se o comando `colorbox`.

```
\colorbox{<cor>}{<texto>}
```

Exemplo 9. *Utilização do comando `colorbox`.*

```
\colorbox{lime}{Texto em fundo verde claro.}
```

Texto em fundo verde claro.

3.11.2 Personalizando Cores

O comando `definecolor` habilita a criação de cores no \LaTeX . Esse comando é utilizado no preâmbulo do documento.

```
\definecolor{<nome>}{<modelo>}{<especificações>}
```

O parâmetro `nome` é o nome da cor, definido pelo usuário. O `modelo` define o modelo de cor a ser utilizado. E, por último, tem-se o parâmetro `especificações`, que depende do modelo de cor escolhido. A Tabela 3.5 apresenta os modelos de cor mais comuns, junto com a estrutura de especificação necessária e um exemplo.

Tabela 3.5 – Modelos e especificações de cores.

Modelo	Especificação	Exemplo
rgb	valores reais do intervalo $[0, 1]$	$\{0, 0.45, 0.34\}$
RGB	valores inteiros do intervalo $[0, 255]$	$\{0, 32, 159\}$
HTML	código hexadecimal de 6 dígitos	$\{AF359A\}$
cmyk	valores reais do intervalo $[0, 1]$	$\{0.66, 0.33, 0, 0.24\}$
wave	valores reais do intervalo $[363, 814]$	$\{485\}$
gray	valores reais do intervalo $[0, 1]$	$\{0.82\}$

Exemplo 10. *Criando e utilizando uma cor no modelo RGB.*

```
\definecolor{bordo}{RGB}{161,0,0} %preâmbulo  
\textcolor{bordo}{Texto utilizando a cor criada no modelo \texttt{RGB}.}
```

Texto utilizando a cor criada no modelo RGB.

Exemplo 11. *Criando e utilizando uma cor no modelo HTML.*

```
\definecolor{verdeEscuro}{HTML}{0F8B41} %preâmbulo
\textcolor{verdeEscuro}{Texto utilizando a cor criada no modelo \texttt{HTML}.}
```

Texto utilizando a cor criada no modelo HTML.

Exemplo 12. *Criando e utilizando uma cor no modelo rgb.*

```
\definecolor{azul}{rgb}{0.06, 0.35, 0.81} %preâmbulo
\textcolor{azul}{Texto utilizando a cor criada no modelo \texttt{rgb}.}
```

Texto utilizando a cor criada no modelo rgb.

Exemplo 13. *Criando e utilizando uma cor no modelo gray.*

```
\definecolor{cinza}{gray}{0.4} %preâmbulo
\textcolor{cinza}{Texto utilizando a cor criada no modelo \texttt{gray}.}
```

Texto utilizando a cor criada no modelo gray.

3.12 Listas

A criação de listas é feita a partir de ambientes. Existem três tipos distintos de listas: enumerada, com marcador e descritiva.

```
\begin{<tipo da lista>}
  \item
\end{<tipo da lista>}
```

Os exemplos a seguir apresentam os tipos de lista.

Exemplo 14. *Lista enumerada com três itens.*

```
\begin{enumerate}
  \item Primeiro item;
  \item Segundo item;
  \item Terceiro item.
```

```
\end{enumerate}
```

1. Primeiro item;
2. Segundo item;
3. Terceiro item.

Exemplo 15. *Lista com marcadores com três itens.*

```
\begin{itemize}
  \item Primeiro item;
  \item Segundo item;
  \item Terceiro item.
\end{itemize}
```

- Primeiro item;
- Segundo item;
- Terceiro item.

Exemplo 16. *Lista descritiva com três itens.*

```
\begin{description}
  \item [I] Primeiro item;
  \item [II] Segundo item;
  \item [III] Terceiro item.
\end{description}
```

- I Primeiro item;
- II Segundo item;
- III Terceiro item.

3.12.1 Listas aninhadas

É possível combinar as listas de forma que seja possível criar listas aninhadas.

Exemplo 17. *Criando uma lista aninhada com o ambiente enumerate.*

```
\begin{enumerate}
  \item Item número um.
  \item Item número dois.
  \begin{enumerate}
    \item Subitem número um.
    \item Subitem número dois.
  \end{enumerate}
\end{enumerate}
```

-
1. Item número um.
 2. Item número dois.
 - (a) Subitem número um.
 - (b) Subitem número dois.

Exemplo 18. *Criando uma lista aninhada com o ambiente description.*

```
\begin{description}
  \item [I] Item número um;
  \begin{description}
    \item [a] Subitem número um;
    \item [b] Subitem número dois;
    \item [c] Subitem número três;
  \end{description}
  \item [II] Item número dois.
\end{description}
```

-
- I** Item número um;
- a** Subitem número um;
 - b** Subitem número dois;
 - c** Subitem número três;
- II** Item número dois.

3.13 Ambiente Verbatim

O ambiente `verbatim` escreve com fonte de máquina de escrever. Caso seja necessário dar destaque em um texto, é possível utilizar este ambiente. Além disso, ele também é frequentemente utilizado para digitar código-fonte dentro do \LaTeX . O Exemplo 19 apresenta um código-fonte redigido na linguagem de programação *Python*, e o Exemplo 20 apresenta outro código-fonte em linguagem *C*.

Exemplo 19. *Código-fonte na linguagem Python.*

```
\begin{verbatim}
print('Digite dois números')
n1 = float(input())
n2 = float(input())
soma = n1 + n2
print(soma)
\end{verbatim}
```

```
print('Digite dois números')
n1 = float(input())
n2 = float(input())
soma = n1 + n2
print(soma)
```

Exemplo 20. *Código-fonte na linguagem C.*

```
\begin{verbatim}
#include <stdio.h>
#include <stdlib.h>
int main() {
    int idade; printf("Qual a sua idade? ");
    scanf("%d", &idade);
    if(idade >= 18 && idade <=30)
        printf("Você está ficando velho!");
    else if(idade <= 18)
        printf("Você ainda é novo!");
    else
        printf("Você está velho!");
    system("pause");
}
\end{verbatim}
```

O Exemplo 21 apresenta um código-fonte digitado na linguagem \LaTeX .

Exemplo 21. *Código-fonte em \LaTeX .*

```
\begin{verbatim}
\textbf{Este é um texto em negrito}.
\textit{Este é um texto em itálico}.
\begin{theorem}
  Este é um teorema importante.
\end{theorem}
\end{verbatim}
```

3.14 Ambiente Minipage

O ambiente `minipage` é muito utilizado para posicionar tabelas ou figuras lado a lado. Este ambiente cria pequenas páginas dentro da própria página.

```
\begin{minipage}[alinhamento]{largura}
  <conteúdo>
\end{minipage}
```

O parâmetro `alinhamento` permite alinhar o conteúdo da página ao centro (`c`), no topo (`t`) e na parte inferior (`b`). Já o parâmetro `largura` permite definir a largura da `minipage`. Além disso, é possível utilizar o comando `hspace` para alterar o espaçamento horizontal entre as `minipages`.

Exemplo 22. *Utilização de duas `minipages` lado a lado, e o comando `hspace` para adicionar um espaçamento entre elas.*

```
\begin{minipage}[t]{7cm}
Conteúdo da \texttt{minipage} da esquerda alinhado no topo. Conteúdo da
→ \texttt{minipage} da esquerda alinhado no topo. Conteúdo da
→ \texttt{minipage} da esquerda alinhado no topo.
\end{minipage}\hspace{1cm}
\begin{minipage}[b]{7cm}
Conteúdo da \texttt{minipage} da direita alinhado na parte inferior. Conteúdo
→ da \texttt{minipage} da direita alinhado na parte inferior. Conteúdo da
→ \texttt{minipage} da direita alinhado na parte inferior.
\end{minipage}
```

Conteúdo da `minipage` da esquerda alinhado no topo. Conteúdo da `minipage` da esquerda alinhado no topo. Conteúdo da `minipage` da esquerda alinhado no topo.

Conteúdo da `minipage` da direita alinhado na parte inferior. Conteúdo da `minipage` da direita alinhado na parte inferior. Conteúdo da `minipage` da direita alinhado na parte inferior.

Vale ressaltar que as `minipages` aparecem lado a lado dado suas dimensões. Caso a soma das duas larguras seja maior do que a largura das margens, a `minipage` que fica à direita é deslocada para baixo. Além disso, caso exista uma linha em branco entre o final da primeira e o início da segunda, a `minipage` também é deslocada para baixo.

3.15 Ambiente `framed`

O ambiente `framed` cria uma caixa de destaque simples ao redor de um texto, com largura igual à da página. Para utilizá-la, é necessário inserir o pacote `framed` no preâmbulo.

```
\usepackage{framed}
```

Este pacote habilita o ambiente `framed`.

Exemplo 23. *Caixa simples criada com o ambiente `framed`.*

```
\begin{framed}
Caixa produzida com o ambiente \texttt{framed}.
\end{framed}
```

Caixa produzida com o ambiente `framed`.

3.16 Criação de Títulos

Além dos títulos criados com os comandos `chapter`, `section` e `subsection`, é possível criar títulos de tamanhos variados com alguns comandos. Os títulos criados com esses comandos não aparecem no sumário. A Tabela 3.6 apresenta os comandos que alteram o tamanho de um

texto. Vale ressaltar que os valores da segunda coluna desta tabela são para um documento que utiliza a fonte 11pt.

Tabela 3.6 – Títulos personalizados.

Comando	Tamanho da fonte	Resultado
Huge	24.88pt	Texto.
huge	20.74pt	Texto.
LARGE	17.28pt	Texto.
Large	14.40pt	Texto.
large	12pt	Texto.
normalsize	10.95pt	Texto.
small	10pt	Texto.
footnotesize	9pt	Texto.
scriptsize	8pt	Texto.
tiny	6pt	Texto.

3.16.1 Título com maketitle

Para criar um título rápido para um documento simples, utiliza-se o comando `maketitle`. Este comando vale para classes de documento como `article`, `report` e `book`.

Exemplo 24. *Utilização do comando `maketitle`.*



Antes de compilar um documento com este comando, é necessário declarar algumas informações no preâmbulo do documento, como nome do autor, data e título.

```
\author{}  
\date{}  
\title{}
```

Capítulo 4

Estrutura do Documento

Este capítulo apresenta informações que possibilitam a definição do *layout* da página, criação de lista de referências, apêndices, citações, sumário, nota de rodapé e *hyperlinks*. Além disso, ao final do capítulo, é possível descobrir como separar o documento em módulos.

4.1 Layout da Página

A formatação geral do documento se dá pelo comando `documentclass`, inserido no preâmbulo. No entanto, tal comando não permite a definição das margens, por exemplo. Para isso, utiliza-se o pacote `geometry`, um pacote descomplicado e versátil.

Exemplo 25. *Definindo as margens do documento conforme as normas da ABNT.*

```
\usepackage[top=3cm,left=3cm,right=2cm,bottom=2cm]{geometry}
```

No Exemplo 25 é possível ver que o pacote `geometry` possui 4 parâmetros opcionais. São estes parâmetros que definem as margens:

- *top* é a margem superior;
- *right* é a margem direita;
- *left* é a margem esquerda;
- *bottom* é a margem inferior.

4.2 Sumário

No \LaTeX , é possível criar o sumário com apenas um comando, inserido logo após o início do documento.

```
\tableofcontents
```

4.3 Notas de Rodapé

Para inserir uma nota de rodapé utiliza-se o comando `footnote` ao lado da palavra a qual deseja-se referenciar¹.

```
\footnote{<texto da nota>}
```

Exemplo 26. *Inserindo o comando de nota de rodapé ao final do parágrafo anterior.*

```
\footnote{Exatamente assim}.
```

4.4 Referências e Citações

Em textos científicos a utilização de citações e lista de referência é indispensável. Para criar uma lista de referências simples, utiliza-se o ambiente `thebibliography`. Outra opção é a utilização do pacote `abntex2cite`. Nesse caso, a lista de referências é criada em um arquivo externo com extensão `.BIB`. No entanto, este material se limitará à criação de lista de referências com o ambiente `thebibliography`.

4.4.1 O ambiente `thebibliography`

O comando `bibitem` é inserido dentro do ambiente `thebibliography`. Sua função é nomear a referência para que depois seja possível criar a citação.

Exemplo 27. *Referência de um artigo científico.*

```
\begin{thebibliography}{3}
\bibitem{dacruz}DA CRUZ, A. B. Comparação entre modelos de previsão de
demanda: estudo de caso de um restaurante de comida japonesa. \textbf{REMAT:
Revista Eletrônica da Matemática}, v. 2, n. 2, p. 180-197, 2016.
\end{thebibliography}
```

Referências Bibliográficas

- [1] DA CRUZ, A. B. Comparação entre modelos de previsão de demanda: estudo de caso de um restaurante de comida japonesa. **REMAT: Revista Eletrônica da Matemática**, v. 2, n. 2, p. 180-197, 2016.

¹Exatamente assim.

4.4.2 Citações

Para fazer a citação do autor da seção anterior, utiliza-se o comando `\cite{dacruz}`. Assim, quando o documento for compilado, será gerada uma referência numérica, da forma [1].

Além da chamada por números [1], existem outros estilos que podem ser usados, como a chamada por autor (AUTOR, ano). Um maior detalhamento sobre referências e citações pode ser encontrado no volume 2 da coleção (L^AT_EX: Noções Intermediárias).

Para citações longas, utiliza-se o pacote `quoting`.

Exemplo 28. *Citação longa utilizando o pacote `quoting`.*

```
\begin{quoting}[rightmargin=0cm,leftmargin=4cm]
\begin{singlespace}
{\small \noindent
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
→ incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis
→ nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
→ consequat.}
\end{singlespace}
\end{quoting}
```

>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

4.4.3 Referência Cruzada

Para fazer referências de capítulos, seções, imagens, equações, entre outros, utilizam-se dois comandos.

```
\label{nome}
\ref{nome}
```

O comando `label` é utilizado para nomear o elemento a ser referenciado. O comando `ref` é utilizado no local no qual a referência aparecerá. O parâmetro `nome` é o nome de chamada dado à referência.

Como no L^AT_EX podemos referenciar quase tudo, recomenda-se identificar o objeto a ser referenciado com um prefixo seguido de dois pontos. A Tabela 4.1 apresenta um conjunto de sugestões de prefixos.

Tabela 4.1 – Identificação de referências cruzadas.

Identificação	Item
ch:	capítulo
sec:	seção
subsec:	seção
fig:	figura
tab:	tabela
eq:	equação

4.5 Inserindo hyperlinks

Para inserir *hyperlinks* utiliza-se o pacote `hyperref`.

```
\usepackage{hyperref}
```

Ao habilitar este pacote, todas as referências internas se tornarão *hyperlinks*. Ou seja, quando um item do sumário ou um elemento criado por meio do comando `ref` for clicado, o usuário será redirecionado ao conteúdo indicado.

É possível criar uma referência interna personalizada, isto é, um texto que redireciona o usuário para outra parte do documento. Para isso, utiliza-se o comando `hyperref`.

```
\hyperref[descrição do label]{descrição desejada}
```

Exemplo 29. Referência interna personalizada que redireciona o usuário para o Capítulo 1.

```
\hyperref[capitulo1]{Clique aqui} para ser direcionado ao Capítulo 1.
```

Clique aqui para ser direcionado ao Capítulo 1.

É importante destacar que, para utilizar este comando, é necessário que já esteja pré-definido um comando *label* identificando o objeto a ser referenciado.

4.5.1 Páginas da web

Para referenciar *links* da internet (*links* externos) utiliza-se o comando `href` com dois parâmetros.

```
\href{<endereço do site>}{<texto>}
```

O primeiro parâmetro refere-se ao *link* e o segundo parâmetro ao texto que aparecerá no documento. Assim, quando o usuário clicar sobre o texto, ele será redirecionado ao site referenciado.

Exemplo 30. *Referência externa que redireciona o usuário ao site do Instituto Federal do Rio Grande do Sul Campus Caxias do Sul.*

```
\href{https://ifrs.edu.br/caxias/}{Site IFRS Caxias}
```

Site IFRS Caxias

4.6 Documento por Módulos

O \LaTeX possui um comando que permite criar um documento por meio de módulos. Esse comando torna-se útil quando o documento criado possui muitas linhas de comando e, por consequência, demorando para compilar. Um documento por módulos funciona da seguinte forma: cria-se um arquivo principal MAIN.TEX com toda a informação do preâmbulo do documento. Em seguida, criam-se arquivos secundários contendo apenas o conteúdo do documento. Por fim, esses arquivos secundários, também com a extensão TEX, são incluídos no arquivo principal MAIN.TEX com o comando `input`, dentro do ambiente `document`. Outra forma de incluir os arquivos é utilizar o comando `include`.

```
\input{<nome do arquivo.tex>}
```

```
\include{<nome do arquivo>}
```

Vale salientar que ao utilizar o comando `input` se torna necessário indicar a extensão do arquivo. Já o `include` dispensa sua utilização.

Capítulo 5

Inserindo Imagens

É possível inserir imagens geradas por outros programas no \LaTeX . Para isso, deve-se usar o pacote `graphicx` no preâmbulo do documento.

```
\usepackage{graphicx}
```

É preciso que as imagens estejam salvas na mesma pasta que o arquivo `TEX`. Para criar um arquivo `PDF` por meio do \LaTeX as figuras devem estar nos formatos `JPG`¹, `PNG`², `EPS`³ ou `PDF`.

O comando para a inserção de imagens é o `includegraphics`, acompanhado de dois parâmetros.

```
\includegraphics[scale=...]{<nome do arquivo>}
```

O parâmetro opcional `scale` permite alterar a escala da imagem original, podendo admitir valores sem unidade. O parâmetro `nome do arquivo` refere-se ao título da figura utilizada.

5.1 Alinhando figuras

É possível alinhar figuras no centro, à esquerda e à direita da página. Para centralizar, utiliza-se o ambiente `center` ou o comando `centering`.

```
\begin{center}  
\includegraphics[scale=...]{<nome do arquivo>}  
\end{center}
```

¹JPG é extensão para o formato de arquivo *Joint Photographic Experts Group*.

²PNG é extensão para o formato de arquivo *Portable Network Graphics*.

³EPS é extensão para o formato de arquivo *Encapsulated PostScript*.

```
\centering{\includegraphics[scale=...]{<nome do arquivo>}}
```

O alinhamento à esquerda é automático. No entanto, caso seja necessário fazer este alinhamento de forma manual, utiliza-se o ambiente `flushleft` ou o comando `flushleft`.

```
\begin{flushleft}  
\includegraphics[scale=...]{<nome do arquivo>}  
\end{flushleft}
```

Para alinhar um conteúdo à direita, utiliza-se o ambiente `flushright` ou o comando `flushright`.

```
\begin{flushright}  
\includegraphics[scale=...]{<nome do arquivo>}  
\end{flushright}
```

Exemplo 31. Figuras alinhadas ao centro (Figura 1), à esquerda (Figura 2) e à direita (Figura 3).

<código-fonte omitido>

Figura 1: Figura centralizada



Figura 2: Figura alinhada à esquerda



Figura 3: Figura alinhada à direita



A próxima seção abordará o ambiente `figure`, que permite enumerar, inserir legendas e referenciar figuras, conforme o Exemplo 31.

5.2 Ambiente Figure

O ambiente `figure` oportuniza inserir legendas, referenciá-las e definir o local de inserção da imagem.

```
\begin{figure}[posição] \centering
\includegraphics[scale=...]{<nome do arquivo>}
\caption{<legenda da figura>}
\end{figure}
```

O parâmetro `posição` admite os valores `h` (*here* - aqui), `t` (topo da página), `b` (final da página), `p` (página separada) e `!` (escolha do \LaTeX). No entanto, a utilização desses valores pode, muitas vezes, causar confusão no documento, de modo que o \LaTeX insere a imagem onde bem entender.

Assim, o pacote `float` aprimora o posicionamento de elementos *floats*, isto é, imagens e tabelas. Ao contrário dos valores de posição apresentados anteriormente, este pacote habilita o valor `H` (letra maiúscula). Assim, ao utilizar `H` como parâmetro de posição, o `float` se posicionará no local desejado.

```
\usepackage{float}
```

5.3 Inserindo figuras lado a lado

Para inserir figuras lado a lado, é possível usar o pacote `subfig`. Além disso, este pacote habilita a inserção de legendas individuais para cada figura.

Exemplo 32. *Utilização básica do pacote `subfig`.*

```
\begin{figure}[h]
\centering
\subfloat[Figura 1]
{\includegraphics[scale=0.1]{ifvertical.jpg}}
\quad
\subfloat[Figura 2]
{\includegraphics[scale=0.1]{ifvertical.jpg}}
\caption{Figuras lado a lado, com legenda individual}
\end{figure}
```



Figura 4: Figuras lado a lado, com legenda individual

Caso seja necessário inserir apenas uma legenda para as duas figuras, basta omitir os comandos `subfloat`.

```
\begin{figure}[H] \centering
\includegraphics[scale=...]{nome do arquivo}
\includegraphics[scale=...]{nome do arquivo}
\caption{legenda}
\end{figure}
```

Observação 2. Quando vários comandos `includegraphics` são incluídos em um único ambiente, o $LaTeX$ interpreta como uma única figura.

Como cada imagem é inserida com um comando `includegraphics`, é possível inserir figuras lado a lado com diferentes escalas.

```
\begin{figure}[H]
\centering
\caption{Figuras lado a lado com escalas diferentes}
\subfloat[Escala 0.1]{\includegraphics[scale=0.1]{if-horizontal.png}}
\quad
\subfloat[Escala 0.2]{\includegraphics[scale=0.2]{if-horizontal.png}}
\quad
\subfloat[Escala 0.3]{\includegraphics[scale=0.3]{if-horizontal.png}}
\end{figure}
```

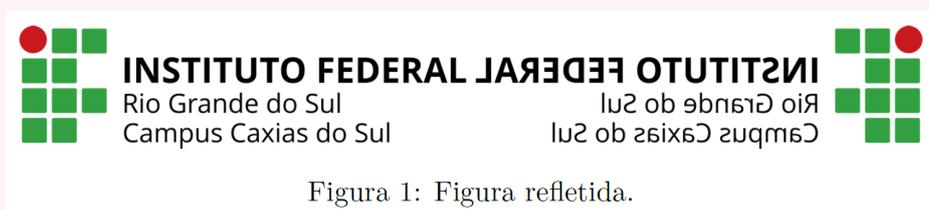


Figure 1: Figuras lado a lado com escalas diferentes

5.4 Refletindo figuras

Para inserir imagens refletidas utiliza-se o comando `reflectbox` antes do comando `includegraphics`.

```
\begin{figure}[H]
\centering
\caption{Figura refletida.}
\includegraphics[scale=0.3]{if-horizontal.png}
\reflectbox{\includegraphics[scale=0.3]{if-horizontal.png}}
\label{figura:refletida}
\end{figure}
```



5.5 Rotacionando figuras

Para rotacionar uma figura é preciso inserir um parâmetro opcional `angle` juntamente com um ângulo em graus no comando `includegraphics`.

Exemplo 33. *Rotacionando uma imagem a 45°.*

```
\begin{figure}[H] \centering
\includegraphics[scale=0.3, angle=45]{if-horizontal.png}
\caption{Rotacionando uma figura}
\end{figure}
```

5.6 Figura ao lado de texto

Para inserir figuras ao lado de texto é necessário inserir o pacote `wrapfig` no preâmbulo.

```
\usepackage{wrapfig}
```

Este pacote habilita o uso de um novo ambiente, o `wrapfigure`.

```
\begin{wrapfigure}[número de linhas]{posicionamento}{largura}
\centering
\includegraphics[scale=...]{nome do arquivo}
\caption{legenda da figura}
\end{wrapfigure}
```

O parâmetro `número de linhas` define o número de linhas que a imagem vai ocupar ao lado do texto. O parâmetro `posicionamento` é a posição da figura, podendo ser `r` (*right* - à direita) ou `l` (*left* - à esquerda). Por último, o parâmetro `largura` define a largura da figura.

Observação 3. *O comando para inserir a imagem deve vir antes do texto o qual queremos que a imagem fique ao lado.*

5.7 Referenciando figuras

Para fazer referência a uma figura utiliza-se o comando `label` dentro do ambiente `figure`. Quando for necessário referenciá-la no texto, utiliza-se o comando `ref`. Alguns editores de \LaTeX exigem que o documento seja compilado mais de uma vez para gerar o arquivo corretamente, como é o caso do \TeX maker.

Capítulo 6

Matemática no L^AT_EX

O L^AT_EX é muito utilizado para a elaboração de textos em linguagem matemática. Nele, é possível escrever, por exemplo, equações, matrizes, derivadas, integrais, de forma eficaz e prática. Além disso, é possível construir listas de exercícios, citar exemplos, definir teoremas, corolários, lemas, entre outros. Este capítulo tem o objetivo de apresentar os principais comandos que a linguagem oferece para a construção de conteúdos que envolvem linguagem matemática.

Antes de tudo, é necessário inserir alguns pacotes no preâmbulo, a fim de ter um aproveitamento melhor das ferramentas que o L^AT_EX dispõe.

```
\usepackage{amsmath, amssymb, amsfonts, amsthm, exscale}
```

6.1 Expressões matemáticas

O L^AT_EX dispõe de três ambientes para produção de expressões matemática: `math`, `displaymath` e `equation`.

6.1.1 Ambiente `math`

Para escrever expressões matemáticas na mesma linha que um texto, utiliza-se o ambiente `math`. Para simplificar, também é possível escrever a expressão entre cifrões (`$$`).

Exemplo 34. *Expressão matemática simples utilizando o ambiente `math`.*

Seja `$$x+y=10$` e `$x \cdot y=1$`.

Seja $x + y = 10$ e $x \cdot y = 1$.

6.1.2 Ambiente `displaymath`

Para que a expressão fique em destaque (centralizada), utiliza-se o ambiente `displaymath`. Também é possível utilizá-lo por meio dos seguintes comandos:

```
\[ expressão matemática \  
$$ expressão matemática $$
```

Exemplo 35. *Expressão matemática simples utilizando o ambiente `displaymath`.*

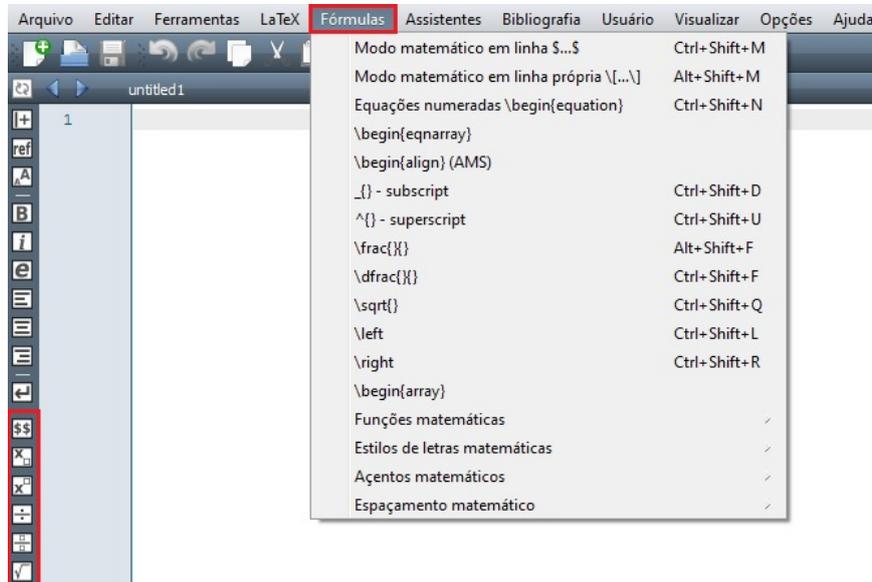
```
$$x+y=10$$
```

$$x + y = 10$$

6.2 Assistente para inserção de fórmulas (T_EXmaker)

O T_EXmaker possui um assistente para auxiliar na inserção de fórmulas matemáticas (Figura 6.1). O assistente está localizado na barra de ferramentas dentro do item FÓRMULAS. Neste item, encontram-se diversas opções de comandos prontos. Basta clicar no escolhido e editá-lo de acordo com a necessidade do usuário. Alguns dos comandos encontrados no item FÓRMULAS, também estão disponíveis na barra vertical à esquerda do monitor, conforme destacado na Figura 6.1.

Figura 6.1 – Assistente para inserção de fórmulas matemáticas.



6.2.1 Ambiente `equation`

O ambiente `equation` é um ambiente que enumera as expressões automaticamente. Da mesma forma que o `displaymath`, as expressões também ficam centralizadas.

Exemplo 36. *Expressões matemáticas enumeradas com o apoio do ambiente `equation`.*

```
\begin{equation}
f(x)=x+4
\end{equation}
\begin{equation}
g(y)=y-19
\end{equation}
```

$$f(x) = x + 4 \tag{6.1}$$

$$g(y) = y - 19 \tag{6.2}$$

O modo matemático não reconhece texto corrido. Para inserir expressões como “seja”, “onde” e “tal que”, utilizam-se alguns comandos específicos, como `textrm`, `text` e `mbox`.

Exemplo 37. *Utilização de texto comum dentro de um ambiente matemático com o comando `textrm`.*

```
$$\mathbb{Q} = \{a/b : \text{onde } a \in \mathbb{Z} \text{ e } b \in \mathbb{Z}^* \}$$
```

$$\mathbb{Q} = \{a/b : \text{onde } a \in \mathbb{Z} \text{ e } b \in \mathbb{Z}^*\}$$

Note que há um espaço de um caractere entre as chaves e o texto, seu uso é necessário para que haja o espaçamento adequado entre os caracteres.

6.3 Fontes no modo matemático

Na matemática, é comum utilizar uma tipografia diferente para indicar, por exemplo, os conjuntos numéricos: \mathbb{N} para os números naturais, \mathbb{Z} para os inteiros e assim por diante. Sendo assim, utilizam-se comandos especiais para essas fontes diferenciadas. A Tabela 6.1 apresenta algumas representações aceitas no modo matemático.

Tabela 6.1 – Tipografia diferenciada para representar conjuntos numéricos.

Código-fonte	Resultado
<code>\mathbb{N}</code>	\mathbb{N}
<code>\mathfrak{N}</code>	\mathfrak{N}
<code>\mathit{N}</code>	N
<code>\mathrm{N}</code>	N
<code>\mathcal{N}</code>	\mathcal{N}
<code>\mathtt{N}</code>	N
<code>\mathbf{N}</code>	\mathbf{N}
<code>\mathsf{N}</code>	N

Observação 4. Vale destacar que os comandos `mathbb` e `mathcal` só aceitam letras maiúsculas.

Exemplo 38. Conjuntos numéricos representados por meio do comando `mathbb`.

<pre> \$ \mathbb{N} \quad \mathbb{Z} \quad \mathbb{Q} \quad \mathbb{R} \quad \mathbb{I} \quad \mathbb{C} \$ </pre>	$\mathbb{N} \quad \mathbb{Z} \quad \mathbb{Q} \quad \mathbb{R} \quad \mathbb{I} \quad \mathbb{C}$
--	---

6.4 Expoentes e índices

Para expressar caracteres sobrescritos ou subscritos, utilizam-se, respectivamente, o acento circunflexo ($\hat{}$) e o sinal de sublinhado ($\underline{}$). Sempre que for necessário mais de um caractere nessas condições, é necessário envolver o conteúdo com chaves. Por exemplo, se for escrito `\mathit{x}^2n`, o \LaTeX entenderá apenas o número 2 como expoente, resultando em x^2n .

Exemplo 39. *Expoentes e índices simples.*

<code>\mathit{x}^3</code>	x^3
<code>\mathit{x}_n</code>	x_n
<code>\mathit{x}^3_n</code>	x_n^3

Exemplo 40. *Expoentes e índices compostos.*

```
$x^{3n} \neq x^{3n}$ \\
$x_{n-1} \neq x_{n-1}$ \\
$x^2_{n-1} \neq x^2_{n-1}$
```

$$x^{3n} \neq x^3n$$

$$x_{n-1} \neq x_n - 1$$

$$x_{n-1}^2 \neq x_n^2 - 1$$

6.5 Frações

Para representar frações utilizam-se os comandos `frac` e `dfrac`. Enquanto o primeiro apresenta a fração na mesma linha na qual está escrita, o comando `dfrac` apresenta-a em modo *displaystyle*, ocupando quantas linhas forem necessárias para exibir a fração.

```
\[ \dfrac{<numerador>}{<denominador>} \]
```

Exemplo 41. *Algumas aplicações dos comandos `frac` e `dfrac`.*

```
$$\frac{1}{x}$$ \\
$x + \frac{2}{1+y}$ \\
$$\frac{x+\frac{1}{2}}{3}$$ \\
$$\dfrac{x+\frac{1}{2}}{3}$$
```

$$\frac{1}{x}$$

$$x + \frac{2}{1+y}$$

$$\frac{x+\frac{1}{2}}{3}$$

$$\frac{x + \frac{1}{2}}{3}$$

6.6 Raízes

No \LaTeX , a representação de raízes se dá pelo comando `sqrt`.

```
\sqrt[<radical>]{<radicando>}
```

Exemplo 42. *Algumas representação de raízes.*

```
$$\sqrt{3}$$ \\
$$\sqrt[3]{27}$$ \\
$$\sqrt[5]{\sqrt{3x}+1}$$
```

$$\sqrt{3}$$

$$\sqrt[3]{27}$$

$$\sqrt[5]{\sqrt{3x} + 1}$$

Exemplo 43. Representação da fórmula da diagonal de um paralelepípedo de dimensões a , b e c .

```
\[ D = \sqrt{a^2 + b^2 + c^2} \]
```

$$D = \sqrt{a^2 + b^2 + c^2}$$

Exemplo 44. Representação da fórmula do volume de um tronco de pirâmide de altura h e áreas das bases A_B e A_b .

```
\[ V = \frac{\pi}{3} \cdot (A_B + \sqrt{A_B \cdot A_b} + A_b) \]
```

$$V = \frac{\pi}{3} \cdot (A_B + \sqrt{A_B \cdot A_b} + A_b)$$

Exemplo 45. Representação da fórmula de Bhaskara.

```
\[ x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \]
```

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

6.7 Limites

Para a representação de limites utiliza-se o comando `lim`.

```
\[ \lim \]
```

Exemplo 46. Representação de um limite de uma função definida por $f(x)$ à medida que x tende a a .

```
\$ \lim_{x \to a} f(x) \$
```

$$\lim_{x \rightarrow a} f(x)$$

Exemplo 47. Representação do mesmo limite do Exemplo 46 utilizando o comando `displaystyle`.

```
\$ \displaystyle \lim_{x \to a} f(x) \$
```

$$\lim_{x \rightarrow a} f(x)$$

Existem duas formas de representar expressões dentro dos ambientes matemáticos do \LaTeX : em linha e em modo `displaystyle`. Quando uma expressão é redigida em linha, o comando `\lim_{x \to a} f(x)` produz $x \rightarrow a$ ao lado do limite. Quando o comando `displaystyle` é adicionado, $x \rightarrow a$ fica posicionado abaixo do limite. Alguns ambientes aplicam automaticamente o `displaystyle`, como é o caso da utilização do cifrão duplo ($\$$).

6.8 Somatórios e Produtórios

Para somatórios e produtórios utilizam-se, respectivamente, os comandos `sum` e `prod`.

Exemplo 48. *Representação genérica de um somatório.*

`\sum\limits_{i=1}^n a_i`

$$\sum_{i=1}^n a_i$$

Exemplo 49. *Representação genérica de um produtório.*

`\prod\limits_{i=1}^n a_i`

$$\prod_{i=1}^n a_i$$

Exemplo 50. *Representação genérica de um produtório.*

`\sum\limits_{i=0}^n C_n^i =`
`\rightarrow C_n^0 + C_n^1 + C_n^2 + \dots +`
`\rightarrow C_n^n = 2^n`

$$\sum_{i=0}^n C_n^i = C_n^0 + C_n^1 + C_n^2 + \dots + C_n^n = 2^n$$

6.9 Derivadas

Para simbolizar derivadas não há um comando específico. Basta utilizar simplesmente $f'(x)$.

Exemplo 51. *Representação das derivadas primeira, segunda e terceira de $f(x) = \sin(x)$.*

`f'(x)=\cos x`
`f''(x)=-\sin x`
`f^{(3)}(x)=-\cos x`

$$f'(x) = \cos x$$

$$f''(x) = -\sin x$$

$$f^{(3)}(x) = -\cos x$$

Exemplo 52. *Representação generalizada de derivadas utilizando a notação de Leibniz.*

`\frac{df}{dx}=f'(x)`
`\frac{\partial f}{\partial x}=f_x`

$$\frac{df}{dx} = f'(x)$$

$$\frac{\partial f}{\partial x} = f_x$$

6.10 Integrais

O símbolo de integral é produzido pelo comando `int`. Para integrais definidas, é possível estabelecer os intervalos de integração.

```
$ \int $
```

Exemplo 53. *Representação genérica de integrais simples indefinidas.*

```
$\int f(x)\,dx$ \\  
$
```

```
$\displaystyle \int f(x)\,dx$
```

$$\int f(x) dx$$

$$\int f(x) dx$$

Exemplo 54. *Representação genérica de integrais simples definidas em um intervalo $[a, b]$.*

```
$\int_{a}^{b} g(x)\,dx$ \\  
$
```

```
$\displaystyle \int_{a}^{b} g(x)\,dx$ \\  
$
```

```
$\int \limits_{a}^{b} g(x)\,dx$ \\  
$
```

```
$\displaystyle \int \limits_{a}^{b} g(x)\,dx$ \\  
$
```

$$\int_a^b g(x) dx$$

$$\int_a^b g(x) dx$$

$$\int_a^b g(x) dx$$

$$\int_a^b g(x) dx$$

Exemplo 55. *Representação genérica de integrais fechadas.*

```
$\oint g(x)\,dx$ \\  
$
```

```
$\displaystyle \oint g(x)\,dx$
```

$$\oint g(x) dx$$

$$\oint g(x) dx$$

Exemplo 56. *Resolução de uma integral indefinida.*

```
$\displaystyle \int x^2\,dx=$
```

```
\to \frac{x^3}{3}+C$
```

$$\int x^2 dx = \frac{x^3}{3} + C$$

Exemplo 57. *Resolução de uma integral definida.*

<pre> \displaystyle \int\limits_{1}^{3} → x^2\,dx = \frac{x^3}{3} → \Big _1^3 = \frac{3^3}{3} - → \frac{1^3}{3} = 9 - \frac{1}{3} = → \frac{26}{3} </pre>	$\int_1^3 x^2 dx = \left. \frac{x^3}{3} \right _1^3 = \frac{3^3}{3} - \frac{1^3}{3} = 9 - \frac{1}{3} = \frac{26}{3}$
---	---

Para integrais duplas, triplas, etc., utilizam-se os comando apresentados na Tabela 6.2. Para tais comandos, o pacote `amsmath` precisa estar definido no preâmbulo.

Tabela 6.2 – Comandos para integrais múltiplas.

Comando	O que vai produzir
<code>\iint</code>	\iint
<code>\iiint</code>	\iiint
<code>\iiiiint</code>	\iiiiint
<code>\idotsint</code>	$\int \cdots \int$

6.11 Linhas e chaves

É possível inserir linhas e chaves abaixo ou acima de uma expressão. Os comandos estão apresentados na Tabela 6.3.

Tabela 6.3 – Comandos para linhas e chaves.

Código-fonte	Resultado
<code>\overbrace{x+y}</code>	$\overbrace{x+y}$
<code>\overbrace{x+y}^{\alpha}</code>	$\overbrace{x+y}^{\alpha}$
<code>\underbrace{x+y}_{\alpha}</code>	$\underbrace{x+y}_{\alpha}$
<code>\overline{x+y}</code>	$\overline{x+y}$
<code>\underline{x+y}</code>	$\underline{x+y}$

6.12 Delimitadores

Muitas vezes é necessário usar parênteses, chaves e colchetes. Para que uma expressão fique entre esses delimitadores e com o tamanho adequado, utilizam-se os comandos `left` e `right`, juntamente com o delimitador desejado. A vantagem da utilização desses comandos é que a altura dos delimitadores se adequa automaticamente à altura dos caracteres da expressão inserida.

Exemplo 58. *Utilização de colchetes com os comandos `left` e `right`.*

```
$$\left [ \frac{1}{3+x} \right ]$$
```

$$\left[\frac{1}{3+x} \right]$$

Às vezes é necessário definir o tamanho dos delimitadores. Para isso, existem alguns comandos manuais, como: `big`, `Big`, `bigg` e `Bigg`.

Exemplo 59. *Utilização do comando `bigg` junto a um delimitador.*

```
$$ k \cdot \bigg( \displaystyle \int  
→ f(x) \, dx \bigg) $$
```

$$k \cdot \left(\int f(x) dx \right)$$

6.13 Estruturas

O comando `newtheorem` permite criar ambientes personalizados para teoremas, corolários, exercícios, lemas, exemplos, definições, proposições, entre outros. Ele é inserido no preâmbulo do documento.

```
\newtheorem{<chamada>}{<saída>}
```

O primeiro parâmetro `chamada` é o nome de chamada do objeto, ou seja, como será indicada a inserção do objeto. O segundo parâmetro `saída` é como ele aparecerá no nosso documento. A vantagem desse comando é que ele enumera automaticamente os ambientes. Recomenda-se inserir todos os comandos `newtheorem` no preâmbulo do documento.

```
\newtheorem{ex}{Exercício}
```

Exemplo 60. Criando um ambiente para uma lista de exercícios.

<pre>\begin{ex} Texto do exercício um. \end{ex} \begin{ex} Texto do exercício dois. \end{ex}</pre>	<p>Exercício 1. <i>Texto do exercício um.</i></p> <p>Exercício 2. <i>Texto do exercício dois.</i></p>
--	---

Exemplo 61. Modelo de lista de exercícios utilizando o ambiente `ex` criado anteriormente.

```
\begin{ex}
Seja  $x \in \mathbb{Z}$ . Demonstre que este  $x$  será sempre menor do que um
número  $x + 1$ .
\end{ex}
\begin{ex}
Seja  $y \in \mathbb{Q}$ . Demonstre que este  $y$  será sempre menor do que um
número  $y + 1$ .
\end{ex}
```

Exercício 3. *Seja $x \in \mathbb{Z}$. Demonstre que este x será sempre menor do que um número $x + 1$.*

Exercício 4. *Seja $y \in \mathbb{Q}$. Demonstre que este y será sempre menor do que um número $y + 1$.*

Também é possível nomear a estrutura criada.

```
\newtheorem{teorema}{Teorema}
```

Exemplo 62. Teorema de Pitágoras.

```
\begin{teorema}[Pitágoras]
O quadrado da hipotenusa de um triângulo retângulo é igual à soma dos
quadrados dos catetos.
\end{teorema}
```

Teorema 1 (Pitágoras). *O quadrado da hipotenusa de um triângulo retângulo é igual à soma dos quadrados dos catetos.*

6.14 Matrizes

Em função de uma matriz ser escrita no formato de uma tabela, este assunto será abordado no capítulo 7, seção 7.4.

6.15 Símbolos

Esta seção tem o objetivo de apresentar um conjunto de símbolos frequentemente utilizados junto à escrita matemática.

Sequência de pontos	
$\backslash\ldots$...
\backslashcdots	...
\backslashvdots	⋮
\backslashddots	⋱

Pontuação			
$,$,	$.$$ ou $\backslash\dotp$.
$;$;	\backslashcdotp	.
$:$ ou \backslashcolon	:		

Letras Gregas					
<code>\alpha</code>	α	<code>\epsilon</code>	ϵ	<code>\phi</code>	ϕ
<code>\beta</code>	β	<code>\varepsilon</code>	ε	<code>\Phi</code>	Φ
<code>\theta</code>	θ	<code>\lambda</code>	λ	<code>\pi</code>	π
<code>\vartheta</code>	ϑ	<code>\gamma</code>	γ	<code>\Pi</code>	Π
<code>\delta</code>	δ	<code>\xi</code>	ξ	<code>\Omega</code>	Ω
<code>\Delta</code>	Δ	<code>\tau</code>	τ	<code>\sigma</code>	σ
<code>\rho</code>	ρ	<code>\Theta</code>	Θ	<code>\Sigma</code>	Σ
<code>\varrho</code>	ϱ	<code>\mu</code>	μ	<code>\kappa</code>	κ
<code>\upsilon</code>	υ	<code>\omega</code>	ω	<code>\eta</code>	η

Acentos no modo matemático			
<code>\hat{v}</code>	\hat{v}	<code>\ddot{v}</code>	\ddot{v}
<code>\vec{v}</code>	\vec{v}	<code>\acute{v}</code>	\acute{v}
<code>\bar{v}</code>	\bar{v}	<code>\check{v}</code>	\check{v}
<code>\breve{v}</code>	\breve{v}	<code>\tilde{v}</code>	\tilde{v}
<code>\grave{v}</code>	\grave{v}	<code>\dot{v}</code>	\dot{v}
<code>\stackrel{\frown}{v}</code>	$\stackrel{\frown}{v}$	<code>\widehat{abc}</code>	\widehat{abc}
<code>\overrightarrow{abc}</code>	\overrightarrow{abc}	<code>\overleftarrow{abc}</code>	\overleftarrow{abc}
<code>\widetilde{abc}</code>	\widetilde{abc}		

Funções					
<code>\sin</code>	sin	<code>\sinh</code>	sinh	<code>\arcsin</code>	arcsin
<code>\cos</code>	cos	<code>\cosh</code>	cosh	<code>\arccos</code>	arccos
<code>\tan</code>	tan	<code>\tanh</code>	tanh	<code>\arctan</code>	arctan
<code>\sec</code>	sec	<code>\sec</code>	sec	<code>\log</code>	log
<code>\csc</code>	csc	<code>\csc</code>	csc	<code>\ln</code>	ln
<code>\cot</code>	cot	<code>\coth</code>	coth	<code>\exp</code>	exp
<code>\lim</code>	lim	<code>\liminf</code>	lim inf	<code>\limsup</code>	lim sup
<code>\min</code>	min	<code>\max</code>	max	<code>\det</code>	det
<code>\sup</code>	sup	<code>\inf</code>	inf	<code>\arg</code>	arg

Setas					
<code>\Leftarrow</code>	\Leftarrow	<code>\leftarrow</code>	\leftarrow	<code>\mapsto</code>	\mapsto
<code>\Rightarrow</code>	\Rightarrow	<code>\rightarrow</code>	\rightarrow	<code>\longmapsto</code>	\longmapsto
<code>\Leftrightarrow</code>	\Leftrightarrow	<code>\leftrightharrow</code>	\leftrightharrow	<code>\hookrightarrow</code>	\hookrightarrow
<code>\Llongleftarrow</code>	\Llongleftarrow	<code>\longleftarrow</code>	\longleftarrow	<code>\hookrightarrow</code>	\hookrightarrow
<code>\Llongrightarrow</code>	\Llongrightarrow	<code>\longrightarrow</code>	\longrightarrow	<code>\nearrow</code>	\nearrow
<code>\Llongleftrightharrow</code>	\Llongleftrightharrow	<code>\longleftrightharrow</code>	\longleftrightharrow	<code>\searrow</code>	\searrow
<code>\Uparrow</code>	\Uparrow	<code>\uparrow</code>	\uparrow	<code>\swarrow</code>	\swarrow
<code>\Downarrow</code>	\Downarrow	<code>\downarrow</code>	\downarrow	<code>\nwarrow</code>	\nwarrow
<code>\Updownarrow</code>	\Updownarrow	<code>\updownarrow</code>	\updownarrow		

Símbolos de relações					
\sim	\approx	\leq	\leq	\in	\in
\simeq	\cong	\geq	\geq	\ni	\ni
\cong	\approx	$<$	$<$	\vdash	\vdash
$=$	$=$	$>$	$>$	\dashv	\dashv
\approx	\approx	\equiv	\equiv	\subset	\subset
\neq	\neq	$\not\equiv$	$\not\equiv$	\supset	\supset
\subseteq	\subseteq	\supseteq	\supseteq	\parallel	\parallel
\perp	\perp	\gg	\gg	\ll	\ll

Operações Binárias					
\pm	\pm	\mp	\mp	\times	\times
\div	\div	$*$	$*$	\star	\star
\cap	\cap	\cup	\cup	\bigtriangleup	\bigtriangleup
\vee	\vee	\wedge	\wedge	\bigtriangledown	\bigtriangledown
\oplus	\oplus	\otimes	\otimes	\odot	\odot
\circ	\circ	\bigcirc	\bigcirc		

Relações Binárias			
\geqslant	\geqslant	\leqslant	\leqslant
\geqq	\geqq	\leqq	\leqq
\sqsubset	\sqsubset	\sqsupset	\sqsupset
\subseteqq	\subseteqq	\supseteqq	\supseteqq
\therefore	\therefore	\because	\because

Relações Binárias de Negação			
$\$\\ngeqslant\$$	$\not\geq$	$\$\\nleqslant\$$	$\not\leq$
$\$\\ngeqq\$$	$\not\geq$	$\$\\nleqq\$$	$\not\leq$
$\$\\nsubseteq\$$	$\not\subseteq$	$\$\\nsupseteq\$$	$\not\supseteq$
$\$\\nsubseteqq\$$	$\not\subseteq$	$\$\\nsupseteqq\$$	$\not\supseteq$
$\$\\nmid\$$	\nmid	$\$\\nparallel\$$	\nparallel

Símbolos variáveis			
$\$\\sum\$$	Σ	$\$\\prod\$$	Π
$\$\\bigcup\$$	\cup	$\$\\bigcap\$$	\cap
$\$\\bigvee\$$	\vee	$\$\\bigwedge\$$	\wedge
$\$\\bigoplus\$$	\oplus	$\$\\bigotimes\$$	\otimes
$\$\\bigodot\$$	\odot	$\$\\bigsqcup\$$	\sqcup

Outros símbolos					
$\$\\forall\$$	\forall	$\$\\exists\$$	\exists	$\$\\nexists\$$	\nexists
$\$\\angle\$$	\angle	$\$\\measuredangle\$$	\sphericalangle	$\$\\sphericalangle\$$	\sphericalangle
$\$\\neg\$$	\neg	$\$\\infty\$$	∞	$\$\\ \$$	\parallel
$\$\\emptyset\$$	\emptyset	$\$\\diagdown\$$	\diagdown	$\$\\partial\$$	∂
$\$\\square\$$	\square	$\$\\triangle\$$	\triangle	$\$\\lozenge\$$	\lozenge
$\$\\blacksquare\$$	\blacksquare	$\$\\blacktriangle\$$	\blacktriangle	$\$\\blacklozenge\$$	\blacklozenge

Capítulo 7

Tabelas

Existem três ambientes para construção de tabelas: `tabbing`, `tabular` e `array`. Além disso, existe outro ambiente chamado `table`, que permite inserir legenda nas tabelas criadas.

7.1 Ambiente `tabbing`

Este ambiente organiza o texto com paradas de posicionamento, ou seja, é possível definir a distância entre uma coluna e outra. Dentro do ambiente `tabbing`, utiliza-se o símbolo de igualdade (=) para indicar os pontos de parada, os caracteres `\>` para fazer mudança de coluna e `\\` para mudança de linha. Vale ressaltar que, neste ambiente, não é possível obter linhas verticais e horizontais.

Exemplo 63. *Tabela simples criada com o ambiente `tabbing`.*

```
\begin{tabbing}
\hspace{2cm}\= \hspace{5cm}\= \kill
NOME \> NÚMERO \> NOTA \\
Amanda \> 1 \> $10,0$ \\
Bruna \> 2 \> $9,0$ \\
Cláudio \> 3 \> $9,5$
\end{tabbing}
```

NOME	NÚMERO	NOTA
Amanda	1	10,0
Bruna	2	9,0
Cláudio	3	9,5

7.2 Ambiente tabular

O ambiente `tabular` possibilita a construção de tabelas com linhas verticais e horizontais. Também é possível especificar o alinhamento do texto. Este ambiente possui dois parâmetros.

```
\begin{tabular}[<posição>]{<características>}
<conteúdo da tabela>
\end{tabular}
```

O parâmetro `posição` admite os caracteres `t` ou `b`. O parâmetro `t` dispõe o topo da tabela alinhada com o texto corrente e `b` dispõe a base da tabela alinhada com o texto corrente. Caso este parâmetro for omitido, a tabela ficará centralizada entre o texto corrente.

Em `características` é possível definir a quantidade de colunas da tabela e o alinhamento do texto. É utilizada uma barra (`|`) para estabelecer uma linha vertical, o caractere `l` para texto alinhado à esquerda (*left*), `r` para texto alinhado à direita (*right*) e `c` para centralizado. Por exemplo: `|c|` indica uma coluna com borda à esquerda e à direita e conteúdo alinhado ao centro, `|cc|` indica duas colunas sem a linha vertical entre elas e `||` determina borda dupla.

No ambiente `tabular`, `&` indica mudança de coluna, `\\` mudança de linha e o comando `hline` insere uma linha horizontal. Também é possível estabelecer um tamanho para colunas. Ao invés de utilizar os parâmetros `c`, `l` ou `r`, é inserido o parâmetro `p{medida}`.

Exemplo 64. *Tabela de disciplinas criada com ambiente `tabular`.*

```
\begin{tabular}[t]{|c|c|} \hline
DISCIPLINAS & CARGA HORÁRIA TOTAL (HORAS) \\ \hline
Geometria Plana & $75$ \\ \hline
Geometria Espacial & $60$ \\ \hline
Geometria Analítica & $60$ \\ \hline
\end{tabular}
```

DISCIPLINAS	CARGA HORÁRIA TOTAL (HORAS)
Geometria Plana	75
Geometria Espacial	60
Geometria Analítica	60

Exemplo 65. Tabela-verdade criada com o ambiente `tabular`.

```
\begin{tabular}[t]{|p{1cm}|p{1cm}|p{1cm}|} \hline


$ & $ & $ \wedge $ \\ \hline
V & V & V \\ \hline
V & F & F \\ \hline
F & V & F \\ \hline
F & F & F \\ \hline
\end{tabular}


```

p	q	$p \wedge q$
V	V	V
V	F	F
F	V	F
F	F	F

7.2.1 Os comandos `cline` e `multicolumn`

Para construir tabelas com células mescladas é possível usar os comandos `cline` e `multicolumn`. O comando `cline` faz a ligação entre duas colunas.

```
\cline{<coluna a - coluna b>}
```

Exemplo 66. Tabela criada utilizando o ambiente `tabular` e o comando `cline`.

```
\begin{tabular}[b]{|c|c|c|}
\hline
{\bf Informação} & {\bf Equação} & {\bf Sistema} \\ \hline
Perímetro &  $x+y=44$  & \\
\cline{1-2}
Área &  $2x+3y=100$  & \\
\hline
\end{tabular}
```

Informação	Equação	Sistema
Perímetro	$x + y = 44$	
Área	$2x + 3y = 100$	

Já o comando `multicolumn` distribui o texto por várias colunas, ou seja, produz um novo espaço com tamanho correspondente à soma das outras. Enquanto o comando `cline` possui apenas um parâmetro, o comando `multicolumn` possui três parâmetros.

```
\multicolumn{<número de colunas>}{<especificação>}{<texto>}
```

O número de colunas se refere ao número de colunas que serão mescladas, a especificação define o alinhamento do texto, podendo dar lugar aos parâmetros `l`, `r` ou `c`. Por fim, o texto é o conteúdo da nova célula.

Exemplo 67. Tabela criada utilizando o ambiente `tabular` e o comando `multicolumn`.

```
\begin{tabular}[t]{|c|c|c|}
\hline
\multicolumn{3}{|c|}{\it $1^{\circ}$ Semestre} \\ \hline
Matemática Elementar I & Geometria Plana & Geometria Analítica \\ \hline
Psicologia da Educação & Didática Geral & \\ \hline
\end{tabular}
```

<i>1° Semestre</i>		
Matemática Elementar I	Geometria Plana	Geometria Analítica
Psicologia da Educação	Didática Geral	

Exemplo 68. Tabela apresentando algumas identidades e relações trigonométricas.

```
\begin{tabular}{|c|c|c|}
\hline
\multicolumn{3}{|c|}{Algumas Identidades Trigonométricas} \\ \hline
\hline
$\sin^2 x + \cos^2 x = 1$ & $\sec^2 x = 1 + \tan^2 x$ & $\csc^2 x = 1 + \cot^2 x$ \\ \hline
\end{tabular}
```

```


$$\sin^2 x = \frac{1 - \cos 2x}{2}$$
 & 
$$\cos^2 x = \frac{1 + \cos 2x}{2}$$
 & \\
\hline

$$\sin 2x = 2 \sin x \cos x$$
 & 
$$\cos 2x = \cos^2 x - \sin^2 x$$
 & \\
\hline
\end{tabular}

```

Algumas Identidades Trigonômicas		
$\sin^2 x + \cos^2 x = 1$	$\sec^2 x = 1 + \tan^2 x$	$\csc^2 x = 1 + \cot^2 x$
$\sin^2 x = \frac{1 - \cos 2x}{2}$	$\cos^2 x = \frac{1 + \cos 2x}{2}$	
$\sin 2x = 2 \sin x \cos x$	$\cos 2x = \cos^2 x - \sin^2 x$	

7.3 Ambiente table

No ambiente `table` é possível definir o local da página que a tabela ficará posicionada, pois muitas vezes uma tabela pode “flutuar no texto”. Aqui é possível utilizar os mesmos parâmetros apresentados na seção 5.2 do Capítulo 5, sendo o uso do parâmetro `H` fortemente recomendado.

Neste ambiente é possível inserir legenda em uma tabela e referenciá-la; o processo é o mesmo que vimos na seção 5.7 do Capítulo 5.

Exemplo 69. *Tabela com legenda.*

```

\begin{table}[H]\centering
\caption{Função  $f(x)=2^x$ }
\begin{tabular}{|c|c|}
\hline
\multicolumn{2}{|c|}{ $f(x)=2^x$ } \\
\hline
↪ \hline
 $x$  &  $y$  \\ \hline
-2 &  $y = \frac{1}{4}$  \\ \hline
-1 &  $y = \frac{1}{2}$  \\ \hline
0 &  $y = 1$  \\ \hline
1 &  $y = 2$  \\ \hline
2 &  $y = 4$  \\ \hline
\end{tabular}
\label{fun}
\end{table}

```

Tabela 7.1 – Função $f(x) = 2^x$

$f(x) = 2^x$	
x	y
-2	$y = \frac{1}{4}$
-1	$y = \frac{1}{2}$
0	$y = 1$
1	$y = 2$
2	$y = 4$

7.4 Ambiente array

O ambiente array possibilita inserir caracteres, construir matrizes, sistemas, entre outros.

```
\begin{array}{<especificações>}
<conteúdo do ambiente>
\end{array}
```

O parâmetro especificações é utilizado para definir o alinhamento do texto, admitindo l, r ou c. Por exemplo, ao utilizar ccc como parâmetro, serão produzidas 3 colunas, todas com o texto centralizado. Para mudança de linha é comum utilizar \\ e para colunas &.

7.4.1 Matrizes

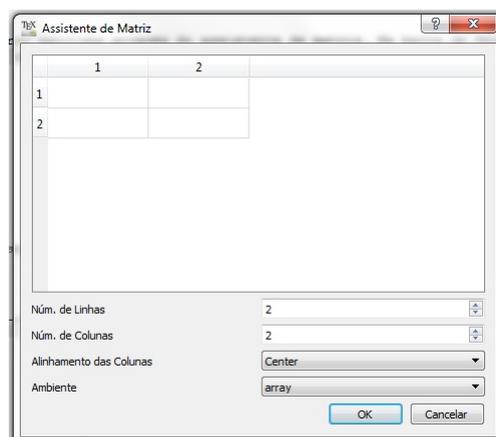
Uma matriz nada mais é do que uma tabela delimitada por colchetes ou parênteses.

Exemplo 70. *Matriz genérica criada com o ambiente array.*

```
$$ \mathbf{A}=
\left[
\begin{array}{cccc}
a_{11} & a_{12} & \dots & a_{1n} \\
a_{21} & a_{22} & \dots & a_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{m1} & a_{m2} & \dots & a_{mn}
\end{array}
\right]
\right] $$
```

Também é possível gerar matrizes por meio do assistente de matriz, disponível no *software* T_EXmaker. O assistente de matriz está localizado no menu ASSISTENTES na barra de ferramentas do *software*. A Figura 7.1 apresenta o assistente de matriz.

Figura 7.1 – Assistente de matriz.



Além de preencher os elementos da matriz, é possível escolher o número de linhas e colunas, alinhamento e também o ambiente para a construção da matriz. O ambiente possui seis opções: `array`, `matrix`, `pmatrix`, `bmatrix`, `vmatrix` e `Vmatrix`. Tais ambiente diferem por seus delimitadores, conforme Figura 7.2. O ambiente `array` funciona do mesmo modo como apresentado no Exemplo 70, sendo necessário definir manualmente o delimitador. Ressalta-se que esses ambientes devem estar entre `$` ou `$$`.

Figura 7.2 – Ambientes para construção de matrizes.

(a) <i>matrix</i>	(b) <i>pmatrix</i>	(c) <i>bmatrix</i>	(d) <i>vmatrix</i>
$\begin{matrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{matrix}$	$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$	$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$	$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$
(e) <i>Vmatrix</i>			
$\left\ \begin{matrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{matrix} \right\ $			

Exemplo 71. *Matriz identidade de ordem 3.*

<pre style="font-family: monospace; font-size: 0.9em;">\begin{displaymath} \mathbf{I}_3= \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array} \right] \end{displaymath}</pre>	$\mathbf{I}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
--	--

7.4.2 Sistemas de equações e funções definidas por mais de uma sentença

Para construir sistemas de equações e expressões de funções definidas por mais de uma sentença, é possível utilizar o ambiente `array` ou o ambiente `cases`. A principal diferença entre os dois é que o ambiente `cases` insere, de forma automática, a chave à esquerda das expressões.

Exemplo 72. *Representação de uma função definida por partes utilizando o ambiente `array`.*

<pre style="font-family: monospace; font-size: 0.9em;">\$\$ f(x)=\left\{ \begin{array}{l} x, & \text{se } x \geq 0 \\ -x, & \text{se } x < 0 \end{array} \right. \$\$</pre>	$f(x) = \begin{cases} x, & \text{se } x \geq 0 \\ -x, & \text{se } x < 0 \end{cases}$
--	---

Exemplo 73. Representação de um sistema linear utilizando o ambiente `cases`.

```


$$\begin{cases} x+y=20 \\ 3x+5y=50 \end{cases}$$


```

$$\begin{cases} x + y = 20 \\ 3x + 5y = 50 \end{cases}$$

Exemplo 74. Representação de um sistema linear utilizando o ambiente `array`.

```


$$\left\{ \begin{array}{l} x + 4y = 10 \\ x - y = 5 \end{array} \right.$$


```

$$\begin{cases} x + 4y = 10 \\ x - y = 5 \end{cases}$$

Exemplo 75. Representação de uma função definida por partes utilizando o ambiente `cases`.

```


$$f(x) = \begin{cases} -3x+2, & \text{se } x < 0 \\ x+2, & \text{se } 0 \leq x < 2 \\ 3x-2, & \text{se } x \geq 2 \end{cases}$$


```

$$f(x) = \begin{cases} -3x + 2, & \text{se } x < 0 \\ x + 2, & \text{se } 0 \leq x < 2 \\ 3x - 2, & \text{se } x \geq 2 \end{cases}$$

7.5 Ambiente `eqnarray`

Para equações grandes ou equações multilinhas, utiliza-se o ambiente `eqnarray`. Nele, as expressões são enumeradas. Neste ambiente, é possível pular linhas e escrever várias expressões alinhadas verticalmente. Ele também funciona como uma espécie de tabela, onde cada linha compreende uma equação, integrando uma tabela com 3 colunas (com organização `rc1`). O ambiente `eqnarray*` não enumera as equações.

Exemplo 76. Utilização do ambiente `eqnarray`.

```


$$\begin{eqnarray} 27x^2y+9xy^2+3xy & = & 3(9x^2y+3xy^2+xy) \\ & = & 3x(9xy+3y^2+y) \\ & = & 3xy(9x+3y+1) \end{eqnarray}$$


```

$$27x^2y + 9xy^2 + 3xy = 3(9x^2y + 3xy^2 + xy) \quad (7.1)$$

$$= 3x(9xy + 3y^2 + y) \quad (7.2)$$

$$= 3xy(9x + 3y + 1) \quad (7.3)$$

Caso não seja necessário enumerar todas as expressões, é possível usar o comando `nonumber` ao lado da linha desejada.

Exemplo 77. *Utilização do comando `nonumber` dentro do ambiente `eqnarray`.*

```
\begin{eqnarray}
27x^2y+9xy^2+3xy & = & 3(9x^2y+3xy^2+xy) \nonumber \\
& = & 3x(9xy+3y^2+y) \nonumber \\
& = & 3xy(9x+3y+1) \\
\end{eqnarray}
```

$$27x^2y + 9xy^2 + 3xy = 3(9x^2y + 3xy^2 + xy)$$

$$= 3x(9xy + 3y^2 + y)$$

$$= 3xy(9x + 3y + 1) \quad (7.4)$$

7.5.1 Outros apontamentos

A quebra de linha em expressões longas (quando a margem direita é ultrapassada) deve ser definida de forma manual.

Exemplo 78. *Expressão longa com quebra de linha manual.*

```
\begin{eqnarray}
\sum\limits_{n=0}^{\infty} (0,9)^n & = & 1+0,9+0,81+0,729+ \nonumber \\
& & +0,6561+0,59049+\{\}\cdots \\
\end{eqnarray}
```

$$\sum_{n=0}^{\infty} (0,9)^n = 1 + 0,9 + 0,81 + 0,729 +$$

$$+0,6561 + 0,59049 + \dots \quad (7.5)$$

Cada equação recebe uma numeração diferente. Caso seja necessário que a numeração seja igual, é preciso utilizar o pacote `subeqnarray`.

Exemplo 79. *Expressões enumeradas com o pacote subeqnarray.*

```
\begin{subeqnarray}
\sin^2 x & = & 1 - \cos^2 x \\
\sec^2 x & = & 1 + \tan^2 x
\end{subeqnarray}
```

$$\sin^2 x = 1 - \cos^2 x \quad (7.6a)$$

$$\sec^2 x = 1 + \tan^2 x \quad (7.6b)$$

Capítulo 8

Caixas personalizadas com o pacote

mdframed

Existem diversas formas de criação de caixas personalizadas no \LaTeX . Este capítulo tem o objetivo de abordar os tópicos essenciais do `mdframed`, um pacote que permite ao usuário a criação de caixas personalizadas simples de forma rápida.

```
\usepackage{mdframed}
```

8.1 O ambiente `mdframed`

Assim que o pacote for incluído no preâmbulo, já é possível criar uma caixa utilizando o ambiente `mdframed`. Por padrão, esta caixa tem largura fixa, uma borda preta sutil e fundo branco.

Exemplo 80. *Caixa personalizada simples.*

```
\begin{mdframed}
```

Caixa personalizada.

```
\end{mdframed}
```

Caixa personalizada.

8.2 Personalizando as caixas

O comando `mdfsetup` permite personalizar as caixas. Para fins de organização do código-fonte, recomenda-se que este comando seja inserido no preâmbulo. No entanto, neste material o comando `mdfsetup` será apresentado ligeiramente antes da caixa criada por facilidade ao apresentar os exemplos.

```
\mdfsetup{<modificações>}
```

Algumas modificações possíveis são alterar a cor e a espessura da borda, alterar a cor do fundo, remover bordas específicas e adicionar sombra.

Exemplo 81. *Caixa personalizada em tons de cinza.*

```
\mdfsetup{
  topline=false,bottomline=false, linewidth=5pt,
  linecolor=gray!50, backgroundcolor=gray!10,
  frametitle=Título da caixa, frametitlebackgroundcolor=gray!20,
  innerbottommargin=25pt, innertopmargin=25pt
}

\begin{mdframed}
Conteúdo da caixa.
\end{mdframed}
```

Título da caixa

Conteúdo da caixa.

8.2.1 Modificando as margens

As margens de uma caixa podem ser omitidas por meio dos comandos `topline`, `bottomline`, `rightline` e `leftline`, juntamente com uma especificação `false`.

Tabela 8.1 – Comandos para omitir as margens.

Código-fonte	Resultado
<code>topline = false</code>	omite a margem superior
<code>bottomline = false</code>	omite a margem inferior
<code>leftline = false</code>	omite a margem da esquerda
<code>rightline = false</code>	omite a margem da direita
<code>hidealllines = true</code>	omite todas as margens

O parâmetro `linewidth` altera a espessura da borda. Ele acompanha uma especificação de medida, sendo que é comum utilizar pontos (`pt`).

Para alterar a cor da borda, utiliza-se o parâmetro `linecolor`, juntamente com a especificação da cor desejada.

Exemplo 82. *Caixa personalizada.*

```
\mdfsetup{
  topline=false, leftline=false, linewidth=3pt, linecolor=gray
}

\begin{mdframed}
Conteúdo da caixa.
\end{mdframed}
```

Conteúdo da caixa.

Exemplo 83. *Caixa personalizada.*

```
\mdfsetup{
  rightline=false, leftline=false, linewidth=5pt, linecolor=orange!50
}

\begin{mdframed}
Conteúdo da caixa.
\end{mdframed}
```

Conteúdo da caixa.

8.2.2 Alterando a cor da fonte e de fundo

Para alterar a cor da fonte, utiliza-se o parâmetro `fontcolor`, juntamente com a cor desejada. Ainda, para alterar a cor de fundo, é necessário utilizar o parâmetro `backgroundcolor`, cujo funcionamento se dá da mesma forma que o comando `fontcolor`.

Exemplo 84. *Caixa com texto em cor azul.*

```
\mdfsetup{
  bottomline=false, topline=false,
  linewidth=2pt, linecolor=blue!50,
  fontcolor=blue!70
}
```

```

\begin{mdframed}
Conteúdo da caixa.
\end{mdframed}

```

Conteúdo da caixa.

Exemplo 85. *Caixa em tons de cinza.*

```

\mdfsetup{
  bottomline=false, topline=false, rightline=false,
  linewidth=5pt, linecolor=gray!50, fontcolor=gray!5,
  backgroundcolor=gray!80
}

\begin{mdframed}
Conteúdo da caixa.
\end{mdframed}

```

Conteúdo da caixa.

8.2.3 Criando um título e subtítulo

É possível também adicionar um título e/ou um subtítulo nas caixas criadas. Para inserir um título, utiliza-se o parâmetro `frametitle`. No entanto, a inserção de um subtítulo se dá de forma diferente, por meio de um comando `mdfsubtitle` inserido diretamente dentro da caixa, no local desejado. Assim, é possível adicionar quantos subtítulos forem necessários, não apenas um.

Exemplo 86. *Caixa com título.*

```

\mdfsetup{
  rightline=false, leftline=false, linewidth=5pt,
  linecolor=orange!50, frametitle=Título da caixa
}

\begin{mdframed}
Conteúdo da caixa.
\end{mdframed}

```

Título da caixa

Conteúdo da caixa.

Exemplo 87. *Caixa com título e subtítulo.*

```
\mdfsetup{
  rightline=false, leftline=false, linewidth=5pt,
  linecolor=blue!50, frametitle=Título da caixa
}

\begin{mdframed}
\mdfsubtitle{Subtítulo da caixa}
Conteúdo da caixa.
\mdfsubtitle{Outro subtítulo da caixa}
Mais conteúdo da caixa.
\end{mdframed}
```

Título da caixa

Subtítulo da caixa

Conteúdo da caixa.

Outro subtítulo da caixa

Mais conteúdo da caixa.

Existem uma série de parâmetros que permitem a personalização de títulos e subtítulos. É possível modificar a cor de fundo, adicionar espaçamento acima e abaixo e adicionar uma linha com espessura definida pelo usuário, também acima e abaixo. Todas esses parâmetros podem ser aplicados tanto ao título quanto ao subtítulo. A Tabela 8.2 apresenta os parâmetros aplicáveis aos títulos, enquanto a Tabela 8.3 apresenta os parâmetros aplicáveis aos subtítulos.

Tabela 8.2 – Comandos para alterar o visual do título.

Código-fonte	Resultado	Valor
<code>frametitlerule</code>	adiciona uma linha abaixo do título	<code>true, false</code>
<code>frametitlerulewidth</code>	altera a espessura da linha criada	<code>zzzzzz</code>
<code>frametitlebackgroundcolor</code>	altera a cor de fundo do título	<code>red!40, orange, ...</code>
<code>frametitleaboveskip</code>	adiciona espaçamento acima do título	<code>12pt, 5mm, ...</code>
<code>frametitlebelowskip</code>	adiciona espaçamento abaixo do título	<code>12pt, 5mm, ...</code>

Tabela 8.3 – Comandos para alterar o visual do subtítulo.

Código-fonte	Resultado	Valor
<code>subtitleaboveline</code>	adiciona uma linha acima	<code>true, false</code>
<code>subtitlebelowline</code>	adiciona uma linha abaixo	<code>true, false</code>
<code>subtitlebackgroundcolor</code>	altera a cor de fundo	<code>red!40, orange, ...</code>
<code>subtitleabovelinewidth</code>	altera a espessura da linha superior	<code>12pt, 5mm, ...</code>
<code>subtitlebelowlinewidth</code>	altera a espessura da linha inferior	<code>12pt, 5mm, ...</code>
<code>subtitleabovelinecolor</code>	altera a cor da linha superior	<code>red!40, orange, ...</code>
<code>subtitlebelowlinecolor</code>	altera a cor da linha inferior	<code>red!40, orange, ...</code>
<code>subtitleaboveskip</code>	adiciona espaçamento acima	<code>12pt, 5mm, ...</code>
<code>subtitlebelowskip</code>	adiciona espaçamento abaixo	<code>12pt, 5mm, ...</code>

Exemplo 88. *Caixa com título personalizado.*

```

\mdfsetup{
  linewidth=4pt, linecolor=green!70!black, frametitle=Título da caixa,
  frametitlerule=true, frametitleaboveskip=10pt, frametitlebelowskip=10pt,
  frametitlebackgroundcolor=green!20,frametitlerulewidth=1pt
}
\begin{mdframed}
Conteúdo da caixa.
\end{mdframed}

```

Título da caixa

Conteúdo da caixa.

Exemplo 89. Caixa com título e subtítulo personalizados.

```
\mdfsetup{
  linewidth=4pt, linecolor=green!70!black, frametitle=Título da caixa,
  frametitlerule=true, frametitleaboveskip=10pt, frametitlebelowskip=10pt,
  frametitlebackgroundcolor=green!25, frametitlerulewidth=1pt,
  subtitleaboveline=true, subtitlebelowline=true,
  subtitleabovelinewidth=1pt, subtitlebelowlinewidth=1pt,
  subtitleabovelinecolor=green!70!black,
  ↪ subtitlebelowlinecolor=green!70!black,
  subtitlebackgroundcolor=green!10
}

\begin{mdframed}
Conteúdo da caixa.
\mdfsubtitle{Subtítulo da caixa}
Conteúdo do subtítulo caixa.
\end{mdframed}
```

Título da caixa

Conteúdo da caixa.

Subtítulo da caixa

Conteúdo do subtítulo caixa.

8.3 O ambiente `newmdenv`

Em alguns casos, é necessário criar mais de um tipo de caixa personalizada em um documento. Assim, nesse caso, o ambiente `mdframed` pode se tornar limitado. O comando `newmdenv` permite a criação de novos ambientes com todos os parâmetros apresentados neste capítulo.

```
\newmdenv[<parâmetros>]{<nome do ambiente>}
```

O nome do ambiente é o nome de chamada do ambiente criado.

```
\begin{<nome do ambiente>}[<parâmetros>]
<conteúdo da caixa>
\end{<nome do ambiente>}
```

Outra grande vantagem do uso desse comando é poder adicionar parâmetros únicos em cada ambiente criado no corpo do documento. Essa funcionalidade é muito utilizada, por exemplo, na criação de títulos.

Exemplo 90. *Utilização básica do ambiente newmdenv.*

```
\newmdenv[
  linecolor=orange!50, rightline=false, leftline=false,
  linewidth=6pt, frametitlebackgroundcolor=orange!20
]{exemplos}

\begin{exemplos}[frametitle={Exemplo 1.}]
Encontre o valor de  $x$ .
\end{exemplos}
```

Exemplo 1.

Encontre o valor de x .

Referências

- [1] PIERINI, A. C. et al. **LaTeX**: produção e apresentação de textos científicos. Santa Maria, 2015. (Apostila)
- [2] SODRÉ, U. **LaTeX com o TeXnicCenter**. Londrina, 2011. (Apostila)
- [3] LATEX. Wikibooks contributors. The Free Textbook Project. 2016. Disponível em: <https://en.wikibooks.org/wiki/LaTeX>. Acesso em: 4 maio 2018.
- [4] DANIEL, M. SCHUBERT, E. **The mdfamed package**. Disponível em: <https://ctan.org/pkg/mdfamed>. Acesso em: 4 maio 2018.
- [5] ARSENEAU, D. **The framed package**. Disponível em: <https://ctan.org/pkg/framed>. Acesso em: 4 maio 2018.