

Emanuel Mendes Queiroz, Giselle Lopes da Cruz, Márcio Antônio de Andrade Bortoloti, Maria Clara Brito dos Reis, Samara Viriato Vilar Dias, Wéllington Moutinho Dias

# Apostila *LaTeX*

Uma introdução ao editor de texto *LaTeX*

23 de outubro de 2023



# Agradecimentos

Os autores agradecem ao Programa de Educação Tutorial Institucional da Univesidade Estadual do Sudoeste da Bahia (PETI/UESB) pelas bolsas de estudo e à *Springer Nature* por disponibilizar o template que tornou possível a criação deste material.



# Apresentação

O Programa de Educação Tutorial Institucional de Matemática (PETIMAT) tem como missão aprimorar os cursos regulares de graduação na Universidade Estadual do Sudoeste da Bahia (UESB), com o objetivo de proporcionar uma formação abrangente e de alta qualidade para os alunos envolvidos, mantendo a integração entre ensino, pesquisa e extensão como um princípio fundamental do ambiente universitário.

O PETIMAT se dedica a elevar o padrão da formação dos estudantes de graduação e promover o sucesso acadêmico. Para alcançar esses objetivos, o programa oferece, a cada semestre, minicursos voltados principalmente para os alunos matriculados no curso de matemática da UESB. Esses minicursos visam estimular a capacitação de futuros profissionais e docentes, proporcionando-lhes uma qualificação acadêmica, científica, técnica e tecnológica.

Dentro desse contexto, o PETIMAT desenvolveu o minicurso "Introdução à Edição de Textos usando LaTeX". O objetivo desse minicurso é apresentar conceitos e noções iniciais relacionados ao LaTeX, bem como fornecer instruções sobre comandos básicos de formatação de texto e introduzir os principais tipos de documentos científicos, tais como artigos, monografias, apresentações (utilizando o beamer) e pôsteres. A intenção é capacitar os participantes para que se tornem autônomos na produção de seus próprios textos.

Este material corresponde às notas de aula e serve como um recurso de consulta para os participantes do minicurso. Nesta apostila, estão disponíveis informações detalhadas e comandos necessários para criar e personalizar esses diversos tipos de documentos. Para ter acesso aos diferentes modelos de documentos criados durante o minicurso, basta consultar o nosso repositório no GitHub:

[https://github.com/petimatematica/curso\\_latex](https://github.com/petimatematica/curso_latex)

Para conhecer mais sobre o trabalho e as publicações do PETIMAT, acesse o nosso site:

<http://www2.uesb.br/programa/petimatematica/>

e nossa conta oficial no Instagram:

<https://www.instagram.com/petimatuesb/>

onde estão dispostas informações detalhadas sobre nossas atividades, projetos, eventos e conteúdos relacionados ao nosso programa.



# Conteúdo

<b>1</b>	<b>Introdução</b>	1
1.1	O que é o LaTeX?	1
1.2	Vantagens de usar o LaTeX	1
1.3	Instalação	2
1.3.1	Instalação do MiKTeX no Windows	2
1.3.2	Instalação do MiKTeX no Linux	2
1.3.3	Instalação do MacTeX no Macbook	2
1.4	Editores LaTeX	3
1.4.1	TeXWorks	3
1.4.2	TeXStudio	3
1.4.3	Overleaf	3
1.5	Tópicos básicos do LaTeX	3
1.5.1	Preâmbulo	3
1.5.2	Elementos iniciais do texto	4
1.6	Pacotes e Extensões	4
1.7	Ambientes	4
<b>2</b>	<b>Formatação de texto</b>	7
2.1	Tamanho e estilo de fonte	7
2.2	Espaçamento entre linhas	8
2.3	Hifenização	9
2.4	Ajuste de margens	9
2.5	Alinhamento do texto	9
2.6	Inserir cabeçalho, rodapé e número de páginas	10
2.6.1	Cabeçalho e rodapé	11
2.6.2	Número de página	12
2.7	Exercícios	12
<b>3</b>	<b>Tópicos Específicos</b>	15
3.1	Acentuação	15
3.2	Seções e subseções	16
3.3	Ambientes de equação	16
3.3.1	Alinhamento de equações	17
3.4	Definição de novos comandos	18

<b>4</b>	<b>Matemática e Símbolos</b>	21
4.1	Símbolos	21
4.1.1	Fonte Mathbb	21
4.1.2	Letras hebraicas	22
4.1.3	Letras gregas	23
4.1.4	Símbolos matemáticos	23
4.1.5	Fonte Mathcal	24
4.1.6	Fonte Mathfrak	24
4.1.7	Delimitadores	24
4.1.8	Operadores	25
4.1.9	Operadores grandes	25
4.1.10	Acentos	25
4.1.11	Miscelânea	25
4.2	Construções básicas	27
4.2.1	Operações aritméticas	27
4.2.2	Subscrever e sobrescrever em ambientes matemáticos	27
4.2.3	Coeficientes binomiais	28
4.2.4	Reticências	28
4.2.5	Integrais	28
4.2.6	Raízes	29
4.2.7	Texto	30
4.2.8	Delimitadores	31
4.2.9	Operadores e novos símbolos	32
4.2.10	Chaves e linhas horizontais	33
4.2.11	Fórmulas emolduradas	34
4.2.12	Função definida por partes	34
4.2.13	Fórmulas muito longas	36
4.2.14	Matrizes	36
4.2.15	Sistemas lineares	39
4.2.16	Exercícios	41
<b>5</b>	<b>Figuras, Tabelas e Diagramas</b>	43
5.1	Figuras	43
5.1.1	Inserção de figuras	43
5.1.2	Posicionamento e dimensionamento de figuras	45
5.1.3	Subfiguras	46
5.2	Tabelas	48
5.2.1	Criação Básica de Tabelas	48
5.2.2	Formatação de Tabelas	49
5.2.3	Mesclagem de Células	51
5.2.4	Aparência das tabelas	52
5.3	Diagramas de Venn	52
5.4	Diagramas Comutativos e representação de Grafos	54
5.4.1	Diagramas Comutativos	54
5.4.2	Grafos	56
5.4.3	Exercícios	58



<b>6 Bibliografias e Citações</b> .....	59
6.1 Construção de Sumário .....	59
6.2 Construção de Índice Remissivo .....	60
6.3 Construção de Lista de Figuras e Tabelas .....	61
6.4 Legendas e Referências cruzadas .....	62
6.5 Sistemas de Gerenciamento de Referências .....	63
Referências .....	65



# Capítulo 1

## Introdução

Neste material, exploraremos o sistema TeX, com foco especial no editor de texto  $\LaTeX$ . Descobriremos como essa ferramenta pode ser utilizada para criar e formatar uma ampla variedade de documentos, abrangendo desde simples artigos até monografias, apresentações em estilo *beamer* e pôsteres informativos. Ao longo deste guia, iremos explorar as ferramentas essenciais que o  $\LaTeX$  oferece, permitindo produzir documentos profissionais.

### 1.1 O que é o LaTeX?

O TeX é uma notável linguagem de marcação, concebida e apresentada por Donald Knuth em 1978. Essa linguagem trouxe consigo um novo padrão de qualidade tipográfica, porém, logo após seu lançamento, surgiu uma demanda por uma abordagem mais simplificada e acessível. Isso se deve ao fato de que, apesar das melhorias na tipografia, o TeX apresentava uma curva de aprendizado acentuada e necessitava de conhecimento em programação, o que restringia sua adoção a um público mais especializado.

Esse cenário deu origem ao  $\LaTeX$ , uma evolução do TeX desenvolvida por Leslie Lamport na década de 1980. O  $\LaTeX$  foi projetado para facilitar a produção de documentos de alta qualidade, em que os autores poderiam focar no conteúdo em vez de se preocupar com os detalhes de formatação. Com comandos mais intuitivos e uma estrutura de documento modular, o  $\LaTeX$  rapidamente se tornou uma ferramenta fundamental para a comunidade acadêmica e científica, onde a apresentação precisa de fórmulas matemáticas e citações bibliográficas era essencial.

O desenvolvimento posterior do  $\LaTeX$  representou uma resposta à necessidade de tornar a composição de documentos tipográficos mais acessível e eficaz, desempenhando um papel significativo na evolução da produção textual, especialmente na esfera acadêmica e científica.

### 1.2 Vantagens de usar o LaTeX

O  $\LaTeX$  oferece diversas vantagens que o tornam uma escolha assertiva para a criação de documentos. Neste contexto, uma de suas principais vantagens é a versatilidade, adequada para a criação de diversos tipos de documentos. Além disso, o  $\LaTeX$  é amplamente reconhecido por sua capacidade de lidar com as notações matemáticas de maneira prática e eficiente, em que é possível escrever equações complexas e símbolos matemáticos com facilidade. Ademais, ao contrário de muitos editores de texto que possuem múltiplas barras de ferramentas, o  $\LaTeX$  oferece uma abordagem mais simplificada, em que é possível editar todo o texto em uma única aba.

## 1.3 Instalação

A instalação do  $\LaTeX$  é dividida em duas partes, pois envolve a configuração de dois componentes essenciais: o compilador e o editor.

Ao escrever um texto em  $\LaTeX$  estamos essencialmente criando um código-fonte que descreve a estrutura e o conteúdo do documento. Esse código-fonte contém comandos que instruem o  $\LaTeX$  sobre como formatar o texto. Para isso, é necessário compilar o texto, ou seja, o  $\LaTeX$  lê o código-fonte, interpreta os comandos e gera um arquivo de saída.

O editor é o responsável por oferecer uma série de recursos que simplificam a criação e a formatação de documentos, como preenchimento automático de comandos, ajuda na identificação de erros no código e atalhos para a compilação instantânea. Além disso, os editores frequentemente organizam a visualização do documento, permitindo a visão da área de edição e do documento final lado a lado, facilitando a identificação de possíveis ajustes.

### 1.3.1 Instalação do MiKTeX no Windows

O MiKTeX é o responsável por compilar as entradas em  $\TeX$  no Windows e gerar o documento já formatado. Para a instalação, acesse o site oficial do MiKTeX

<https://miktex.org/download>

e escolha o sistema operacional correspondente ao seu dispositivo. Com isso, basta clicar em "Download" e, em seguida, executar o arquivo de instalação já baixado. Após isso clique em "Avançar" no termo com as condições para o uso e selecione "Complete MiKTeX" para baixar a versão completa. o MixTeX será baixado para o local selecionado anteriormente.

### 1.3.2 Instalação do MiKTeX no Linux

É importante observar que o MikTeX não possui uma versão nativa para sistemas Linux. Em virtude disso, recomendamos o tutorial disposto no site oficial do MiKTeX

<https://miktex.org/download>

em que será possível encontrar informações específicas para usuários do Linux, incluindo instruções passo a passo sobre como configurar e utilizar o MiKTeX neste sistema.

### 1.3.3 Instalação do MacTeX no Macbook

Acesse o site oficial do MacTeX:

<https://tug.org/mactex/mactex-download.html>.

Baixe o arquivo de instalação *MacTeX.pkg* e em seguida, execute-o. Após isso, o assistente de instalação do MacTeX será aberto. Siga as instruções na tela para concluir a instalação. Elas incluirão a aceitação do contrato de licença e a escolha do diretório de instalação. Vale ressaltar que o usuário deve verificar a versão do seu Macbook para baixar a versão do MacTeX correspondente.

## 1.4 Editores LaTeX

### 1.4.1 TeXWorks

Acesse o site oficial do TeXWorks

<https://www.tug.org/texworks/>

e baixe a versão apropriada para o seu sistema operacional (Windows, Mac ou Linux). Após isso, execute o arquivo e siga as instruções para concluir a instalação padrão.

### 1.4.2 TexStudio

Acesse o site oficial do TexStudio

<https://www.texstudio.org/>

e baixe a versão apropriada para o seu sistema operacional (Windows, Mac ou Linux). Após isso, execute o arquivo e siga as instruções para concluir a instalação padrão.

### 1.4.3 Overleaf

O Overleaf é uma plataforma online para escrita em LaTeX que não requer instalação local. Basta criar uma conta no site do Overleaf

<https://www.overleaf.com/project>

e começar a escrever imediatamente, pois ele entende a linguagem  $\LaTeX$  e formata o texto automaticamente, sendo possível exportá-lo para um arquivo em PDF ou outros formatos para imprimir ou compartilhar.

## 1.5 Tópicos básicos do LaTeX

O  $\LaTeX$  oferece uma maneira estruturada de criar documentos. Através dos comandos e ambientes, podemos definir a formatação, a organização e os elementos do nosso documento de forma precisa e eficaz.

### 1.5.1 Preâmbulo

No  $\LaTeX$ , o preâmbulo representa as configurações iniciais do código onde são definidas diretrizes globais, formatação e informações cruciais que permeiam todo o documento a ser criado. Ele é o responsável por dar forma, estilo e personalidade à criação textual. Nele, não deve ser escrito o conteúdo principal do texto, mas devem ser declaradas as regras que vão orientar a apresentação, como a escolha do tipo de documento ou os pacotes a serem utilizados, que abordaremos posteriormente. No preâmbulo, deve ser definido o tipo de documento que está sendo criado, como um artigo, um relatório ou um livro. Isso pode ser feito com o comando `\documentclass{}`.

### 1.5.2 Elementos iniciais do texto

No  $\text{\LaTeX}$ , um documento é organizado por meio de comandos específicos. Para iniciar o texto, deve ser utilizado o comando `\begin{document}` e para finalizar, o comando `\end{document}`. Tudo entre esses comandos será processado e exibido no documento.

A barra invertida (`\`) é usada para indicar comandos no  $\text{\LaTeX}$ . No preâmbulo, o comando `\title{}` é utilizado para definir o título do documento, já o comando `\author{}` define o nome do autor e `\date{}` define a data. Essas informações serão usadas posteriormente para criar o cabeçalho do texto.

Outro fator importante é o símbolo `%`, usado para indicar comentários ao longo da escrita. Quando usado, qualquer texto seguido de `%` em uma linha não será processado pelo compilador, e portanto, não aparecerá no documento final.

### 1.6 Pacotes e Extensões

Os Pacotes e extensões no  $\text{\LaTeX}$  são conjuntos de comandos e funcionalidades pré-definidos que permitem a inclusão de recursos adicionais ao seu documento. Eles abrangem desde a manipulação de imagens, tabelas, equações matemáticas complexas até a inserção de códigos-fonte e algoritmos. Essas extensões são criadas por membros da comunidade  $\text{\LaTeX}$  para facilitar tarefas específicas e adicionar recursos especializados. Através da incorporação de pacotes e extensões, pode-se adicionar funcionalidades avançadas, formatar elementos específicos e aprimorar significativamente a qualidade e a estética do trabalho.

### 1.7 Ambientes

Um ambiente é uma estrutura delimitada que define a maneira que o conteúdo deve ser formatado e exibido. O  $\text{\LaTeX}$  oferece uma variedade de ambientes que permite formatar diferentes partes do documento de maneira específica. Esses ambientes fornecem estruturas de textos, listas, equações e outros elementos. O símbolo `$`, por exemplo, desempenha um papel fundamental na marcação do *ambiente matemático*. Quando envolvemos uma expressão ou equação entre `$$`, o  $\text{\LaTeX}$  reconhece que o conteúdo deve ser interpretado como código matemático. Ao longo deste material, serão abordados outros ambientes. Abaixo, seguem exemplos de ambientes que não necessitam de outros comandos e pacotes no preâmbulo.

#### Listas não numeradas

```
\begin{itemize}
...
\end{itemize}
```

#### Listas numeradas

```
\begin{enumerate}
  \item{}
...
\end{enumerate}
```

---

Neste exemplos, o comando utilizado para adicionar os itens da lista é definido por `\item{}`.





## Capítulo 2

# Formatação de texto

Em todo texto é necessário elementos para formatá-lo de acordo com as necessidades de cada escritor. Assim, neste capítulo, apresentaremos os tópicos básicos para formatação de qualquer texto em  $\LaTeX$ .

### 2.1 Tamanho e estilo de fonte

Um dos aspectos mais importantes quando se está formatando um texto é o tamanho da fonte. Abaixo, iremos mostrar as principais formas de fazer essas alterações em um texto utilizando o  $\LaTeX$ .

Comando	Resultado
<code>\tiny{Exemplo}</code>	Exemplo
<code>\scriptsize{Exemplo}</code>	Exemplo
<code>\footnotesize{Exemplo}</code>	Exemplo
<code>\small{Exemplo}</code>	Exemplo
<code>\normalsize{Exemplo}</code>	Exemplo
<code>\large{Exemplo}</code>	Exemplo
<code>\Large{Exemplo}</code>	Exemplo
<code>\LARGE{Exemplo}</code>	Exemplo
<code>\huge{Exemplo}</code>	Exemplo
<code>\Huge{Exemplo}</code>	Exemplo

Para deixar o texto em negrito é necessário usar o comando `\textbf`. Veja o exemplo:

#### Texto em negrito

```
\begin{document}
  \textbf{Exemplo}
\end{document}
```

Para deixar o texto em itálico é necessário usar o comando `\textit`. Veja o exemplo:

### Texto em itálico

```
\begin{document}
  \textit{Exemplo}
\end{document}
```

Para deixar o texto sublinhado é necessário usar o comando `\underline`. Veja o exemplo:

### Texto sublinhado

```
\begin{document}
  \underline{Exemplo}
\end{document}
```

Para deixar o texto com alguma cor diferente é necessário que no preâmbulo esteja o pacote `xcolor` e em seguida no texto com o comando `\textcolor{cor}{texto}`, sendo o primeiro argumento a cor que deseja e no segundo o texto que deseja colorir. Veja o exemplo a seguir:

### Cor no texto

```
\usepackage{xcolor}
\begin{document}
  \textcolor{red}{Texto}
\end{document}
```

## 2.2 Espaçamento entre linhas

Para alterar o espaçamento entre linhas é preciso adicionar no preâmbulo do texto o pacote `setspace` e no corpo do texto adicionar um dos seguintes comandos:

- `\singlespacing` - Espaçamento simples
- `\onehalfspacing` - Espaçamento de 1,5
- `\doublespacing` - Espaçamento duplo

### Espaçamento entre linhas

```
\usepackage{setspace}
\begin{document}
  \singlespacing
  Copiar texto desejado. \\
  O texto terá espaçamento simples
```

```
%Para alterar espaçamento basta adicionar o outro
\onehalfspacing
Copiar texto desejado.\\
O texto terá espaçamento de 1,5
\end{document}
```

---

Se você deseja alterar o espaçamento apenas para um trecho específico de texto (e não para todo o documento), você pode usar os ambientes `singlenspace`, `onehalfspace` e `doublepace`.

## 2.3 Hifenização

Em um texto em  $\text{\LaTeX}$  a separação silábica é feita de forma automática, visto que a linguagem já é automatizada para que possa ser feito em qualquer idioma. Basta utilizar no preâmbulo o pacote `\usepackage[portuguese]{babel}`. No entanto, caso se queira fazer a hifenização de forma manual de uma palavra é preciso utilizar o comando `\hyphenation{e-xem-plo}` no preâmbulo.

## 2.4 Ajuste de margens

Em um texto em  $\text{\LaTeX}$  é preciso ajustar as margens de acordo com as necessidades de cada escritor. Assim, para configurar as margens é necessário inserir no preâmbulo o pacote `\usepackage{geometry}`. Veja o exemplo abaixo:

### Ajustando margens

```
\usepackage[left=2cm,right=2cm,top=3cm,bottom=3cm]{geometry}
%Defina a configuração das margens

\begin{document}
  Inserir texto.
\end{document}
```

---

## 2.5 Alinhamento do texto

O alinhamento em um texto é importante, pois distribui o texto uniformemente entre as margens. Para se ter um alinhamento à esquerda, à direita ou centralizado em um trecho do texto é preciso inserir o texto dentro dos ambientes citados abaixo.

### Alinhamento

```
\begin{document}
```

```

\begin{flushleft}
Trecho alinhado à esquerda
\end{flushleft}

\begin{flushright}
Trecho alinhado à direita
\end{flushright}

\begin{center}
Trecho centralizado
\end{center}
\end{document}

```

---

Para se fazer o alinhamento do texto todo é necessário inserir os seguintes comandos antes do texto:

### Alinhamento em todo texto

```

\begin{document}
\raggedleft %alinhamento à esquerda
\raggedright %alinhamento à direita
\centering %centralizado
%Selecione algum desses alinhamentos e inicie a escrita.

Escrever texto
\end{document}

```

---

Para que o texto seja escrito de forma justificada, é necessário inserir no preâmbulo o pacote `ragged2e` e utilizar o comando `\justifying` antes de iniciar a escrita. Veja o exemplo abaixo:

### Alinhamento

```

\usepackage{ragged2e}
\begin{document} \justifying

Escrever texto que será justificado.

\end{document}

```

---

## 2.6 Inserir cabeçalho, rodapé e número de páginas

Em diversos textos, é necessário que tenha cabeçalho, rodapé e indicação do número de páginas para uma melhor organização do texto. Na escrita  $\LaTeX$  essas configurações são feitas de forma simples e rápida.

### 2.6.1 Cabeçalho e rodapé

Para inserir cabeçalho e rodapé no texto em  $\text{\LaTeX}$  é necessário inserir no preâmbulo o pacote `fancyhdr` e, em seguida, ainda no preâmbulo, colocar `\pagestyle{fancy}` para que o estilo da página siga a fórmula dada pelo pacote `fancyhdr`. Veja o exemplo

#### Cabeçalho e rodapé

```
\pagestyle{fancy} % Define o estilo de página como "fancy"
\fancyhf{} % Limpa todos os estilos de cabeçalho e rodapé atuais

% Configuração do cabeçalho
\fancyhead[L]{Cabeçalho à esquerda}
\fancyhead[C]{Cabeçalho no centro}
\fancyhead[R]{Cabeçalho à direita}

% Configuração do rodapé
\fancyfoot[L]{Rodapé à esquerda}
\fancyfoot[C]{Rodapé no centro}
\fancyfoot[R]{Rodapé à direita}

% Linha superior do cabeçalho
\renewcommand{\headrulewidth}{0.4pt}

% Linha inferior do rodapé
\renewcommand{\footrulewidth}{0.4pt}
```

Para fazer as configurações para o estilo, podemos usar os comandos `lhead`, `rhead`, `lfoot`, `cfoot` e `rfoot` para que seja possível definir o conteúdo que vai ser inserido no cabeçalho e no rodapé em páginas distintas do texto. Já para inserir em uma página específica, basta usar o comando `pagestyle{fancy}`. Veja o exemplo de como colocar cabeçalho e rodapé:

#### Exemplo

```
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhf{}
\fancyhead[L]{Cabeçalho à esquerda}
\fancyhead[C]{Cabeçalho no centro}
\fancyhead[R]{Cabeçalho à direita}
\fancyfoot[L]{Rodapé à esquerda}
\fancyfoot[C]{Rodapé no centro}
\fancyfoot[R]{Rodapé à direita}
\renewcommand{\headrulewidth}{0.4pt}
\renewcommand{\footrulewidth}{0.4pt}
\begin{document}
```

Seu conteúdo vai aqui.

```
\end{document}
```

---

## 2.6.2 Número de página

Para inserir número nas páginas no texto em L<sup>A</sup>T<sub>E</sub>X, é preciso inserir no preâmbulo o pacote `fancyhdr`, já visto anteriormente. Em seguida, para exibir os números nas páginas, basta inserir o comando `\thepage`. Veja exemplo:

### Número nas páginas

```
\usepackage{fancyhdr}

\pagestyle{fancy}
\fancyhf{}
\fancyhead[L]{Cabeçalho à esquerda}
\fancyhead[C]{Cabeçalho no centro}
\fancyhead[R]{Cabeçalho à direita}
\fancyfoot[L]{Rodapé à esquerda}
\fancyfoot[C]{\thepage} % Insere o número da
página no centro do rodapé
\fancyfoot[R]{Rodapé à direita}
\renewcommand{\headrulewidth}{0.4pt}
\renewcommand{\footrulewidth}{0.4pt}

\begin{document}

Seu conteúdo vai aqui.

\end{document}
```

---

Neste exemplo, o comando `\fancyfoot[C]{\thepage}` insere o número da página no centro do rodapé.

## 2.7 Exercícios

1. Escreva o texto abaixo.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus vitae bibendum libero. Sed vestibulum metus vitae massa fringilla, vel interdum nisi dictum. *Maecenas vel neque sed orci venenatis tristique. Fusce ut condimentum libero, in tincidunt libero*. Sed varius, arcu in fermentum hendrerit, sapien urna fringilla nulla, id consequat sapien odio eu

justo. Vivamus auctor suscipit lectus.

2. Deixe o texto acima centralizado.

3. Coloque espaçamento entre as linhas de 1,5.





## Capítulo 3

### Tópicos Específicos

Tendo em vista o que foi visto no capítulo anterior, é possível perceber que em um texto em  $\text{\LaTeX}$  a formatação é algo muito importante para que o texto final fique da forma desejada. Neste capítulo, será discutido sobre alguns tópicos específicos necessários para a escrita.

#### 3.1 Acentuação

Em  $\text{\LaTeX}$ , existem duas maneiras de se acentuar as palavras. A primeira forma é manualmente e a segunda é da forma tradicional. Para escrever de forma manual, existe um modelo universal em  $\text{\LaTeX}$ . Veja o exemplo abaixo:

##### Acentuação manual

```
\begin{document}
%Colocar ~, ç
Acentua\c c\~ao

%Colocar ´, ‘, ^
S\’ilaba %aspas simples
\‘a %o acento grave
Jud\^o %o acento circunflexo

\end{document}
```

Já para colocar a acentuação de forma direta pelo teclado basta acrescentar no preâmbulo o pacote `\usepackage[utf8]{inputenc}`.

### 3.2 Seções e subseções

Dentro do ambiente `document`, podemos usar comandos como `section{}` para criar seções no documento e `subsection{}` para criar subseções. As seções ajudam a organizar o conteúdo em partes distintas, facilitando a leitura e compreensão. Veja o exemplo:

#### Seção e subseção

```
\begin{document}
  \section{Nome da seção}
  Texto

  \subsection{Nome da subseção}
  Texto
\end{document}
```

---

### 3.3 Ambientes de equação

Em textos matemáticos é inevitável a escrita de equações. E uma das vantagens da escrita em  $\text{\LaTeX}$  é o fato das equações ficarem organizadas. No  $\text{\LaTeX}$  podemos utilizar os seguintes ambientes para a escrita matemática: `equation`, `eqnarray`, `array`, `align`, `gather` ou utilizar `\$equação\$`. Veja um exemplo abaixo:

#### Ambiente de equação

No ambiente `equation` não é necessário usar `$$`, pois já é um ambiente matemático.

```
\begin{document}
% Equação numerada
  \begin{equation}
```

```

x+y=56
\end{equation}

% Equação não numerada
\begin{equation*}
x+y=56
\end{equation*}
\end{document}

```

---

### Resultado

Equação numerada

$$x + y = 56 \tag{3.1}$$

Equação não numerada

$$x + y = 56$$


---

#### 3.3.1 Alinhamento de equações

Ao escrever as equações em um texto, às vezes é necessário alinhá-las. No  $\text{\LaTeX}$ , para se alinhar as equações, é preciso utilizar o pacote `\usepackage{amsmath}` e o ambiente `align`.

No ambiente `align` não precisa usar `$$`, pois já é um ambiente matemático. Assim, quando temos duas ou mais equações, podemos alinhá-las da seguinte forma:

#### Alinhar equação

```

\begin{document}
\begin{align*}
2x + 3y &= 7

```

```

3x - 2y &= 4
\end{align*}

\end{document}

```

---

Obteremos o seguinte resultado:

$$2x + 3y = 7$$

$$3x - 2y = 4$$

Para especificar onde será o alinhamento na próxima linha se utiliza & antes da parte que deseja alinhar

### 3.4 Definição de novos comandos

Na escrita matemática podem aparecer Teoremas, Lemas, Corolários, Proposições, entre outros. No  $\text{\LaTeX}$ , é possível definir esses ambientes no preâmbulo e inseri-los ao longo do texto de acordo com a necessidade. Veja o exemplo abaixo:

#### Novos comandos

```

\newtheorem{definition}{Definição}[section]
\newtheorem{theorem}{Teorema}[section]
\newtheorem{cor}{Corolário}[section]
\newtheorem{lemma}{Lema}[section]
\newtheorem{prop}{Proposição}[section]
\newtheorem{Ex}{Exemplo}[section]
\newtheorem{Remark}{Observação}[section]

\begin{document}
\begin{theorem}

```

```
Enunciado do teorema
```

```
\end{theorem}
```

```
% Para os outros comandos se faz da mesma forma
```

```
\end{document}
```

---

## Resultado

*Teorema* Enunciado do teorema

□

---



## Capítulo 4

# Matemática e Símbolos

Neste capítulo, apresentaremos os símbolos e operadores utilizados nas seções seguintes. Equações básicas serão construídas e, por fim, exercícios serão propostos ao leitor para que o mesmo possa verificar o seu progresso.

### 4.1 Símbolos

Apesar do nome do capítulo ser chamado de "Matemática e Símbolos", os símbolos serão apresentados primeiro por razões didáticas. Apesar disso, não se espera e nem se recomenda que o leitor memorize todos os comandos. Use esta seção apenas como um material para consulta, pois acreditamos que a memorização deve ocorrer em virtude da experiência de uso do  $\text{\LaTeX}$ .

A lista a seguir não é completa, por isso, acesse o site <https://app.mettzer.com/latex>, os apêndices A, B e C de [2], ou ainda, [3] para mais informações. Para ter acesso a todos os símbolos a seguir é necessário usar os pacotes `amsmath`, `amsthm`, `amsfonts` e `amssymb`.

#### 4.1.1 Fonte `Mathbb`

ABCDEFGHIJKLMNOPQRSTUVWXYZ

##### 4.1.1.1 Fonte `Mathbf`

**ABCDEFGHIJKLMNOPQRSTUVWXYZ**

**abcdefghijklmnopqrstuvwxyz**

### 4.1.2 Letras hebraicas

א \aleph ב \beth ג \daleth ד \gimel



### 4.1.3 Letras gregas

$\alpha$ \alpha	$\Theta$ \Theta	$\Xi$ \varXi	$\Upsilon$ \Upsilon
$\beta$ \beta	$\vartheta$ \vartheta	$\pi$ \pi	$\Upsilon$ \varUpsilon
$\Gamma$ \Gamma	$\Theta$ \varTheta	$\Pi$ \Pi	$\phi$ \phi
$\Gamma$ \varGamma	$\iota$ \iota	$\varpi$ \varpi	$\Phi$ \Phi
$F$ \digamma	$\kappa$ \kappa	$\Pi$ \varPi	$\varphi$ \varphi
$\delta$ \delta	$\varkappa$ \varkappa	$\rho$ \rho	$\Phi$ \varPhi
$\Delta$ \Delta	$\lambda$ \lambda	$\varrho$ \varrho	$\chi$ \chi
$\Delta$ \varDelta	$\Lambda$ \Lambda	$\sigma$ \sigma	$\psi$ \psi
$\epsilon$ \epsilon	$\Lambda$ \varLambda	$\Sigma$ \Sigma	$\Psi$ \Psi
$\epsilon$ \varepsilon	$\mu$ \mu	$\varsigma$ \varsigma	$\Psi$ \varPsi
$\zeta$ \zeta	$\nu$ \nu	$\Sigma$ \varSigma	$\omega$ \omega
$\eta$ \eta	$\xi$ \xi	$\tau$ \tau	$\Omega$ \Omega
$\theta$ \theta	$\Xi$ \Xi	$\upsilon$ \upsilon	$\Omega$ \varOmega

### 4.1.4 Símbolos matemáticos

$\in$	$\leq$	$\sim$	$\equiv$
$\subseteq$	$\perp$	$\ni$	$\geq$
$\cong$	$\supset$	$\supseteq$	$\parallel$
$\therefore$	$\geqslant$	$\because$	$\pm$
$\cdot$	$\circ$	$\bigcirc$	$\div$
$\cap$	$\cup$	$\sqcap$	$\sqcup$
$\vee$	$\oplus$	$\otimes$	$\odot$
$\dagger$	$\setminus$	$\neq$	$\notin$
$\ngtr$	$\nleq$	$\ngeq$	$\nleqslant$
$\Leftrightarrow$	$\supsetneq$	$\Leftarrow$	$\Rightarrow$
$\Leftrightarrow$	$\subsetneq$	$\partial$	$\infty$
$\Rightarrow$	$\emptyset$	$\forall$	$\exists$
$\Leftarrow$	$\nabla$	$\neg$	$\complement$
$\subset$	$\approx$	$\leqslant$	$\times$
$\bmod$	$\wedge$	$\amalg$	$\less$
$\ngeqslant$	$\mapsto$	$\emptyset$	$\top$
$\nexists$			

#### 4.1.5 Fonte Mathcal

*ABCDEFGHIJKLMNOPQRSTUVWXYZ*

#### 4.1.6 Fonte Mathfrak

**ABCDEFGHIJKLMNOPQRSTUVWXYZ**

**abcdefghijklmnopqrstuvwxyz**

#### 4.1.7 Delimitadores

<code>{\lbrace</code>	<code>}\rbrace</code>	<code>\backslash</code>	<code>\langle</code>
<code>\rangle</code>	<code>\ </code>	<code>\lceil</code>	<code>\rceil</code>
<code>\lfloor</code>	<code>\rfloor</code>		

### 4.1.8 Operadores

<code>\arcsin</code>	<code>\cos</code>	<code>\cosh</code>	<code>\det</code>	<code>\dim</code>
<code>\exp</code>	<code>\ker</code>	<code>\ln</code>	<code>\inf</code>	<code>\lim</code>
<code>\liminf</code>	<code>\limsup</code>	<code>\min</code>	<code>\max</code>	<code>\sin</code>

### 4.1.9 Operadores grandes

<code>\prod</code>	<code>\sum</code>	<code>\bigcap</code>	<code>\bigcup</code>
<code>\bigoplus</code>	<code>\bigotimes</code>	<code>\int</code>	<code>\oint</code>
<code>\iint</code>	<code>\iiint</code>	<code>\iiiiiint</code>	<code>\int \dots \int</code>
<code>\coprod</code>	<code>\bigwedge</code>	<code>\bigvee</code>	<code>\bigsqcup</code>

### 4.1.10 Acentos

`\acute{a}` `\bar{a}` `\ddot{a}` `\grave{a}`

`\hat{a}` `\tilde{a}` `\vec{a}`

### 4.1.11 Miscelânea

`\hbar` `\ell` `\imath` `\jmath` `\smallint` `\clubsuit`  
`\diamondsuit` `\heartsuit` `\spadesuit` `\pounds` `\mho`

`\hslash` `\eth` `\measuredangle` `\sphericalangle`

## 4.2 Construções básicas

### 4.2.1 Operações aritméticas

Começaremos com adição, subtração, multiplicação e divisão,

$$a + b \quad a+b \quad a - b \quad a-b \quad ab \quad ab \quad a/b \quad a/b$$

há outras duas formas de indicar uma multiplicação ou outra de divisão:

$$a \cdot b \quad a \cdot b \quad a \times b \quad a \times b \quad a \div b \quad a \div b$$

frações podem ser inscritas com o comando `\frac{numerador}{denominador}`,

$$\frac{x+y+z}{x-z+1} \quad \text{\frac{x+y+z}{x-z+1}}$$

Observamos que se este comando for escrito numa linha de texto, ou seja, fora do ambiente `equation`, por exemplo, teremos este resultado  $\frac{x+y+z}{x-z+1}$  ao invés deste  $\frac{x+y+z}{x-z+1}$ , de forma geral podemos optar por qual estilo usar, o primeiro resultado é chamado de *inline-style* e o segundo de *display-style*. No geral, podemos determinar qual estilo usar com os comandos `\tfrac` (*inline-style*) e `\dfrac` (*display-style*). Assim é possível obter o resultado abaixo,

$$\frac{5x}{y+1} \quad \text{\tfrac{5x}{y+1}}$$

### 4.2.2 Subscrever e sobrescrever em ambientes matemáticos

Para subscrever e sobrescrever usa-se `_` e `^`, respectivamente. A expressão que se desejar subscrever ou sobrescrever deve estar entre chaves `{}`. Veja os exemplos:

$$x_1 \quad x_{\{1\}} \quad x^2 \quad x^{\{2\}} \quad x^{a_1} \quad x^{\{a_{\{1\}}\}} \quad x_{i+1} \quad x_{\{i+1\}}$$

$$x_i^2 \quad x^{\{2\}}_{\{i\}} \quad x_i^2 \quad x_{\{i\}}^{\{2\}} \quad x_{i,j}^{a_i} \quad x_{\{i\}_{\{j\}}^{\{2\}}\}^{\{a^{\{2\}}_{\{i\}}\}}$$

Cabe salientar que o antepenúltimo e o penúltimo exemplo nos mostram que não é necessário seguir uma ordem em específico para a subscrição e a sobrescrição, já o último exemplo no diz que é possível realizar o processo de sobrescrição em textos já subscritos, por exemplo. Não obstante, caso o que se deseja subscriver ou sobrescrever tenha apenas um caractere, as chaves podem ser omitidas, apesar disso, tal procedimento é considerado uma má prática. Em certas ocasiões, precisamos escrever <sup>†</sup> ou <sub>1</sub>, para isso basta usar  $\{\}^{\dagger}$  e  $\{\}_{1}$ , respectivamente, ou seja, as chaves com nenhum caractere entre elas servem como uma base para a sobrescrição e subscrição.

### 4.2.3 Coeficientes binomiais

Um coeficiente binomial pode ser escrito com o comando `\binom`. Exemplo:

$$\binom{n}{p} = \frac{n!}{p!(n-p)!} \quad \text{\binom{n}{p}=\frac{n!}{p!(n-p)!}}$$

### 4.2.4 Reticências

As reticências podem ser escritas de duas formas ... (*low*) e ... (*centered*) com os comandos `\ldots` e `\cdots`, respectivamente. Caso `\dots` seja utilizado o  $\LaTeX$  decidirá uma das duas formas baseado no caractere que antecede o comando. Por exemplo o comando `T(x_1, x_2, \dots, x_n)=T(x_1)+T(x_2)+\dots+T(x_n)` resulta em,

$$T(x_1, x_2, \dots, x_n) = T(x_1) + T(x_2) + \dots + T(x_n).$$

Além disso, temos os comandos `\vdots` (reticências na horizontal) e `\ddots` (reticências na diagonal), daremos um exemplo adiante, quando discutiremos a construção de matrizes.

### 4.2.5 Integrais

Podemos escrever uma integral usando o comando `\int`,

$$\int \cos x \, dx = \sin x + C \quad \text{\int \cos x\, dx = \sin x + C}$$

para as integrais definidas, basta usar a noção de subscrição e sobrescrição abordada na seção 4.2.1, assim,

$$\int_0^1 x^2 dx = \frac{1}{3} \quad \backslash\text{int}_{\{0\}}^{\{1\}} x^2 \backslash, dx = \backslash\text{frac}\{1\}\{3\}$$

já em relação às integrais impróprias, precisamos saber apenas que  $\infty$  pode ser escrito com o comando `\infty`,

$$\int_{-\infty}^{\infty} e^{-x^2} dx \quad \backslash\text{int}_{\{-\infty\}}^{\{\infty\}} e^{\{-x^2\}} \backslash, dx$$

ademais, é possível forçar que o símbolo de integral esteja entre os limites de integração graças ao comando `\limits` aplicando no exemplo anterior, temos:

$$\int_{-\infty}^{\infty} e^{-x^2} dx \quad \backslash\text{int}\backslash\text{limits}_{\{-\infty\}}^{\{\infty\}} e^{\{-x^2\}} \backslash, dx$$

aplicando o que acabamos de ver com as variações do comando `\int` podemos escrever integrais múltiplas,

$$\iiint_{\Omega} f(w, x, y, z) dw dx dy dz$$

$$\backslash\text{iiint}\backslash\text{limits}_{\{\Omega\}} f(w, x, y, z) \backslash, dw \backslash, dx \backslash, dy \backslash, dz \backslash$$

$$\int_0^{\sqrt{\pi}} \int_0^x \int_0^{xz} x^2 \sin y dy dz dx$$

### Integral tripla.

$$\backslash\text{int}^{\{\sqrt{\pi}\}}_{\{0\}} \backslash\text{int}^{\{x\}}_{\{0\}} \backslash\text{int}^{\{xz\}}_{\{0\}} x^2 \backslash\sin y \backslash, dy \backslash, dz \backslash, dx \backslash$$

### 4.2.6 Raízes

O comando `\sqrt` produz uma raiz quadrada, por exemplo,  $\sqrt{5}$  é gerado por `\sqrt{5}` ou ainda,

$$\sqrt{\frac{x^2}{x^2+1}} \quad \backslash\text{sqrt}\{\backslash\text{frac}\{x^2\}\{x^2+1\}\}$$

para escrevermos uma raiz enésima basta utilizarmos o seguinte argumento opcional, como em `\sqrt[3]{5}` de modo que obteremos  $\sqrt[3]{5}$ , além disso, podemos deslocar o radical usando `\leftroot` e `\uproot`,

$$\sqrt[3]{5} \quad \text{\code{\sqrt[\leftroot{2} \uproot{2} 3]{5}}}$$

Vale destacar que utilizando um número positivo como argumento de `\leftroot` o radicando é deslocado uma determinada quantidade de espaço para a esquerda, caso seja um número negativo, este deslocamento ocorre à direita, a mesma lógica se aplica ao comando `\uproot`.

#### 4.2.7 Texto

O  $\LaTeX$  permite que se escreva textos em ambientes matemáticos, através do comando `\text`. Por exemplo:

$$A \cup B = \{x; x \in A \text{ ou } x \in B\}$$

é obtido por `A\cup B = \{x;\text{ }x\in A \text{ ou } x\in B\}`. O pacote `amsfonts` habilita o comando `\mathbb` que acessa uma fonte que tem como característica possuir letras com duplo traço que são usadas na matemática para denotar conjuntos numéricos, por exemplo,

$$\mathbb{N} \subset \mathbb{Z} \subset \mathbb{R}$$

$$\text{\code{\mathbb{N} \subset \mathbb{Z} \subset \mathbb{R}}}$$

há ainda outros estilos matemáticos de fontes que são acessadas por `\mathcal` e `\mathfrak` e que possuem sintaxe parecida com o comando `\mathbb`.

Ademais, existem quatro tamanhos de fontes em ambientes matemáticos, a saber:

$$\begin{array}{l} \prod \text{\code{\displaystyle \coprod}} \quad \prod \text{\code{\textstyle \coprod}} \\ \prod \text{\code{\scriptstyle \coprod}} \quad \prod \text{\code{\scriptscriptstyle \coprod}} \end{array}$$

dessa forma podemos selecionar um tamanho de fonte independentemente se o que gostaríamos escrever possui um comando específico ou não para tal, como vimos com as frações e os comandos `\tfrac` e `\dfrac`.

Por fim, apresentaremos alguns comandos de espaçamento,



```

♠ ♣      \spadesuit \thinspace \clubsuit
♠ ♣      \spadesuit \thickspace \clubsuit
♠ ♣      \spadesuit \quad \clubsuit
♠ ♣      \spadesuit \qqquad \clubsuit

```

### 4.2.8 Delimitadores

Observe o seguinte exemplo,

$$f(x) = \left(\frac{1}{2}\right)^x \quad f(x) = (\frac{1}{2})^x$$

note que o delimitador não está englobando completamente a fração. Para corrigir este problema, basta usar os comandos `\left` e `\right` para obter,

$$f(x) = \left(\frac{1}{2}\right)^x \quad f(x) = \left(\frac{1}{2}\right)^x$$

por fim, apresentaremos outro exemplo,

$$\left[\sum_i a_i\right]^{1/p} \quad \left[\sum_i a_i\right]^{1/p}$$

observe que `\left` e `\right` foram os responsáveis por "esticar" o delimitador por uma certa quantidade que é variável, ou seja, depende de qual expressão o delimitador precisa abranger. Devido a isso, existem outros comandos que possuem a mesma função mas que "esticam" por uma quantidade fixa,

```

| | | | |

```

```

| \quad \big | \quad \Big | \quad \bigg | \quad \Bigg |

```

geralmente, são usados caso os delimitadores esticados por `\left` e `\right` ou são muito longos ou muito curtos como é o caso do exemplo anterior, assim:

$$\left[\sum_i a_i\right]^{1/p} \quad \left[\sum_i a_i\right]^{1/p}$$

#### Comparação

$$\left[\sum_{i} a_{i}\right]^{1/p} \quad \quad \quad \bigg[\sum_{i} a_{i}\bigg]^{1/p}$$


---

como podemos ver o resultado da direita certamente é melhor.

#### 4.2.9 Operadores e novos símbolos

Quando tentamos escrever no L<sup>A</sup>T<sub>E</sub>X o comando correspondente a  $\sin x$ , é possível pensar em `\sin x` como sendo uma resposta plausível, entretanto o que obteremos será *sinx*, para evitar esta situação escrevemos `\sin x`.

Além disso, dentre os operadores, um dos quais provavelmente será muito utilizado será o `\lim`, pois com ele podemos escrever expressões como esta,

$$\lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}, \quad \text{\lim_{h \to 0} \frac{f(x+h) - f(x)}{h}},$$

no estilo *inline* teríamos  $\lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$ . Observe que  $h \rightarrow 0$  estava debaixo de `\lim` no primeiro exemplo, mas agora está subscrito, podemos forçar esta subscrição no estilo *displayed* usando o comando `\nolimits`,

$$\lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h},$$

com o comando `\lim\nolimits_{h \to 0} \frac{f(x+h) - f(x)}{h}`.

Novos operadores matemáticos podem ser criados no pré-âmbulo, a partir do comando `\DeclareMathOperator` assim, se fizermos `\DeclareMathOperator {\mat}{Matema}` e no corpo do texto invocarmos o novo operador com `\mat A` obteremos *Matema A*. Caso o comando `\mat` já existisse uma mensagem de erro seria exibida.

O comando `\substack` provê limites multilineares para operadores grandes, por exemplo:

$$\sum_{\substack{i < n \\ i \text{ par}}} x_i^2 \quad \text{\sum_{ \substack{ i < n \\ i \text{ par} } } x_{i}^2}$$

com `\sideset` é possível escrever,

$$\prod_{\alpha}^{\beta} \prod_{\gamma}^{\delta} \prod_{\pi}^{\xi} \text{\sideset{}{\_}\alpha^{\beta}} \text{\sideset{}{\_}\gamma^{\delta}} \text{\prod_{\pi}^{\xi}}$$

e conseqüentemente, escrever expressões como a seguir,

$$\sum'_{\substack{n < k \\ n \text{ ímpar}}} n E_n$$

$$\text{\sideset{}{\_}\sum_{\substack{n < k \\ n \text{ ímpar}}} n E_n}$$

Além disso, temos `\overset` e `\underset` que são capazes de criar novos símbolos,

$$\underset{\max}{X} \stackrel{\text{def}}{=} \max X \quad \underset{\max}{X} \overset{\text{def}}{=} \max X$$

#### 4.2.10 Chaves e linhas horizontais

O comando `\overbrace` permite a seguinte construção,

$$\overbrace{x^2 + 2x + 1 - y^2} = (x + 1)^2 - y^2$$

$$\overbrace{x^2 + 2x + 1} - y^2 = (x + 1)^2 - y^2$$

mas talvez o fato mais interessante seja o fato de que podemos escrever textos matemáticos acima desta chave horizontal,

$$\overbrace{a_1 + \dots + a_n}^{\sum_{i=1}^n} + \sum_{i=n+1}^{2n} a_i = \sum_{i=1}^{2n} a_i$$

#### **Overbrace com marcador.**

$$\overbrace{a_1 + \dots + a_n}^{\sum_{i=1}^n} + \sum_{i=n+1}^{2n} a_i = \sum_{i=1}^{2n} a_i$$


---

para a chave horizontal está debaixo de uma expressão matemática basta usar o comando `\underbrace`,

$$\underbrace{a + \dots + a}_{n \text{ parcelas}} = na \quad \text{\underbrace{a+\dots+a}_{n} \text{\text{ parcelas}}}$$

temos ainda `\underline` e `\overline` para criar linhas horizontais,

$$\overline{\overline{X}} = \underline{X} \quad \text{\overline{\overline{X}}} = \underline{X}$$

#### 4.2.11 Fórmulas emolduradas

Com o comando `\boxed` é possível escrever fórmulas emolduradas da seguinte forma,

$$\boxed{\sin(a \pm b) = \sin a \cos b \pm \sin b \cos a}$$

$$\text{\boxed{ \sin (a\pm b) = \sin a\cos b \pm \sin b\cos a}}$$

#### 4.2.12 Função definida por partes

Para escrever uma função definida por partes no  $\text{\LaTeX}$  é preciso usar o ambiente `cases`,

$$|x| = \begin{cases} x, & \text{se } x \geq 0 \\ -x, & \text{se } x < 0 \end{cases}$$

da seguinte forma,

#### Função modular

```
\begin{equation*}
|x| =
\begin{cases}
x, & \text{\text{ se } } x \ge 0 \\
-x, & \text{\text{ se } } x < 0
\end{cases}
```

```
-x,&\text{ se } x < 0  
\end{cases}  
\end{equation*}
```

---

observe que o símbolo & é o responsável por alinhar o texto gerado pelo comandos `\text{ se } x \ge 0` e `\text{ se } x < 0`, além disso `\\` indica a quebra de linha.

### 4.2.13 Fórmulas muito longas

Podemos escrever fórmulas muito longas com o ambiente `multline`,

$$\begin{aligned}(x - y + z)^5 = & x^5 - 5x^4y + 5x^4z + 10x^3y^2 - 20x^3yz + 10x^3z^2 - 10x^2y^3 \\ & + 30x^2y^2z - 30x^2yz^2 + 10x^2z^3 + 5xy^4 - 20xy^3z + 30xy^2z^2 \\ & - 20xyz^3 + 5xz^4 - y^5 + 5y^4z - 10y^3z^2 + 10y^2z^3 - 5yz^4 + z^5\end{aligned}$$

#### Comando `multline`.

```
\begin{multline*}
(x-y+z)^{5} = x^{5} - 5x^{4}y + 5x^{4}z + 10x^{3}y^{2}
- 20x^{3}yz + 10x^{3}z^{2} - 10x^{2}y^{3} \\
+ 30x^{2}y^{2}z - 30x^{2}yz^{2} + 10x^{2}z^{3} + 5xy^{4}
- 20xy^{3}z + 30xy^{2}z^{2} \\
- 20xyz^{3} + 5xz^{4} - y^{5} + 5y^{4}z - 10y^{3}z^{2} 0
+ 10y^{2}z^{3} - 5yz^{4} + z^{5}
\end{multline*}
```

este comando faz com que o trecho da equação que está na primeira linha esteja alinhado à esquerda, centralizado para todas as linhas a seguir, exceto pela última que ficará alinhada a direita.

### 4.2.14 Matrizes

As matrizes podem ser escritas com o ambiente `pmatrix`,

$$A = \begin{pmatrix} -1 & 1 \\ 2 & 377 \end{pmatrix}$$

### Criação de uma matriz.

```
\begin{equation*}
  A =
  \begin{pmatrix}
    -1 & 1 \\
    2 & 377
  \end{pmatrix}
\end{equation*}
```

cabe salientar que o ambiente `matrix` entrega o mesmo resultado do que `pmatrix` mas sem nenhum delimitador, além disso `vmatrix` e `vmatrix` possuem os caracteres `— —` e `[ ]` como delimitadores, respectivamente, como é mostrado abaixo

$$\begin{array}{ccc}
 -1 & 1 & \left| \begin{array}{cc} -1 & 1 \\ 2 & 377 \end{array} \right| & \left[ \begin{array}{cc} -1 & 1 \\ 2 & 377 \end{array} \right] \\
 2 & 377 & & 
 \end{array}$$

podemos alinhar os números da primeira coluna dessa matriz com o comando `\phantom` responsável por inserir espaços vazios na horizontal,

$$A = \begin{pmatrix} -1 & 1 \\ 2 & 377 \end{pmatrix}$$

### Comando *phantom*.

```
\begin{equation*}
  A =
  \begin{pmatrix}
    -1 & 1 \\
    \phantom{-}2 & 377
  \end{pmatrix}
\end{equation*}
```

```
\end{pmatrix}
\end{equation*}
```

---

temos ainda o `\vphantom` responsável por inserir espaços vazios na vertical,

$$\begin{matrix} \overline{a} & \overline{b} & & \overline{a} & \overline{b} \\ 1 & 2 & & 1 & 2 \end{matrix}$$

### Comando `vphantom`.

```
\begin{equation*}
\begin{matrix}
\overline{a} & \overline{b} \\
1 & 2
\end{matrix}
\quad
\begin{matrix}
\overline{\vphantom{B}a} & \overline{b} \\
1 & 2
\end{matrix}
\end{matrix}
\end{equation*}
```

---



por fim, como prometido, usaremos os comandos `\hdots`, `\vdots` e `\ddots` para nos ajudar a construir uma matriz,

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} \end{pmatrix}$$

### Matriz genérica.

```
\begin{equation*}
  \begin{pmatrix}
    a_{11} & a_{12} & a_{13} & \hdots & a_{1n} \\
    a_{21} & a_{22} & a_{23} & \hdots & a_{2n} \\
    a_{31} & a_{32} & a_{33} & \hdots & a_{3n} \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    a_{m1} & a_{m2} & a_{m3} & \hdots & a_{mn}
  \end{pmatrix}
\end{equation*}
```

#### 4.2.15 Sistemas lineares

Um sistema linear pode ser facilmente construído com o comando `\cases`

$$\begin{cases} 3\alpha + 2\beta - 5\gamma = 6 \\ 7\alpha - 11\beta + 13\gamma = 5 \\ -17\alpha + 19\beta - 23\gamma = 4 \end{cases}$$

```
\begin{equation*}
  \begin{cases}
    \phantom{-1}3\alpha + \phantom{1}2 \beta \\
    - \phantom{1}5 \gamma = 6 \\
    \phantom{-1}7\alpha - 11\beta + 13 \gamma = 5 \\
    -17\alpha + 19\beta - 23 \gamma = 4
  \end{cases}
\end{equation*}
```

---

**4.2.16 Exercícios**

4.1

$$A = B \Leftrightarrow (A \subset B) \wedge (B \subset A)$$

4.2

$$\overline{A \cup B} = \overline{A} \cap \overline{B}$$

4.3

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

4.4

$$a_n = \begin{cases} a_1 = a & a \in \mathbb{R} \\ a_n = a_{n-1} + r & r \in \mathbb{R}, \forall n \geq 2 \end{cases}$$

4.5

$$\sum_{k=1}^n \binom{n}{k} = 2^n$$

4.6 Equação retirada em [5]:

$$\lim a_n = L := \forall \epsilon > 0, \exists n_0 \in \mathbb{N}; n > n_0 \Rightarrow |a_n - L| < \epsilon$$

4.7 Equação retirada em [5]:

$$\lim_{x \rightarrow a} f(x) = L := \forall \epsilon > 0, \exists \delta > 0; 0 < |x - a| < \delta \Rightarrow |f(x) - L| < \epsilon$$

4.8 Equação retirada em [2]:

$$\sum_{i=1}^{\lfloor \frac{n}{3} \rfloor} \left( \frac{x_{i,i+1}^{i^2}}{\binom{i+3}{3}} \right) \frac{\sqrt{\mu(i)^{\frac{3}{2}}(i^2 - 1)}}{\sqrt[3]{\rho(i) - 2} + \sqrt[3]{\rho(i) - 1}}$$

4.9 Equação retirada em [2]:

$$\det \mathbf{K}(t = 1, t_1, \dots, t_n) = \sum_{I \in \mathbf{n}} (-1)^{|I|} \prod_{i \in I} t_i \prod_{j \in I} (D_j + \lambda_j t_j) \det \mathbf{A}^\lambda(\bar{I}|\bar{I}) = 0$$

4.10 Equação retirada em [2]:

$$\int_{\mathcal{D}} |\overline{\partial u}|^2 \Phi_0(z) e^{\alpha|z|^2} \geq \int_{\mathcal{D}} |u|^2 \Phi_0 e^{\alpha|z|^2} + c_5 \delta^{-2} \int_{\mathcal{A}} |u|^2 \Phi_0 e^{\alpha|z|^2}$$

4.11 Equação retirada em [2]:

$$\Theta_i = \bigcup (\Theta(\overline{a \wedge b}, \overline{a \wedge b} \mid a, b \in B_i) \vee \bigcup (\Theta(\overline{a \vee b}, \overline{a \vee b} \mid a, b \in B_i)$$

4.12 Equação retirada em [6]:

$$-\frac{\hbar^2}{2m} \frac{\partial^2 \Psi(x, t)}{\partial x^2} + V(x, t) \Psi(x, t) = i\hbar \frac{\partial \Psi(x, t)}{\partial t}$$

4.13 Equação retirada em [7]:

$$\frac{D\mathbf{V}}{Dt} = \frac{\partial \mathbf{V}}{\partial t} + \nabla \mathbf{V} \cdot \mathbf{V} = -\frac{1}{\rho} \nabla p + \mathbf{G} + \nu \nabla^2 \mathbf{V} + \frac{1}{3} \nu \nabla(\nabla \cdot \mathbf{V})$$

4.14

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}$$

4.15 Equação retirada em [4]:

$$S = \int_{\alpha}^{\beta} 2\pi y \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2} dt$$

4.16 Equação retirada em [4]:

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x - a)^n$$

4.17

$$\begin{cases} 3x + 7y + 4z = 1 \\ 2x + 8y + 6z = 2 \\ 5x + 9y + z = 3 \end{cases} \Rightarrow \begin{bmatrix} 7 & 7 & 4 \\ 2 & 8 & 6 \\ 5 & 9 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

## Capítulo 5

# Figuras, Tabelas e Diagramas

Neste capítulo, será discutido sobre como inserir e formatar figuras, tabelas e diagramas no  $\text{\LaTeX}$ . Existem diversas preferências a respeito de formatação e *layout* relacionados a esses elementos, e vamos discutir os mais básicos a fim de trazer uma perspectiva geral para o leitor.

### 5.1 Figuras

Para a maioria dos tipos de arquivo científicos, a inserção de figuras é essencial. O  $\text{\LaTeX}$  oferece diferentes ferramentas que podem ser utilizadas para inserir figuras e configurá-las para se obter o que precisa.

#### 5.1.1 Inserção de figuras

Para inserir figuras, devemos utilizar o pacote `\usepackage{graphicx}` no preâmbulo do documento.

O ambiente utilizado para inserir figuras é o ambiente `figure`, delimitado por `\begin{figure}` e `\end{figure}`. Este comando deve ser inserido no corpo do texto e delimita o trecho do código onde a figura deve ser inserida juntamente a outros comandos. Além disso, possibilita legendar uma figura produzida no próprio  $\text{\LaTeX}$  ou por outro programa e flutuá-la de forma que a perda de espaço seja mínima.

O comando `\includegraphics[escala]{nome da figura}` será utilizado na inserção da figura. Este comando deve ser inserido entre o `\begin{figure}` e `\end{figure}`, onde o parâmetro `[escala]` entre colchetes indica a porcentagem de escala, por exemplo, escala igual a 1 implica que a figura terá 100% das suas dimensões originais, escala igual a 0.5 implica em reduzir a imagem pela metade. Já o parâmetro entre chaves `{nome da figura}` deve conter a localização da figura. Para que a figura seja centralizada, utilizamos o comando `\centering` antes de incluí-la.

Para adicionar uma legenda à figura, deve-se utilizar o comando `\caption{legenda}` logo após inserir a figura, antes de inserir `\end{figure}`.

### Exemplo

```
\usepackage{graphicx}

\begin{document}

\noindent Veja na figura a seguir uma superfície determinada
pela função  $f(x,y)=\frac{x^2}{5}+\frac{y^2}{5}-4$ 

\begin{figure}[h]
\centering
\includegraphics[scale=0.22]{grafico_tridimensional.png}
\captionsetup{font=small, justification=centering}
\caption{Gráfico da função  $f(x,y)$ .}
\end{figure}

\end{document}
```

### Resultado

Veja na figura a seguir uma superfície determinada pela função  $f(x, y) = \frac{x^2}{5} + \frac{y^2}{5} - 4$

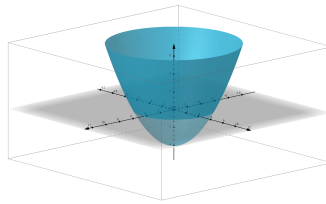


Figura 5.1: Gráfico da função  $f(x, y)$ .

### 5.1.2 Posicionamento e dimensionamento de figuras

O ambiente `figure` também permite que possamos reposicionar a figura baseado em alguns comandos. Utilizando `\begin{figure}[onde]`, o argumento “onde” se refere ao local onde deve ser colocada a figura, podendo ser uma combinação das seguintes letras:

- **h (here)** - indicará que a figura deve ser inserida exatamente naquele local;
- **t (top)** - indicará que a figura deve ser inserida no topo da próxima página;
- **b (bottom)** - indicará que a figura deve ser inserida na parte inferior da página;
- **p (page)** - indicará que a figura deve ser inserida em uma página separada;
- **!** - indicará que devem ser ignorados alguns parâmetros internos e reforça o posicionamento desejado;
- **H** - Caso o **h** não consiga posicionar uma figura no local desejado nem mesmo com o uso do **!**, deve-se utilizar o **H**. Este comando exige o pacote `\usepackage{float}` no preâmbulo.

Para inclinar uma imagem devemos adicionar o parâmetro *angle* ao lado do comando *scale* na linha de código `\includegraphics`. O parâmetro *angle* =  $\theta$  gira a imagem em  $\theta$  graus no sentido anti-horário. Para girar a imagem no sentido horário, deve-se utilizar um número negativo. O comando que deve ser utilizado é:

$$\includegraphics[*scale*, *angle*]{nome da imagem}.$$

Ao invés de trabalhar apenas alterando a escala das figuras, podemos redimensioná-las de outras formas, utilizando como parâmetros medidas fixas de largura e/ou altura para a figura. O comando que deve ser utilizado é:

$$\includegraphics[width=x cm, height=y cm]{nome da imagem},$$

onde *width* indica o comprimento e *height* a altura da figura. Se apenas um dos parâmetros *width* ou *height* for inserido, o outro será dimensionado para manter a proporção.

### 5.1.3 Subfiguras

Muitas vezes, é necessário incluir várias figuras relacionadas em um mesmo ambiente, como gráficos lado a lado ou imagens que devem ser agrupadas para comparação. A inserção de subfiguras pode ser realizada usando o pacote `subcaption`, que oferece recursos avançados para criar subfiguras de maneira fácil e personalizável.

O pacote `subcaption` permite criar ambientes de subfiguras dentro de um ambiente `figure`, permitindo controlar o layout, a legenda e a numeração das subfiguras independentemente. Aqui estão alguns parâmetros comuns usados no ambiente de subfiguras:

- `width`: Define a largura da subfigura. Pode ser especificado em unidades como `cm`.
- `caption`: Define a legenda da subfigura.
- `label`: Define uma etiqueta para referência cruzada.

#### Exemplo

```
\usepackage{graphicx}
\usepackage{subcaption} % Pacote para subfiguras

\begin{document}

\begin{figure}[H]
  \centering

  \begin{subfigure}[b]{0.4\textwidth}
    \centering
    \includegraphics[width=\textwidth]{grafico1.png}
    \caption{Gráfico 1.}
  \end{subfigure}

```



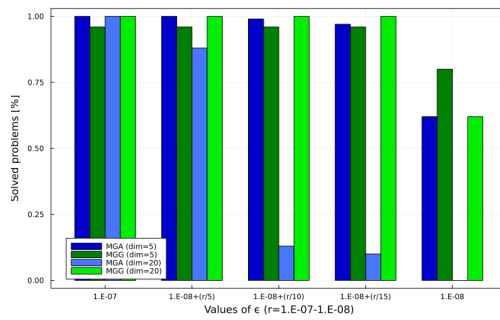
```
\hfill
\begin{subfigure}[b]{0.4\textwidth}
  \centering
  \includegraphics[width=\textwidth]{grafico2.png}
  \caption{Gr\'afico 2.}
\end{subfigure}
\begin{subfigure}[b]{0.4\textwidth}
  \centering
  \includegraphics[width=\textwidth]{grafico3.png}
  \caption{Gr\'afico 3.}
\end{subfigure}

\caption{An\'alise de gr\'aficos}
\label{fig:subcaption_example}
\end{figure}

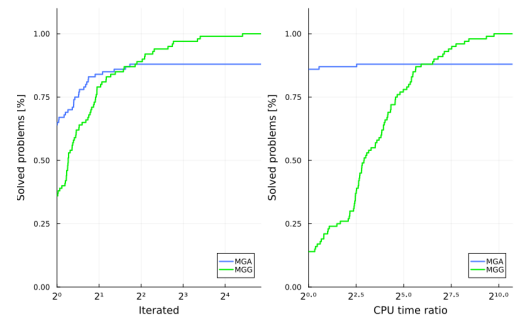
\end{document}
```

---

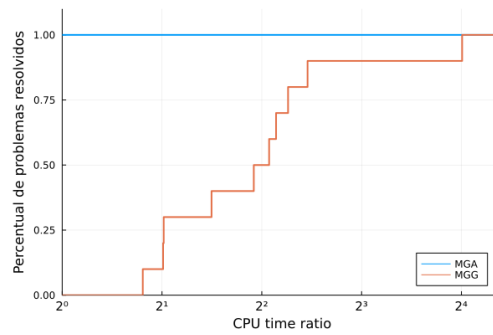
**Resultado**



(a) Gráfico 1.



(b) Gráfico 2.



(c) Gráfico 3.

Figura 5.2: Análise de gráficos

## 5.2 Tabelas

Tabelas e quadros são elementos essenciais em muitos documentos, pois permitem a organização e apresentação de informações de maneira estruturada. O LaTeX oferece um ambiente poderoso para criar e personalizar tabelas de acordo com suas necessidades.

### 5.2.1 Criação Básica de Tabelas

Para a criação de tabelas, deve-se usar o ambiente `table`. Este ambiente delimita a região do código onde serão inseridas as configurações para a tabela.

Para inserir a tabela, deve-se usar o ambiente `tabular` internamente ao ambiente `table`. Este comando delimita a tabela, para determinar o parâmetro de número de colunas.

Para o alinhamento do conteúdo de cada coluna, utilizamos as letras, **r** (**right**), **l** (**left**) e **c** (**center**) onde **r** alinha o conteúdo de uma coluna à direita, **l** à esquerda e **c** centraliza.

Para determinarmos a quantidade de colunas, basta adicionar um desses parâmetros para cada coluna e o número de letras de alinhamento será igual a número de colunas.

### Exemplo

```
\begin{table}[H]
  \centering
  \begin{tabular}{c c}
    célula 1 & célula 2 \\
    célula 3 & célula 4
  \end{tabular}
\end{table}
```

### Resultado

célula 1	célula 2
célula 3	célula 4

## 5.2.2 Formatação de Tabelas

A formatação de tabelas é muito importante para melhorar a legibilidade e a estética. Abaixo estão alguns dos principais comandos de formatação de tabelas.

- **Linhas Horizontais:** Use `\hline` para criar linhas horizontais.
- **Colunas Verticais:** Use `|` entre as letras do argumento do ambiente `tabular` para criar linhas verticais.

- **Espaçamento:** Use `>{\centering\arraybackslash}p{largura}` para definir espaçamento entre colunas.

### Exemplo

```

\begin{table}
\centering
\begin{tabular}{|>{\centering\arraybackslash}p{2.5cm}|
>{\centering\arraybackslash}p{3cm}|
>{\centering\arraybackslash}p{3.5cm}|}
\hline
\textbf{Item} & \textbf{Quantidade} & \textbf{Preço (R\$)} \\ \hline
Maçãs & 10 & 2.50 \\ \hline
Laranjas & 8 & 3.00 \\ \hline
Uvas & 5 & 4.75 \\ \hline
\end{tabular}
\end{table}

```

### Resultado

Item	Quantidade	Preço (R\$)
Maçãs	10	2.50
Laranjas	8	3.00
Uvas	5	4.75

### 5.2.3 Mesclagem de Células

A mesclagem de células pode ser realizada com o pacote `multirow` para mesclagem vertical e `multicolumn` para mesclagem horizontal.

#### Exemplo

```

\begin{table}
\centering
\begin{tabular}{|c|c|c|}
\hline
\multicolumn{2}{|c|}{\textbf{Grupo 1}}
& \multirow{2}{*}{\textbf{Grupo 2}} \\
\cline{1-2}
\textbf{Célula 1} & \textbf{Célula 2} & \\
\hline
Dado 1 & Dado 2 & Dado 3 \\
\hline
\multirow{2}{*}{\textbf{Células mescladas}}
& \multicolumn{2}{c}{Dado 4} \\
& \multicolumn{2}{c}{Dado 5} \\
\hline
\end{tabular}
\end{table}

```

#### Resultado

Grupo 1		Grupo 2
Célula 1	Célula 2	
Dado 1	Dado 2	Dado 3
Células mescladas	Dado 4 Dado 5	

### 5.2.4 Aparência das tabelas

Vários elementos da tabela podem ser modificados para obter um documento de boa aparência. Pode-se modificar, por exemplo, a espessura, a cor da linha e a cor do plano de fundo das células de uma tabela.

É comum usar duas cores de forma alternada em tabelas para melhorar a legibilidade. Isto pode ser feito em  $\LaTeX$  com o pacote `xcolor`, inserindo no preâmbulo o comando `\usepackage[pacote de cores]{xcolor}`. Para inserir as cores, deve-se inserir o comando

```
\rowcolors{\varphi}{cor1!80!cor2!50...}{cor1!70!cor2!40...}
```

antes do ambiente `tabular`. Esse comando leva três parâmetros, cada um dentro de chaves, sendo o primeiro a linha para começar a colorização, o segundo a cor das linhas ímpares, o terceiro a cor das linhas pares. As cores podem ser misturadas da seguinte forma: o nome da cor seguido de exclamação (exemplo: `blue!`) e a porcentagem que define sua intensidade em um valor numérico de 0 a 100 seguido de exclamação (exemplo: `90!`).

Utilizando o comando `\arrayrulecolor`, é possível colorir as linhas de contorno das tabelas. Este comando deve ser inserido no preâmbulo e as cores devem ser inseridas da mesma forma que no pacote `xcolor`.

Podemos também colorir células específicas em uma tabela facilmente através do comando `\cellcolor{cores}`. As mesmas observações sobre a seleção de cores mencionadas nos comandos anteriores são válidas para este. Este comando deve ser inserido diretamente na célula que se quer colorir.

Muitas vezes, a criação e personalização de tabelas no  $\LaTeX$  podem ser trabalhosas. O site <https://www.tablesgenerator.com> é uma alternativa para contornar essa dificuldade, pois permite a criação de tabelas de forma automatizada e gera o código em  $\LaTeX$ .

## 5.3 Diagramas de Venn

Os diagramas de Venn são uma ferramenta visual poderosa para representar relações entre conjuntos. No  $\LaTeX$ , podemos criar esses diagramas usando o pacote `venndiagram`, que nos permite criar diagramas de Venn de maneira simples e eficaz.

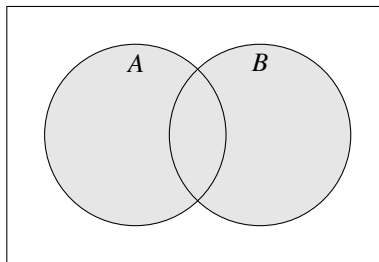
Primeiramente, o comando `\usepackage{venndiagram}` deve ser inserido no preâmbulo.

Para criar um diagrama de Venn com 2 conjuntos, pode-se usar o ambiente `venndiagram2sets`. Deve-se usar os comandos `\fillA`, `\fillB` e `\fillAB` para preencher as regiões correspondentes aos conjuntos  $A$ ,  $B$  e à interseção  $AB$ , respectivamente.

#### Exemplo

```
\begin{venndiagram2sets}
  \fillA \fillB
\end{venndiagram2sets}
```

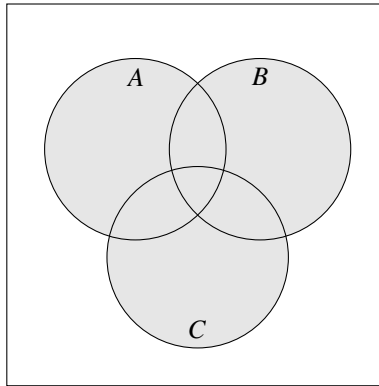
#### Resultado



Para um diagrama com 3 conjuntos, pode-se utilizar o ambiente `venndiagram3sets` e seguir a mesma estrutura.

#### Exemplo

```
\begin{venndiagram3sets}
  \fillA \fillB \fillC
\end{venndiagram3sets}
```

**Resultado**

O pacote **venndiagram** oferece várias opções de personalização, incluindo:

- **labelA, labelB, labelC**: Para definir rótulos para os conjuntos.
- **labelOnlyA, labelOnlyB, labelOnlyC**: Para definir rótulos para regiões específicas.
- **labelOnlyAB, labelOnlyAC, labelOnlyBC, labelOnlyABC**: Para rótulos em interseções específicas.

## 5.4 Diagramas Comutativos e representação de Grafos

### 5.4.1 Diagramas Comutativos

Um diagrama comutativo é uma representação gráfica usada em matemática e outras disciplinas para ilustrar relações entre objetos e suas transformações de maneira que o resultado seja independente do caminho escolhido. Em  $\LaTeX$ , é possível criar diagramas comutativos de forma elegante usando pacotes específicos, como **tikz-cd**.

Para utilizar o pacote **tikz-cd**, você deve inserir no seu preâmbulo o comando `\usepackage{tikz-cd}`. O ambiente que será utilizado é o **tikzcd**. A seguir, mostraremos um exemplo simples.

**Exemplo**

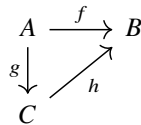
```
\begin{tikzcd}
A \arrow{r}{f} \arrow{d}[swap]{g} & B \end{tikzcd}
```



```
C \arrow{ur}[swap]{h}
\end{tikzcd}
```

---

### Resultado



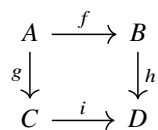
Neste exemplo, temos 3 objetos:  $A$ ,  $B$ ,  $C$  e setas (flechas) que representam as transformações entre eles. O comando `\arrow` é usado para criar flechas e as letras dentro delas representam os nomes das transformações. A seguir, é apresentado um exemplo com 4 objetos:

### Exemplo

```
\begin{tikzcd}
A \arrow{r}{f} \arrow[swap]{d}{g} & B \arrow{d}{h} \\
C \arrow{r}{i} & D
\end{tikzcd}
```

---

### Resultado



### 5.4.2 Grafos

Os grafos são uma ferramenta fundamental em matemática e ciência da computação para representar relações entre objetos. Em  $\text{\LaTeX}$ , é possível criar representações gráficas de grafos usando o pacote **tikz**, que oferece uma ampla gama de funcionalidades para desenhar grafos personalizados.

Primeiramente, deve-se inserir no preâmbulo o comando `\usepackage{tikz}`.

Aqui estão alguns conceitos e comandos básicos para a construção de grafos usando o **tikz**:

- **Definindo um ambiente de grafo:** Para criar um ambiente para o seu grafo, use o ambiente **tikzpicture**;
- **Nós (Vértices):** Para adicionar nós ao seu grafo, use o comando `\node` seguido das coordenadas do nó e do nome do nó;
- **Arestas:** Para criar arestas entre os nós, use o comando `\draw` seguido das coordenadas dos nós de origem e destino;

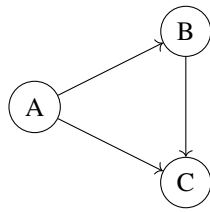
#### Exemplo

```
\begin{tikzpicture}
  \node[draw, circle] (A) at (0,0) {A};
  \node[draw, circle] (B) at (2,1) {B};
  \node[draw, circle] (C) at (2,-1) {C};

  \draw[->] (A) -- (B);
  \draw[->] (A) -- (C);
  \draw[->] (B) -- (C);
\end{tikzpicture}
```

---

#### Resultado



---

Abaixo segue um exemplo de um grafo com 4 nós:

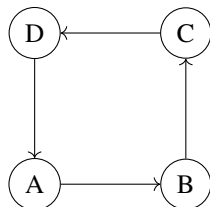
### Exemplo

```
\begin{tikzpicture}
  \node[draw, circle] (A) at (0, 0) {A};
  \node[draw, circle] (B) at (2, 0) {B};
  \node[draw, circle] (C) at (2, 2) {C};
  \node[draw, circle] (D) at (0, 2) {D};

  \draw[->] (A) -- (B);
  \draw[->] (B) -- (C);
  \draw[->] (C) -- (D);
  \draw[->] (D) -- (A);
\end{tikzpicture}
```

---

### Resultado



O **tikz** permite que você crie grafos mais complexos com diferentes estilos de arestas, cores, formas e tamanhos de nós. Você pode personalizar ainda mais o seu grafo para atender às suas necessidades específicas.

### 5.4.3 Exercícios

#### 5.1

Tabela 5.1: Minha Tabela Exemplar

Cabeçalho 1	Cabeçalho 2	Cabeçalho 3	Cabeçalho 4	Cabeçalho 5
Dado 1,1	Dado 1,2	Dado 1,3	Dado 1,4	Dado 1,5
Dado 2,1	Dado 2,2	Dado 2,3	Dado 2,4	Dado 2,5
Dado 3,1	Dado 3,2	Dado 3,3	Dado 3,4	Dado 3,5
Dado 4,1	Dado 4,2	Dado 4,3	Dado 4,4	Dado 4,5
Dado 5,1	Dado 5,2	Dado 5,3	Dado 5,4	Dado 5,5

#### 5.2

Tabela 5.2: Tabela com Mesclagem de Células

Mesclado 2x1		Cabeçalho 3	Mesclado 1x2	
Cabeçalho 1	Cabeçalho 2	Mesclado 2x1	Cabeçalho 4	Cabeçalho 5
Dado 1,1	Dado 1,2		Dado 1,4	Dado 1,5
Dado 2,1	Dado 2,2	Dado 2,3	Dado 2,4	Dado 2,5
Dado 3,1	Dado 3,2	Dado 3,3	Dado 3,4	Dado 3,5

## Capítulo 6

# Bibliografias e Citações

Neste capítulo, exploraremos a elaboração de elementos importantes para documentos acadêmicos, incluindo a criação de sumários, índices remissivos e outros recursos relevantes.

### 6.1 Construção de Sumário

O  $\LaTeX$  é projetado para simplificar a criação de documentos estruturados, e isso inclui a geração automática de sumários. Cada vez que é criada uma seção, subseção, subsubseção e assim por diante, o  $\LaTeX$  rastreia essas estruturas para criar uma hierarquia organizada que será usada para gerar o sumário.

Para criar um sumário, deve-se incluir o comando `\tableofcontents` no local onde é desejado que o sumário apareça (normalmente após o `\begin{document}`, o título e o resumo). Esse comando é responsável por gerar o sumário baseado nas seções e subseções do documento.

Por padrão, o  $\LaTeX$  inclui todas as seções, subseções e subsubseções no sumário. Entretanto, é possível personalizar o nível de profundidade exibido no sumário usando o comando `\setcounter{tocdepth}` seguido pelo valor apropriado.

Os valores permitidos são de 0 a 3:

- 0: Capítulos;
- 1: Capítulos e seções;
- 2: Capítulos, seções e subseções;
- 3: Capítulos, seções, subseções e subsubseções.

Por exemplo, para incluir apenas capítulos e seções no sumário, é utilizado o seguinte comando:

#### Construção de Sumário

```

\begin{document}
\setcounter{tocdepth}{1}
\tableofcontents % Gera o sumário
\section{Introdução}
Esta é a introdução do documento.
\subsection{Objetivos}
Nesta subseção, discutimos os objetivos.
\subsubsection{Objetivos específicos}
Nesta subsubseção, discutimos os objetivos específicos.
\end{document}

```

---

Para definir o sumário em português, pode-se usar o pacote `babel`, que é um pacote de internacionalização que permite configurar diversos aspectos do documento, incluindo os títulos como "Sumário", "Capítulo", entre outros. Para isso, basta usar `\usepackage[brazilian]{babel}` ou `\usepackage[portuguese]{babel}`.

## 6.2 Construção de Índice Remissivo

No  $\LaTeX$ , o pacote `makeidx` é usado para criar índices remissivos de forma automática. Para ativá-lo, deve-se incluir a seguinte linha no preâmbulo do documento:

### Construção de Índice Remissivo

```

\usepackage{makeidx}
\makeindex

```

---

O primeiro comando importa o pacote `makeidx`, enquanto o segundo comando, `\makeindex`, instrui o  $\LaTeX$  a criar o índice remissivo. Para marcar um termo ou palavra para inclusão no índice, basta usar o comando `\index{termo}`, durante o texto que o termo se encontra. Por exemplo:

### Inclusão de um termo no índice remissivo

O `\index{LaTeX}` é um sistema de formatação de documentos ...

---

Após marcar os termos relevantes com o comando `\index`, é necessário compilar o documento com uma compilação adicional para que o índice seja gerado.

Os termos marcados são coletados durante a compilação e organizados em ordem alfabética. Além disso, os números das páginas em que os termos aparecem são registrados. O índice é então inserido no documento no local onde foi colocado o comando `\printindex`. Esta ferramenta é muito importante em textos acadêmicos, pois facilita a busca rápida de informações relevantes no documento.

## 6.3 Construção de Lista de Figuras e Tabelas

No  $\text{\LaTeX}$ , é possível gerar automaticamente listas de figuras e tabelas no início do documento, o que torna a navegação do leitor mais fácil e organizada. Para fazer isso, pode-se usar os comandos `\listoffigures` e `\listoftables`.

Antes de tudo, deve-se usar o ambiente `figure` para inserir figuras e o ambiente `table` para inserir tabelas, como foi visto no capítulo anterior. Além disso, dentro desses ambientes, é importante utilizar o comando `\caption{}` para adicionar uma legenda descritiva, pois ela aparecerá na lista criada.

No início do documento, depois do `\tableofcontents` (se houver um sumário), insira os comandos `\listoffigures` e `\listoftables`. Isso fará com que o  $\text{\LaTeX}$  gere automaticamente as listas de figuras e tabelas.

### Construção de lista de figuras e de tabelas

```
\tableofcontents
```

```
\listoffigures
```

```
\listoftables
```

---

As legendas usadas com o comando `\caption` serão automaticamente incluídas nas listas de figuras e tabelas. O  $\LaTeX$  coleta essas legendas e as exibe nas listas correspondentes no início do documento. As listas mostrarão os números das figuras e tabelas, seguidos pelos seus respectivos títulos.

## 6.4 Legendas e Referências cruzadas

O  $\LaTeX$  utiliza os ambientes `figure` e `table` para inserir figuras e tabelas, respectivamente. Esses ambientes são “flutuantes”, o que significa que o  $\LaTeX$  decide onde posicioná-los da maneira mais esteticamente agradável, levando em consideração a formatação geral do documento. Dentro dos ambientes `figure` e `table`, o comando `\caption` é usado para adicionar legendas descritivas às figuras e tabelas. Uma legenda deve ser clara e concisa, descrevendo o conteúdo de maneira informativa.

O comando `\label` é usado para criar rótulos para figuras e tabelas. Esses rótulos são únicos e permitem criar referências cruzadas para esses elementos em diferentes partes do documento. Veja um exemplo abaixo:

### Como rotular uma figura

```
\begin{figure}[opção]
\includegraphics[opção]{nome.jpg}
\caption{Legenda}
\label{Rótulo}
\end{figure}
```

---

Para fazer referência a uma figura ou tabela em outras partes do documento, pode-se usar o comando `\ref`. O  $\LaTeX$  substituirá automaticamente o número da figura ou tabela correspondente no local da referência. Por exemplo:

### Referência cruzada com figura

A Figura `\ref{Rótulo}` evidencia que...

---



## 6.5 Sistemas de Gerenciamento de Referências

Os sistemas de gerenciamento de referências, como BibTeX e BibLaTeX, são ferramentas utilizadas para organizar e citar referências bibliográficas em documentos  $\LaTeX$ . Eles simplificam o processo de gerenciamento, formatação e inclusão de citações.

O BibTeX é um sistema tradicional que utiliza arquivos `.bib` para armazenar informações bibliográficas. Cada entrada no arquivo `.bib` descreve um item bibliográfico, como um livro, um artigo, entre outros. Essas entradas têm campos específicos, como autor, título, ano, editora, etc.

O BibLaTeX é uma evolução do BibTeX, oferecendo capacidades mais avançadas de personalização. Ele é mais flexível na formatação de estilos de citação e é totalmente integrado ao  $\LaTeX$ . Ele permite estilos de citação altamente personalizados, tornando-o ideal para documentos que seguem diferentes convenções de estilo.

Para criar um banco de dados bibliográfico no formato `.bib`, pode-se criar um arquivo de texto simples com extensão `.bib`. Cada entrada começa com um tipo (por exemplo, `@book`, `@article`, `@inproceedings`) seguido por campos como `author`, `title`, `year`, etc. Veja exemplos a seguir de entradas `.bib`:

### Exemplo de citação para livro

```
@book{exemploLivro,  
  author = {Autor},  
  title = {Título do livro},  
  year = {ano de publicação},  
  publisher = {Editora},  
}
```

---

### Exemplo de citação para artigo

```
@article{exemploArtigo,  
  author = {Autor},  
  title = {Título do artigo},
```

```
year = {ano de publicação},  
publisher = {Editora}  
}
```

---

Para inserir a seção de referências utilizando essas ferramentas, basta usar o seguinte comando:

### Criação de Referências

```
\bibliography{Nome} %Nome do arquivo com extensão .bib  
\bibliographystyle{abbrv}
```

---

Vale ressaltar que o argumento `abbrv` utilizado no comando `\bibliographystyle{}` é um estilo de referência, mas é possível utilizar outros estilos. No BibTeX, existem quatro estilos de bibliografia que são comumente utilizados, são eles: `abbrv`, `alpha`, `unsrt` e `plain`. O estilo `abbrv` abrevia os nomes dos autores e títulos dos periódicos. O estilo `alpha` organiza as referências em ordem alfabética. O estilo `unsrt` lista as referências na ordem em que são citadas no texto, sem classificação por ordem alfabética. O estilo `plain` lista as referências na ordem em que são citadas no texto e a entrada da bibliografia pode ser referenciada como [1] ou [ABC20], dependendo das configurações.

Usando BibTeX ou BibLaTeX, as citações são incorporadas no texto por meio de comandos específicos. No caso do BibTeX, o comando `\cite` é amplamente utilizado. Veja um exemplo abaixo de como inserir a citação no texto:

### Citação no texto

```
Este resultado pode ser visto em \cite{exemploLivro}.
```

---

Para o BibLaTeX, existem comandos adicionais para maior flexibilidade:

- `\parencite{}`: Citação entre parênteses (muito utilizado);
- `\textcite{}`: Citação no formato autor-data;
- `\footcite{}`: Citação em nota de rodapé.

## Referências

1. Broy, M.: Software engineering — from auxiliary to key technologies. In: Broy, M., Dener, E. (eds.) *Software Pioneers*, pp. 10-13. Springer, Heidelberg (2002)
2. Grätzer, G. (2007). *More Math into*. New York: Springer.
3. Downes, M. (2002). *Short math guide for LATEX*. American Mathematical Society.
4. Stewart, J. (2011). *Calculus*. Cengage Learning.
5. Lima, E. L. (2004). *Análise real (Vol. 1)*. Rio de Janeiro: Impa.
6. F. Silva. (2020, October 7). A equação de Schrodinger: Uma mecânica ondulatória. Retrieved September 10, 2023, from [https://edisciplinas.usp.br/pluginfile.php/5744597/mod\\_resource/content/1/Aula%206%20A%20equa%C3%A7%C3%A3o%20de%20Schrodinger.pdf](https://edisciplinas.usp.br/pluginfile.php/5744597/mod_resource/content/1/Aula%206%20A%20equa%C3%A7%C3%A3o%20de%20Schrodinger.pdf)
7. Baliño. (n.d.). Equação de Navier-Stokes. Retrieved September 14, 2023, from [https://edisciplinas.usp.br/pluginfile.php/2967734/mod\\_resource/content/1/Navier\\_Stokes.pdf](https://edisciplinas.usp.br/pluginfile.php/2967734/mod_resource/content/1/Navier_Stokes.pdf)