



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE CIÊNCIAS EXATAS E DA TERRA
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO
MESTRADO ACADÊMICO EM SISTEMAS E COMPUTAÇÃO



Manual do Modelo PPgSC de Dissertações e Teses em \LaTeX

Bruno Motta de Carvalho

Natal-RN

Maio de 2021

Bruno Motta de Carvalho

Manual do Modelo PPgSC de Dissertações e Teses
em \LaTeX

Qualificação de Mestrado apresentada ao Programa de Pós-Graduação em Sistemas e Computação do Departamento de Informática e Matemática Aplicada da Universidade Federal do Rio Grande do Norte como requisito parcial para a obtenção do grau de Mestre em Sistemas e Computação.

Linha de pesquisa:

Processamento Gráfico e Inteligência Computacional

Orientador

Hari Seldon

Coorientador

Salvor Hardin

PPgSC – PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO

DIMAP – DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA

CCET – CENTRO DE CIÊNCIAS EXATAS E DA TERRA

UFRN – UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

Natal-RN

Maio de 2021

Qualificação de Mestrado sob o título *Manual do Modelo PPgSC de Dissertações e Teses em L^AT_EX* apresentada por Bruno Motta de Carvalho e aceita pelo Programa de Pós-Graduação em Sistemas e Computação do Departamento de Informática e Matemática Aplicada da Universidade Federal do Rio Grande do Norte, sendo aprovada por todos os membros da banca examinadora abaixo especificada:

Prof. Dr. Hari Seldon
Orientador
Departamento de Informática e Matemática Aplicada
Universidade Federal do Rio Grande do Norte

Prof. Dr. Salvor Hardin
Coorientador
Departamento de Informática e Matemática Aplicada
Universidade Federal do Rio Grande do Norte

Prof. Dr. Gaal Dornick
Departamento de Informática e Matemática Aplicada
Universidade Federal do Rio Grande do Norte

Prof. Dr. Kwisatz Haderach
Departamento de Presciência
Universidade de Caladan

Natal-RN, 6 de Maio de 2021.

Dedicatória

Dedico este trabalho a todos os apreciadores de \LaTeX do PPgSC.

Agradecimentos

Agradeço a todos os autores de pacotes \TeX e \LaTeX . Sem sua dedicação e engenhosidade, não teríamos essas maravilhosas ferramentas gratuitas de diagramação de texto.

Agradeço ainda aos professores João Marcos de Almeida, Marcel Vinícius Medeiros Oliveira, Monica Magalhães Pereira e ao aluno Vitor Rodrigues Greati pelas sugestões ofertadas.

Epígrafe

Those who believe in telekinetics, raise my hand.

Kurt Vonnegut Jr.

Manual do Modelo PPgSC de Dissertações e Teses em \LaTeX

Autor: Bruno Motta de Carvalho

Orientador: Hari Seldon

Coorientador: Salvor Hardin

Resumo

Este documento descreve o modelo de dissertações e teses do [Programa de Pós-graduação em Sistemas e Computação \(PPgSC\)](#) da [Universidade Federal do Rio Grande do Norte \(UFRN\)](#), sua organização e as razões que embasaram algumas escolhas. O principal objetivo deste documento é facilitar a escrita dos documentos por parte dos alunos, e homogeneizar a diagramação dos mesmos. Ele não se propõe a ser um guia inicial para quem não conhece \LaTeX , mas indica referências que podem ser utilizadas tanto para usuários novatos como para os que possuem larga experiência em diagramação usando \LaTeX . A maioria dos pacotes incluídos neste modelo são brevemente descritos e alguns exemplos são incluídos juntamente com os códigos \LaTeX que os produzem. Este modelo ficará disponível com os arquivos fonte, assim os usuários terão acesso ao código \LaTeX usado para gerar todos os exemplos, como figuras, tabelas, algoritmos e códigos. Sugiro que você identifique os pacotes que você deve utilizar e comente a inclusão do restante, de modo a diminuir o tempo de processamento do documento, e inclua-os a medida que os necessite.

Palavras-chave: L^AT_EX, PPgSC, UFRN.

L^AT_EX PPgSC Model for Dissertations and Theses

Author: Bruno Motta de Carvalho

Advisor: Hari Seldon

Co-advisor: Salvor Hardin

Abstract

This document describes the dissertations and theses of [UFRN's PPgSC](#), its structure and the reasons that supported some of the choices made. The main goal of this document is to facilitate the writing of documents by the students, and to homogenize their typesetting. This document does not aim to be an introductory guide for those who do not know L^AT_EX, but it lists references that can be used for those who are novices to the task of writing documents in L^AT_EX as well as for those who are experienced. The majority of the packets used in the model are briefly described and some examples are included in this document, as well as some examples with the L^AT_EX code which produces them. This model will be available together with the L^AT_EX source code used to produce all the figures, tables, algorithms and listings. I propose that you identify the packets you will need and comment out the instructions that load the ones you will not need, thus, reducing the document's processing time, and include packets individually as you detect the need for them.

Keywords: L^AT_EX, [PPgSC](#), [UFRN](#).

Lista de Figuras

1.1	Estrutura de arquivos do modelo PPgSC. Essa figura foi gerada usando TikZ (ver Capítulo 5) e os símbolos da fonte <code>fontawesome</code>	8
3.1	Tirinha cômica extraída da página phdcomics.com	18
3.2	Exemplo de subfiguras usando o pacote <code>subfig</code> . Representação de uma imagem digital. (a) Imagem em escala de cinza. (b) Imagem colorida. Imagem extraída de (Barbosa 2020).	19
3.3	Exemplo de subfiguras usando o ambiente <code>tabular</code> . Representação de uma imagem digital. (a) Imagem em escala de cinza. (b) Imagem colorida. Imagem extraída de (Barbosa 2020).	20
4.1	Exemplo de uso da ferramenta <i>online</i> detexify (https://detexify.kirelabs.org/classify.html).	33
5.1	Exemplo simples de árvore gerada usando o pacote <code>qtree</code>	41
5.2	Exemplo simples de gráfico elaborado usando TikZ.	43
5.3	Exemplo de gráfico gerado com TikZ.	44
5.4	Exemplo de duas redes de <i>switching</i> Banyan-Omega geradas usando a biblioteca <code>switching-architectures</code> do pacote <code>sa-tikz</code> . Exemplo extraído de (Fiandrino 2014).	45

5.5	Exemplo de grafo de dependência criado usando <code>tikz-dependency</code> . Exemplo extraído de (Pighin 2012).	46
5.6	Exemplo de grafo criado usando <code>tikz-network</code> . Exemplo extraído de (Hackl 2019).	47
5.7	Exemplo de grafo em multinível criado usando <code>tikz-network</code> . Exemplo adaptado de (Hackl 2019).	47
5.8	Exemplo de gráfico 3D impresso usando <code>pgfplots</code>	48
5.9	Exemplo de mapa mental criado usando biblioteca <code>mindmaps</code> para TikZ. O mapa mostra todos os capítulos contidos neste documento e expande alguns capítulos em suas seções.	50
5.10	Exemplo de desenho produzido com o auxílio do pacote <code>tikz-3dplot</code> . Exemplo extraído de (Hein 2012).	51
5.11	Exemplo de grafo em multinível criado usando <code>tikz-network</code> . Exemplo extraído de http://mirrors.ctan.org/info/visualtikz/VisualTikZ.pdf	53
5.12	Ilustração do algoritmo recursivo de subdivisão de curvas de Bézier. . . .	56

Lista de Tabelas

1.1	Exemplos de glifos do pacote <code>fontawesome</code>	6
3.1	Resultados de uma tarefa de agrupamento. Adaptada de (Marques 2018).	21

Lista de Códigos

3.1	Exemplo de imagem carregada usando o comando <code>\includegraphics</code> . . .	18
3.2	Código usado para organizar subfiguras usando o pacote <code>subfig</code>	20
3.3	Código usado para organizar subfiguras usando o ambiente <code>tabular</code> . . .	21
3.4	Código usado para gerar a Tabela 3.1.	22
3.5	Exemplo de código \LaTeX usado para configuração do <code>algorithm2e</code>	24
3.6	Exemplo de código definido por <code>algorithm2e</code> usado para gerar o Algoritmo 3.1.	25
3.7	Comando usado para gerar uma lista de <i>listings</i> ou códigos.	26
3.8	Exemplo de código inserido em um <i>listing</i>	27
4.1	Código \LaTeX usado para gerar a Equação 4.1.	30
4.2	Código \LaTeX usado para gerar a unidade kg m s^{-2}	31
4.3	Exemplo do uso do comando <code>\mathclap</code>	32
4.4	Exemplo do uso do pacote <code>bussproofs</code> . Exemplo extraído de (Buss 2011). .	34
4.5	Comandos necessários para o uso do pacote <code>lplfitch</code> (Etchemendy, Barker-Plummer e Zach 2013) com as classes do modelo KOMA-Script (KOMA).	35
4.6	Comandos necessários para o uso do pacote <code>lplfitch</code> (Etchemendy, Barker-Plummer e Zach 2013) com as classes do modelo KOMA.	36
4.7	Exemplo do uso do pacote <code>natded</code>	38

4.8	Exemplo do uso do pacote <code>zed-csp</code>	39
5.1	Exemplo de código \LaTeX usado para gerar a árvore da Figura 5.1.	41
5.2	Exemplo de código \LaTeX usado para configuração do <code>TikZ</code>	42
5.3	Código \LaTeX usado para gerar exemplo de gráfico da Figura 5.2.	43
5.4	Código \LaTeX usado para gerar exemplo de gráfico da Figura 5.3.	44
5.5	Código \LaTeX usado para gerar exemplo de gráfico da Figura 5.5 usando a biblioteca definida pelo pacote <code>tikz-dependency</code>	46
5.6	Código \LaTeX usado para gerar exemplo de gráfico da Figura 5.7 usando a biblioteca definida pelo pacote <code>tikz-network</code>	48
5.7	Código \LaTeX usado para gerar exemplo de gráfico da Figura 5.8 usando o pacote <code>pgfplots</code>	49
5.8	Código <code>TikZ</code> , contendo comandos definidos no pacote <code>tikz-3dplot</code> , usado para gerar a Figura 5.10	51
5.9	Código <code>TikZ</code> , contendo comandos definidos no pacote <code>tikz-3dplot</code> , usado para gerar a Figura 5.10	52
5.10	Código \LaTeX usado para gerar exemplos de curvas fractais da Figura 5.11 usando a biblioteca <code>decoration.fractal</code> do <code>TikZ</code>	54
5.11	Código <code>Asymptote</code> usado para gerar a Figura 5.12.	57
6.1	Código \LaTeX usado para definir as informações referentes ao pacote <code>changes</code>	59
6.2	Código \LaTeX usado para definir o formato do documento como <code>final</code> ao invés de <code>draft</code> com relação ao pacote <code>changes</code>	60
6.3	Código do <code>changes</code> usado para sugerir a adição de texto.	60
6.4	Código do <code>changes</code> usado para sugerir a remoção de texto.	61
6.5	Código do <code>changes</code> usado para sugerir a troca de texto.	61
6.6	Código do <code>changes</code> usado para destacar texto.	61
6.7	Código do <code>changes</code> usado para destacar texto.	61

6.8	Código do <code>changes</code> que mostra um comentário dentro de um comando de adição.	62
7.1	Carregamento do pacote <code>hyperref</code> com as opções usadas neste modelo e associação de informações que serão adicionadas ao arquivo <code>Portable Document Format (PDF)</code>	63
7.2	Carregamento do pacote <code>BIBL_AT_EX</code> com as opções usadas neste modelo. .	65
8.1	Definição de um símbolo como entrada de glossário.	68
8.2	Parte das definições de acrônimos usados neste documento, localizadas no arquivo <code>editaveis/Acrônimos.tex</code>	69
.1	Exemplo de código <code>L_AT_EX</code> usado para carregar um arquivo de apêndice. .	83

Lista de Acrônimos

scrbook (classe livro do ambiente KOMA). [10](#), [14](#), [67](#)

CCET Centro de Ciências Exatas e da Terra. [2](#)

CTAN Comprehensive TeX Archive Network. [3–5](#), [41](#), [53](#), [61](#)

DIMAp Departamento de Informática e Matemática Aplicada. [2](#)

EPS Encapsulated PostScript. [55](#)

GIF Graphic Interchange Format. [57](#)

IDE Integrated Development Environment. [12](#)

JPG Joint expert Photography Group. [17](#)

KOMA KOMA-Script. [xii](#), [2](#), [3](#), [10](#), [11](#), [35](#), [36](#), [67](#)

MPEG Moving Picture Experts Group. [57](#)

Overleaf . [4](#), [13](#), [15](#), [26](#), [28](#), [49](#), [55](#), [74](#)

PDF Portable Document Format. [xiv](#), [6](#), [16](#), [17](#), [40](#), [47](#), [53](#), [55–57](#), [59](#), [63](#)

PGF Portable Graphics Format. [40](#), [41](#)

PNG Portable Network Graphics. [17](#)

PPgSC Programa de Pós-graduação em Sistemas e Computação. [vii](#), [viii](#), [1](#), [2](#), [4](#), [7](#), [10](#),
[75](#)

SVG Scalable Vector Graphics. [53](#), [55](#)

TikZ *TikZ ist kein Zeichenprogramm*. [3](#), [40](#), [41](#)

UFRN Universidade Federal do Rio Grande do Norte. [vii](#), [viii](#), [2](#), [7](#), [10](#)

UTF Unicode Transformation Format. [4](#), [5](#)

WYSIWYM What You See Is What You Mean. [33](#)

xindy flexible **indeing** system. [72](#)

Sumário

Lista de Acrônimos	xv
1 O Modelo PPgSC de Dissertações e Teses	1
1.1 Pacotes	4
1.2 Codificação de Entrada e Fontes	4
1.3 Estrutura de Arquivos	7
1.4 Linguagens	7
1.5 Variáveis	9
1.6 Novos Comandos	10
1.7 Arquivos Auxiliares	11
1.8 Como Usar Este Modelo	12
2 Diagramação e Características do Texto	13
2.1 Espaçamento e Indentação	13
2.2 Ajustes Finos	14
2.3 Cores	14
2.4 Contadores	14
2.5 Listas	15

3	Objetos Float - Figuras, Tabelas, Algoritmos e Código Fonte	16
3.1	Figuras	17
3.1.1	Sub-figuras	19
3.2	Tabelas	21
3.3	Algoritmos	23
3.4	Código	25
4	Equações, Provas e Especificações	28
4.1	Equações	28
4.2	Provas	34
4.3	Especificações	37
5	Desenhos e Animações	40
5.1	Qtree	40
5.2	TikZ e Pacotes Auxiliares	41
5.3	Asymptote	55
6	Correções	59
7	Referências	63
8	Glossário, Acrônimos e Índice	67
9	Considerações Finais	75
	Bibliografia	77
	Apêndice 1	83
	Índice	84

1 O Modelo PPgSC de Dissertações e Teses

O $\text{T}_{\text{E}}\text{X}$ (Knuth [1984](#)) é um sistema de diagramação de textos que foi projetado por Donald Knuth e lançado em 1978 mas teve sua primeira versão finalizada somente em 1989. O $\text{T}_{\text{E}}\text{X}$ foi projetado com dois objetivos em mente: o de permitir que qualquer pessoa possa produzir livros com alta qualidade de diagramação com um esforço mínimo, e o de prover um sistema que gera exatamente a mesma saída, independente do computador utilizado. Quando Leslie Lamport começou a utilizar $\text{T}_{\text{E}}\text{X}$, ele começou a escrever macros para facilitar o uso de $\text{T}_{\text{E}}\text{X}$, gerando o $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ (Lamport [1994](#)) ($\text{L}^{\text{A}}\text{Mport's } \text{T}_{\text{E}}\text{X}$).

A qualidade de textos produzidos usando $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ já é amplamente conhecida, e na minha opinião, você deve usar $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ principalmente se precisa gerar documentos técnicos e que contenham equações. Além disso, o $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ é altamente configurável e automatiza boa parte das tarefas de numeração e referências, sendo ainda um sistema que oferece várias distribuições gratuitas para diversos sistemas operacionais.

Você pode ver em algum lugar referências a $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$. O $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ nada mais é do que a versão atual do $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$. Uma nova versão, chamada de $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 3$, vem sendo desenvolvida há mais de uma década e deve ser lançada em alguns anos. Essas versões são referentes a linguagem, seus comandos e estrutura interna, e não às versões dos processadores $\text{T}_{\text{E}}\text{X}$ e $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

Este capítulo descreve a estrutura geral do modelo $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ do [PPgSC](#), que foi

feito usando a classe `scrbook` da família de pacotes [KOMA](#) (Kohm 2020). As principais razões por trás desta escolha são uma maior flexibilidade em sua configuração e um menor número de conflitos com outros pacotes, quando comparado com as classes `base` `book` e `memoir`.

Algumas explicações sobre os objetivos e formato deste documento são necessárias para que você compreenda como foi feito e possa utilizá-lo da melhor maneira. O principal objetivo deste documento é a homogeneização das dissertações e teses escritas no âmbito do PPgSC da UFRN, sendo que o PPgSC é ligado diretamente ao [Centro de Ciências Exatas e da Terra \(CCET\)](#) e possui a grande maioria de seus professores lotados no [Departamento de Informática e Matemática Aplicada \(DIMAp\)](#). O segundo objetivo da elaboração do modelo e deste manual é o auxílio a ser dado aos alunos na escrita de suas qualificações, dissertações e teses.

Dados estes objetivos, decidi escrever este manual contendo informações sobre os pacotes, variáveis e parâmetros utilizados, no formato de uma dissertação, embora não seja o mais adequado para a escrita de um manual. Assim, você pode ter uma ideia melhor de como seu documento será organizado. Como consequência ou efeito colateral do uso de um modelo de dissertação para escrever esse manual, alguns nomes fantasia foram utilizados na sua escrita de modo a poupar terceiros do uso de seus nomes.

Espero que apreciem este documento e que o mesmo os auxiliem na escrita de seus trabalhos. Gostaria de salientar que tenho uma boa experiência com \LaTeX , mas estou bem longe de me considerar um *expert* na matéria. Sugestões e correções são bem vindas.

Existem muitas referências de excelente qualidade sobre como elaborar documentos em \LaTeX . Listo a seguir algumas delas com breves descrições de seus conteúdos e objetivos. Os livros mais antigos ainda servem como textos base para os comandos do \LaTeX , embora o conteúdo sobre pacotes esteja desatualizado. No caso dos pacotes, o

mais aconselhado é o uso dos manuais oficiais, guias rápidos e exemplos disponíveis na Internet.

- TeX StackExchange (<https://tex.stackexchange.com/>) - Extremamente útil, este sítio coleta dúvidas de usuários e respostas de especialistas em todo o mundo. Se você tem alguma dúvida sobre TeX ou L^AT_EX, ela provavelmente já foi perguntada e respondida lá.
- [Comprehensive TeX Archive Network \(CTAN\)](#) - Repositório de pacotes L^AT_EX, também possui muitos manuais com vários exemplos de uso dos pacotes.
- *L^AT_EX (2nd Ed.): A Document Preparation System: User's Guide and Reference Manual* (Lamport 1994). Livro texto do criador do L^AT_EX, Leslie Lamport, que descreve a linguagem L^AT_EX, composta de comandos de alto nível, também chamados de macros, que simplificam o uso de TeX.
- *The L^AT_EX Companion* (Mittelbach et al. 1999) - Ótima referência sobre L^AT_EX e vários pacotes, embora esteja desatualizada em relação a novos pacotes.
- *L^AT_EX Graphics Companion, The (2nd Edition)* (Goossens et al. 2007) - Referência antiga, porém detalhada sobre como lidar com gráficos em documentos L^AT_EX, comandos de desenho do PSTricks, dentre outros. Como a referência anterior, esta serve como texto introdutório.
- *Typesetting Tables with L^AT_EX* (Voss 2011) - Esse livro se dedica exclusivamente a formatação de tabelas em L^AT_EX.
- *L^AT_EX for Complete Novices* (N. L. C. Talbot 2012) - Este livro introdutório é um bom guia para quem tem pouca experiência escrevendo documentos em L^AT_EX e está disponível gratuitamente no endereço <https://www.dickimaw-books.com/latex/novices/novices-report.pdf>
- *L^AT_EX Cookbook* (Kottwitz 2015) - Livro recente, que inclui material sobre pacotes usados nesse modelo, como KOMA, *TikZ ist kein Zeichenprogramm* (TikZ), pgfplots,

e BIB \LaTeX . Indicado para quem já tem um bom conhecimento sobre \LaTeX .

- ([Overleaf](#)) - Este sítio de escrita colaborativa de documentos em \LaTeX possui uma variada gama de artigos descrevendo o uso de vários comandos e pacotes, e é uma ótima opção para o compartilhamento de textos com seu(ua) orientador(a).

1.1 Pacotes

Ao longo deste documento, descreverei brevemente vários pacotes e algumas de suas funcionalidades e sintaxes. O objetivo deste manual é facilitar a escrita de documentos em \LaTeX no modelo [PPgSC](#), e o não detalhar os vários pacotes que foram sugeridos, testados e incluídos neste modelo. É importante frisar que provavelmente você não precisará utilizar a maioria dos pacotes citados aqui, e que basta comentar as linhas `\usepackage{pacote}` do arquivo `./fixos/pacotes.tex`.

A grande maioria dos pacotes mencionados aqui estão disponíveis no [CTAN](#) e podem ser acessados em <https://ctan.org/>. Ao longo deste documento, incluirei links para os manuais oficiais dos pacotes, bem como para outras referências que proveêm conteúdo mais aprofundado sobre os mesmos.

1.2 Codificação de Entrada e Fontes

Vários pacotes que controlam a codificação de entrada e carregam fontes utilizadas no modelo são descritos a seguir.

- `inputenc`

O pacote `inputenc` permite que o usuário especifique um padrão de codificação da entrada, i.e., dos caracteres. Existem dezenas de opções de codificação. Neste modelo, usamos o padrão [Unicode Transformation Format \(UTF\)](#), definido no arquivo `Pacotes.tex`, e selecionado pelo comando:

`\usepackage[utf8]{inputenc}`

O uso da codificação UTF permite que seu documento use caracteres de várias linguagens, inclusive as que possuem caracteres não Latinos, além de vários símbolos usados em expressões matemáticas. Apesar do comando acima definir o conjunto de caracteres UTF como possíveis entradas, o mapeamento contido no arquivo `utf8.def` não contém mapeamentos de todos os possíveis caracteres UTF. Isso acontece devido ao imenso número de caracteres UTF que podem aparecer em um documento. Eu menciono isso porque os caracteres UTF não mapeados em `utf8.def` irão produzir uma mensagem de erro. Se isso acontecer, você deve incluir o mapeamento para este novo glifo¹ no arquivo `utf8ienc.dtx` e carregá-lo no seu documento. Não acredito que você passará por essa experiência, a não ser que deseje incluir glifos de linguagens Asiáticas.

O manual deste pacote pode ser acessado em <http://mirrors.ctan.org/macros/latex/base/inputenc.pdf> (Jeffrey e Mittelbach 2020).

- **fontenc**

O pacote **fontenc** permite que se selecione padrões de codificação de fontes usadas no documento. Neste modelo definimos as fontes como tendo codificação T1, que utiliza 8 bits, provendo espaço para 256 glifos. Isso permite que palavras com letras com acentos possam ser hifenizadas e que se possa copiar palavras acentuadas de outros documentos e os caracteres corretos sejam colados no seu documento. Além disso, alguns outros símbolos, como $>$, podem exibir um comportamento inesperado.

O comando utilizado aqui é:

`\usepackage[T1]{fontenc}`

Esse pacote não possui um manual específico no CTAN pois faz parte do núcleo do L^AT_EX.

¹Glifo é a representação pictorial de um caractere.

- **fontawesome**

O pacote **fontawesome** fornece acesso a um grande número de ícones relacionados com a *web*. Dependendo do tema de sua dissertação/tese, esses símbolos podem ser úteis para dar um toque mais profissional em alguns desenhos ou descrições.

```
\usepackage{fontawesome}
```

Abaixo estão alguns dos glifos definidos em **fontawesome**. O manual deste pacote pode ser acessado em <http://mirrors.ctan.org/fonts/fontawesome/doc/fontawesome.pdf> (Danaux e Gandy 2016). Um exemplo de seu uso neste documento pode ser visto na Tabela 1.1.

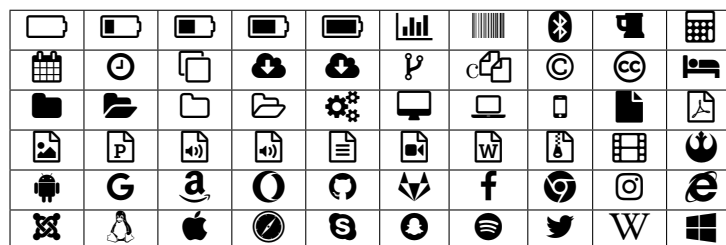


Tabela 1.1: Exemplos de glifos do pacote **fontawesome**.

- **cmap**

O pacote **cmap** provê tabelas de mapeamento de caracteres que permitem que arquivos gerados usando pdfL^AT_EX sejam buscáveis e seu conteúdo possa ser copiado na maioria dos visualizadores de arquivos PDF.

```
\usepackage{cmap}
```

- **lmodern**

O pacote **lmodern** provê a fonte Latin Modern, usada no modelo, e é carregado usando o comando abaixo.

```
\usepackage{lmodern}
```

1.3 Estrutura de Arquivos

Organizamos todos os arquivos do modelo em vários diretórios, de modo a compartimentalizar os arquivos de acordo com suas características e isolar os arquivos que não necessitam ser alterados por você.

Abaixo temos um exemplo da estrutura de arquivos utilizada para gerar um documento com 5 capítulos. Os nomes dos arquivos dos capítulos são de sua escolha e devem ser alterados nos comandos que os carregam, no arquivo principal, `DissertacaoPPgSC.tex`, cujo nome também pode ser mudado por você.

A estrutura de arquivos do modelo pode ser vista na Figura 1.1 e mostra os arquivos `.tex` que se localizam na pasta `capitulos`, que contêm o código fonte $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ dos capítulos da dissertação/tese. Já o diretório `editaveis`, como o nome sugere, agrupa os arquivos `.tex` que devem ser alterados por você para que o documento tenha as informações específicas de seu trabalho e sua defesa. Já os arquivos do diretório `fixos` mostra os arquivos `.tex` que não devem ser alterados por você, exceto em caso de extrema necessidade. Finalmente, o diretório `imagens` agrupa os diretórios que contêm os arquivos de imagens, organizados por capítulos, e com um diretório específico para os logotipos da `UFRN` e `PPgSC`. Essa figura foi gerada usando símbolos da fonte `fontawesome`, vista anteriormente, e o pacote `TikZ` (Seção 5.2).

1.4 Linguagens

O pacote `babel` gerencia regras tipográficas para uma grande game de linguagens. Usando este pacote, um documento pode selecionar uma ou mais linguagens para serem usadas, e alternar entre as linguagens quando necessário.

O comando utilizado neste modelo usa a opção `brazil` (como pode ser visto abaixo), que define os nomes dos elementos como Conteúdo, Lista de Figuras, etc.

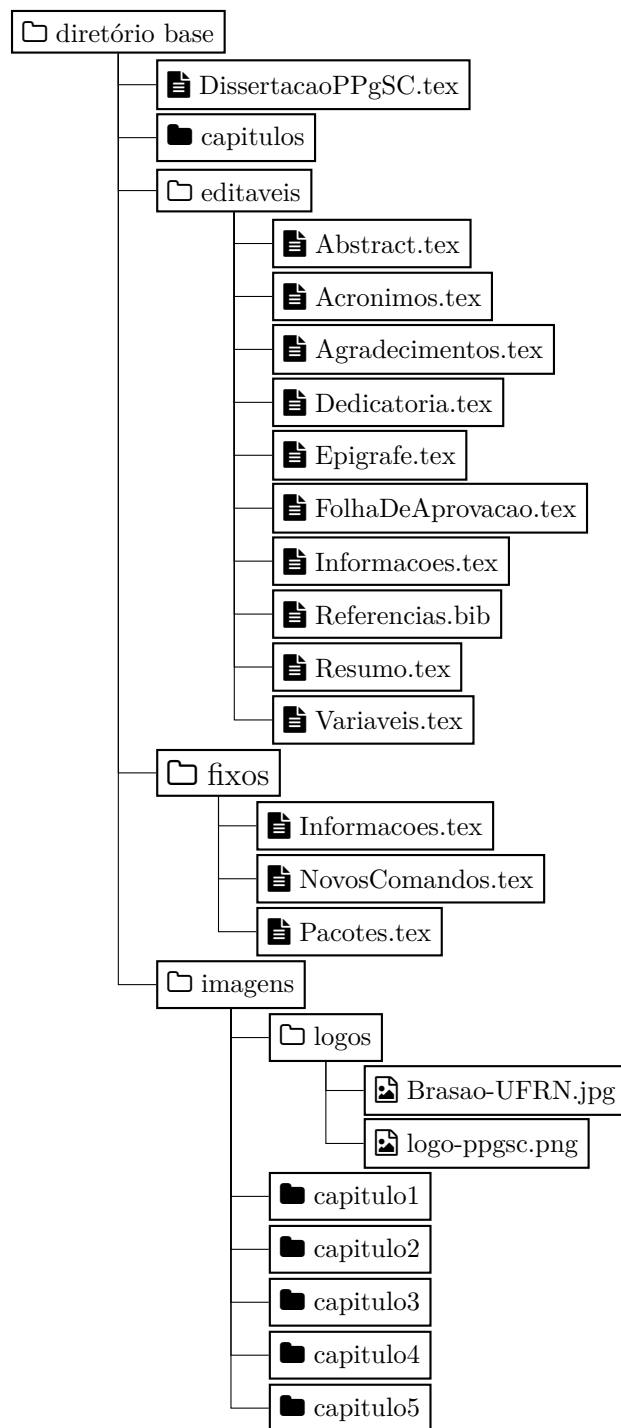


Figura 1.1: Estrutura de arquivos do modelo PPgSC. Essa figura foi gerada usando *TikZ* (ver Capítulo 5) e os símbolos da fonte *fontawesome*.

`\usepackage[brazil]{babel}`

Na realidade, qualquer das opções `brazil`, `brazilian`, `portuges` ou `portuguese` são aceitas e têm o mesmo efeito. O manual do `babel` pode ser acessado em <http://mirrors.ctan.org/macros/latex/required/babel/base/babel.pdf> (Johannes L. Braams 2021).

Entretanto, para que o `babel` funcione com Português é preciso que você também tenha o pacote `babel-portuges` instalado. Este pacote é o que realmente define as macros específicas e é carregado automaticamente pelo `babel`. Garanta também que o pacote `hyphen-portuguese` esteja instalado. O manual do `babel-portuges` pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/babel-contrib/portuges/portuges.pdf> (Johannes L. Braams 2020).

1.5 Variáveis

O modelo define algumas variáveis de modo a facilitar a geração das páginas iniciais do documento, e que são definidas no arquivo `./fixos/variaveis.tex`. Os nomes, tipos e significados das variáveis são:

- `PPgSC-Proposta` - Variável do tipo booleano que indica se o documento é um documento de exame preliminar (qualificação de mestrado ou proposta de doutorado) ou se é um documento de exame final (dissertação ou tese); Valor default: `false`.
- `PPgSC-Tese` - Variável do tipo booleano que indica se o documento é uma tese de doutorado; Valor default: `false`.
- `PPgSC-Ingles` - Variável do tipo booleano que indica se a linguagem usada na escrita do documento é o Inglês; Valor default: `false`.
- `CO-orientador` - Variável do tipo booleano que indica se o aluno(a) possui Coordenador(a); Valor default: `false`.

- `signSkip` - Variável numérica que indica o espaço usado no espaçamento vertical de uma linha de assinatura; Valor default: 13cm.
- `signWidth` - Variável numérica que indica o comprimento de uma linha de assinatura; Valor default: 10cm.
- `signThickness` - Variável numérica que indica a espessura de uma linha de assinatura; Valor default: 0,4pt.

É importante lembrar que os comandos, variáveis e macros em \LaTeX são *case sensitive*, i.e., se você não usar letras minúsculas e maiúsculas nos lugares corretos, o \LaTeX não vai reconhecer os comandos e variáveis.

1.6 Novos Comandos

Alguns novos comandos foram criados para facilitar a diagramação, como a capa do documento, a folha de assinaturas, e o *Abstract* e o Resumo, que não fazem parte da classe (classe livro do ambiente KOMA) (`scrbook`) da KOMA.

O arquivo `./fixos/informacoes.tex` contém informações imutáveis sobre o programa e a instituição, enquanto que o arquivo `./editaveis/informacoes.tex` contém informações específicas do trabalho, como autor, data, orientador, coorientador (se for o caso). Essas informações são utilizadas para gerar os elementos abaixo.

- Capa - Página gerada automaticamente. Utiliza imagens dos logotipos da [UFRN](#) e do [PPgSC](#) e informações do arquivo `./editaveis/informacoes.tex`.
- Folha de rosto - Página gerada automaticamente. Utiliza informações do arquivo `./editaveis/informacoes.tex`.
- Folha de assinaturas - Página que precisa ser ajustada manualmente, incluindo os nomes dos membros da banca que não são o orientador e coorientador no arquivo `./editaveis/FolhaDeAprovacao.tex`.

- Abstract - Novo *environment* (ambiente) definido no modelo devido a sua ausência no KOMA. Está definido no arquivo `./fixos/NovosComandos.tex`.
- Resumo - Novo *environment* (ambiente) definido no modelo devido a sua ausência no KOMA. Está definido no arquivo `./fixos/NovosComandos.tex`.

1.7 Arquivos Auxiliares

Um dos problemas existentes no T_EX que não foi resolvido na implementação do ϵ -T_EX (L^AT_EX 2_ε) foi o suporte a somente 18 manipuladores de arquivos para escrita (*write handles*). Esse número pode parecer grande, mas muitos desses manipuladores são reservados, como o manipulador 0 para o arquivo `.log`. O T_EX usa o manipulador 1 para o arquivo `.aux`, o 2 para o `partaux`, e um manipulador para cada lista, como as geradas pelos comandos `\tableofcontents`, `\listoffigures` e `\listoftables`. Além disso, o L^AT_EX usa manipuladores para pacotes como `\makeindex`, `hyperref`, `minted`, `TikZ` e `glossaries`, que usa mais de um manipulador.

O problema aparece quando seu documento usa muitos desses pacotes que utilizam arquivos para armazenar informações que são utilizadas em passos extra do processador L^AT_EX para formatar corretamente seu documento. Eventualmente, você pode receber a mensagem abaixo durante o processamento de seu documento.

`!No room for a new \write`

Por algum tempo, a solução mais simples adotada era a da utilização de LuaL^AT_EX ao invés de pdfL^AT_EX ou X_YL^AT_EX, eliminando esta restrição e limitando o número de manipuladores de arquivos abertos de acordo com o sistema operacional. O pacote `scrwfile`, do KOMA, altera o *kernel* do L^AT_EX, permitindo que pdfL^AT_EX e X_YL^AT_EX também possam utilizar mais do que 18 manipuladores de arquivos. Para mais detalhes, leia o Capítulo 14 de (Kohm 2020).

1.8 Como Usar Este Modelo

Para começar a utilizar este modelo de dissertações/teses, copie toda a estrutura de arquivos e comece a editar os arquivos que contêm informações sobre seu documento. Comece pelos arquivos do diretório **editaveis**, que podem ser vistos na Figura 1.1. Caso não deseje utilizar um ou mais elementos localizados nesse diretório, como **Dedicatória** ou **Agradecimentos**, comente sua inclusão no arquivo principal, o `DissertacaoPPgSC.tex`.

O nome do arquivo principal pode ser alterado por você, bem como os nomes dos arquivos de referências bibliográficas e dos capítulos. Apenas se certifique que alterou seus nomes ao carregá-los no arquivo principal. A estrutura de organização das imagens também é sugerida, e pode ser alterada caso deseje. Sugiro que incluam novos pacotes no arquivo principal, caso necessitem, embora, em alguns casos, os autores indiquem a necessidade de precedência no carregamento de diferentes pacotes. Nesse caso, é mais prudente seguir as indicações dos autores dos pacotes que desejam usar.

Finalmente, não se esqueça de configurar sua [Integrated Development Environment \(IDE\)](#) para que execute a sequência correta de comandos. Por exemplo, caso esteja usando `BIBLATEX` com `biber`, certifique-se que sua [IDE](#) chama o `biber` e não o `BIBTEX` (Capítulo 7). Em alguns casos, também é necessário incluir o flag `-shell-escape` na execução do `pdfLATEX`, como no caso do pacote `minted` (Seção 3.4).

2 Diagramação e Características do Texto

Neste capítulo descrevo brevemente algumas funções e comandos dos pacotes utilizados para realizar tarefas de diagramação e que controlam características do texto. Os espaçamentos de bordas, cabeçalho e rodapé são definidos internamente e não devem ser alterados.

2.1 Espaçamento e Indentação

O primeiro pacote trata de indentação e é bastante simples. Caso queira acessar e ler todas as quatro linhas de código do pacote `indentfirst`, você pode acessá-lo em <http://mirrors.ctan.org/macros/latex/required/tools/indentfirst.pdf> (D. P. Carlisle 1995). Este pacote faz uma indentação obrigatória do primeiro parágrafo após o título de uma seção.

Já o pacote `setspace` permite que se controle o espaçamento entre linhas de maneira bem simples, usando os comandos `\singlespacing`, `\onehalfspacing` e `\doublespacing`. Devido a sua simplicidade, esse pacote não possui manual. É importante dizer que existem outras maneiras de definir o espaçamento entre linhas em um documento, como utilizando os comandos `\baselineskip` ou `\linespread`, como pode ser visto nas páginas do Overleaf em <https://www.overleaf.com/learn/latex/>

`paragraph_formatting`.

2.2 Ajustes Finos

O pacote `microtype` provê uma interface para extensões micro-tipográficas introduzidas pelo pdfL^AT_EX, como protrusão de caracteres¹, expansão de fontes e ajustes finos entre palavras. Esse é um pacote que também pode ser usado com bons resultados na produção de artigos, e seu manual está disponível em <http://mirrors.ctan.org/macros/latex/contrib/microtype/microtype.pdf> (Schlicht 2021).

2.3 Cores

O uso de cores para sinalizar mudanças ou chamar a atenção do leitor é algo comum e efetivo. Neste modelo, usamos o pacote `xcolor`. Assim, usando o comando:

```
\textcolor{red}{Texto com cor alterada.}
```

pode-se produzir a seguinte saída:

Texto com cor alterada.

Para maiores detalhes sobre o pacote `xcolor`, como opções do pacote, modelos de cores suportados e nomes de cores pré-definidos, consulte o manual no endereço <http://mirrors.ctan.org/macros/latex/contrib/xcolor/xcolor.pdf> (Kern 2016).

2.4 Contadores

Neste modelo, escolhemos numerar os objetos float (figuras, tabelas e algoritmos) por capítulo. No modelo ABNT_EX, essas contagens eram feitas de modo global. No `scrbook`, a opção de contagem por capítulo é a padrão.

¹A protrusão de caracteres move caracteres (geralmente pontuação), parcialmente ou integralmente, para a margem, de modo a criar uma aparência visualmente mais suave.

Caso estivesse usando o modelo ABNT_{TEX}, você deveria usar os comandos abaixo para configurar a contagem por capítulos:

- `\counterwithin{figure}{chapter}` - Define numeração de figuras por capítulo;
- `\counterwithin{table}{chapter}` - Define numeração de tabelas por capítulo;
- `\counterwithin{algocf}{chapter}` - Define numeração de algoritmos por capítulo;

Porém, se o seu processador \LaTeX for anterior a Abril de 2018, para usar o comando `counterwithin` você deve usar o pacote `chngctr`.

Você pode resetar e acessar valores de contadores e até criar novos contadores. Um bom guia inicial de como usar contadores pode ser visto no [Overleaf](https://www.overleaf.com/learn/latex/Counters) (<https://www.overleaf.com/learn/latex/Counters>). Entretanto, sugiro que tenha cuidado ao manipular contadores, de modo a evitar problemas de numerações erradas em referências.

2.5 Listas

Quando utilizamos os ambientes `itemize`, `enumerate` e `description` do \LaTeX , nós fazemos uso dos padrões de numeração definidos pelo \LaTeX . Caso desejemos alterar cesses padrões, podemos usar o pacote `enumitem`, que permite que o usuário controle o layout dos três ambientes citados acima, incluindo espaçamento, rótulos e numeração.

Você pode, por exemplo, remover o espaço vertical em uma lista usando a opção `nosep`, como vemos abaixo no caso da definição de um ambiente `enumerate`:

`\begin{enumerate}[nosep]`

Para maiores detalhes, consulte o manual do pacote, disponível em <http://mirrors.ctan.org/macros/latex/contrib/enumitem/enumitem.pdf> (Bezoz 2019).

3 Objetos Float - Figuras, Tabelas, Algoritmos e Código Fonte

Neste capítulo, irei falar sobre objetos do tipo `float`, que recebem este nome porque “flutuam” no documento, e têm seus lugares finais influenciados por sugestões dadas pelos autores, mas que o \LaTeX tem a decisão final sobre onde os colocar. Os principais objetos `float` são as figuras, tabelas, algoritmos e listagens de códigos.

As seções a seguir contêm exemplos do uso desses quatro tipos de objetos `float`, bem como de alguns pacotes auxiliares que foram incorporados a este modelo de dissertação/tese.

O pacote `pdfscape` adiciona o suporte [PDF](#) ao ambiente *landscape* (orientação paisagem) do pacote `lscape`. Páginas marcadas com o atributo que indica essa orientação serão rotacionadas e mostradas em modo paisagem pelos visualizadores de arquivos [PDF](#). O manual desse pacote pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/pdfscape/pdfscape.pdf> (Oberdiek 2019).

O pacote `float` melhora a interface para a definição de objetos `float`, introduzindo os objetos *boxed float*, *ruled float* e *plaintop float*. O primeiro tipo de objeto cria floats com um retângulo ao redor dos objetos, enquanto que os dois últimos tipos são mais usados para mostrar códigos. Entretanto, o pacote `minted`, mostrado na Seção 3.4 provê uma visualização muito mais elegante, de modo que sugiro que use o `minted` para diagramar seus códigos em \LaTeX .

O pacote `float` ainda define a opção `H` para colocação de *floats*, que força o \LaTeX a colocar um objeto `float` exatamente naquele lugar (deixando assim de ser um objeto `float`), mesmo que isso implique em deixar uma parte da página anterior em branco, sem texto. Use essa opção com parcimônia, pois ela pode quebrar o seu texto e gerar uma diagramação esquisita. O manual do pacote `float` pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/float/float.pdf> (Lingnau 2001).

O pacote `adjustbox` pode ser usado para ajustar conteúdo dentro de uma caixa “virtual”, alterando sua escala, orientação e cortar parte do conteúdo. Esse pacote pode ser aplicado a qualquer objeto `float` ou até a texto. Os textos destacados dentro de caixas e centralizados que aparecem nos Capítulos 1, 2 e outros ainda não vistos, foram produzidos usando o pacote `adjustbox`. Você também pode usar o `adjustbox` para diminuir o tamanho de uma tabela, por exemplo, quando ela passa um pouco da largura máxima da área reservada para o texto, e uma diminuição da fonte usada gera letras muito pequenas, difíceis de se ver. Nesse caso, um pequeno ajuste do tamanho da tabela pode ser a melhor opção. Você deve usar essa opção com cuidado, pois uma mudança muito grande pode afetar a qualidade da saída.

3.1 Figuras

Como recomendo o uso do processador $\text{pdf}\text{\LaTeX}$, devo informar que o $\text{pdf}\text{\LaTeX}$ permite carregar imagens nos formatos [PDF](#), [Portable Network Graphics \(PNG\)](#) e [Joint expert Photography Group \(JPG\)](#). Algumas ferramentas, como LyX, fazem a conversão *on-the-fly*, facilitando a tarefa do usuário mas adicionando tempo ao processamento do texto. Eu sugiro que você converta suas imagens para um desses formatos antes de carregá-las, economizando tempo de conversão durante a compilação do código \LaTeX .

O pacote `graphicx` se baseia no pacote `graphics` para prover uma interface para argumentos opcionais para o comando `\includegraphics`. O pacote `graphicx` faz

parte do grupo de pacotes `latex-graphics`, que é uma das coleções obrigatórias¹ de \LaTeX .

Na Figura 3.1 vemos um exemplo do uso do comando `\includegraphics` para a inclusão de uma imagem no objeto `float figure`. Os comandos utilizados para gerar essa figura podem ser vistos no Código 3.1.



Figura 3.1: Tirinha cômica extraída da página phdcomics.com.

```
1 \begin{figure}[ht]
2   \centering
3   \includegraphics[width=12cm]{./imagens/capitulo3/phd020808s}
4   \caption{Tirinha cômica extraída da página \url{phdcomics.com}.}
5   \label{fig:phdcomics}
6 \end{figure}
```

Código 3.1: Exemplo de imagem carregada usando o comando `\includegraphics`.

O manual disponível em <http://mirrors.ctan.org/macros/latex/required/graphics/grfguide.pdf> (D. P. Carlisle 2020) se refere à coleção `latex-graphics` e descreve os pacotes `color`, `graphics` e `graphicx` enquanto que o manual acessível em <http://mirrors.ctan.org/macros/latex/required/graphics/graphics.pdf> (D. P. Carlisle e S. P. Q. Rahtz 2020) descreve o pacote `graphics`. Sugiro a leitura do primeiro

¹As coleções obrigatórias de \LaTeX implicam que toda distribuição \LaTeX deve possuí-las.

manual, principalmente das opções descritas em sua Seção 4.4, que tratam da formatação das imagens carregadas pelo comando `\includegraphics`.

3.1.1 Sub-figuras

O pacote `subfig` provê suporte para a manipulação e referenciamento de subfiguras e subtabelas, permitindo que elas possam ser referenciadas e/ou descritas separadamente ou até mesmo listadas separadamente na Lista de Figuras. Um exemplo simples de figura composta por subfiguras e que foi contruída usando o pacote `subfig` pode ser vista na Figura 3.2. Os comandos necessários para gerar esta figura podem ser vistos no Código 3.2.

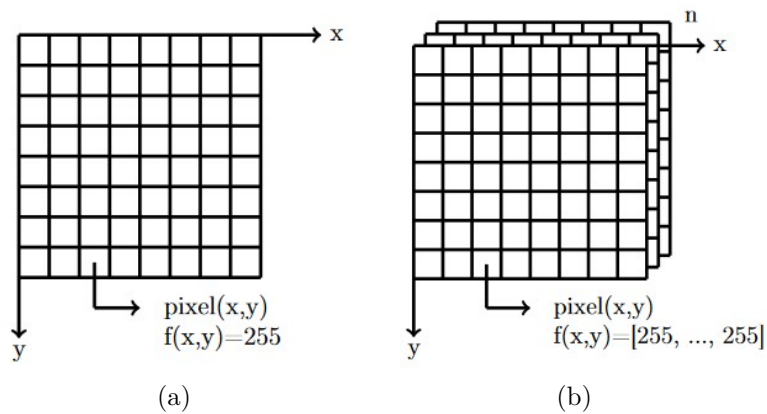


Figura 3.2: Exemplo de subfiguras usando o pacote `subfig`. Representação de uma imagem digital. (a) Imagem em escala de cinza. (b) Imagem colorida. Imagem extraída de (Barbosa 2020).

Como alternativa, você pode usar o ambiente `tabular` para organizar as subfiguras e seus rótulos. A Figura 3.3 foi criada usando este outro modo de organizar subfiguras. O Código 3.3 mostra os comandos usados para gerá-la.

```

1 \begin{figure}[ht]
2   \centering
3   \subfloat[]{}
4     \includegraphics[height=5cm]{imagens/capitulo2/imagemCinza.jpg}}
5   \subfloat[]{}
6     \includegraphics[height=5cm]{imagens/capitulo2/imagemColorida.jpg}}
7   \caption{Exemplo de subfiguras usando o pacote \texttt{subfig}.
8   Representação de uma imagem digital. (a) Imagem em escala de cinza.
9   (b) Imagem colorida. Imagem extraída de \parencite{Barbosa2020}.}
10  \label{fig:subfig}
11 \end{figure}

```

Código 3.2: Código usado para organizar subfiguras usando o pacote `subfig`.

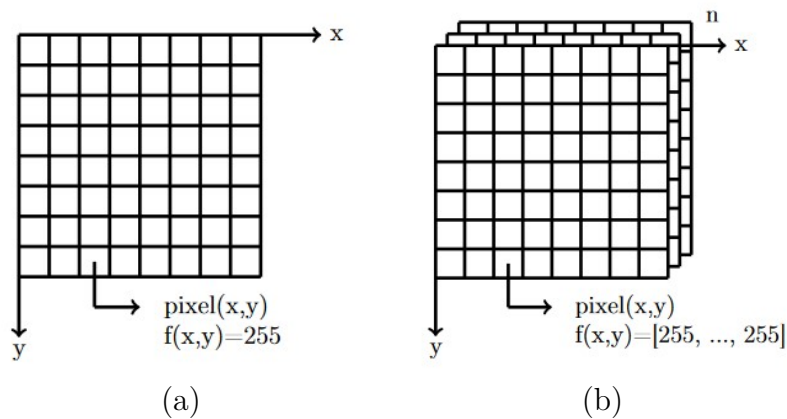


Figura 3.3: Exemplo de subfiguras usando o ambiente `tabular`. Representação de uma imagem digital. (a) Imagem em escala de cinza. (b) Imagem colorida. Imagem extraída de (Barbosa 2020).

Você pode comparar visualmente os resultados das duas opções descritas acima observando as Figuras 3.2 e 3.3. Lembre-se que no caso do pacote `subfig`, você pode referenciar e listar as subfiguras separadamente. Para maiores detalhes sobre o pacote `subfig`, consulte seu manual, que está disponível em <http://mirrors.ctan.org/macros/latex/contrib/subfig/subfig.pdf> (Cochran 2005).

```

1 \begin{figure}[ht]
2   \begin{center}
3     \begin{tabular}{cc}
4       \includegraphics[height=5cm]{imagens/capitulo2/imagemCinza.jpg} &
5       \includegraphics[height=5cm]{imagens/capitulo2/imagemColorida.jpg} \\
6       \\
7       (a) & (b)
8     \end{tabular}
9   \end{center}
10  \caption{Exemplo de subfiguras usando o ambiente \texttt{tabular}.
11  Representação de uma imagem digital. (a) Imagem em escala de cinza.
12  (b) Imagem colorida. Imagem extraída de \parencite{Barbosa2020}.}
13  \label{fig:subfigtabular}
14 \end{figure}

```

Código 3.3: Código usado para organizar subfiguras usando o ambiente `tabular`.

3.2 Tabelas

Tabelas são um outro tipo de objeto `float` presente no \LaTeX . Existem páginas, capítulos de livros e até livros completos dedicados a criação de tabelas em \LaTeX . Geralmente, se utiliza um ambiente `tabular` dentro de um objeto `float` do tipo `table`. Esse ambiente `tabular` é responsável por informar quantas colunas uma tabela terá, e por sua tabulação, organizando os dados usando delimitadores pré-definidos.

Tecido	Distância e Tamanho	Acurácia	Especificidade	Sensibilidade	Coefficiente de Dice
Granulação	$9 \times 9_E$	$0,9252 \pm 0,0796$	$0,8961 \pm 0,1520$	$0,8478 \pm 0,1942$	$0,8796 \pm 0,1699$
	$11 \times 11_E_CSR$	$0,9292 \pm 0,0755$	$0,8828 \pm 0,1673$	$0,8983 \pm 0,0914$	$0,9224 \pm 0,0650$
Necrótico	$9 \times 9_E$	$0,9595 \pm 0,0518$	$0,9739 \pm 0,0436$	$0,8758 \pm 0,8800$	$0,8215 \pm 0,3155$
	$11 \times 11_E_CSR$	$0,9591 \pm 0,0514$	$0,9741 \pm 0,0379$	$0,8963 \pm 0,0638$	$0,9037 \pm 0,1195$
Esfacelo	$9 \times 9_E$	$0,9346 \pm 0,0840$	$1,0000 \pm 0,0000$	$0,8018 \pm 0,1489$	$0,8825 \pm 0,0983$
	$11 \times 11_E_CSR$	$0,9336 \pm 0,0854$	$1,0000 \pm 0,0000$	$0,8111 \pm 0,1378$	$0,8707 \pm 0,1089$
Todos	$9 \times 9_E$	$0,9482 \pm 0,0457$	$0,9784 \pm 0,0309$	$0,8932 \pm 0,0771$	$0,9234 \pm 0,0673$
	$11 \times 11_E_CSR$	$0,9491 \pm 0,0423$	$0,9788 \pm 0,0298$	$0,8952 \pm 0,0717$	$0,9247 \pm 0,0625$

Tabela 3.1: Resultados de uma tarefa de agrupamento. Adaptada de (Marques 2018).

A Tabela 3.1 (adaptada de (Marques 2018)) mostra resultados de uma tarefa de classificação. Neste exemplo, usei o pacote `multirow` (Oostrum, Bache e Leichter 2021), que permite criar multilinhas e multicolumnas, centralizando o texto dentro dessas células

compostas. O Código 3.4 mostra os comandos usados para gerar a Tabela 3.1.

```

1 \begin{table}[H]
2   \centering
3   \resizebox{\textwidth}{!}{%
4     \begin{tabular}{|l|l|l|l|l|l|l|l|}
5       \hline
6       Tecido & Distância e Tamanho & Acurácia & Especificidade
7       & Sensibilidade & Coeficiente de Dice \\ \hline
8       \multirow{2}{*}{Granulação} & $9 \times 9$ \_E & $0,9252
9       \pm 0,0796$ & \textbf{$0,8961 \pm 0,1520$} & $0,8478 \pm 0,1942$
10      & $0,8796 \pm 0,1699$ \\ \cline{2-6}
11      & $11 \times 11$ \_CSR & \textbf{$0,9292 \pm 0,0755$} &
12      $0,8828 \pm 0,1673$ & \textbf{$0,8983 \pm 0,0914$} &
13      \textbf{$0,9224 \pm 0,0650$} \\ \hline
14      \multirow{2}{*}{Necrótico} & $9 \times 9$ \_E &
15      \textbf{$0,9595 \pm 0,0518$} & $0,9739 \pm 0,0436$ &
16      $0,8758 \pm 0,8800$ & $0,8215 \pm 0,3155$ \\ \cline{2-6}
17      & $11 \times 11$ \_E\_CSR & $0,9591 \pm 0,0514$ &
18      \textbf{$0,9741 \pm 0,0379$} & \textbf{$0,8963 \pm
19      0,0638$} & \textbf{$0,9037 \pm 0,1195$} \\ \hline
20      \multirow{2}{*}{Esfacelo} & $9 \times 9$ \_E &
21      \textbf{$0,9346 \pm 0,0840$} & \textbf{$1,0000 \pm
22      0,0000$} & $0,8018 \pm 0,1489$ & \textbf{$0,8825 \pm
23      0,0983$} \\ \cline{2-6}
24      & $11 \times 11$ \_E\_CSR & $0,9336 \pm 0,0854$ &
25      \textbf{$1,0000 \pm 0,0000$} & \textbf{$0,8111 \pm
26      0,1378$} & $0,8707 \pm 0,1089$ \\ \hline
27      \multirow{2}{*}{Todos} & $9 \times 9$ \_E & $0,9482
28      \pm 0,0457$ & $0,9784 \pm 0,0309$ & $0,8932 \pm
29      0,0771$ & $0,9234 \pm 0,0673$ \\ \cline{2-6}
30      & $11 \times 11$ \_E\_CSR & \textbf{$0,9491 \pm 0,0423$}
31      & \textbf{$0,9788 \pm 0,0298$} & \textbf{$0,8952 \pm
32      0,0717$} & \textbf{$0,9247 \pm 0,0625$} \\ \hline
33     \end{tabular}%
34   }
35   \caption{Melhores resultados do agrupamento, adaptada de
36   \parencite{Marques2018}.}
37   \label{tab:resultadosVitor}
38 \end{table}

```

Código 3.4: Código usado para gerar a Tabela 3.1.

Além do estilo padrão de tabelas do \LaTeX , que é bem permissivo, pode-se utilizar o pacote `booktabs` (Fear 2020), que é conhecido pelo estilo de suas tabelas, similar a tabelas presentes em livros. Entretanto, o `booktabs` tem algumas restrições que foram impostas por escolhas de diagramação feitas pelos seus autores, como a impossibilidade de se usar linhas verticais separando colunas de uma tabela, a adição de um espaço acima e abaixo de linhas horizontais, a existência de linhas de diferentes espessuras e a impossibilidade do uso de linhas de separação duplas. Assim como as tabelas do estilo padrão do \LaTeX , as tabelas do `booktabs` podem ter suas linhas ou colunas coloridas usando os pacotes `xcolor` ou `colortbl`. O manual do pacote `booktabs` pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/booktabs/booktabs.pdf> (Fear 2020).

Caso você tenha problemas no início para gerar suas tabelas, você pode utilizar algumas das páginas na Internet que permitem a criação de tabelas em \LaTeX de modo interativo, como o Tables Generator (<https://www.tablesgenerator.com/>) e o \LaTeX Tables (<https://www.latex-tables.com/>). Algumas delas ainda permitem que se importem dados de arquivos de vários tipos, como `.csv`, `.xls` e `.ods`.

Se você desejar criar tabelas muito elaboradas, podendo inclusive conter ilustrações, então eu sugiro que considere criá-las usando `TikZ`, que será abordado no Capítulo 5. Vários exemplos de tabelas criadas usando `TikZ` estão disponíveis na Internet.

3.3 Algoritmos

O pacote `algorithm2e` define um ambiente para escrever algoritmos em \LaTeX 2 ϵ , que são definidos como objetos *float* como figuras e tabelas. A apresentação dos algoritmos é bastante configurável. As opções mostradas no Código 3.5 indicam que os algoritmos serão numerados por capítulo (`algochapter`), terão suas linhas numeradas (`linesnumbered`), exceto por comentários e entrada/saída, imprime linhas verticais de-

limitando blocos (`lined`), usa as palavras chaves em Português (`portuguese`) e escolhe o estilo `ruled` como padrão para mostrar os algoritmos.

```
1 \usepackage[algochapter, linesnumbered, lined, portuguese, ruled]
2 {algorithm2e}
```

Código 3.5: Exemplo de código L^AT_EX usado para configuração do `algorithm2e`.

Algoritmo 3.1: LeftRotate(T, x)

```
1  $y = x.right$ 
2  $x.right = y.left$ 
3 if  $y.left \neq T.nil$  then
4   |  $y.left.p = x$ 
5 end
6  $y.p = x.p$ 
7 if  $x.p == T.nil$  then
8   |  $T.root = y$ 
9 end
10 else if  $x == x.p.left$  then
11   |  $x.p.left = y$ 
12 end
13 else
14   |  $x.p.right = y$ 
15 end
16  $y.left = x$ 
17  $x.p == y$ 
```

No Algoritmo 3.1 vemos o código usado para realizar a rotação à esquerda em torno de um nó em uma árvore rubro-negra. Note o efeito das opções mencionadas acima na formatação do algoritmo. No Código 3.6 vemos os comandos definidos no pacote `algorithm2e` que foram usados para gerar o Algoritmo 3.1. O comando da Linha 2 foi usado para diminuir o espaçamento entre linhas, já que este documento está usando

```

1 \begin{algorithm}[ht]
2   \setstretch{1.35}
3   $y = x.right$ \\\
4   $x.right = y.left$ \\\
5   \If{$y.left \neq T.nil$}{
6     $y.left.p = x$}
7   $y.p = x.p$ \\\
8   \If{$x.p == T.nil$}{
9     $T.root = y$}
10  \ElseIf{$x == x.p.left$}{
11    $x.p.left = y$}
12  \Else{$x.p.right = y$}
13    $y.left = x$ \\\
14    $x.p == y$
15    \caption{LeftRotate($T,x$)}
16    \label{alg:left-rotate}
17 \end{algorithm}

```

Código 3.6: Exemplo de código definido por `algorithm2e` usado para gerar o Algoritmo 3.1.

espaçamento duplo.

Existem várias opções de formatação e numeração dos algoritmos, deste modo, sugiro que você leia o manual, que está disponível em <http://mirrors.ctan.org/macros/latex/contrib/algorithm2e/doc/algorithm2e.pdf> (Fiorio 2017) e teste os estilos disponíveis para que escolha o que lhe agrada mais.

3.4 Código

O pacote `minted` define os ambientes `minted` e `listings` para receber blocos de código. O primeiro gera o código e coloca em um retângulo com cor de fundo (*background*) que pode ser redefinida, enquanto que o segundo coloca o código em uma caixa do tipo *float*².

²Um objeto do tipo *float* é um objeto que se move no documento de acordo com a escolha do *kernel* do L^AT_EX para gerar a melhor diagramação possível.

O usuário pode então usar o comando mostrado no Código 3.7 para gerar uma lista de códigos ou *listings*. Esse comando deve ser chamado no *frontmatter* do documento, junto com as listas de figuras, tabelas e algoritmos.

```
1 \texttt{listoflistings}
```

Código 3.7: Comando usado para gerar uma lista de *listings* ou códigos.

O pacote `minted` provê suporte para mais de 300 linguagens de programação. Para obter uma lista de todas elas, digite o comando abaixo em um terminal:

```
pygmentize -L lexers
```

⚠ Cuidado!

É importante mencionar que o pacote `minted` usa `Pygments`, um pacote de realçamento de texto escrito em Python. Como esse é um comando externo, você tem que habilitar a execução de comandos externos em sua ferramenta de edição \LaTeX (caso esteja usando uma) e usar o flag `--shell-escape` no comando do processador utilizado, no nosso caso, o `pdf\text{\LaTeX}`.

O exemplo do Código 3.8 mostra um exemplo do uso dos ambientes `minted` e `listing` para mostrar um código em C em um objeto `float`.

Para mais detalhes, você pode consultar o manual do `minted` ou o guia básico do `minted` no [Overleaf](http://mirrors.ctan.org/macros/latex/contrib/minted/minted.pdf), disponíveis em <http://mirrors.ctan.org/macros/latex/contrib/minted/minted.pdf> (Poore 2017) e https://www.overleaf.com/learn/latex/Code_Highlighting_with_minted, respectivamente.

```

1  #include <stdio.h>
2  #include <math.h>
3  void main() {
4      int cont=0, n, i;
5      printf("Digite um número: ");
6      scanf("%d", &n);
7      for(i=2; i<= floor(sqrt(n)); i++){
8          printf("i = %d\n", i);
9          if (n%i == 0) {
10             cont++;
11             break;
12         }
13     }
14     if (cont)
15         printf("%d não é primo\n", n);
16     else
17         printf("%d é primo\n", n);
18 }

```

Código 3.8: Exemplo de código inserido em um *listing*.

4 Equações, Provas e Especificações

O intuito deste capítulo é o de introduzir os pacotes matemáticos que foram sugeridos por diversos professores, de acordo com suas necessidades, e mostrar exemplos simples de como usá-los. Ele não deve ser visto como texto introdutório, mediano ou avançado sobre como se deve usar os ambientes matemáticos disponíveis em \LaTeX . Existem vários excelentes livros e manuais físicos e digitais, pagos e gratuitos, que lidam com este extenso tópico, de modo que não usarei espaço neste capítulo para tal.

Como sugestões de referências sobre o uso de comandos e pacotes \LaTeX para a escrita de fórmulas matemáticas, elenco as seguintes:

1. Short Math Guide for \LaTeX - Disponível em <http://mirrors.ctan.org/info/short-math-guide/short-math-guide.pdf>
2. \LaTeX Cookbook (Kottwitz 2015) - Livro disponível em papel e eletronicamente.
3. Overleaf - O [Overleaf](https://www.overleaf.com/learn/latex/mathematical_expressions) separa pequenos textos introdutórios e com exemplos para diferentes tópicos relacionados à definição de equações. Um bom lugar para iniciar a leitura é https://www.overleaf.com/learn/latex/mathematical_expressions.

4.1 Equações

Um dos principais objetivos de Knuth quando projetou o \TeX era o de permitir a construção de fórmulas matemáticas de modo simples, mas que tivessem qualidade profissional quando impressas. Então, ele definiu sintaxes para vários símbolos matemáti-

cos e criou comandos para a representação de equações. Para que \TeX ou \LaTeX gere os símbolos matemáticos, eles precisam saber que o texto é matemático. Os textos matemáticos pode ser do tipo texto (ou *inline*), isto é, os símbolos são colocados na linha de texto, ou equação, onde os símbolos são colocados na linha própria deles.

No caso dos textos matemáticos *inline*, nós sinalizamos um texto matemático usando os delimitares $\backslash($ e $\backslash)$, no caso do \LaTeX e $\$$ e $\$$, no caso do \TeX e \LaTeX . Deve-se ter cuidado para não colocar equações ou símbolos muito altos que acabem gerando em espaçamento muito grande entre linhas, o que pode gerar uma diagramação não agradável visualmente. Um exemplo de equação simples que pode caber em uma linha é $E = mc^2$, que foi gerada com o código $\backslash(E = mc^2 \backslash)$.

Os textos matemáticos no formato equação podem ser numerados ou não. No caso das equações não numeradas, você pode usar os delimitadores $\backslash[$ e $\backslash]$ no caso do \LaTeX e $\$$ e $\$$, no caso do \TeX e \LaTeX . Caso esteja usando **amsmath** (Society 2020) ou **mathtools** (Morten Høgholm 2021), você pode usar a versão “*” do ambiente **equation**, que suprime a numeração e contagem dos objetos. O mesmo vale para os outros ambientes definidos em **mathtools**, como **align**, **flalign**, **gather** e **multilined**.

Para usar a numeração das equações, você deve colocar os códigos das suas equações dentro de um ambiente (*environment*) **equation**, que foi usado para gerar a Equação 4.1, que define a equação da interação da luz no ray-tracing recursivo, como pode ser visto no Código 4.1. Nesse caso, por causa do tamanho da equação, eu utilizei o ambiente *multlined* do **mathtools** (também incluído neste modelo) para quebrar a equação em duas linhas. Caso deseje alinhar os termos da equação, você pode usar o ambiente **aligned**, também definido no **mathtools**. Neste exemplo, eu escolhi usar a representação das setas que indicam vetores do pacote **esvector**, que permite que se selecione um dentre oito possíveis estilos de setas. Eu escolhi usar as setas do **esvector** somente nos vetores \vec{N} e \vec{V} , para que você pudesse comparar com as aparências dos vetores \vec{L} e \vec{R} ,

que foram criados usando as setas do `mathtools`. Caso necessite usar muitos símbolos matemáticos, sugiro que você veja os símbolos definidos pelo pacote `amsfonts` (<http://mirrors.ctan.org/fonts/amsfonts/doc/amsfonts.pdf>) nos sub-pacotes `amssymb` (<http://mirrors.ctan.org/fonts/amsfonts/doc/amssymb.pdf>), `euscript` (<http://mirrors.ctan.org/fonts/amsfonts/doc/euscript.pdf>) e `eufrak` (<http://mirrors.ctan.org/fonts/amsfonts/doc/eufrak.pdf>).

$$I_{\lambda} = \underbrace{I_{a\lambda} K_a O_{d\lambda}}_{\text{ambiente}} + \sum f_{att} I_{p\lambda} \left[\underbrace{k_d O_{d\lambda} (\vec{N} \cdot \vec{L})}_{\text{difusa}} + \underbrace{k_s O_{s\lambda} (\vec{R} \cdot \vec{V})^n}_{\text{especular}} \right] + k_s \underbrace{\left[\underbrace{(1 - k_t) O_{s\lambda}}_{\text{refletida}} + \underbrace{k_t O_{t\lambda} I_{t\lambda}}_{\text{refratada}} \right]}_{\text{recursivo}} \quad (4.1)$$

```

1 \begin{equation}
2   \begin{multlined}
3     I_{\lambda} = \underbrace{I_{a\lambda} K_a O_{d\lambda}}
4     \}_{\text{ambiente}} + \sum f_{att} I_{p\lambda} \left[
5     \underbrace{k_d O_{d\lambda} \left( \vec{N} \cdot \vec{L} \right)}_{\text{difusa}} + \underbrace{k_s
6     O_{s\lambda} \left( \vec{R} \cdot \vec{V} \right)^n}_{\text{especular}} \right] \backslash\backslash
7     + \underbrace{k_s \left[ \underbrace{\left( 1 - k_t \right)}_{\text{refletida}}
8     O_{s\lambda} \right] + \underbrace{k_t O_{t\lambda} I_{t\lambda}}_{\text{refratada}}}_{\text{recursivo}}
9   \end{multlined}
10   \label{eq:raytracing}
11 \end{equation}

```

Código 4.1: Código L^AT_EX usado para gerar a Equação 4.1.

Um pacote que pode auxiliá-lo eventualmente é o `siunitx`, que provê um conjunto de ferramentas para a diagramação de quantidades de um modo consistente. Por exemplo,

a unidade kg m/s^2 pode ser gerada usando um dos comandos mostrados no Código 4.2. No primeiro modo, o literal, o `siunitx` converte os símbolos “.” e “~” nos espaços correspondentes e posiciona corretamente os subscritos e sobrescritos, enquanto que o modo “textual” usa o significado das unidades ao invés da aparência ditada pelo modo literal. O manual do pacote `siunitx` pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/siunitx/siunitx.pdf> (Wright 2020).

```
1 \si{kg.m/s^2}
2 \si{\kilo\gram\meter\per\square\second}
```

Código 4.2: Código L^AT_EX usado para gerar a unidade kg m s^{-2} .

O pacote `amsmath` provê uma grande gama de melhorias para a organização e impressão de expressões matemáticas. Por exemplo, ele define novos ambientes para a diagramação de matrizes, amplia o leque de opções para espaçamento em equações, melhora a representação pictorial de letras com acentos, define setas extensíveis, dentre outros. Para mais detalhes, consulte o manual em <http://mirrors.ctan.org/macros/latex/required/amsmath/amslldoc.pdf> (Society 2020).

O pacote `mathtools` provê uma série de ferramentas projetadas para melhorar a aparência de documentos que contenham muitos símbolos matemáticos. Ele carrega o pacote `amsmath`, passando parâmetros para esse outro pacote quando necessário, além de definir vários novos símbolos, novos ambientes para equações e permitir que se numere somente equações referenciadas de forma automática. Ele ainda corrige alguns erros presentes no pacote `amsmath`. Como ele carrega o pacote `amsmath` quando é carregado, você pode omitir o carga do pacote `amsmath` quando usa o `mathtools`.

O exemplo abaixo mostra como podemos usar `mathtools` para ajustar o subscrito de um somatório, removendo espaçamento extra que faz a equação parecer desconectada. A Equação 4.2 mostra a saída original enquanto que a Equação 4.3 mostra a versão

usando o comando `\mathclap`, cujo uso pode ser visto no Código 4.3.

$$T = \sum_{1 \leq i \leq j \leq n} X_{ij} \quad (4.2)$$

$$T = \sum_{1 \leq i \leq j \leq n} X_{ij} \quad (4.3)$$

```

1 \begin{equation}
2   T = \sum_{\mathclap{1 \leq i \leq j \leq n}} X_{ij}
3   \label{eq:sum_mathclap}
4 \end{equation}

```

Código 4.3: Exemplo do uso do comando `\mathclap`.

O pacote `mathtools` ainda define novos ambientes de matrizes, similares aos definidos no `amsmath`, mas que permitem que o alinhamento dos elementos seja configurado e não sempre centralizado, como no `amsmath`.

O pacote ainda provê vários comandos para que se façam ajustes finos em elementos de equações. Para conhecê-los, acesse o manual do `mathtools` em <http://mirrors.ctan.org/macros/latex/contrib/mathtools/mathtools.pdf> (Morten Høgholm 2021).

Existem vários editores de equações online, que permitem que você teste a escrita de suas equações e veja os resultados. Você ainda terá que digitar os comandos das equações como se estivesse em um documento \LaTeX . Alguns exemplos dessas ferramentas são o \TeX Equation Editor <http://atomurl.net/math/>, o CodeCogs (<https://www.codecogs.com/latex/eqneditor.php>), o HostMath (<https://www.hostmath.com/>) e o Latex4technics (<https://www.latex4technics.com/>).

Algumas ferramentas de edição de código matemático para \LaTeX foram desenvolvidas para serem executadas localmente, como o EqualX <https://equalx.sourceforge.io/> e o AxMath <https://www.axsoft.co/>, sendo que esta última é uma solução paga, com o custo de US\$ 12.00 no momento da escrita deste parágrafo.

Detexify

classify

symbols

Want a Mac app?

Lucky you. The Mac app is finally stable enough. See how it works on [Vimeo](#). Download the latest version [here](#).

Restriction: In addition to the LaTeX command the unlicensed version will copy a reminder to purchase a license to the clipboard when you select a symbol.

You can purchase a license here:

Buy Detexify for Mac

Score: 0.20810505590822606

```
\usepackage{ marvosym }
\Lightning
textmode
```

Score: 0.216548086558558

```
\usepackage{ stmaryrd }
\Rbag
mathmode
```

Score: 0.22057537064278826

```
\usepackage{ amssymb }
\mathcal{S}
mathmode
```

Score: 0.2273911917630418

```
\xi
mathmode
```

Score: 0.23454487276981645

```
\usepackage{ stmaryrd }
\lightning
mathmode
```

The symbol is not in the list? [Show more](#)

Figura 4.1: Exemplo de uso da ferramenta *online* detexify (<https://detexify.kirelabs.org/classify.html>).

Além dessas ferramentas, você pode utilizar o LyX (<https://www.lyx.org/>), um editor de textos [What You See Is What You Mean \(WYSIWYM\)](#), que está disponível para Linux, Windows e Mac OS, e que é construído em cima do \LaTeX , e que permite se construir facilmente não só equações usando uma interface gráfica, mas também ambientes tabulares complexos para inserção em tabelas.

Finalmente, gostaria de indicar uma ferramenta simples, mas que pode ser bastante útil quando você quer incluir um símbolo matemático no seu documento mas esqueceu do nome dele em \LaTeX . Se este é o seu caso, visite a página do detexify <https://detexify.kirelabs.org/classify.html> e desenhe uma aproximação do símbolo no

janela indicada e um software de classificação vai indicar quais símbolos são mais parecidos com o seu desenho, qual é o nome do comando que representa o símbolo e o pacote ao qual ele pertence. Um exemplo do uso desta ferramenta pode ser visto na Figura 4.1.

4.2 Provas

Existem vários pacotes que auxiliam na criação de provas de diversos estilos. Neste modelo, incorporamos alguns desses pacotes, os quais descrevemos brevemente a seguir. O primeiro pacote a ser apresentado é o **bussproofs**, um pacote que permite que se construa árvores de provas no estilo de cálculo de seqüentes e outros sistemas de provas. Esse pacote define o comando `\fCenter`, que define um ponto central para uso no alinhamento horizontal. Abaixo vemos um exemplo do uso do pacote **bussproofs**, seguido dos comandos usados para gerá-lo, no Código 4.4, inclusive do comando `\fCenter`.

$$\frac{\frac{\Gamma, A, B \rightarrow B}{\Gamma, A \rightarrow (B \rightarrow C)}}{\Gamma \rightarrow (A \rightarrow (B \rightarrow C))}$$

```

1 \def\fCenter{\mbox{\Large$\rightarrow$}}
2 \begin{prooftree}
3   \Axiom$\Gamma, A, B\ \fCenter\ B$
4   \UnaryInf$\Gamma, A\ \fCenter\ (B \rightarrow C)$
5   \UnaryInf$\Gamma\ \fCenter\ (A \rightarrow (B \rightarrow C))$
6 \end{prooftree}

```

Código 4.4: Exemplo do uso do pacote **bussproofs**. Exemplo extraído de (Buss 2011).

O endereço <http://mirrors.ctan.org/macros/latex/contrib/bussproofs/testbp2.pdf> permite que se acesse um pequeno documento contendo exemplos do uso do pacote enquanto que o manual pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/bussproofs/BussGuide2.pdf> (Buss 2011).

Já o pacote **lplfitch** provê macros para a diagramação de provas por dedução natural no estilo “Fitch”, com as sub-provas indentadas e linhas de escopo. A tentativa

do uso desse pacote com pacotes das classes de documentos do KOMA gerou erros por conflito devido ao uso de pacotes descontinuados pelo T_EX mas que ainda funcionam com classes mais antigas como a `book` e `article`. Para que este pacote funcione com as classes do modelo KOMA, as duas linhas do Código 4.5 devem ser definidas no preâmbulo do seu documento.

```

1 \DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
2 \DeclareOldFontCommand{\bf}{\normalfont\bfseries}{\mathbf}

```

Código 4.5: Comandos necessários para o uso do pacote `lplfitch` (Etchemendy, Barker-Plummer e Zach 2013) com as classes do modelo KOMA.

O exemplo abaixo mostra uma prova no estilo produzido pelo pacote `lplfitch`, que pode ser incluído em um objeto `float figure` ou apresentado sozinho. Devido a como o pacote `lplfitch` foi implementado, sugiro que você coloque comandos escolhendo o espaçamento simples antes e duplo após a prova, como pode ser visto no Código 4.6. Caso isso não seja feito, os espaços entre as linhas das provas ficam muito grandes e o efeito não é muito bom.

1. $\forall x (Cube(x) \rightarrow Small(x))$	
2. $\exists x Cube(x)$	
3. $\boxed{a} Cube(a)$	
4. $Cube(a) \rightarrow Small(a)$	\forall Elim: 1
5. $Small(a)$	\rightarrow Elim: 4, 3
6. $\exists x Small(x)$	\exists Intro: 5
7. $\exists x Small(x)$	\exists Elim: 2, 3–6
8. $\exists x Cube(x) \rightarrow \exists x Small(x)$	\rightarrow Intro: 2–7
9. $(\forall x (Cube(x) \rightarrow Small(x)) \rightarrow (\exists x Cube(x) \rightarrow \exists x Small(x)))$	\rightarrow Intro: 1–8

O manual do pacote `lplfitch` pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/lplfitch/lplfitch.pdf> (Etchemendy, Barker-Plummer

```

1 \singlespacing
2 \fitchprf{}{
3   \subproof{\pline[1.]{\uni{x}{(Cube(x)\lif Small(x))}}}{
4     \subproof{\pline[2.]{\exi{x}{Cube(x)}}}{
5       \boxedsubproof[3.]{a}{Cube(a)}{
6         \pline[4.]{Cube(a)\lif Small(a)}[\lalle{1}]\
7         \pline[5.]{Small(a)}[\life{4}{3}]\
8         \pline[6.]{\exi{x}{Small(x)}}[\lexii{5}]
9       }
10      \pline[7.]{\exi{x}{Small(x)}}[\lexie{2}{3--6}]
11    }
12    \pline[8.]{\exi{x}{Cube(x)}\lif \exi{x}{Small(x)}}[\lifi{2--7}]
13  }
14  \pline[9.]{\brokenform{(\uni{x}{(Cube(x)\lif Small(x))}\lif){
15    \formula{(\exi{x}{Cube(x)} \lif \exi{x}{Small(x))}}}\lifi{1--8}]
16 }
17 \doublespacing

```

Código 4.6: Comandos necessários para o uso do pacote `lplfitch` (Etchemendy, Barker-Plummer e Zach 2013) com as classes do modelo KOMA.

e Zach 2013).

Diferente dos pacotes vistos acima, o pacote `natded` implementa ferramentas para criar provas de dedução natural nos estilos de Jaśkowski e Kalish-Montague. O exemplo abaixo mostra uma prova escrita no estilo de Kalish-Montague (exemplo extraído de (Francis Jeffry Pelletier 2020a)) e os comandos usados para gerá-la podem ser vistos no Código 4.7. A Linha 2 do Código 4.7 foi incluída para diminuir o espaçamento entre linhas pois o pacote `natded` gera caixas com linhas não contíguas quando o espaçamento é duplo, como neste modelo.

1.	<i>Show</i> $((P \rightarrow Q) \wedge (\neg R \rightarrow \neg Q)) \rightarrow (P \rightarrow R)$	2 – 13 Conditionalization
2.	$((P \rightarrow Q) \wedge (\neg R \rightarrow \neg Q))$	Supposition
3.	<i>Show</i> $(P \rightarrow R)$	4 – 13 Conditionalization
4.	P	Supposition
5.	$((P \rightarrow Q) \wedge (\neg R \rightarrow \neg Q))$	2 Repeat
6.	$(P \rightarrow Q)$	5 Simplification
7.	Q	4, 6 Modus Ponens
8.	$(\neg R \rightarrow \neg Q)$	5 Simplification
9.	<i>Show</i> R	10 – 13 Reductio ad Absurdum
10.	$\neg R$	Supposition
11.	$(\neg R \rightarrow \neg Q)$	8 Repeat
12.	$\neg Q$	10 , 11 Modus Ponens
13.	Q	7 Repeat

O pacote `natded` disponibiliza o manual em <http://mirrors.ctan.org/macros/latex/contrib/natded/natded.pdf> (Francis Jeffrey Pelletier 2020a), de onde foi extraído o exemplo acima, e a documentação extra em http://mirrors.ctan.org/macros/latex/contrib/natded/extended_doc.pdf (Francis Jeffrey Pelletier 2020b).

4.3 Especificações

Especificações formais usam notações matemáticas para modelar precisamente as propriedades que um sistema computacional deve possuir. Existem várias linguagens criadas para descrever essas propriedades. Neste modelo, incorporamos os pacotes das linguagens CSP, Z e Circus. Caso você não necessite usá-las, comente os comandos

\usepackage correspondentes.

```

1 \begin{equation*}
2   \setstretch{1.3}
3   \KMproof{
4     \cbbblk{
5       \proofline{((( P \rightarrow Q ) \land (\neg R \rightarrow
6         \neg Q ) ) \rightarrow ( P \rightarrow R ) ) }{2 -- 13
7         Conditionalization}
8     }{
9       \proofline{(( P \rightarrow Q ) \land (\neg R \rightarrow
10        \neg Q ) ) }{Supposition}
11     \cbbblk{
12       \proofline{( P \rightarrow R ) }{4 -- 13 Conditionalization}
13     }{
14       \proofline{ P }{ Supposition }
15       \proofline{(( P \rightarrow Q ) \land (\neg R \rightarrow
16        \neg Q ) ) }{2 Repeat}
17       \proofline{( P \rightarrow Q ) }{5 Simplification}
18       \proofline{ Q }{4, 6 Modus Ponens }
19       \proofline{(\neg R \rightarrow \neg Q ) }{5 Simplification}
20     \cbbblk{
21       \proofline{ R }{10 -- 13 Reductio ad Absurdum }
22     }{
23       \proofline{\neg R }{ Supposition}
24       \proofline{(\neg R \rightarrow \neg Q ) }{8 Repeat}
25       \proofline{\neg Q }{10 , 11 Modus Ponens}
26       \proofline{ Q }{7 Repeat}
27     }
28   }
29 }
30 }
31 \end{equation*}

```

Código 4.7: Exemplo do uso do pacote `natded`.

O pacote `zed-csp` é um pacote que foi desenvolvido baseado no pacote original `zed`, que implementa a diagramação no formato da linguagem Z e incluiu as definições para CSP. O exemplo abaixo mostra o uso do ambiente de definição genérica (`gendef`), criado pelo pacote `zed-csp`, e os comandos usados para gerar este exemplo, extraído de

(Spivey 1990), podem ser vistos no Código 4.8.

$[X, Y]$	=====
$first : X \times Y \rightarrow X$	
$\forall x : X; y : Y \bullet$	
$first(x, y) = x$	

```

1 \begin{gendif}[X,Y]
2   first: X \cross Y \fun X
3 \where
4   \forall x: X; y: Y @ \
5 \t1    first(x,y) = x
6 \end{gendif}

```

Código 4.8: Exemplo do uso do pacote `zed-csp`.

O pacote `zed-csp` possui dois manuais, um para a linguagem CSP <http://mirrors.ctan.org/macros/latex/contrib/zed-csp/csp2e.pdf> (Davies 2020) e o outro para Z <http://mirrors.ctan.org/macros/latex/contrib/zed-csp/zed2e.pdf> (Spivey 1990).

Existe um outro pacote para a criação de especificações em Z, o `objectz`. Entretanto eu não carreguei ambos os pacotes para verificar se `objectz` e `zed-csp` são compatíveis, já que definem ambientes com os mesmos nomes. Caso precise usar o `objectz`, eu sugiro que apenas substitua o pacote `zed-csp`. O manual do `objectz` pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/objectz/ozguide.pdf> (King 1990).

5 Desenhos e Animações

Neste capítulo, descreverei brevemente os pacotes de auxílio a desenhos que são recomendados para uso com este modelo. Como parte ainda experimental, adicionei um pacote que permite incluir animações em documentos [PDF](#), embora elas só possam ser visualizadas em leitores de [PDF](#) que processam JavaScript.

A gama de pacotes que facilitam a criação de desenhos em \LaTeX é imensa, e vai desde grandes ambientes de programação a pequenos pacotes auxiliares. Aqui, vamos ver brevemente alguns desses ambientes e pacotes auxiliares. Esse capítulo dá uma maior ênfase ao ambiente de geração de ilustrações definidos pela dupla de linguagens [Portable Graphics Format \(PGF\)](#)/[TikZ](#), sendo que [PGF](#) é uma linguagem de alto nível e [TikZ](#) é um conjunto de macros de alto nível que usa [PGF](#).

5.1 Qtree

O pacote `qtree` oferece suporte para o desenho de árvores, e é comumente utilizado em aplicações de linguística. Ele emprega uma sintaxe simples usando colchetes e calcula automaticamente os tamanhos dos ramos. A Figura [5.1](#) foi gerada usando o comando mostrado do Código [5.1](#).

Para maiores detalhes e outros exemplos do uso de `qtree`, você pode utilizar <http://mirrors.ctan.org/macros/latex/contrib/qtree/qtreenotes.pdf> (Siskind e Dimitriadis [2008](#)).

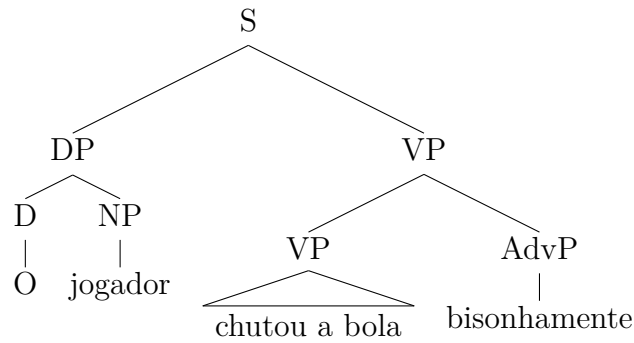


Figura 5.1: Exemplo simples de árvore gerada usando o pacote qtree.

```

1 \Tree [.S [.DP [.D O ] [.NP jogador ] ] [.VP \qroof{chutou a bola}.VP
2 [.AdvP bisonhamente ] ] ]

```

Código 5.1: Exemplo de código \LaTeX usado para gerar a árvore da Figura 5.1.

5.2 TikZ e Pacotes Auxiliares

O pacote `tikz` (geralmente escrito em documentos como *TikZ*) é provavelmente a ferramenta mais potente para a criação de gráficos em \LaTeX . Ele implementa várias funções de desenho e serve como base para vários outros pacotes associados que criam facilidades para que se produzam desenhos com certas especialidades. Till Tantau projetou as linguagens `PGF` e *TikZ*, sendo que o nome *TikZ* representa o acrônimo recursivo “*TikZ ist kein Zeichenprogramm*”, que em Português significa “*TikZ não é um programa de desenho*”.

Uma busca recente feita por mim no [CTAN](#) com o termo “tikz” gerou uma lista com 205 pacotes. Como o intuito aqui é o de introduzir o uso de *TikZ* e não cobrir de um grande número de pacotes, veremos aqui apenas alguns exemplos.

Outra opção bastante utilizada para desenhar em \LaTeX é o pacote `PSTricks`, que também é muito bom. Entretanto, devido a sua melhor compatibilidade com o `pdf \LaTeX` , escolhi o *TikZ* como pacote básico de desenho para este modelo.

Como no caso dos outros pacotes, o *TikZ* pode ser incluído simplesmente usando

o comando:

`\usepackage{tikz}`

Abaixo, no Código 5.2 nós vemos alguns comandos que podem ser necessários executar antes de se usar o TikZ, dependendo do tipo de desenho que deseje produzir. O comando da Linha 1 carrega o pacote `bclogo` e informa que o pacote gráfico a ser utilizado é o TikZ, pois o `bclogo` também funciona com o PSTricks. As Linhas 2 e 3 carregam dois pacotes que se baseiam no TikZ para realizar desenhos de grafos de dependência e redes, respectivamente, enquanto que as Linhas 4 e 5 carregam dois pacotes que foram implementados em formato de bibliotecas TikZ. Finalmente, as Linhas 6, 7 e 8 definem e configuram *layers* (camadas), que são necessárias para alguns casos.

```
1 \usepackage[tikz]{bclogo}
2 \usepackage{tikz-dependency}
3 \usepackage{tikz-network}
4 \usetikzlibrary{switching-architectures}
5 \usetikzlibrary{mindmap}
6 \pgfdeclarelayer{background}
7 \pgfdeclarelayer{foreground}
8 \pgfsetlayers{background,main,foreground}
```

Código 5.2: Exemplo de código \LaTeX usado para configuração do TikZ.

O pacote define o ambiente `tikzfigure`, que delimita o código de seu desenho. Geralmente colocamos o ambiente `tikzfigure` dentro de um ambiente `figure` para criar um objeto `float`, como vimos no Capítulo 3.

O desenho de linhas com e sem setas é extremamente simples, assim como o de outras primitivas gráficas, e as propriedades associadas a esses elementos. Abaixo, na Figura 5.2, vemos um exemplo simples de desenho feito usando várias características de primitivas, e que foi gerado pelo Código 5.3. Na Linha 1 do código, um ambiente `tikzpicture` foi criado e teve a escala 2 associada a ele, ou seja, o gráfico terá o dobro do tamanho definido internamente. A Linha 2 desenha as linhas do sistema de coordenadas

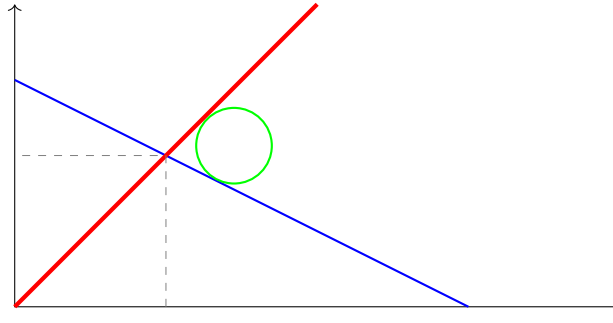


Figura 5.2: Exemplo simples de gráfico elaborado usando TikZ.

e a opção `<->` diz ao TikZ que setas devem ser desenhadas no início e final das linhas. As opções mostradas nas Linhas de 3 a 6 estabelecem as propriedades das primitivas desenhadas, como cor, espessura e padrão. Note como é simples desenhar uma linha, apenas definindo as coordenadas¹ dos pontos entre parenteses e alternando eles com os caracteres `--`.

```

1 \begin{tikzpicture}[scale=2]
2   \draw [<->] (0,2) -- (0,0) -- (4,0);
3   \draw [blue, thick] (0,1.5) -- (3,0);
4   \draw [red, ultra thick] (0,0) -- (2,2);
5   \draw [dashed, help lines] (1,0) -- (1,1) -- (0,1);
6   \draw [green, thick] (1.45,1.065) circle [radius=0.25];
7 \end{tikzpicture}

```

Código 5.3: Código L^AT_EX usado para gerar exemplo de gráfico da Figura 5.2.

O TikZ também nos permite desenhar facilmente gráficos de equações como pode ser visto na Figura 5.3, que foi gerada com o Código 5.4. Observe como é simples se gerar gráficos com funções conhecidas. Caso você queira desenhar gráficos de funções geradas por sequências de pontos, é só carregar as coordenadas dos pontos no formato TikZ e ligá-los por retas. Note, entretanto, que nesse caso, uma ampliação do pdf iria salienta a falta de suavidade do gráfico, dependendo da amostragem utilizada na geração dos pontos.

¹As coordenadas no TikZ são expressas em centímetros.

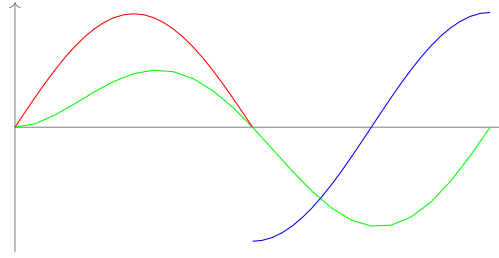


Figura 5.3: Exemplo de gráfico gerado com TikZ.

```

1 \begin{tikzpicture}[yscale=1.5]
2   \draw [help lines, ->] (0,0) -- (6.5,0);
3   \draw [help lines, ->] (0,-1.1) -- (0,1.1);
4   \draw [green,domain=0:2*pi] plot(\x,{(sin(\x r)*ln(\x+1))/2});
5   \draw [red,domain=0:pi] plot(\x,{sin(\x r)});
6   \draw [blue,domain=pi:2*pi] plot(\x,{cos(\x r)*exp(\x/exp(2*pi))});
7 \end{tikzpicture}

```

Código 5.4: Código L^AT_EX usado para gerar exemplo de gráfico da Figura 5.3.

Os manuais do TikZ podem ser acessados em <http://cremeronline.com/LaTeX/minimaltikz.pdf> (Crémer 2011) (manual introdutório) e <http://mirrors.ctan.org/graphics/pgf/base/doc/pgfmanual.pdf> (Tantau 2020) (manual oficial). Eu recomendo que tenha o manual oficial do TikZ, caso precise gerar desenhos com frequência para os seus documentos. Ele é extremamente detalhado e contém muitos exemplos. Além disso, sugiro a página <https://texample.net/tikz/examples/>, que armazena muitos exemplos de desenhos gerados com vários pacotes, e que podem servir de base para algum desenho que precise gerar.

- SA-TikZ - O pacote `sa-tikz` define a biblioteca Sa-TikZ que auxilia no desenho de arquiteturas de *switching* (comutação) e define os modelos Clos, Benes e Banyan e algumas variações destes modelos. O pacote permite que se configure aspectos da rede como as dimensões do módulo, a distância entre módulos e a fonte usada. Por exemplo, a Figura 5.4 mostra uma rede de *switching* Banyan-Omega. Para maiores detalhes, consulte o manual, que está disponível em <http://mirrors.ctan.org/>

graphics/pgf/contrib/sa-tikz/doc/sa-tikz-doc.pdf (Fiandrino 2014).

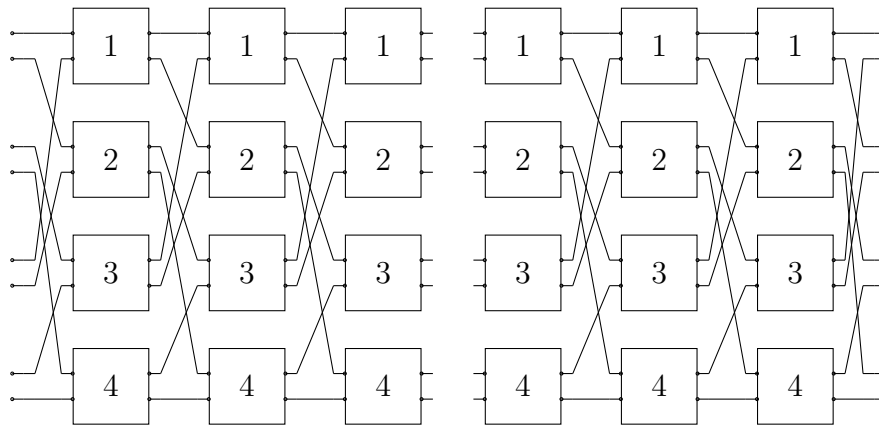


Figura 5.4: Exemplo de duas redes de *switching* Banyan-Omega geradas usando a biblioteca `switching-architectures` do pacote `sa-tikz`. Exemplo extraído de (Fiandrino 2014).

- Bclogo - O pacote `bclogo` permite que se criem caixas coloridas com a inclusão de logotipos, o que é interessante para chamar a atenção para alguns trechos do documento. Esse pacote depende do pacote `mdframed`. Assim, mensagens importantes, como a mostrada abaixo, podem ter a devida atenção dos leitores.

Você Sabia?

Você sabia que o consumo moderado de cerveja aumenta a densidade dos ossos em humanos? Um outro estudo mostrou que bebedores moderados têm um menor risco de doenças cardiovasculares do que os abstêmios!

O manual do pacote `bclogo` (em Francês) pode ser acessado em <http://mirrors.ctan.org/graphics/bclogo/doc/bclogo-doc.pdf> (Maxime C. Hupin 2016).

- TikZ-Dependency - O pacote `tikz-dependency` é um pacote que facilita a criação de grafos de dependência, comumente utilizados em algumas áreas de pesquisa, como grafos e processamento de linguagem natural.

Ele permite que facilmente se definam estilos para os nós, arestas e rótulos, facilitando enormemente a produção desses grafos. A Figura 5.5 mostra um exemplo simples feito

usando `tikz-dependency`. Seu manual pode ser acessado em <http://mirrors.ctan.org/graphics/pgf/contrib/tikz-dependency/tikz-dependency-doc.pdf> (Pighin 2012).

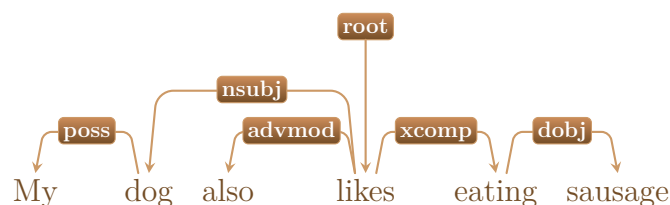


Figura 5.5: Exemplo de grafo de dependência criado usando `tikz-dependency`. Exemplo extraído de (Pighin 2012).

```

1 \begin{dependency}[theme=copper]
2   \begin{deptext}[column sep=0.2cm]
3     My \&[.5cm] dog \& also \&[.7cm] likes \&[.4cm] eating
4     \& sausage \\
5   \end{deptext}
6   \depedge{2}{1}{poss}
7   \depedge{4}{2}{nsubj}
8   \depedge{4}{3}{advmod}
9   \depedge{4}{5}{xcomp}
10  \depedge{5}{6}{dobj}
11  \deproot{4}{root}
12 \end{dependency}

```

Código 5.5: Código \LaTeX usado para gerar exemplo de gráfico da Figura 5.5 usando a biblioteca definida pelo pacote `tikz-dependency`.

- TikZ-Network - Existem várias ferramentas que facilitam o desenho de redes ou grafos, como Xfig ou Inkscape. Você pode utilizar uma dessas ferramentas e salvar o conteúdo desejado, de preferência em um formato vetorial, para posteriormente adicioná-las ao seu documento.

Uma abordagem diferente permite que você desenhe sua rede ou grafo diretamente no documento \LaTeX , o que possibilita a realização de ajustes de estilos e tamanhos de fontes, adição de equações, e outras tarefas, sem que se precise retornar a um

software externo. O pacote `tikz-network` permite que se crie e manipule desenhos de redes ou gráficos de maneira simples, gerando gráficos escaláveis que mantêm a qualidade quando o arquivo PDF é ampliado.

A Figura 5.6 mostra um exemplo simples de rede que foi gerada carregando-se dois arquivos de configuração, um contendo os nós e o outro, as arestas.

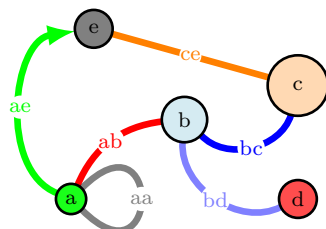


Figura 5.6: Exemplo de grafo criado usando `tikz-network`. Exemplo extraído de (Hackl 2019).

Esse pacote permite que se crie redes mais complexas, como a rede em multinível mostrada na Figura 5.7, que foi gerada usando os comandos vistos no Código 5.6. O manual desse pacote pode ser acessado em <http://mirrors.ctan.org/graphics/pgf/contrib/tikz-network/tikz-network.pdf> (Hackl 2019).

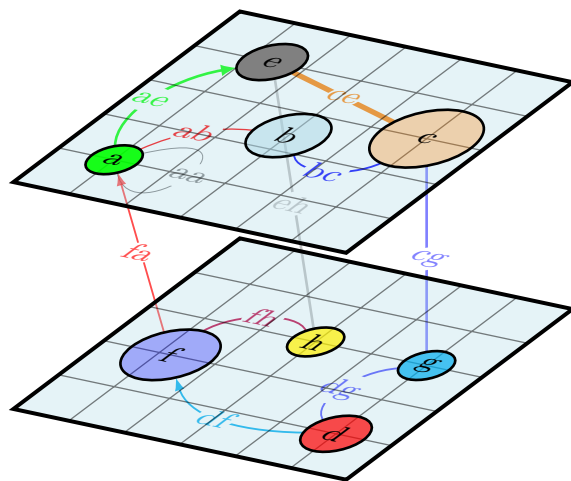


Figura 5.7: Exemplo de grafo em multinível criado usando `tikz-network`. Exemplo adaptado de (Hackl 2019).

```

1 \begin{tikzpicture}[multilayer=3d, scale=1.5]
2   \begin{Layer}[layer=1]
3     \Plane[x=-.5,y=-.5,width=2.5,height=3,grid=5mm]
4   \end{Layer}
5   \begin{Layer}[layer=2]
6     \Plane[x=-.5,y=-.5,width=2.5,height=3,grid=5mm]
7   \end{Layer}
8   \Vertices{capitulos/ml-vertices.csv}
9   \Edges{capitulos/ml-edges.csv}
10 \end{tikzpicture}

```

Código 5.6: Código L^AT_EX usado para gerar exemplo de gráfico da Figura 5.7 usando a biblioteca definida pelo pacote `tikz-network`.

- PGFPlots - O pacote `pgfplots` permite criar facilmente gráficos de alta qualidade em escalas lineares e logarítmicas em 2D e 3D usando uma interface amigável. A Figura 5.8 mostra um gráfico gerado usando esse pacote, com a sequência de comandos mostradas no código 5.7.

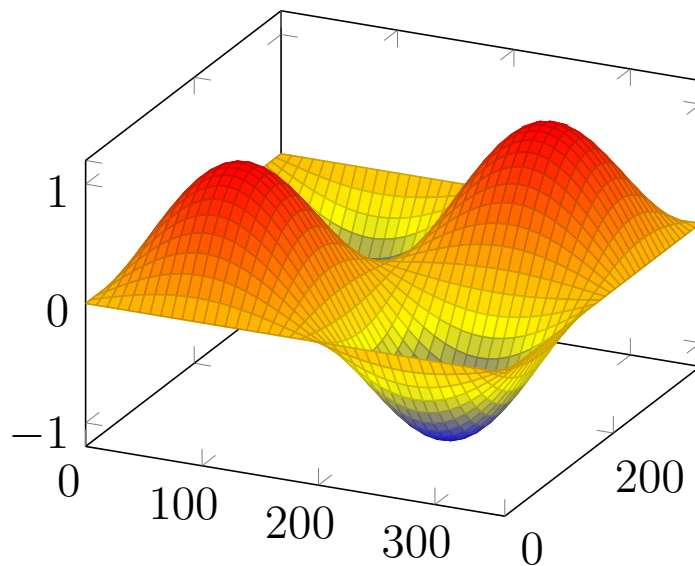


Figura 5.8: Exemplo de gráfico 3D impresso usando `pgfplots`.

Algumas rotinas chamadas por este pacote, principalmente as 3D, podem ser demoradas, acarretando em aumento razoável do tempo de compilação do seu documento.

```

1 \begin{tikzpicture}[scale=1.5]
2   \begin{axis}
3     \addplot3 [ surf, domain=0:360, samples=40, ] {sin(x)*sin(y)};
4   \end{axis}
5 \end{tikzpicture}

```

Código 5.7: Código \LaTeX usado para gerar exemplo de gráfico da Figura 5.8 usando o pacote `pgfplots`.

Deste modo, analise se esta é a melhor opção ou se você deveria gerar os gráficos 3D fora do \LaTeX e importá-los.

O manual desse pacote pode acessado em <http://mirrors.ctan.org/graphics/pgf/contrib/pgfplots/doc/pgfplots.pdf> (Feuersänger 2020), e muitos exemplos podem ser encontrados na Internet.

- Como mais um exemplo de tipo específico de desenho, apresento uma figura gerada com o auxílio da biblioteca `mindmaps`. Essa biblioteca pode ser usada juntamente com o `TikZ` para criar mapas mentais, que são diagramas usados para organizar informações visualmente. A Figura 5.9 mostra um exemplo que elenca os capítulos desse documento, detalhando as subseções de alguns capítulos.

Certos cuidados devem ser tomados ao criar esses mapas mentais. Muitos ramos ou textos longos criam problemas na diagramação, que devem ser corrigidos alterando a escala e os ângulos entre os ramos. Mais detalhes podem ser acessados no tutorial para iniciantes em `TikZ` do [Overleaf](https://www.overleaf.com/learn/latex/LaTeX_Graphics_using_TikZ:_A_Tutorial_for_Beginners_(Part_5)%E2%80%94Creating_Mind_Maps), que pode ser acessado em [https://www.overleaf.com/learn/latex/LaTeX_Graphics_using_TikZ:_A_Tutorial_for_Beginners_\(Part_5\)%E2%80%94Creating_Mind_Maps](https://www.overleaf.com/learn/latex/LaTeX_Graphics_using_TikZ:_A_Tutorial_for_Beginners_(Part_5)%E2%80%94Creating_Mind_Maps).

- `TikZ-3dplots` - O pacote `tikz-3dplot` permite definir sistemas de coordenadas tridimensionais para uso com desenhos `TikZ`. O usuário pode especificar a orientação do sistema de coordenadas principal para desenhar e um segundo sistema de coordenadas para realizar rotações e translações em relação ao sistema de coordenadas principal.

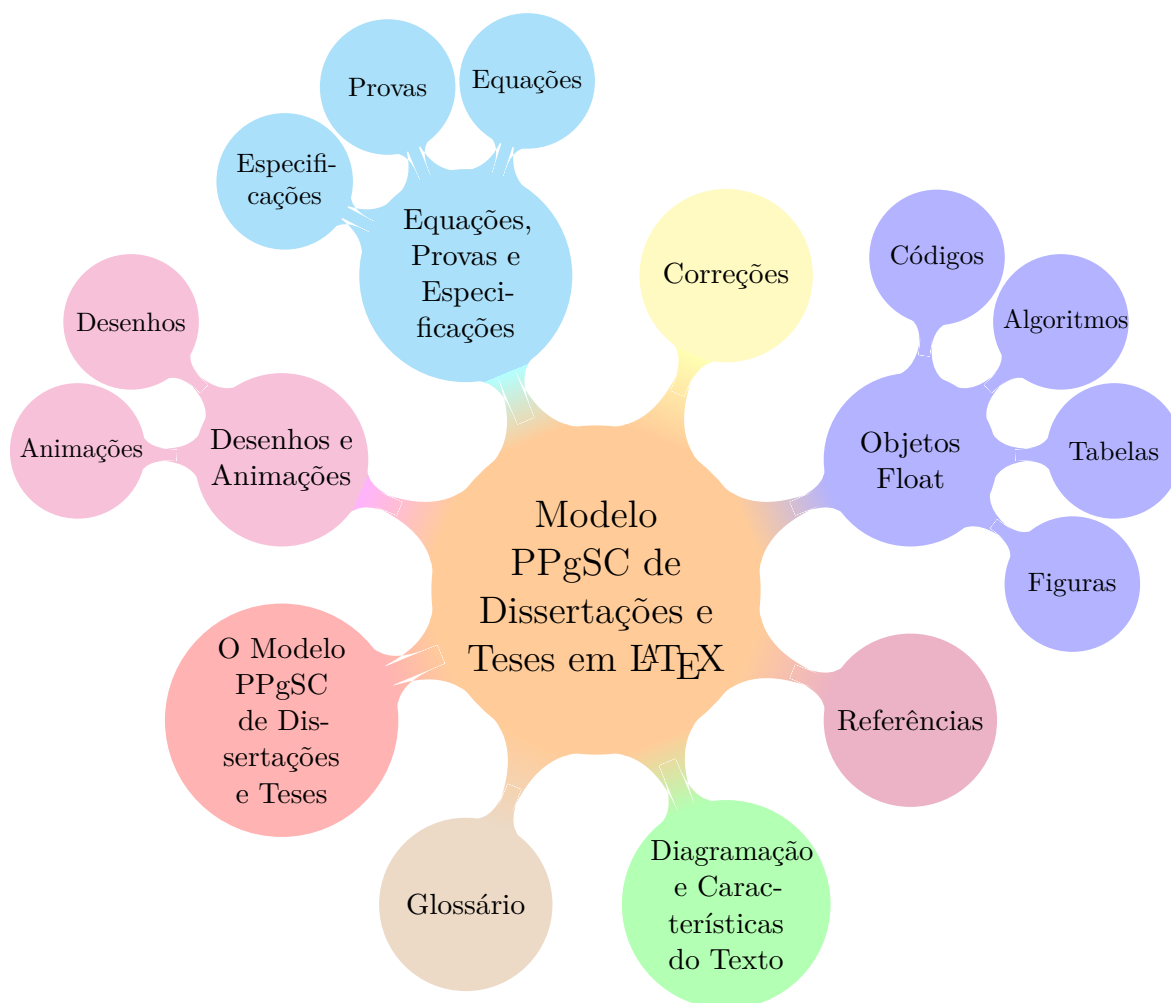


Figura 5.9: Exemplo de mapa mental criado usando biblioteca `mindmaps` para `TikZ`. O mapa mostra todos os capítulos contidos neste documento e expande alguns capítulos em suas seções.

O pacote ainda permite que se use coordenadas polares esféricas para desenhar.

É importante ressaltar que tudo o que você pode fazer com o pacote `tikz-3dplot` pode ser feito usando `TikZ` puro. A ideia é a de que o pacote `tikz-3dplot` facilita a tarefa de criação de desenhos tridimensionais e suas projeções bidimensionais.

Antes de desenhar uma figura, você deve definir a transformação para o sistema de coordenadas principal, que determina a posição da câmera virtual, além de definir variáveis que serão utilizadas em seus desenhos. Os comandos mostrados no Código 5.8 mostram as definições da posição da câmera virtual e de três ângulos.

```

1 \begin{tikzpicture}[scale=5,tdplot_main_coords]
2     \tdplotsetmaincoords{60}{110}
3     \pgfmathsetmacro{\rvec}{.8}
4     \pgfmathsetmacro{\thetavec}{30}
5     \pgfmathsetmacro{\phivec}{60}
6 \end{tikzpicture}

```

Código 5.8: Código TikZ, contendo comandos definidos no pacote `tikz-3dplot`, usado para gerar a Figura 5.10

A Figura 5.10 mostra um exemplo de gráfico produzido usando esse pacote, enquanto que o Código 5.9 apresenta o código usado para desenhá-lo.

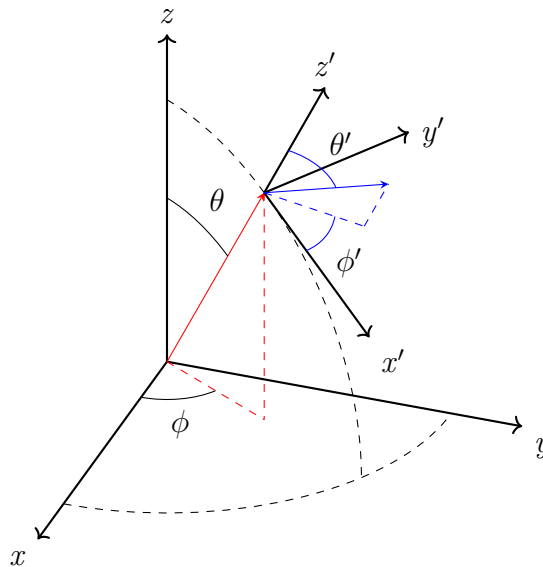


Figura 5.10: Exemplo de desenho produzido com o auxílio do pacote `tikz-3dplot`. Exemplo extraído de (Hein 2012).

O manual do `tikz-3dplot` pode ser acessado em http://mirrors.ctan.org/graphics/pgf/contrib/tikz-3dplot/tikz-3dplot_documentation.pdf (Hein 2012).

Existem várias *threads* de questões sobre desenhos 3D usando TikZ na área de T_EX do StackExchange <https://stackexchange.com>, seja com o auxílio de `tikz-3dplot` ou não. Recomendo que faça uma busca não só nessa página, mas também em páginas como a do T_EXample.net (<https://texample.net/>) antes de começar a

criar seus gráficos 3D.

```

1 \begin{tikzpicture}[scale=5,tdplot_main_coords]
2   \coordinate (O) at (0,0,0);
3   \draw[thick,->] (0,0,0) -- (1,0,0) node[anchor=north east]{$x$};
4   \draw[thick,->] (0,0,0) -- (0,1,0) node[anchor=north west]{$y$};
5   \draw[thick,->] (0,0,0) -- (0,0,1) node[anchor=south]{$z$};
6   \tdplotsetcoord{P}{\rvec}{\thetavec}{\phivec}
7   \draw[-stealth,color=red] (O) -- (P);
8   \draw[dashed,color=red] (O) -- (Pxy);
9   \draw[dashed,color=red] (P) -- (Pxy);
10  \tdplotdrawarc{(O)}{0.2}{0}{\phivec}{anchor=north}{$\phi$}
11  \tdplotsetthetaplanecoords{\phivec}
12  \tdplotdrawarc[tdplot_rotated_coords]{(0,0,0)}{0.5}{0}%
13  {\thetavec}{anchor=south west}{$\theta$}
14  \draw[dashed,tdplot_rotated_coords] (\rvec,0,0) arc (0:90:\rvec);
15  \draw[dashed] (\rvec,0,0) arc (0:90:\rvec);
16  \tdplotsetrotatedcoords{\phivec}{\thetavec}{0}
17  \tdplotsetrotatedcoordsorigin{(P)}
18  \draw[thick,tdplot_rotated_coords,->] (0,0,0)
19  -- (.5,0,0) node[anchor=north west]{$x'$};
20  \draw[thick,tdplot_rotated_coords,->] (0,0,0)
21  -- (0,.5,0) node[anchor=west]{$y'$};
22  \draw[thick,tdplot_rotated_coords,->] (0,0,0)
23  -- (0,0,.5) node[anchor=south]{$z'$};
24  \draw[-stealth,color=blue,tdplot_rotated_coords] (0,0,0) --(.2,.2,.2);
25  \draw[dashed,color=blue,tdplot_rotated_coords] (0,0,0) -- (.2,.2,0);
26  \draw[dashed,color=blue,tdplot_rotated_coords] (.2,.2,0) --(.2,.2,.2);
27  \tdplotdrawarc[tdplot_rotated_coords,color=blue]{(0,0,0)}{0.2}{0}%
28  {45}{anchor=north west,color=black}{$\phi'$}
29  \tdplotsetrotatedthetaplanecoords{45}
30  \tdplotdrawarc[tdplot_rotated_coords,color=blue]{(0,0,0)}{0.2}{0}%
31  {55}{anchor=south west,color=black}{$\theta'$}
32 \end{tikzpicture}

```

Código 5.9: Código TikZ, contendo comandos definidos no pacote tikz-3dplot, usado para gerar a Figura 5.10

- TikZ-qtree - O pacote tikz-qtree implementa uma parte dos comandos do pacote qtree usando TikZ. A sintaxe é a mesma do qtree, e segundo os autores, as características mais básicas daquele pacote estão implementadas. Além disso, comandos

TikZ podem ser incorporados na descrição das árvores. O manual desse pacote pode ser acessado em <http://mirrors.ctan.org/graphics/pgf/contrib/tikz-qtree/tikz-qtree-manual.pdf> (Chiang 2012).

- **Animate** - O pacote `animate` permite criar animações em **PDF** usando JavaScript e em **Scalable Vector Graphics (SVG)**. Infelizmente, este pacote só funciona os visualizadores **PDF** que suportam JavaScript, como o Acrobat Reader ©. No caso de animações **SVG**, essa ferramenta permite que elas sejam usadas em navegadores. Essa pode ser uma poderosa ferramenta na demonstração de algo dinâmico. Para os interessados, sugiro que estudem e testem os exemplos do manual desse pacote, acessível em <http://mirrors.ctan.org/macros/latex/contrib/animate/animate.pdf> (Grahm 2020).
- **Outros pacotes** - Existem vários outros pacotes de desenho que se baseiam no TikZ disponíveis gratuitamente, mas que não estão armazenados no **CTAN**. Um deles é o `tkz-2d`, que pode ser baixado de <https://texample.net/tikz/examples/tkz-2d/> (Matthes 2007). Além disso, temos várias bibliotecas prontas para uso com o TikZ, como a `decoration.fractals`, que nos permite desenhar facilmente curvas fractais como a curva de Koch de níveis 0, 1, 2 e 3, mostradas na Figura 5.11.

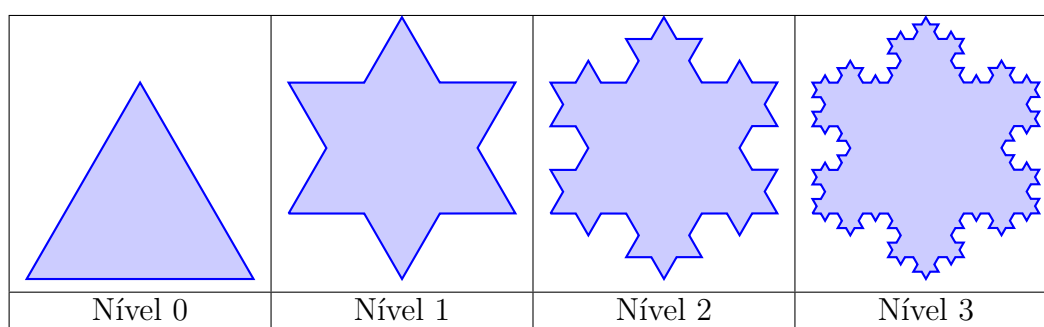


Figura 5.11: Exemplo de grafo em multinível criado usando `tikz-network`. Exemplo extraído de <http://mirrors.ctan.org/info/visualtikz/VisualTikZ.pdf>.

Os comandos necessários para desenhar as curvas e organizá-las no ambiente tabular

podem ser vistos no Código 5.10. Note a simplicidade e elegância do código recursivo usado para desenhar as curvas.

```

1 \begin{center}
2 \begin{tabular}{|c|c|c|c|} \hline
3   \begin{tikzpicture}[decoration=Koch snowflake,draw=blue,
4     fill=blue!20,thick]
5     \filldraw (0,0) -- ++(60:3) -- ++(-60:3) -- cycle ;
6   \end{tikzpicture}
7   &
8   \begin{tikzpicture}[decoration=Koch snowflake,draw=blue,
9     fill=blue!20,thick]
10    \filldraw decorate{ (0,0) -- ++(60:3) -- ++(-60:3) -- cycle };
11  \end{tikzpicture}
12  &
13  \begin{tikzpicture}[decoration=Koch snowflake,draw=blue,
14    fill=blue!20,thick]
15    \filldraw decorate{ decorate{ (0,0) -- ++(60:3) -- ++(-60:3)
16      -- cycle }};
17  \end{tikzpicture}
18  &
19  \begin{tikzpicture}[decoration=Koch snowflake,draw=blue,
20    fill=blue!20,thick]
21    \filldraw decorate{ decorate{ decorate{ (0,0) -- ++(60:3)
22      -- ++(-60:3) -- cycle }}};
23  \end{tikzpicture}
24  \\ \hline
25  Nível 0 & Nível 1 & Nível 2 & Nível 3 \\ \hline
26 \end{tabular}
27 \end{center}

```

Código 5.10: Código L^AT_EX usado para gerar exemplos de curvas fractais da Figura 5.11 usando a biblioteca `decoration.fractal` do TikZ.

Eu recomendo que sempre que precise desenhar algo mais complexo em TikZ, faça uma busca na Internet para avaliar se não existe algum código disponível que possa ajudá-lo. Em alguns casos, existem vários pacotes que têm propostas similares, como os pacotes `forest` e `tikz-qtree`. Nestes caso, você deve pesquisar e definir qual é a melhor opção para você.

Outra possibilidade é a utilização de ferramentas de desenho que salvam em formato TikZ. Existem várias ferramentas que permitem se criar desenhos vetoriais e salvá-los na linguagem TikZ. Dentre estas, destaco a Geogebra (<https://www.geogebra.org/>), uma ótima ferramenta para criar e visualizar funções, objetos geométricos e superfícies, e que possui uma página introdutória no [Overleaf](#) ensinando a gerar código TikZ. Além dela, você pode utilizar as ferramentas TikZit (<https://tikzit.github.io/>) e TikzEdt (<http://www.tikzedt.org/>), que permitem que você crie ilustrações usando linguagem TikZ ou comandos interativos, e visualize os efeitos das alterações imediatamente.

Uma outra possibilidade é a utilização do Inkscape, um programa de desenho vetorial, que salva os desenhos em formato [SVG](#), seguido da conversão dos arquivos SVG em TikZ usando uma ferramenta externa, como o SVG2TikZ <https://github.com/xyz2tex/svg2tikz> já que o Inkscape não dá mais suporte a essas conversões.

5.3 Asymptote

Asymptote é uma poderosa linguagem descritiva de gráficos vetoriais para desenhos técnicos. Ela foi inspirada em METAPOST mas possui uma sintaxe parecida com C++. Os programas escritos em Asymptote devem ser processados usando o programa `asy`, que gerará uma saída no formato [Encapsulated PostScript \(EPS\)](#) (default) ou no formato [PDF](#) (caso explicitamente especificado). Abaixo vemos as chamadas usadas para processar o arquivo `test.asy` em uma saída dos tipos [EPS](#) e [PDF](#).

```
asy -V test
asy -V -f pdf test
```

Entretanto, o Asymptote, permite que se coloque o código Asymptote dentro do arquivo \LaTeX , encapsulado no ambiente `asy`. Então, após o processamento do arquivo pelo processador \LaTeX , o Asymptote deve ser executado e depois o o processador \LaTeX

deve ser executado novamente, como visto abaixo. Lembro novamente, que este fluxo de processamento deve ser escolhido nas opções de sua ferramenta de edição \LaTeX ou executado diretamente na linha de comando.

```
pdflatex DissertacaoPPgSC
asy DissertacaoPPgSC-*.asy
pdflatex DissertacaoPPgSC
```

Diferente do TikZ , o Asymptote usa a unidade de tamanho ponto (`point`), que equivale a $0,035\text{cm}$. Para usar centímetro como a unidade do Asymptote, deve-se usar o comando abaixo:

```
unitsize(1cm);
```

A Figura 5.12 mostra um exemplo de gráfico gerado usando Asymptote, que representa o algoritmo recursivo de subdivisão de curvas de Bézier. Neste caso, optei por gerar o gráfico em formato **PDF** fora da ferramenta \LaTeX e importá-la usando o comando `\includegraphics`.

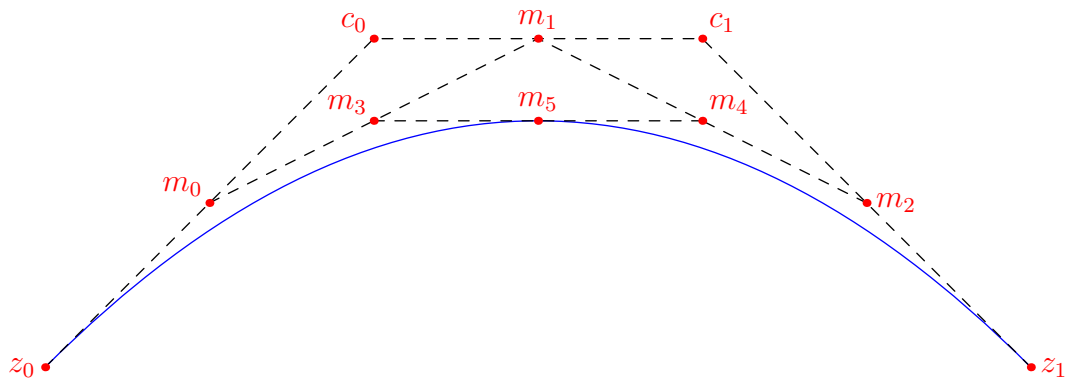


Figura 5.12: Ilustração do algoritmo recursivo de subdivisão de curvas de Bézier.

O Código 5.11 mostra os comandos em Asymptote usados para gerar a Figura 5.12. Note como é simples e intuitivo desenhar retas e posicionar os nomes dos pontos, e a definição da função auxiliar `midpoint`.

O Asymptote também permite que se construa animações dos gráficos gerados

```

1  import beziercurve;
2
3  pair midpoint(pair a, pair b) {return interp(a,b,0.5);}
4
5  pair m0=midpoint(z0,c0);
6  pair m1=midpoint(c0,c1);
7  pair m2=midpoint(c1,z1);
8
9  draw(m0--m1--m2,dashed);
10 dot("$m_0$",m0,NW,red);
11 dot("$m_1$",m1,N,red);
12 dot("$m_2$",m2,red);
13
14 pair m3=midpoint(m0,m1);
15 pair m4=midpoint(m1,m2);
16 pair m5=midpoint(m3,m4);
17
18 draw(m3--m4,dashed);
19 dot("$m_3$",m3,NW,red);
20 dot("$m_4$",m4,NE,red);
21 dot("$m_5$",m5,N,red);

```

Código 5.11: Código Asymptote usado para gerar a Figura 5.12.

usando seu pacote `animation`, que empilha múltiplas imagens em um [Graphic Interchange Format \(GIF\)](#) animado ou um vídeo [Moving Picture Experts Group \(MPEG\)](#) usando o programa `convert` do software ImageMagick. Assim como no caso do `TikZ`, essas animações só podem ser visualizadas em leitores de [PDF](#) que suportem JavaScript ou em browsers.

O Asymptote provê uma interface gráfica bem simples, a `xasy`, que permite que se veja imediatamente o resultado da manipulação dos elementos gráficos. Como a interface não é tão completa, o usuário pode preferir salvar o código do gráfico e completar a programação fora da ferramenta.

Uma boa referência inicial é o tutorial disponível em https://asymptote.sourceforge.io/asymptote_tutorial.pdf (Staats III 2021). Caso precise de infor-

mações mais detalhadas, você pode acessar o manual do Asymptote em <http://mirrors.ctan.org/graphics/asymptote/doc/asymptote.pdf> (Hammerlindl, Bowman e Prince 2020a). Finalmente, você também pode obter o cartão de referência dos comandos Asymptote no endereço <http://mirrors.ctan.org/graphics/asymptote/doc/asyRefCard.pdf> (Hammerlindl, Bowman e Prince 2020b).

6 Correções

Durante a escrita de uma dissertação ou tese, é importante que os orientadores possam sinalizar correções e comentar trechos de texto, figuras, tabelas e outros elementos. Isto pode ser feito por anotações feitas diretamente no arquivo [PDF](#) ou usando um pacote que implemente marcações de correções.

Neste modelo, selecionamos o pacote `changes` para este fim. O pacote `changes` permite que se notifique mudanças no documento com identificação do indivíduo que as fez, facilitando a comunicação quando mais de duas pessoas estão alterando o documento.

O Código [6.1](#) descreve os comandos usados para configurar o pacote para funcionar com este documento, definindo os nomes dos usuários que irão interagir com o documento e o modo atual do documento, o `draft`.

```
1 \usepackage[draft,markup=underlined]{changes}
2 \definechangesauthor[name={Bruno},color=violet]{Bruno}
3 \definechangesauthor[name={Hari},color=purple]{Hari}
4 \definechangesauthor[name={Salvor},color=olive]{Salvor}
```

Código 6.1: Código \LaTeX usado para definir as informações referentes ao pacote `changes`.

O pacote `changes` permite que se altere da versão `draft`, que exibe os comentários e mudanças, para a versão `final` com a alteração de apenas um comando, trocando a Linha 1 do Código [6.1](#) pela linha do Código [6.2](#).

Na primeira definição mostrada, usamos o parâmetro `draft`, indicando que a versão atual do documento mostrará os comentários e mudanças, e a parâmetro `markup`

```
1 \usepackage[final]{changes}
```

Código 6.2: Código L^AT_EX usado para definir o formato do documento como **final** ao invés de **draft** com relação ao pacote **changes**.

com o estilo **underlined** associado a ele. Os estilos de **markup** (marcação) disponíveis são:

- **default** - marcação padrão para comentários e textos adicionados, deletados e destacados;
- **underlined** - sublinhado para textos adicionados, sublinhado ondulado para textos destacados, e padrão para textos deletados e comentários;
- **bfit** - negrito para textos adicionados, itálico para textos deletados, e padrão para textos deletados e comentários;
- **nocolor** - sem cores para marcações, sublinhado para textos adicionados, sublinhado ondulado para textos destacados, e padrão para textos deletados e comentários;

O pacote ainda permite que você defina individualmente o estilo de marcação para cada um dos quatro tipos de alterações disponíveis. Para mais detalhes, consulte o manual do pacote.

Os cinco comandos usados para efetuar mudanças e fazer comentários são apresentados com exemplos simples.

- `\added` - Marca textos adicionados. Adicione isso porque é necessário.^{Hari} O Código 6.3 mostra como isso foi feito.

```
1 \added[id=Hari]{Adicione isso porque é necessário.}
```

Código 6.3: Código do **changes** usado para sugerir a adição de texto.

- `\deleted` - Marca textos deletados. ~~Texto removido porque estava errado.~~^{Salvor} O Código 6.4 mostra como isso foi feito.

[Bruno 1]

Os comentários no estilo todo ficam localizados na borda do documento.

```
1 \deleted[id=Salvor]{Texto removido porque estava errado.}
```

Código 6.4: Código do `changes` usado para sugerir a remoção de texto.

- `\replaced` - Marca textos deletados e suas reposições. Importante para que se visualize sugestões de correções de texto.^{Hari} O Código 6.5 mostra como isso foi feito.

```
1 \replaced[id=Hari]{Importante para que se visualize sugestões de
2 correções de texto.}{Não é importante.}
```

Código 6.5: Código do `changes` usado para sugerir a troca de texto.

- `\highlight` - Destaca trechos de texto como este aqui.^{Bruno} O Código 6.6 mostra como isso foi feito.

```
1 \highlight[id=Bruno]{como este aqui.}
```

Código 6.6: Código do `changes` usado para destacar texto.

- `\comment` - Cria um comentário e coloca-o no documento de acordo com a opção definida. Neste caso, estamos usando a opção `todo`, como visto no comentário inserido acima. O Código 6.7 mostra como isso foi feito.

```
1 \comment[id=Bruno]{Veja o manual do pacote \texttt{changes} para ver as
2 outras opções.}
```

Código 6.7: Código do `changes` usado para destacar texto.

É importante frisar que comentários podem ser incluídos nos outros quatro comandos. Inclua exemplo. Os comandos usados para fazer tal alteração podem ser vistos no Código 6.8.

No sítio [CTAN](http://mirrors.ctan.org/macros/latex/contrib/changes/changes) existem dois manuais para o pacote `changes`, um sem o código fonte do pacote (<http://mirrors.ctan.org/macros/latex/contrib/changes/changes>).

[Bruno 2]

Veja o manual do pacote `changes` para ver as outras opções.

[Hari 1]

Veja no manual.


```
1 \added[id=Hari, comment={Veja no manual.}]{Inclua exemplo.}
```

Código 6.8: Código do `changes` que mostra um comentário dentro de um comando de adição.

[english.pdf](#)) (Kleinod 2020) e outro com o código fonte (<http://mirrors.ctan.org/macros/latex/contrib/changes/changes.english.withcode.pdf>).

7 Referências

O pacote `hyperref` (S. Rahtz e Oberdiek 2021) estende a funcionalidade de todos os comandos que implementam referências cruzadas do \LaTeX , como sumário, bibliografia, listas de figuras e tabelas, para produzir comandos especiais que geram *links* de hipertexto. O pacote permite ainda a criação de *links* para documentos externos e URLs.

O Código 7.1 mostra os comandos usados para carregar o pacote `hyperref`, com as opções `colorlinks`, que colore os *links* das referências, `hyperindex`, que indica que o índice deve conter *hyperlinks*, `plainpages=false`, que força que as âncoras das páginas sejam nomeadas em números arábicos, `pdfusetitle`, que lê as informações de título, autor, etc. e as adiciona ao arquivo PDF, e `pdflang=pt-BR`, que identifica a linguagem do documento como sendo Português do Brasil. O segundo comando, na Linha 3, associa informações aos campos que serão lidos pelo pacote `hyperref`, caso a opção `pdfusetitle` esteja ativa, e associados aos campos de informação do arquivo PDF.

```
1 \usepackage[colorlinks, hyperindex, plainpages=false, pdfusetitle,  
2 pdflang=pt-BR]{hyperref}  
3 \hypersetup{pdftitle={Modelo PPgSC de Dissertações e Teses em LaTeX},  
4 pdfauthor={Bruno Motta de Carvalho},  
5 pdfsubject={Manual do modelo LaTeX do PPgSC-UFRN},  
6 pdfkeywords={LaTeX, diagramação, Modelo PPgSC}}
```

Código 7.1: Carregamento do pacote `hyperref` com as opções usadas neste modelo e associação de informações que serão adicionadas ao arquivo PDF.

Você pode facilmente adicionar *links* para URLs usando o comando `\url`. As definições e exemplos do uso de outros comando e macros podem ser acessados no manual do pacote, disponível em <http://mirrors.ctan.org/macros/latex/contrib/hyperref/doc/hyperref-doc.pdf> (S. Rahtz e Oberdiek 2021).

O pacote `BIBLATEX` provê ferramentas bibliográficas avançadas para uso em conjunto com `LATEX`, sendo uma completa reimplementação das ferramentas disponibilizadas pela distribuição `LATEX`. O `BIBLATEX` trabalha em conjunto com o programa de *backend* `biber`, que é usado para processar entradas em formato `BibTEX`. De posse das entradas, o `BIBLATEX` então ordena as referências, gera rótulos e a saída da bibliografia.

A formatação da bibliografia é controlada por macros `LATEX` padrão e pode ser configurada para a criação de novos estilos de bibliografia, bem como estilos de citação. Esse pacote ainda provê suporte a múltiplas bibliografias, que podem ser ordenadas por tópicos ou separadamente, com ordens diferentes. Ele ainda provê suporte completo a Unicode. Esse pacote possui vários pacotes incompatíveis, que na sua maioria são pacotes de referências bibliográficas ou referências cruzadas. Isso acontece para que definições contrastantes não resultem em comportamentos inesperados no processamento do seu documento. Como exemplo, cito os pacotes `backref`, `chapterbib`, `citeref` e `natbib`.

As principais desvantagens do `BIBLATEX` são que alguns periódicos, conferências e editoras podem não aceitar documentos que usem `BIBLATEX`, se tiverem seu próprio estilo com seu arquivo `.bst` compatível com `natbib`, além da dificuldade da inclusão de bibliografias criadas por `BIBLATEX` em um documento, como algumas editoras exigem. Para realizar esta última tarefa, o usuário tem que comentar os comandos do `BIBLATEX` e usar um outro pacote para carregar as referências no formato `BibTEX`. Essa não é uma preocupação no nosso caso, mas incluí essa explicação para o caso de você se deparar com esse problema quando submetendo um artigo para publicação.

O Código 7.2 mostra o carregamento do pacote `BIBLATEX` com as opções

`bibstyle`, que determina o estilo das referências bibliográficas, `citestyle`, que determina o estilo das citações no texto, `maxcitenames`, que determina o número máximo de autores que aparecerão nas citações, `maxbibnames`, que determina o número máximo de autores que aparecerão nas referências, `hyperref`, que indica que se deve transformar as referências e citações em links clicáveis, `backref`, que indica que referências reversas da bibliografia para o texto serão incluídas, e `backrefstyle`, que indica que qualquer sequência de três ou mais páginas consecutivas deve ser comprimida para uma faixa de valores. Para conhecer outras opções e comandos disponibilizados pelo BIB_{La}TEX, consulte seu manual em <http://mirrors.ctan.org/macros/latex/contrib/biblatex/doc/biblatex.pdf> (Kime, Wemheuer e Lehman 2020).

```
1 \usepackage[bibstyle=authoryear, citestyle=authoryear, maxcitenames=3,  
2 maxbibnames=20, hyperref=true, backref=true, backrefstyle=three]  
3 {biblatex}
```

Código 7.2: Carregamento do pacote BIB_{La}TEX com as opções usadas neste modelo.

O BIB_{La}TEX ainda permite que se use o Bib_{TeX} como *backend*, usando-o para ordenar as referências, mas não permite formatação de arquivos `.bst`, que determinam estilos de referências bibliográficas. Por outro lado, biber permite que se trabalhe com muito mais entradas e tipos de campos de dados nos arquivos `.bib`, funciona com arquivos `.bib` codificados com UTF-8 e permite um maior controle da ordenação das referências. Para maiores detalhes, consulte o manual em <http://mirrors.ctan.org/biblio/biber/documentation/biber.pdf> (Philip Kime 2020).

▲ Cuidado!

Apesar de ter selecionado o biber como opção de processamento de bibliografia nas opções do T_EXstudio 3.1.1, o mesmo não executou automaticamente o biber. Deste modo, tive que executá-lo na linha de comando de um terminal.

Para repetir o comportamento definido neste modelo usando o `BIBTEX`, você deve incluir o `natbib`, que define muitos arquivos de estilo `.bst`. Apesar da linguagem definida pelo `BIBTEX` para a criação de estilos de bibliografia ser complicada, você pode usar a ferramenta `makebst` para criar seu próprio estilo. Seu manual está disponível em <http://mirrors.ctan.org/macros/latex/contrib/custom-bib/makebst.pdf>, enquanto que maiores detalhes sobre `natbib` podem ser vistos no seu manual, disponível em <http://mirrors.ctan.org/macros/latex/contrib/natbib/natbib.pdf> Daly 2010. Finalmente, você precisaria usar o pacote `backref` para habilitar as referências reversas da bibliografia para o texto, algo que é feito diretamente pelo `BIBLATEX`, no nosso caso. O manual do `backref` pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/hyperref/doc/backref.pdf> (D. Carlisle e S. Rahtz 2021).

8 Glossário, Acrônimos e Índice

O pacote escolhido para gerenciar a criação e manipulação de índices, listas de símbolos, acrônimos e glossário foi o `glossaries`. Esse pacote precisa das definições em Português contidas no `glossaries-portuges`, caso você esteja escrevendo seu documento em Português. Você precisa confirmar que os dois pacotes estão instalados no seu sistema caso queira usá-los. Caso você deseje utilizar características mais avançadas, sugiro o pacote `glossaries-extra`, que estende as funcionalidades providas pelo `glossaries` e permite, por exemplo, utilizar o pacote `bib2gls`, brevemente descrito adiante.

É importante frisar que as definições do pacote `glossaries-portuges` contidas no arquivo `glossaries-portuges.dtx` são automaticamente carregadas quando um dos outros dois pacotes é carregado e detecta a opção `portugues`, `portuges`, `brazil` ou `brazilian` no carregamento do pacote `babel`.

▲ Problema Comum

Lembre-se de que caso esteja usando um editor integrado L^AT_EX e não esteja utilizando o T_EX para a ordenação das entradas dos glossários, você deve se certificar que um comando para a geração do glossário deve ser executado na cadeia de comandos do editor, como `makeglossaries`, `makeglossaries-lite` ou `bib2gls`.

O pacote `glossaries` gera conflitos com as classes padrão do L^AT_EX `book` e `memoir`, sendo que a classe `memoir` é utilizada como base da classe `abntex2`. Entretanto, o pacote `glossaries` não gera conflitos com a classe `scrbook` da KOMA, o que nos

possibilita usá-lo aqui.

O pacote `glossaries` nos permite construir listas de acrônimos, de símbolos e índices remissivos, embora não seja tão utilizada neste último caso. Aqui eu explicarei apenas o uso básico desse pacote. Para maiores detalhes, consulte o guia introdutório e o manual do pacote, que estão disponíveis em <http://mirrors.ctan.org/macros/latex/contrib/glossaries/glossaries-user.pdf> (N. L. Talbot 2020b) e <http://mirrors.ctan.org/macros/latex/contrib/glossaries/glossariesbegin.pdf> (N. L. Talbot 2020c), respectivamente.

Depois das definições serem carregadas em algum momento no preâmbulo de seu documento, i.e., antes do `\begin{document}`, você pode gerar as referências a eles usando um dos comandos abaixo, onde `chave` é o apelido do acrônimo a ser incluído, e o segundo comando converte a primeira letra do termo para maiúsculo.

`\gls{chave}`

`\Gls{chave}`

Você pode ainda especificar um símbolo usando um rótulo (*label*) para referenciá-lo e o campo `symbol`, você deve referenciá-lo usando o comando `\glsymbol`. O Código 8.1 mostra um exemplo de definição de símbolo, onde o campo `sort` define o string pelo qual este símbolo será ordenado.

```
1 \newglossaryentry{emptyset}
2 {
3   name={\ensuremath{\emptyset}},
4   sort={conjunto vazio},
5   description={conjunto contendo zero elementos}
6 }
```

Código 8.1: Definição de um símbolo como entrada de glossário.

Você pode concentrar suas definições em um só arquivo ou as dividir de acordo

com sua organização preferida, mas lembre-se de incluir os arquivos de definições usando os comandos `\loadglsentries` ou `\input`.

O Código 8.2 mostra algumas linhas do arquivo `Acronimos.tex`, que contém as definições dos acrônimos usados neste documento (veja a Figura 1.1). Nos comandos `\newacronym` abaixo, o primeiro campo entre chaves indica o apelido do acrônimo, i.e., o nome pelo qual você o identificá, enquanto que o segundo e terceiro campos identificam as formas curta e longa do acrônimo. No nosso modelo, o nome longo é exibido na primeira menção do acrônimo e o curto nas subsequentes. Entretanto, este é um comportamento que pode ser modificado usando o comando `\setacronymstyle`, caso deseje.

```
1 \newacronym{ppgsc}{PPgSC}{Programa de Pós-graduação em Sistemas e
2 Computação}
3 \newacronym{koma}{KOMA}{KOMA-Script}
4 \newacronym{ctan}{CTAN}{Comprehensive TeX Archive Network}
5 \newacronym{scrbook}{\texttt{scrbook}}{(classe livro do ambiente KOMA)}
6 \newacronym{ufrn}{UFRN}{Universidade Federal do Rio Grande do Norte}
7 \newacronym{dimap}{DIMAp}{Departamento de Informática e Matemática
8 Aplicada}
9 \newacronym{ccet}{CCET}{Centro de Ciências Exatas e da Terra}
10 \newacronym{overleaf}{Overleaf}{}
11 \newacronym{utf}{UTF}{Unicode Transformation Format}
12 \newacronym{pdf}{PDF}{Portable Document Format}
13 \newacronym{tikz}{Ti\textit{k}Z}{Ti\textit{k}Z \textit{ist kein
14 Zeichenprogramm}}
```

Código 8.2: Parte das definições de acrônimos usados neste documento, localizadas no arquivo `editaveis/Acronimos.tex`.

O pacote ainda permite que se definam termos de glossários com descrições que podem ser longas, usando as macros `\newglossaryentry` e `\longnewglossaryentry`. Você pode ainda definir plurais para os termos e usá-los através dos comandos abaixo:

<code>\glspl{chave}</code>
<code>\Glspl{chave}</code>

Caso deseje imprimir os glossários sem localização de páginas onde os símbolos aparecem, use um dos comandos abaixo:

```
\printunsrtglossary  
\printunsrtglossaries
```

No caso específico da lista de acrônimos, você pode gerá-la usando um dos comandos abaixo, que são equivalentes.

```
\printacronyms[title=Lista de Acrônimos, toctitle=Lista de Acrônimos]  
\printglossary[type=acronym, title=Lista de Acrônimos,  
toctitle=Lista de Acrônimos]
```

▲ Lembre-se!

Caso deseje somente utilizar o glossário de acrônimos, use a opção `nomain` quando carregando o pacote `glossaries`. Isso desabilita o uso do glossário principal.

A maioria dos usuários prefere exibir uma lista automaticamente ordenada contendo apenas as entradas citadas no documento. O pacote `glossaries` provê três opções: o uso do \TeX , do comando `makeindex` ou do pacote `xindy` para ordenar as entradas incluídas no texto. Além dessas opções você pode usar o pacote `bib2gls`, que só funciona com o `glossaries-extra`, que também permite que as entradas não sejam ordenadas. Além destas possibilidades, o pacote `glossaries-extra` oferece outras opções de configuração para usuários mais avançados. Seu manual pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/glossaries-extra/glossaries-extra-manual.pdf> (N. L. Talbot 2020a).

O uso do \TeX é a opção mais simples mas é bem ineficiente e a ordenação é feita pelos códigos dos caracteres minúsculos, o que funciona para alfabetos latinos,

como o Português, mas falha para outros. Outro problema é que este método usa um comparador ASCII e falha quando há comandos no campo *key*, usado na ordenação. Para usar esse método basta adicionar `\makenoidxglossaries` ao preâmbulo e `\printnoidxglossaries` no local onde você quer colocar seu glossário. Lembre-se que o glossário só aparecerá corretamente após você processar seu documento duas ou até três vezes. Isso acontece porque listas como o sumário, listas de algoritmos, figuras e tabelas só são geradas após a primeira execução do processador \TeX e incluídas na segunda execução. Como a inclusão dessas listas pode afetar as páginas das referências dos glossários, pode ser necessário executar o processador \TeX uma terceira vez. Seu editor integrado \LaTeX deve controlar isso, e caso esteja usando a linha de comando, você deve fazê-lo.

Se você tem um glossário grande, ou se seus termos usam caracteres não-Latinos, caracteres Latinos estendidos ou comandos no campo *key*, você deve usar uma das três outras opções, mas preferencialmente as duas últimas caso utilize caracteres não-Latinos e/ou caracteres Latinos estendidos.

A segunda opção de ordenação envolve a utilização da aplicação `makeindex`. O comando `makeindex` faz parte de todas as distribuições mais recentes do \TeX mas foi programada para funcionar somente com o alfabeto Latino não estendido. Para usar esse método basta adicionar `\makeglossaries` ao preâmbulo e `\printglossaries` no local onde você quer colocar seu glossário. Esta opção não permite uma mistura de métodos de ordenação, i.e., todos os glossários devem usar o mesmo método dentre ordem de palavra/letra, de uso ou de definição. Se você precisar ordens diferentes para glossários diferentes, você deverá executar `makeindex` explicitamente para cada um deles. Lembre-se que o comando `makeindex` deve ser executado após a primeira execução do processador \TeX . Você pode executar `makeindex` indiretamente através dos scripts `makeglossaries` (Perl) e `makeglossaries-lite` (Lua), evitando ter de identificar as

extensões dos arquivos a serem processados. Os comandos abaixo mostram a diferença entre as utilizações dos três programas.

```
makeindex -s exemplo.ist -o exemplo.gls exemplo.glo
makeglossaries exemplo
makeglossaries-lite exemplo
```

O software **flexible indexing system** (**xindy**) é um programa altamente configurável usado para a ordenação e formatação de índices, tendo sido escrito para ser o sucessor de **makeindex**, e para funcionar com diferentes programas, mas especialmente L^AT_EX e troff. Além de processar sem erros linguagens com caracteres especiais, como Português, Francês, Alemão e Islandês, o **xindy** permite que se definam novos tipos de localização (algo útil quando escrevendo manuais) e indexação hierárquica. Como **xindy** é um *script* Perl, você deve ter Perl instalada em seu sistema. Caso você precise de um esquema de indexação mais complexo, sugiro que leiam a documentação disponível em <http://www.xindy.org/>.

Já o pacote **bib2gls** provê uma aplicação Java de linha de comando que pode ser usada para extrair informações de glossários armazenadas em um arquivo **.bib** e convertê-las em comandos de definição de entradas de glossário. Esse pacote deve ser usado com o pacote **glossaries-extra**, que deve ser carregado usando a opção **record**. Assim como o **xindy**, o **bib2gls** permite que se definam divisões lógicas de glossários.

O pacote **bib2gls** provê ainda o programa **convertgls2bib**, que converte arquivos **.tex** contendo definições de glossários para o formato **.bib** usado pelo **bib2gls**. Uma vez convertidos para o formato **.bib**, suas definições de glossários podem ser mantidas usando uma interface gráfica que manipule **.bib** files, como o JabRef (<https://www.jabref.org/>) e o KBibTeX (<https://userbase.kde.org/KBibTeX>). O manual do **bib2gls** pode ser acessado em <http://mirrors.ctan.org/support/bib2gls/bib2gls-begin.pdf>.

Uma outra opção disponível com o pacote `glossaries-extra` é a de não ordenar as entradas. Para tal, você deve carregar o pacote `glossaries-extra` com a opção `sort=none` e as definições na ordem desejada. Esta opção não exige um comando de ativação, como o `\makeglossaries`.

A performance dos métodos é algo a ser considerado se você tiver muitas entradas em seu glossário. Caso tenha poucas entradas, o método que usa o `TeX` pode ser usado tranquilamente. Porém, no caso de muitas entradas ele pode adicionar um tempo considerável a execução do `LATEX` ou `pdfLATEX`. Alguns testes são descritos em <https://www.dickimaw-books.com/gallery/glossaries-performance.shtml#all>. Para maiores detalhes sobre a incorporação destes programas no processamento de seu documento, consulte <https://www.dickimaw-books.com/latex/buildglossaries/>. Por isso, preste atenção no aviso abaixo!

▲ Cuidado!

Algumas vezes, você precisa deletar o arquivo `.glsdefs` antes de executar a sequência `pdflatex`, `makeindex`, `pdflatex` e `pdflatex`. Isso acontece em alguns casos quando há um erro no processamento do glossário e execuções posteriores do `pdflatex` não alteram o arquivo com os erros gerados pelo `makeglossaries`.

Como mencionado acima, o pacote `glossaries` também pode ser usado para gerar índices remissivos, embora seja mais comum usar um pacote de geração de índices como o `imakeidx`. Se você decidir usar a opção `index` do pacote `glossaries`, ela habilitará o comando `\newterm` que funciona como o comando `\newglossaryentry`, exceto pelo fato que atribui uma descrição vazia ao termo definido e cria um novo “glossário” para o índice.

Caso deseje usar o pacote `imakeidx`, você deve incluir no preâmbulo o comando abaixo, caso queira criar um índice com duas colunas e título “Índice”:

```
\makeindex[title=Índice, columns=3, intoc=true]
```

e simplesmente adicionar as palavras chave como uma referência após o seu uso no texto, como no exemplo abaixo, usando o comando:

`\index{termo}`

Então, ao final do seu documento, a inclusão do comando abaixo imprimirá o índice remissivo de seu documento, que, neste caso está precedido pelo comando usado para definir o texto a ser impresso antes do índice.

```
\indexprologue{texto a ser incluído antes das entradas do índice}

\printindex
```

Eu utilizei o `imakeidx` nesse documento e no código fonte dele, nos arquivos `.tex`, você pode ver as referências que são incluídas no índice. Algumas delas são repetidas na Lista de Acrônimos, o que pode parecer um exagero. Entretanto, a inclusão dessas duas listas serve como exemplo da utilização desses pacotes. Você deve decidir, juntamente com seu(s) orientador(es) o que deseja usar no seu documento.

É importante ressaltar que esse pacote é parte da distribuição mínima do \LaTeX e que como padrão, o índice não aparece no Sumário. Caso deseje que o mesmo apareça no Sumário, adicione a opção `intoc` no comando `\makeindex`, como mostrado acima. Para mais detalhes sobre o uso desse pacote, sugiro o guia introdutório do [Overleaf](https://www.overleaf.com/learn/latex/indices#Indices_on_Overleaf), que pode ser acessado em https://www.overleaf.com/learn/latex/indices#Indices_on_Overleaf, e o manual do pacote `imakeidx`, que pode ser acessado em <http://mirrors.ctan.org/macros/latex/contrib/imakeidx/imakeidx.pdf> (Gregorio 2016).

9 Considerações Finais

Antes de iniciar a construção do modelo \LaTeX do [PPgSC](#) e a escrita deste documento achava que sabia bastante sobre \LaTeX . Ledo engano. A quantidade de pacotes e opções para a diagramação de textos, ilustrações, referências e outros elementos é imensa e as possibilidades de configurações dos mesmos é absurda. Até mesmo em tarefas simples como a parametrização de comandos usando programação em \TeX e \LaTeX , como com `if-then-else`, possuem particularidades com as quais eu não estava familiarizado, como seu comportamento com comando expansíveis e não expansíveis. Eu apanhei muito durante esse período mas aprendi bastante.

A depuração de erros em \LaTeX é complicada. Às vezes um erro em um local acaba gerando uma mensagem de erro pelo processador em outro lugar distante do inicial. O uso de ferramentas de edição algumas vezes complica a depuração, visto que elas eventualmente escondem ou postergam problemas. Caso encontre um erro que não consegue eliminar ou entender porque está acontecendo, sugiro que o processe usando a linha de comando e examine qual a mensagem de erro sem o uso da ferramenta. Em alguns casos, nem isso resolve. Eu passei várias horas, distribuídas ao longo de vários dias, tentando identificar o que causava a diminuição de espaços entre os números e os títulos de capítulos, que inclusive afetava o espaçamento no Sumário! Então, decidi desabilitar vários pacotes de cada vez e examinar o resultado, até isolar o pacote que estava causando o problema.

Finalmente, faça uma busca com termos relevantes ao erro que está tentando

eliminar, pois é bastante provável que alguém já teve um problema similar e o ajudaram a resolvê-lo. Por exemplo, busque ajuda na área T_EX do StackExchange <https://tex.stackexchange.com>, que é um fórum bastante ativo que conta com muitos usuários com larga experiência em T_EX e L^AT_EX.

Outro aspecto importante é o tempo de processamento. A medida que você inclui pacotes e funcionalidades, você o aumenta, é claro. Como na grande maioria dos casos, vários dos pacotes mencionados aqui não serão utilizados, eu sugiro que você desabilite o carregamento de vários pacotes e os inclua a medida que identifique que necessita deles. Outra opção é desabilitar a inclusão de capítulos nos quais você não esteja trabalhando no momento e habilitá-las posteriormente.

Finalmente, peço que sugestões de inclusões de novas funcionalidades, exemplos e de correções a este documento sejam encaminhadas para bruno@dimap.ufrn.br. Elas serão levadas em consideração na elaboração de novas versões do modelo e manual. Espero que o modelo e este documento facilitem a elaboração de sua dissertação/tese.

Bibliografia

- Barbosa, Fellipe M. C. (2020). “Medição Precisa de Áreas de Feridas Crônicas Usando *Structure From Motion*”. Diss. de mestr. Universidade Federal do Rio Grande do Norte (ver pp. [19](#), [20](#)).
- Bezoz, Javier (2019). *Customizing lists with the enumitem package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [15](#)).
- Buss, Sam (2011). *LaTeX for Logicians. bussproofs.sty: A User Guide*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [34](#)).
- Carlisle, D. P. (1995). *The indentfirst package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [13](#)).
- (2020). *Packages in the ‘graphics’ bundle*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [18](#)).
- Carlisle, D. P. e S. P. Q. Rahtz (2020). *The graphics package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [18](#)).
- Carlisle, David e Sebastian Rahtz (2021). *Back referencing from bibliographical citations*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [66](#)).
- Chiang, David (2012). *tikz-qtree: better trees with TikZ*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [53](#)).
- Cochran, Steven Douglas (2005). *The Subfig Package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [20](#)).

- Crémer, Jacques (2011). *A very minimal introduction to TikZ*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [44](#)).
- Daly, Patrick W. (2010). *Natural Sciences Citations and References*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [66](#)).
- Danaux, Xavier e Dave Gandy (2016). *The fontawesome package: High quality web icons*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [6](#)).
- Davies, Jim (2020). *Setting real-time CSP*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [39](#)).
- Etchemendy, John, Dave Barker-Plummer e Richard Zach (2013). *The lplfitch package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver pp. [35](#), [36](#)).
- Fear, Simon (2020). *Publication quality tables in L^AT_EX*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [23](#)).
- Feuersänger, Christian (2020). *Manual for Package pgfplots*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [49](#)).
- Fiandrino, Claudio (2014). *Sa-TikZ*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [45](#)).
- Fiorio, Christophe (2017). *algorithm2e.sty — package for algorithms*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [25](#)).
- Francis Jeffrey Pelletier, Mohammad M. Ajallooeian and (2020a). *A L^AT_EX Package for Natural Deduction Proofs in the Jaśkowski and Kalish–Montague Styles*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver pp. [36](#), [37](#)).
- (2020b). *Kalish/Montague and Jaśkowski Natural Deduction*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [37](#)).
- Goossens, Michel, Frank Mittelbach, Sebastian Rahtz, Denis Roegel e Herbert Voss (2007). *LaTeX Graphics Companion, The (2nd Edition) (Tools and Techniques for Computer Typesetting)*. Addison-Wesley Professional. ISBN: 0321508920 (ver p. [3](#)).

- Grahn, Alexander (2020). *The animate Package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [53](#)).
- Gregorio, Enrico (2016). *The package imakeidx*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [74](#)).
- Hackl, Jürgen (2019). *TikZ-network Manual*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [47](#)).
- Hammerlindl, Andy, John Bowman e Tom Prince (2020a). *Asymptote: the Vector Graphics Language*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [58](#)).
- (2020b). *Asymptote: the Vector Graphics Language*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [58](#)).
- Hein, Jeff (2012). *The tikz-3dplot Package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [51](#)).
- Jeffrey, Alan e Frank Mittelbach (2020). *inputenc.sty*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [5](#)).
- Johannes L. Braams, Javier Bezos (2020). *Babel: The Portuguese language*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [9](#)).
- (2021). *Babel: Localization and internationalization*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [9](#)).
- Kern, Uwe (2016). *Extending L A TEX's color facilities: the xcolor package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [14](#)).
- Kime, Philip, Moritz Wemheuer e Philipp Lehman (2020). *The biblalex Package: Programmable Bibliographies and Citations*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [65](#)).
- King, Paul (1990). *Printing Z and Object-Z L^AT_EX documents*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [39](#)).

- Kleinod, Ekkart (2020). *The changes-package: Manual change markup — version 3.2.2*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. 62).
- Knuth, Donald E. (1984). *The TeXbook*. USA: Addison-Wesley Professional. ISBN: 978-0201134483 (ver p. 1).
- Kohm, Markus (2020). *KOMA - Script: a versatile L^AT_EX 2_ε bundle*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver pp. 2, 11).
- Kottwitz, Stefan (2015). *L^AT_EX Cookbook*. Packt Publishing. ISBN: 978-1784395148 (ver pp. 3, 28).
- Lamport, Leslie (1994). *L^AT_EX (2nd Ed.): A Document Preparation System: User's Guide and Reference Manual*. USA: Addison-Wesley Longman Publishing Co., Inc. ISBN: 0201529831 (ver pp. 1, 3).
- Lingnau, Anselm (2001). *An Improved Environment for Floats*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. 17).
- Marques, Vitor Godeiro (2018). “Ulcer Segmentation and Tissue Classification Using Color Texture Clustering”. Monografia (Bacharelado em Ciência da Computação). Diss. de mestr. Universidade Federal do Rio Grande do Norte (ver p. 21).
- Matthes, Alain (2007). *The TikZ and PGF Packages: Manual for Version 3.1.8b*. TeXample.net. Alemanha (ver p. 53).
- Maxime C. Hupin, Patrick F. Radin and (2016). *bclogo: Extension L^AT_EX pour la réalisation de boîtes colorées avec logos*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. 45).
- Mittelbach, Frank, Michel Goossens, Johannes Braams, David Carlisle e Chris Rowley (1999). *The L^AT_EX Companion (Tools and Techniques for Computer Typesetting)*. USA: Addison-Wesley Longman Publishing Co., Inc. ISBN: 0201362996 (ver p. 3).
- Morten Høgholm, Lars Madsen (2021). *The mathtools package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver pp. 29, 32).

- Oberdiek, Heiko (2019). *The pdfscape package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [16](#)).
- Oostrum, Pieter van, Øystein Bache e Jerry Leichter (2021). *The multirow, bigstrut and bigdelim packages*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [21](#)).
- Philip Kime, François Charette (2020). *biber: A backend bibliography processor for biblatex*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [65](#)).
- Pighin, Daniele (2012). *The TikZ-dependency package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [46](#)).
- Poore, Geoffrey M. (2017). *The minted package: Highlighted source code in L^AT_EX*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [26](#)).
- Rahtz, Sebastian e Heiko Oberdiek (2021). *Hypertext marks in L^AT_EX: a manual for hyperref*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver pp. [63](#), [64](#)).
- Schlicht, R. (2021). *The microtype package: Subliminal refinements towards typographical perfection*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [14](#)).
- Siskind, Jeffrey Mark e Alexis Dimitriadis (2008). *Qtree, a L^AT_EX tree-drawing package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [40](#)).
- Society, American Mathematical (2020). *User's Guide for the amsmath Package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver pp. [29](#), [31](#)).
- Spivey, Mike (1990). *A guide to the zed style option*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [39](#)).
- Staats III, Charles (2021). *An Asymptote tutorial*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [57](#)).
- Talbot, Nicola L C (2012). *L^AT_EX for Complete Novices*. Dickimaw Books (ver p. [3](#)).

- Talbot, Nicola L.C. (2020a). *glossaries-extra.sty v1.45: an extension to the glossaries package*. Dickimaw Books. Inglaterra (ver p. [70](#)).
- (2020b). *The glossaries package v4.46: a guide for beginners*. Dickimaw Books. Inglaterra (ver p. [68](#)).
- (2020c). *User Manual for glossaries.sty v4.46*. Dickimaw Books. Inglaterra (ver p. [68](#)).
- Tantau, Till (2020). *The TikZ and PGF Packages: Manual for Version 3.1.8b*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [44](#)).
- Voss, Herbert (2011). *Typesetting Tables with L^AT_EX*. UIT Cambridge Ltd. ISBN: 978-1906860257 (ver p. [3](#)).
- Wilson, Peter, Herries Press e Will Robertson (2020). *The appendix Package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [83](#)).
- Wright, Joseph (2020). *siunitx — A comprehensive (SI) units package*. CTAN - The Comprehensive TeX Archive Network. Alemanha (ver p. [31](#)).

Apêndice 1

Um apêndice deve conter material complementar a sua dissertação/tese, que serve para complementar a argumentação do autor sobre seu trabalho. O conteúdo de um apêndice deve ter sido elaborado pelo autor. Caso o autor deseje incluir conteúdo complementar que auxilie sua argumentação, mas que tenha sido elaborado por outras pessoas, o mesmo deve adicionar esse conteúdo em um anexo.

Os apêndices devem ser localizados após as referências, e os anexos após os apêndices, caso existam. Ambos devem aparecer antes do índice remissivo. É desejável que ambos apareçam no sumário. Para incluir apêndices, você deve ter o pacote **appendix** (Wilson, Press e Robertson 2020) instalado, cujo manual está disponível em <https://ctan.dcc.uchile.cl/macros/latex/contrib/appendix/appendix.pdf>.

Caso não necessite de um apêndice, comente os comandos abaixo no arquivo `DissertacaoPPgSC.tex`.

```
1 \begin{appendices}
2   % Apêndice A (arquivo capitulos/apendice1.tex)
3   \input{capitulos/apendice1.tex}
4 \end{appendices}
```

Código .1: Exemplo de código L^AT_EX usado para carregar um arquivo de apêndice.

Índice

- .aux, [11](#)
- .bib, [65](#), [72](#)
- .bst, [64–66](#)
- .csv, [23](#)
- .log, [11](#)
- .ods, [23](#)
- .tex, [7](#), [72](#), [74](#)
- .xls, [23](#)
- L^AT_EX Tables, [23](#)
- T_EX Equation Editor, [32](#)
- KOMA-Script, [35](#)
- B_BT_EX, [64–66](#)
- KOMA-Script, [2](#), [3](#), [10](#),
[11](#), [35](#), [67](#)
- LuaL^AT_EX, [11](#)
- METAPOST, [55](#)
- X_YL^AT_EX, [11](#)
- biber, [64](#), [65](#)
- pdfL^AT_EX, [11](#), [14](#), [17](#), [26](#),
[41](#), [73](#)
- B_BL^AT_EX, [65](#)
- B_BL^AT_EX, [64–66](#)
- ABNT_EX, [14](#), [15](#)
- abntex2, [67](#)
- added, [60](#)
- adjustbox, [17](#)
- Alemão, [72](#)
- algochapter, [23](#)
- algorithm2e, [23](#), [24](#)
- align, [29](#)
- aligned, [29](#)
- amsfonts, [30](#)
- amsmath, [29](#), [31](#), [32](#)
- amssymb, [30](#)
- animation, [57](#)
- article, [35](#)
- asy, [55](#)
- Asymptote, [55](#), [56](#), [58](#)
- AxMath, [32](#)
- babel, [7](#), [9](#), [67](#)
- babel-portuges, [9](#)
- background, [25](#)
- backref, [64–66](#)
- backrefstyle, [65](#)
- baselineskip, [13](#)
- bclogo, [45](#)
- bfit, [60](#)
- bib2gls, [67](#), [70](#), [72](#)
- biblatex, [4](#)
- bibstyle, [65](#)
- book, [35](#), [67](#)
- booktabs, [23](#)
- boxed float, [16](#)
- brazil, [7](#), [9](#), [67](#)
- brazilian, [9](#), [67](#)
- bussproofs, [34](#)

C, 26
changes, 59, 61
chapterbib, 64
chngetr, 15
Circus, 37
citeref, 64
citestyle, 65
cmap, 6
CO-orientador, 9
CodeCogs, 32
color, 18
colorlinks, 63
colortbl, 23
comment, 61
convertgls2bib, 72
counterwithin, 15
CSP, 37–39
CTAN, 5, 61

default, 60
deleted, 60
description, 15
detexify, 33
doublespacing, 13
draft, 59

enumerate, 15
enumitem, 15

environment, 11, 29
EPS, 55
EqualX, 32
equation, 29
esvector, 29
eufrak, 30
euscript, 30

figure, 35
final, 59
flalign, 29
float, 16, 17, 21, 23, 25, 26, 35, 42
fontawesome, 6
fontenc, 5
forest, 54
Francês, 72
frontmatter, 26

gather, 29
gendef, 38
GIF, 57
glossaries, 11, 67, 70, 73
glossaries-extra, 67, 70, 72
glossaries-portuges, 67
glssymbol, 68
graphics, 17, 18

graphicx, 17, 18

highlight, 61
HostMath, 32
hyperindex, 63
hyperlinks, 63
hyperref, 11, 63, 65
hyphen-portuguese, 9

imakeidx, 73
includegraphics, 17, 19
indentfirst, 13
index, 73
Inkscape, 46, 55
input, 69
inputenc, 4
intoc, 74
Islandês, 72
itemize, 15

JabRef, 72
Java, 72
JavaScript, 40, 53, 57
JPG, 17

KBibTeX, 72
kernel, 11, 25

landscape, 16
latex-graphics, 18

Latex4technics, [32](#)
 lined, [24](#)
 linesnumbered, [23](#)
 linespread, [13](#)
 listing, [26](#), [27](#)
 listings, [25](#), [26](#)
 listoffigures, [11](#)
 listoftables, [11](#)
 lmodern, [6](#)
 loadglsentries, [69](#)
 longnewglossaryentry, [69](#)
 lplfitch, [34](#), [35](#)
 lscape, [16](#)
 LyX, [33](#)

 makebst, [66](#)
 makeglossaries, [67](#), [71](#)
 makeglossaries-lite, [67](#),
 [71](#)
 makeidx, [73](#)
 makeindex, [11](#), [70](#), [71](#)
 makenoidxglossaries, [71](#)
 markup, [59](#)
 mathclap, [32](#)
 mathtools, [29](#), [31](#), [32](#)
 maxbibnames, [65](#)
 maxcitenames, [65](#)

 mdframed, [45](#)
 memoir, [67](#)
 microtype, [14](#)
 mindmaps, [49](#)
 minted, [11](#), [16](#), [25](#), [26](#)
 MPEG, [57](#)
 multicolumns, [21](#)
 multilined, [29](#)
 multilinhas, [21](#)
 multirow, [21](#)
 multlined, [29](#)

 natbib, [64](#), [66](#)
 natded, [36](#), [37](#)
 newacronym, [69](#)
 newglossaryentry, [69](#), [73](#)
 newterm, [73](#)
 nocolor, [60](#)
 nosep, [15](#)

 objectz, [39](#)
 onehalfspacing, [13](#)
 Overleaf, [13](#), [15](#), [26](#), [28](#),
 [49](#)

 partaux, [11](#)
 PDF, [16](#), [17](#), [47](#), [53](#),
 [55–57](#), [59](#)
 pdf, [40](#), [63](#)

 pdflang, [63](#)
 pdfscape, [16](#)
 pdfusetitle, [63](#)
 Perl, [72](#)
 PGF, [40](#)
 PGFPlots, [3](#), [48](#), [49](#)
 plainpages, [63](#)
 plaintop float, [16](#)
 PNG, [17](#)
 portuges, [9](#), [67](#)
 portugues, [67](#)
 portuguese, [9](#)
 Português, [67](#), [72](#)
 PPgSC-Ingles, [9](#)
 PPgSC-Tese, [9](#)
 printglossaries, [71](#)
 printnoidxglossaries, [71](#)
 PSTricks, [3](#), [41](#), [42](#)
 Pygment, [26](#)
 Python, [26](#)

 qtree, [40](#)

 record, [72](#)
 referências bibliográficas,
 [64](#)
 referências cruzadas, [64](#)
 replaced, [61](#)

ruled, 24	subfig, 19 , 20	tikz-network, 47 , 48
ruled float, 16	subfiguras, 19 , 20	tikz-qtree, 52 , 54
	subtabelas, 19	TikzEdt, 55
sa-tikz, 44	SVG, 53 , 55	TikZit, 55
scrbook, 14 , 67	switching, 44	tkz-2d, 53
scrwfile, 11		
setacronymstyle, 69	table, 21	underlined, 60
setspace, 13	tableofcontents, 11	UTF, 5
SGV2TikZ, 55	Tables Generator, 23	xasy, 57
signSkip, 10	tabular, 19	xcolor, 14 , 23
signThickness, 10	TikZ, 3 , 7 , 11 , 23 , 40–44 ,	Xfig, 46
signWidth, 10	49 , 52 , 55	xindy, 70
singlespacing, 13	TikZ-Dependency, 45	Z, 37–39
siunitx, 30	TikZ-Network, 46	zed, 38 , 39
StackExchange, 51 , 76	tikz-3dplot, 49–51	zed-csp, 38 , 39