

# L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> Via Exemplos

Sadao Massago<sup>1</sup>

Versão 0.5, abril de 2018

<sup>1</sup>DFQM-UFSCar – Campus de Sorocaba, SP (<http://dfqm.sorocaba.ufscar.br/>)

Copyright ©2018 Sadao Massago. Todos os direitos reservados. Este documento é software livre; podendo ser redistribuído e/ou modificado de acordo com os termos da Licença Pública da GNU como publicado pela Free Software Foundation; versão 2 da Licença, ou (se for sua opção) qualquer versão posterior. Veja <https://opensource.org/licenses/gpl-license>.

# Prefácio

O  $\text{\LaTeX}2_{\epsilon}$  é um sistema de processamento de documentos implementados sobre o  $\text{\TeX}$ .

O objetivo deste documento é apresentar o  $\text{\LaTeX}2_{\epsilon}$  de forma gradativa, usando os exemplos. Assim, o documento contém muitos exemplos e suas saídas, o que aumentou consideravelmente as suas páginas.

Este documento está dividido em 17 capítulos mais os apêndices.

Do Capítulo 1 até Capítulo 13 utilizam somente os pacotes da `base` e de `required` na qual qualquer sistema  $\text{\LaTeX}$  devem conter. Do capítulo 14 a 17 utilizam os pacotes que não sejam da `base` e de `required` para incrementar a funcionalidade. Nos apêndices, serão tratados alguns recursos extras que podem ser interessantes, mas que julgar que não sejam de interesse de todos.

Capítulo 1 é uma breve introdução sobre o sistema.

Capítulo 2 trata do primeiro documento em  $\text{\LaTeX}2_{\epsilon}$ , incluindo introdução à escrita de textos.

Capítulo 3 introduz sobre fórmulas matemáticas.

Capítulo 4 trata sobre estrutura de textos.

Capítulo 5 é uma continuação do Capítulo 3, tratando sobre fórmulas matemáticas.

Capítulo 6 trata sobre definições de comandos e ambientes.

Capítulo 7 discute sobre divisão lógica de documentos, conhecido como “seccionamentos”.

Capítulo 8 trata de definições e uso dos ambientes do tipo teoremas.

Capítulo 9 discute sobre a criação de figuras e tabelas flutuantes, tabelas longas e inclusão de imagem externa.

Capítulo 10 trata sobre ajuste das fontes e espaçamentos.

Capítulo 11 trata sobre referências bibliográficas e índice remissivos.  $\text{\BibTeX}$  também será introduzido neste capítulo.

Capítulo 12 trata sobre medidas e contadores.

Capítulo 13 explora várias possibilidades dentro dos pacotes de `base` e `required` que não foram discutidos nos capítulos anteriores.

Capítulo 14 discute o uso de pacotes fora do `base` e `required` para ajuste de documentos. A partir deste capítulo, serão apresentados pacotes fora do `base` e `required`. Isto significa que os pacotes indicados podem precisar de instalação a parte.

Capítulo 15 discute o pacote `xcolor` especial para cores, e `tikz` especial para criar ilustrações.

Capítulo 16 ilustra a criação de `poster` e `slides` de apresentação.

Capítulo 17 apresenta o uso de  $X_{\text{q}}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  e  $\text{L}^{\text{u}}\text{a}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ , considerado a próxima geração de  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .

Apêndice A é sobre como escrever um pacote (arquivo de estilos).

Apêndice B trata de editor  $\text{L}^{\text{Y}}\text{X}$  que permite elaborar documentos de forma mais visual como os editores para escritórios, em vez de editar diretamente o código fonte em  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .

Apêndice C trata da mala direta e criação do caderno de resumos, interessantes para organizadores de eventos.

Apêndice D trata de pacotes úteis para elaborar provas e lista de exercícios.

Apêndice E trata de pacotes interessantes para desenvolver projetos tal como escrever um livro ou similar.

Sorocaba, abril de 2018.

Sadao Massago <sadao@ufscar.br>

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Uma breve história . . . . .	1
1.2	Como usar $\text{\LaTeX}$ . . . . .	1
<b>2</b>	<b>Iniciando um Documento</b>	<b>3</b>
2.1	Primeiro documento . . . . .	3
2.2	Mensagem de erro e correção . . . . .	5
2.3	Caracteres especiais . . . . .	5
<b>3</b>	<b>Introdução às Fórmulas Matemáticas</b>	<b>7</b>
3.1	Fórmula <code>textstyle</code> e <code>displaystyle</code> . . . . .	7
3.2	Modo <code>displaystyle</code> no meio do texto . . . . .	9
3.3	Equação enumerada e referências cruzadas . . . . .	9
<b>4</b>	<b>Estrutura do Texto</b>	<b>11</b>
4.1	Alinhamentos . . . . .	11
4.2	Notas de rodapé e ênfase de texto . . . . .	12
4.3	Listas . . . . .	13
4.4	Tabelas . . . . .	15
4.5	Ambiente de tabulação . . . . .	20
4.6	Textos de citações, versos e <code>verbatim</code> . . . . .	21
4.7	Caixa <code>minipage</code> . . . . .	24
4.8	Colunas múltiplas de texto . . . . .	26
<b>5</b>	<b>Aprofundando nas Fórmulas Matemáticas</b>	<b>29</b>
5.1	Usando algumas fontes matemáticas . . . . .	29
5.2	Texto, função por partes e matrizes . . . . .	31
5.3	Delimitadores, chaves e integrais . . . . .	32
5.4	Quebrando fórmulas em várias linhas . . . . .	34
5.5	Nome sobre setas e delimitador empilhados . . . . .	37
5.6	Subequações . . . . .	38
5.7	Acentuação no modo matemático . . . . .	38

<b>6</b>	<b>Definindo Comandos e Ambientes</b>	<b>41</b>
6.1	Definindo comandos . . . . .	41
6.2	Criando ambientes . . . . .	43
6.3	Quebrando o código em várias linhas . . . . .	44
<b>7</b>	<b>Divisão Lógica de Documentos</b>	<b>45</b>
7.1	Capítulos, seções e similares . . . . .	45
7.2	Capa, conteúdo frontal e principal . . . . .	47
7.3	Limpendo o verso das páginas . . . . .	50
7.4	Efetutando pequenos ajustes . . . . .	52
<b>8</b>	<b>Teoremas e Similares</b>	<b>53</b>
8.1	Criando ambiente para teoremas . . . . .	53
8.2	Parâmetros opcionais . . . . .	55
<b>9</b>	<b>Figuras, Tabelas e Imagens Externas</b>	<b>57</b>
9.1	Figuras flutuantes . . . . .	57
9.2	Tabelas flutuantes . . . . .	59
9.3	Tabelas longas . . . . .	60
9.4	Imagem externa . . . . .	61
9.5	Desenhando sobre a imagem externa . . . . .	64
9.6	Caixas gráficas . . . . .	66
<b>10</b>	<b>Ajuste das Fontes e de Espaçamento</b>	<b>69</b>
10.1	Seleção da família de fontes . . . . .	69
10.2	Seleção de formas e peso das fontes . . . . .	70
10.3	Tamanho das fontes . . . . .	71
10.4	Ajuste de fontes no modo matemático . . . . .	72
<b>11</b>	<b>Referências Bibliográficas e Índice Remissivo</b>	<b>75</b>
11.1	Referências bibliográficas . . . . .	75
11.2	Usando o BIBTEX . . . . .	77
11.3	Índice remissivo . . . . .	80
<b>12</b>	<b>Medidas e Contadores</b>	<b>83</b>
12.1	Unidade de medidas e espaçamentos . . . . .	83
12.2	Medidas . . . . .	85
12.3	Contadores . . . . .	86
<b>13</b>	<b>Mais Alguns Cuidados e Ajustes</b>	<b>89</b>
13.1	Comandos frágeis . . . . .	89
13.2	Babel e nomes . . . . .	90
13.3	Espaçamento entre linhas . . . . .	94

13.4	Sobre hifenização . . . . .	95
13.5	Comandos com “@” e “*” . . . . .	95
13.6	Trocando fontes . . . . .	101
13.7	Trocando marcador da lista itemizada . . . . .	101
13.8	Cores no $\text{\LaTeX}$ . . . . .	101
<b>14</b>	<b>Usando Pacotes Fora do base e required</b>	<b>103</b>
14.1	Configuração das páginas . . . . .	103
14.2	Estilo europeu . . . . .	105
14.3	Cabeçalho e títulos . . . . .	106
14.4	Links . . . . .	110
14.5	Trocando fontes . . . . .	112
14.6	Figuras e similares . . . . .	113
14.7	Escrevendo medidas internacionais . . . . .	116
14.8	Enfeites e código fontes . . . . .	119
14.9	Mais sobre referências bibliográficas . . . . .	125
<b>15</b>	<b>Gráfico e Diagramas</b>	<b>129</b>
15.1	Misturando cores . . . . .	129
15.2	Criando ilustrações gráficas . . . . .	130
<b>16</b>	<b>Produzindo Poster e Slides</b>	<b>141</b>
16.1	Poster . . . . .	141
16.2	Slides . . . . .	145
<b>17</b>	<b>Usando <math>\text{\XeLaTeX}</math> e <math>\text{\LuaLaTeX}</math></b>	<b>151</b>
17.1	$\text{\LuaLaTeX}$ e $\text{\XeLaTeX}$ . . . . .	151
17.2	Fontes no $\text{\XeLaTeX}$ / $\text{\LuaLaTeX}$ . . . . .	154
17.3	Usando em conjunto com <code>biblatex</code> . . . . .	156
<b>A</b>	<b>Desenvolvendo Pacotes</b>	<b>161</b>
A.1	Criando pacotes . . . . .	161
A.2	Preenchendo o documento para teste . . . . .	166
<b>B</b>	<b>Usando o Editor <math>\text{\LyX}</math></b>	<b>169</b>
B.1	Visualização do documento final . . . . .	169
B.2	ABC do $\text{\LyX}$ . . . . .	170
B.3	Formatando textos . . . . .	172
B.4	Matemática . . . . .	172
<b>C</b>	<b>Para Organizadores do Evento</b>	<b>175</b>
C.1	Mala direta no $\text{\LaTeX}$ . . . . .	175
C.2	Caderno de resumos . . . . .	179

---

<b>D Para Professores</b>	<b>183</b>
D.1 Lista de exercícios e provas . . . . .	183
D.2 Com e sem respostas . . . . .	184
<b>E Para Projetos</b>	<b>187</b>
E.1 Pacote standalone . . . . .	187
E.2 Dividindo o documento em vários arquivos . . . . .	189
<b>Referências Bibliográficas</b>	<b>191</b>
<b>Índice Remissivo</b>	<b>193</b>

# Lista de Exemplos

2.1	ex02-inicial.tex . . . . .	3
2.2	ex02-caracteres.tex . . . . .	5
3.1	ex03-formulas.tex . . . . .	7
3.2	ex03-simbolos.tex . . . . .	8
3.3	ex03-displaystyle.tex . . . . .	9
3.4	ex03-ref.tex . . . . .	10
4.1	ex04-alinhamento.tex . . . . .	11
4.2	ex04-nova-linha.tex . . . . .	12
4.3	ex04-rodape.tex . . . . .	12
4.4	ex04-enfase.tex . . . . .	13
4.5	ex04-listas.tex . . . . .	13
4.6	ex04-sublistas.tex . . . . .	14
4.7	ex04-enumerate.tex . . . . .	15
4.8	ex04-tabular.tex . . . . .	16
4.9	ex04-multicolumn.tex . . . . .	17
4.10	ex04-cline.tex . . . . .	17
4.11	ex04-dcolumn.tex . . . . .	18
4.12	ex04-tabularx.tex . . . . .	19
4.13	ex04-tabbing.tex . . . . .	20
4.14	ex04-quote.tex . . . . .	21
4.15	ex04-quotation.tex . . . . .	21
4.16	ex04-verse . . . . .	22
4.17	ex04-verbatim.tex . . . . .	23
4.18	ex04-verb.tex . . . . .	23
4.19	ex04-minipage.tex . . . . .	24
4.20	ex04-minipage-fbox.tex . . . . .	25
4.21	ex04-parbox.tex . . . . .	25
4.22	ex04-multicols.tex . . . . .	26
5.1	ex05-mat-fontes-basico.tex . . . . .	29
5.2	ex05-mat-text.tex . . . . .	31
5.3	ex05-matriz.tex . . . . .	31
5.4	ex05-delimitador.tex . . . . .	32

5.5	ex05-delimitador-grande.tex . . . . .	33
5.6	ex05-bracos.tex . . . . .	33
5.7	ex05-integrais.tex . . . . .	33
5.8	ex05-split.tex . . . . .	34
5.9	ex05-cases-aligned.tex . . . . .	35
5.10	ex05-gather.tex . . . . .	35
5.11	ex05-multiline.tex . . . . .	36
5.12	ex05-align-star.tex . . . . .	37
5.13	ex05-stackrel.tex . . . . .	37
5.14	ex05-binom.tex, parte c . . . . .	38
5.15	ex05-subequacao.tex . . . . .	38
5.16	ex05-acentos.tex . . . . .	40
6.1	ex06-newcommand.tex . . . . .	42
6.2	ex06-parametro-opcional.tex . . . . .	42
6.3	ex06-ambiente.tex . . . . .	43
6.4	ex06-comando-multlinhas.tex . . . . .	44
7.1	ex07-capitulo.tex . . . . .	45
7.2	ex07-capitulo-star.tex . . . . .	46
7.3	ex07-matter.tex . . . . .	47
7.4	ex07-maketitle.tex . . . . .	48
7.5	ex07-article.tex . . . . .	49
7.6	ex07-clearpage.tex . . . . .	51
8.1	ex08-teorema.tex . . . . .	53
8.2	ex08-teorema-parametro.tex . . . . .	55
9.1	ex09-figura.tex . . . . .	57
9.2	ex09-figura-fbox.tex . . . . .	58
9.3	ex09-tabela-flutuante.tex . . . . .	59
9.4	ex09-tabela-tabular.tex . . . . .	59
9.5	ex09-longtable.tex . . . . .	60
9.6	ex09-imagem.tex . . . . .	61
9.7	ex09-imagem-rotacao.tex . . . . .	63
9.8	ex09-imagem-lado.tex . . . . .	63
9.9	ex09-imagem-sobreposicao.tex . . . . .	64
9.10	ex09-picture.tex . . . . .	65
9.11	ex09-scale.tex . . . . .	66
9.12	ex09-rotate.tex . . . . .	67
10.1	ex10-family.tex . . . . .	69
10.2	ex10-series.tex . . . . .	70
10.3	ex10-size.tex . . . . .	72
10.4	ex10-fontes-mat.tex . . . . .	73

---

11.1	ex11-bib.tex . . . . .	75
11.2	ex11-cite.tex . . . . .	76
11.3	ex11-bibtex.bib . . . . .	77
11.4	ex11-index.tex . . . . .	80
12.1	ex12-espacos.tex . . . . .	84
12.2	ex12-contadores.tex . . . . .	86
12.3	ex12-newcounter.tex . . . . .	87
13.1	ex13-fragil.tex . . . . .	89
13.2	ex13-babel.tex . . . . .	90
13.3	ex13-babel-caption.tex . . . . .	92
13.4	ex13-star.tex . . . . .	97
13.5	ex13-parametro-finilizacao.tex . . . . .	98
13.6	ex13-keyval . . . . .	99
13.7	ex13-keyval-env . . . . .	100
13.8	ex13-color.tex . . . . .	102
14.1	ex14-lettrine.tex . . . . .	105
14.2	ex14-fancyhdr.tex . . . . .	106
14.3	ex14-titlesec.tex . . . . .	107
14.4	ex14-caption.tex . . . . .	109
14.5	ex14-float.tex . . . . .	113
14.6	ex14-subcaption.tex . . . . .	114
14.7	ex14-wrapfig.tex . . . . .	114
14.8	ex14-pdflscape.tex . . . . .	115
14.9	ex14-num.tex . . . . .	116
14.10	ex14-si.tex . . . . .	117
14.11	ex14-sisetup.tex . . . . .	118
14.12	ex14-fancybox.tex . . . . .	119
14.13	ex14-framed.tex . . . . .	120
14.14	ex14-lineno.tex . . . . .	120
14.15:	ex14-listings.tex . . . . .	122
14.16	ex14-showexpl.tex . . . . .	123
14.1	ex14-showexpl-completo . . . . .	124
14.2	ex14-natbib.tex . . . . .	125
15.1	ex15-xcolor.tex . . . . .	129
15.2	ex15-tikz-preamble.tex . . . . .	130
15.3	ex15-tikz:basico.tex . . . . .	131
15.4	ex15-tikz-draw.tex . . . . .	131
15.5	ex15-tikz-parametro.tex . . . . .	132
15.6	ex15-tikz-coordenadas.tex . . . . .	132
15.7	ex15-tikz-intersection.tex . . . . .	133

---

15.8	ex15-tikz-circle.tex	134
15.9	ex15-tikz-grafico.tex	135
15.10	ex15-tikz-funcao-tabelada.tex	136
15.11	ex15-tikz-pintar.tex	137
15.12	ex15-tikz-matriz.tex	138
15.13	ex15-tikz-diagrama.tex	139
16.1	ex16-poster.tex	141
16.2	ex16-slides.tex	146
17.1	ex17-lualatex.tex	152
17.2	ex17-biblatex.bib	156
17.3	ex17-biblatex.tex	158
A.1	ex-a-estilo.sty	162
A.2	ex-a-estilo.tex	165
A.3	ex-a-lipsum.tex	166
A.4	ex-a-blindtext.tex	167
C.1	ex-c-lista-nomes.csv	176
C.2	ex-c-certificado.tex	176
C.3	ex-c-conferencia1.tex	180
C.4	ex-c-caderno.tex	180
D.1	ex-d-exsheets.tex	183
D.2	ex-d-answers.tex	184
E.1	ex-e-standalone-fig.tex	187
E.2	ex-e-standalone.tex	188
E.3	ex-e-subfiles-principal.tex	189
E.4	ex-e-subfiles-capitulo1.tex	189

# Capítulo 1

## Introdução

$\LaTeX$  é um sistema de diagramação de documentos profissional largamente utilizados, desenvolvido sobre o  $\TeX$ . Para quem entende inglês, os livros impressos recomendados são [Lam94] e [GMS04]. Outros textos recomendados que podem ser lidos gratuitamente são [OPHS18], [Tea00] e [wik18]. Note que a tradução em português brasileiro do [OPHS18] costuma estar desatualizado em relação à versão original. Então prefira a versão em inglês.

### 1.1 Uma breve história

O  $\TeX$ [Knu86] e a fonte padrão `Computer Modern`, foram desenvolvidos pelo *Donald Knuth* em 1977 a 1985. Dos sistemas implementados sobre o  $\TeX$ , o mais usado é o  $\LaTeX$ [Lam86] desenvolvido pela equipe de *Leslie Lamport*, concluído em 1985. A versão mais utilizada do  $\LaTeX$  foi concluído em 1994 que é o  $\LaTeX 2_{\epsilon}$ [Lam94]. Em 2004, o  $\TeX$  começou a suportar oficialmente o `utf-8`, facilitando a elaboração de documentos multi-idíomas. Em 2008, o recurso de sincronização do documento fonte com o PDF tornou fácil e PDF foi substituindo a saída DVI do  $\TeX$  na diagramação. Em 2010 foi lançado a primeira versão estável do  $\text{Lua}\TeX$ , considerado como o sucessor do  $\text{PDF}\TeX$ . O  $\text{Lua}\TeX$  permite usar fonte do sistema, assim como estender a funcionalidade com a linguagem `script Lua`.

### 1.2 Como usar $\LaTeX$

O  $\TeX$  é um sistema de compilação de documentos e o  $\LaTeX$  é um conjunto de macros (instruções) para automatizar e facilitar a diagramação de documentos.

O documento é preparado como arquivo texto num editor de texto, compilado pelo  $\LaTeX$ , visualizado e corrigido e compilado novamente, até obter o resultado desejado. O arquivo fonte do documento costuma ser editado no editor próprio para  $\LaTeX$  tais como `TeXMaker` (<http://www.xmlmath.net/texmaker/>) e `TeXStudio` (<https://www.texstudio.org/>), ambos disponível livremente em várias plataformas. Além de recursos para facilitar a escrever documentos, eles contam com botões de compilação e visualização, assim como sincronismo de PDF como o código fonte. Para quem quer a funcionalidade mais próxima dos aplicativos de escritó-

rios que vê como ficará enquanto escreve e quer elaborar o documento usando botões e menus em vez de digitar comandos, poderá optar pelo LyX (<https://www.lyx.org/>) que também é livre e suporta várias plataformas. Para evitar erros de escrita, é recomendável que ative o corretor ortográfico com idioma desejado, o que depende de cada editor. Para compilar, deverá instalar alguma distribuição de T<sub>E</sub>X. Em geral, o mais recomendado é o TeXLive para Linux, MacTeX para Mac OSX e MikTeX para Windows, todos são livres.

Evite usar o espaço no nome de arquivos para L<sup>A</sup>T<sub>E</sub>X, pois o sincronismo do código fonte com o PDF (posicionar PDF na posição correspondente à linha de código fonte e vice-versa) pode tornar parcial.

# Capítulo 2

## Iniciando um Documento

Neste capítulo, vamos tratar do básico de como começar um documento  $\LaTeX$ .

### 2.1 Primeiro documento

Um documento em  $\LaTeX$  inicia-se com o comando `\documentclass` que especifica qual tipo de documentos será diagramado. Em seguida, será especificado os pacotes (conjunto de instruções) adicionais a serem carregados e também será realizada algumas configurações. Esta parte do documento é chamado de `preamble` (preâmbulo) do documento. Depois inicia o conteúdo do documento com `\begin{document}`, escreve o corpo do documento e finaliza o documento com `\end{document}`.

O código do primeiro documento e sua saída em pdf (que está dentro da moldura) seria como do Exemplo 2.1.

Exemplo 2.1: ex02-inicial.tex

```
\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}

\begin{document}
Este é primeiro parágrafo.
Continuando o primeiro parágrafo.

Este é o segundo parágrafo.

Espaço extra são eliminados.
\end{document}
```

Este é primeiro parágrafo. Continuando o primeiro parágrafo.  
Este é o segundo parágrafo.  
Espaço extra são eliminados.

O comando do  $\text{\LaTeX}$  inicia com o caractere especial “ $\backslash$ ”. O primeiro comando `\documentclass` tem como parâmetros delimitados pelos colchetes e outro com chaves. O que é delimitado pelo chaves é o parâmetro obrigatório (aqui, é o `article`). Os parâmetros colocados entre colchetes, separados pela vírgula são os opcionais. Neste exemplo, são `12pt` e `a4paper`. Opcionais significa que pode ou não colocar tais opções. O parâmetro obrigatório do comando `\documentclass` é tipo de documentos. Aqui foi escolhido o `article` (artigo) que tem como objetivo, colocar maior quantidade de informações no espaço limitado. Outro tipo de documento bastante usados é o `bookbook` (livro). Os parâmetros opcionais usados foram `12pt` que é o tamanho de letra em `12pt` e `a4paper` que é o tamanho do papel em padrão `A4`.

Depois segue com sequência de comandos `\usepackage`. O `\usepackage` carrega o pacote (conjunto de instruções) que configura o documento ou disponibiliza os comandos específicos. O  $\text{\LaTeX}$  dispõe uma grande quantidade de pacotes, uma para cada situação. No Exemplo 2.1, foram carregados os pacotes `fontenc`, `inputenc` e `babel`, todos eles com um parâmetro. O primeiro pacote `fontenc` é usado para especificar a codificação das fontes de letras. A opção `T1` indica que a fonte está em `T1` que dispõe de letras de 8-bits (acentuadas). Esta opção é útil para definir a regra de hifenização local do documento no `preamble` e efetuar busca de texto com letras acentuadas no PDF final. A seguir, o pacote `inputenc` especifica a codificação do documento fonte (o que está editando). Atualmente, o `utf8` é o recomendado, mas antigamente usava o `latin1` para português brasileiro. Note que `inputenc` foi colocado depois da `fontenc`. Assim, a faixa coberta de caracteres é maior, não limitando somente a faixa de `T1`.

O terceiro e último pacote deste exemplo é o `babel` com a opção `brazil`. O pacote `babel` seleciona a regra de hifenização e nomes dos elementos (como figura, capítulo, etc) para idioma especificada. A opção `brazil` ou `brazilian` escolhe o português brasileiro (não confundir com a opção `portuguese` que escolherá português de Portugal).

Depois encontra o `\begin{document}`. O comando `\begin` inicia um ambiente. Um ambiente é uma configuração que será aplicado nos trechos entre `\begin{<ambiente>}` e `\end{<ambiente>}`. O `\begin{document}` e `\end{document}` determina o ambiente de documento na qual seus conteúdos serão colocados no arquivo PDF.

A quebra de linha não efetua quebra de linha na saída. Para que tenha um novo parágrafo de fato, deverá pular uma linha.

Note que, se tiver mais de um espaço, o  $\text{\LaTeX}$  interpretará como um único espaço.

Para gerar o arquivo PDF, deverá processar o  $\text{\LaTeX}$ . Salve o arquivo com extensão `tex` e clique no botão de `Compile` no caso de `TeXMaker` e botão `LaTeX` no caso de `TeXStudio` (na versão mais nova, é um triângulo verde).

## 2.2 Mensagem de erro e correção

Quando ocorre erro de compilação, o `TeXMaker` mostrará em vermelho na parte de baixo e indicará a linha onde ocorreu o erro. Maioria dos editores para `LaTeX` posicionará automaticamente na linha do primeiro erro quando compila. Para demais erros, ao clicar nas mensagens de erros em vermelho, posicionará automaticamente na linha correspondente a tais erros. Para melhorar a precisão da localização de erros, quebre o parágrafo em várias linhas (lembre-se que quebra de linha não afeta o documento final).

Assim, corrige os erros e compila de novo, até sumir com todos erros.

Quando compila o documento, o `TeXMaker` posicionará na página do PDF, correspondente aonde fica o cursor no editor de texto. Para localizar qual código gerou uma determinada parte de PDF, clique no botão direito sobre parte do PDF e escolha “clique para ir para a linha”.

## 2.3 Caracteres especiais

Existem vários caracteres especiais reservados para os comandos e similares do `LaTeX`. Por exemplo, “\” é usado para iniciar um comando, chaves é usado para indicar os parâmetros, etc. Para inserir estes caracteres especiais no documento, deverá usar os comandos especiais de `LaTeX`.

No código fonte do Exemplo 2.2, foi colocado somente os trechos que ficam no corpo do documento. Para que o arquivo compile, deverá colocar entre `\begin{document}` e `\end{document}` do arquivo válido como do Exemplo 2.1.

Exemplo 2.2: ex02-caracteres.tex

```
% esta é comentário
Alguns caracteres especiais:
\textbackslash, \$, \#, \%, \&, \_, \{, \}

Alguns acentos:
\'a, \^a, \^e, \'i, \~a, \'u

“Abrindo e fechando aspas”

Logo do \LaTeX{} e do \TeX
```

```
Alguns caracteres especiais: \, $, #, %, &, _, {, }
Alguns acentos: á, à, ê, í, ã, ü
“Abrindo e fechando aspas”
Logo do LaTeX e do TeX
```

O Exemplo 2.2 mostra alguns comandos para produzir caracteres especiais.

Note que as letras com a acentuação direta pelo teclado é suportado, desde que `\usepackage[utf8]{inp}` ou equivalentes sejam colocados no `preamble` do documento. No exemplo acima, foi mostrado a acentuação do modo `TEX` que pode ser usados em alguns casos especiais, tais como dentro do arquivo de referências bibliográficas, ou arquivo de pacotes. Note que `\'` acentua a letra seguinte, mas “i” tem pingo e “í” (com acento) não tem pingo. `\i` é o comando para produzir “i” sem pingo. Note que acento agudo no modo `TEX` é produzido por apóstrofos e não pelo acento agudo. Apóstrofos é aberto por um acento agudo e fechado pelo apóstrofos. Aspas é aberto pelos dois acentos agudos e fechado pelos dois apóstrofos (fiquem atentos de que fechamento de aspas é dois apóstrofos e não é aspas).

No `LATEX`, quando encontra “%”, o restante desta linha será considerado como comentário e é ignorado completamente.

Comentário é importante para inserir observações sobre o código, ou desativar um trecho do código. Em geral, quem quer remover um trecho do código de `LATEX` no documento, simplesmente comenta o trecho, pois se algum dia quiser ativar, é só remover o “%” do comentário.

Para comentar/descomentar um trecho maior no `TeXMaker`, selecione o trecho e use o `Editar->Comentar` e `Editar->Descomentar`.

Quando editar um documento, existem palavras que devem aparecer grudados (não pode ficar primeira parte no final de linha e outra no começo da linha) como no caso de enumeração de páginas, exemplos, teoremas, etc. Neste caso, usa-se o til (`~`) em vez de espaço. Por exemplo, no caso de `página~1`, não acontece de página ficar no final de uma linha e `1` ficar no começo da próxima linha.

Por último, quando insere comandos e precisa ter espaço depois dele, coloque um par de chaves. Por exemplo, `\LaTeX produz` ficaria como “`LATEX`produz” (grudados) enquanto que `\LaTeX{} produz` ficará como “`LATEX` produz” (com espaço correto). Também lembra que no comando de `LATEX`, maiúsculo e minúsculo são distinguidas. Portanto, `Latex` ou `\latex` resultarão em erros em vez de produzir logotipo de `LATEX`.

# Capítulo 3

## Introdução às Fórmulas Matemáticas

Neste capítulo, veremos um pouco sobre fórmulas matemáticas.

### 3.1 Fórmula `textstyle` e `displaystyle`

Uma fórmula matemática (ou modo matemático) `textstyle` ou `inlinestyle` é uma fórmula no meio do texto. Por exemplo, denotaremos uma sequência por  $\{x_n\}$  e seu limite por  $\lim_n \rightarrow \infty x_n$  é modo `textstyle`, pois fórmulas estão no meio do texto. Agora as fórmulas matemáticas que ocupam linha separada de texto como em

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

é denominado de fórmula matemática em `displaystyle`.

A fórmula no modo `textstyle` devem ficar delimitados entre “\$” ou “\ (“ e “\)”, ou ainda poderá usar o ambiente `math`. O mais usado é delimitar com “\$”.

A fórmula no modo `displaystyle` devem ficar delimitados entre “\$\$” (dois dólares) ou “\[” e “\]”, ou ainda poderá usar o ambiente `displaymath`. Recomendado que delimite com “[” e “\]”, pois delimitar com “\$\$” (dois dólares) dificultará a depuração (achar erros).

Exemplo 3.1: ex03-formulas.tex

Solução da equação  $ax^2+bx+x=0$  é dado pela fórmula

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

Elemento da matriz  $A$  costuma ser denotado por  $a_{ij}$

Solução da equação  $ax^2 + bx + c = 0$  é dado pela fórmula

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

Elemento da matriz  $A$  costuma ser denotado por  $a_{ij}$

No modo matemático, “^” é usado para indicar a potência e “\_” indica o índice. Outros comandos utilizados foram, `\frac` que produz frações, onde primeiro parâmetro é numerador e segundo, o denominador. O comando `\sqrt` produz raiz do argumento. O comando `\pm` é o símbolo  $\pm$ . Na matemática, muitos símbolos são usados e o  $\text{\LaTeX}$  dispõe de comandos para cada uma desses símbolos. Em geral, o editor para  $\text{\LaTeX}$  dispõe de painel de inserção dos símbolos matemáticos para ajudar na elaboração do documento. No caso de **TeXMaker**, tem o botão do painel lateral esquerdo permitem ativar tais painéis, organizados em grupos. As letras gregas no  $\text{\LaTeX}$  são produzidas pelo comando com nome em inglês. Se nome começar em maiúsculo, será letra grega maiúscula.

Note que, quando os parâmetros de um comando for mais de um (mais de uma letra, por exemplo), deverá colocar entre chaves, como foram feitos para `\frac` e `\sqrt`.

Isto vale também para expoentes e índices, como foi feito no elemento da matriz (com índices de duas letras).

O comando `\sqrt` aceita o parâmetro opcional (delimitado pelos colchetes) para poder produzir raiz  $n$ -ésima. O `\sqrt[n]{x}` produz  $\sqrt[n]{x}$ . O Exemplo 3.2 ilustra algumas letras gregas e símbolos. Note que uma tabela completa de símbolos do  $\text{\LaTeX}$  está em <http://tug.ctan.org/info/symbols/comprehensive/symbols-a4.pdf>.

#### Exemplo 3.2: ex03-simbolos.tex

Algumas letras gregas minúsculas: `\alpha`, `\beta`, `\gamma`, `\lambda`, `\pi`.

Algumas letras gregas Maiúsculas: `\Alpha`, `\Beta`, `\Gamma`, `\Lambda`, `\Pi`.

Alguns símbolos:

`\le`, `\ge`, `\neq`, `\in`, `\notin`, `\exists`, `\nexists`, `\to`, `\infty`, `\forall`.

Algumas letras gregas minúsculas:  $\alpha, \beta, \gamma, \lambda, \pi$ .

Algumas letras gregas Maiúsculas:  $A, B, \Gamma, \Lambda, \Pi$ .

Alguns símbolos:  $\leq, \geq, \neq, \in, \notin, \exists, \nexists, \rightarrow, \infty, \forall$ .

## 3.2 Modo `displaystyle` no meio do texto

As fórmulas no meio do texto (`textstyle`) são produzidos de forma que economize a sua altura. Por isso, elementos que normalmente ficariam empilhados, ficariam como índice e expoente. Por exemplo,  $\lim_{n \rightarrow \infty} x_n$  no modo `textstyle` ficaria como  $\lim_{n \rightarrow \infty} x_n$ . Para que uma fórmula fique como `displaystyle` no meio do texto, coloque `\displaystyle` no começo das fórmulas. O Exemplo 3.3 ilustra o caso.

Exemplo 3.3: `ex03-displaystyle.tex`

Temos que  $\lim_{n \rightarrow \infty} x_n = 0$  *%textstyle*  
e  $\lim_{n \rightarrow \infty} x_n = 0$  *%displaystyle*

Outros casos  $\sum_{i=0}^n i = 0 + 1 + \dots + n$   
e  $\sum_{i=0}^n i = 0 + 1 + \dots + n$ .

Forçando a colocar encima/embaixo

$\int_a^b f(x) dx$

Forçar a não colocar encima/embaixo

$\sum_{k=0}^n x_k = 1$

Temos que  $\lim_{n \rightarrow \infty} x_n = 0$  e  $\lim_{n \rightarrow \infty} x_n = 0$

Outros casos  $\sum_{i=0}^n i = 0 + 1 + \dots + n$  e  $\sum_{i=0}^n i = 0 + 1 + \dots + n$ .

Forçando a colocar encima/embaixo

$$\int_I f = \int_a^b f(x) dx$$

Forçar a não colocar encima/embaixo

$$\sum_{k=0}^n x_k = 1$$

Para forçar a colocar elementos embaixo em vez de como índice quando usa o “\_”, usa-se o comando `\limits`, como no Exemplo 3.3.

Note que, para colocar no modo `textstyle` nas fórmulas `displaystyle`, existe o comando `\textstyle` que é usado de forma similar a `\displaystyle`. Nesta família de comandos, também existe o comando `\scriptstyle` que tenta reduzir o tamanho das fórmulas.

## 3.3 Equação enumerada e referências cruzadas

Uma equação enumerada é produzido pelo ambiente `equation`. Quando algum comando do L<sup>A</sup>T<sub>E</sub>X enumera automaticamente (equações, seções, capítulos, figuras, tabelas, itens de istas enu-

meradas, teoremas, etc), sua enumeração pode ser guardado e usado no outro lugar, denominado de referência cruzada. Para tanto, coloca-se o comando `\label{nome do rótulo}` que produz um rótulo, armazenando a enumeração onde produz a enumeração e `\ref{nome do rótulo}` aonde quer colocar esta enumeração. O Exemplo 3.4 ilustra o uso de referência cruzada.

Exemplo 3.4: ex03-ref.tex

```
\begin{equation}\label{eq:pitagoras}
a^2=b^2+c^2
\end{equation}
Pela Equação~\ref{eq:pitagoras} da página~\pageref{eq:pitagoras}\ldots
```

$$a^2 = b^2 + c^2 \tag{1}$$

Pela Equação 1 da página 10...

Como já mencionado, o ambiente `equation` produz enumeração automática na equação. Assim, poderá criar um rótulo para referenciar o número desta equação com o comando `\label`. O nome do rótulo escolhido para esta equação foi `eq:pitagoras`. Tome cuidado para não usar espaços, letras acentuadas ou símbolos especiais, o que dificulta a depuração ou causa erros. No Exemplo 3.4, “:” foi usado em vez do espaço para separar palavras. Note que foi colocado o prefixo “eq” para rótulo da equação. Como pode ter teoremas, equações, figuras, etc que podem sugerir o mesmo nome, é costume usar algum prefixo para cada categoria, o que facilitará a lembrar o elemento que é associado ao rótulo. Para referenciar (usar o número) da equação, foi usado o comando `\ref` e para referenciar a página, foi usado o comando `\pageref`. Note que os editores especializados para  $\text{\LaTeX}$ , costuma ter recursos de listar os rótulos existentes para referências, o que facilitaria a editoração e prevenir erros. Também note o uso de “~” em vez do espaço para usar referências, o que evita que nome “Equação” fique no final de uma linha e numeração no começo da próxima linha.

Quando usa a referência cruzada, as enumerações sempre estarão coerentes e se usar o pacote `hyperref`, terão link automático no documento PDF.

# Capítulo 4

## Estrutura do Texto

Neste capítulo, vamos ver sobre a formatação de textos.

### 4.1 Alinhamentos

O alinhamento do texto padrão no  $\text{\LaTeX}$  é justificado, isto é, alinhado tanto a direita como a esquerda. Para alinhar somente a esquerda ou a direita, usa-se o ambiente `flushleft` e `flushright`. Para centralizar, usa-se o ambiente `center`. O Exemplo 4.1 ilustra os alinhamentos de texto.

Exemplo 4.1: ex04-alinhamento.tex

```
Parágrafo normal.
\begin{center}
Parágrafo centralizado.
\end{center}
\begin{flushright}
Parágrafo alinhado a direita.
\end{flushright}
```

Parágrafo normal.

Parágrafo centralizado.

Parágrafo alinhado a direita.

Quando quer alinhar dentro do ambiente (o que tem `\begin` e `\end`, poderá colocar como comando `\center`, `\flushleft` e `\flushright`, respectivamente que o alinhamento se aplicará até obter o final do ambiente em questão.

Para ter trecho justificado em ambos lados dentro do ambiente com alinhamento a esquerda, direita ou centralizada, costuma colocar dentro do ambiente `minipage` que veremos na Seção 4.7.

Não é o alinhamento, mas para incluir linhas em branco, deverá colocar “\” intercalado com linhas em branco. Para indicar uma nova linha (finalizar uma linha sem esperar atingir final de linha), coloca se “\” ou `\newline`. Para inserir quebra de linhas, mas que justifique, deverá usar o `\linebreak`, o que é ilustrado no Exemplo 4.2

Exemplo 4.2: ex04-nova-linha.tex

Esta é a primeira linha \\  
e esta e a segunda linha

\

Esta é a primeira linha `\linebreak`  
e esta é a segunda linha.

Esta é a primeira linha  
e esta e a segunda linha

Esta é a primeira linha  
e esta é a segunda linha.

Note o uso de “\” para inserir linha em branco.

## 4.2 Notas de rodapé e ênfase de texto

Nota de rodapé é colocado pelo comando `\footnote{<texto>}` que colocará marca de rodapé no local e `<texto>` na parte inferior da página atual.

Ele é usado para colocar alguma observação, mas não quer que conste como conteúdo do documento. Por exemplo, explicação dos termos que aparece, podem ser colocados como rodapé. Veja o Exemplo 4.3

Exemplo 4.3: ex04-rodape.tex

No meio do texto, podemos colocar a nota de rodapé `\footnote{nota que fica na parte inferior da página}` para explicações adicionais tais como significado da palavra, ou fonte que foi usada.

No meio do texto, podemos colocar a nota de rodapé<sup>1</sup> para explicações adicionais tais como significado da palavra, ou fonte que foi usada.

<sup>1</sup>nota que fica na parte inferior da página

Para enfatizar texto dentro do contexto como os termos a ser definidos, usa-se o comando `\emph{texto}` ou o ambiente `em`. Este comando (ou ambiente) alterna entre fontes romano reto e itálico para que o trecho seja enfatizada. Note que os ambientes `em` tem a versão de comandos `\em` que pode ser usados dentro do outro ambiente (como os comandos de alinhamento). Caso não esteja dentro do ambiente, poderá delimitar simplesmente pelos chaves e colocar o comando no começo dele. Isto está ilustrado no Exemplo 4.4.

Exemplo 4.4: ex04-enfase.tex

```
\emph{Enfatizado} ou
\begin{em}enfazizado\end{em}
ou
\begin{center}
\em Centralizado e enfatizado.
\end{center}
ou
{\em Enfatizado.}
Texto normal.
```

*Enfatizado* ou *enfazizado* ou

*Centralizado e enfatizado.*

ou *Enfatizado.* Texto normal.

## 4.3 Listas

O texto pode ter várias estruturas básicas, tais como listas, citações, etc.

Uma delas é listas.

Existem três tipos básicos de listas que são especificados pelos ambientes `enumerate` (lista enumerada), `itemize` (lista de itens) e `description` (lista de descrições). Dentro de cada ambiente, cada item é iniciado com o comando `\item`. O Exemplo 4.5 ilustra algumas listas.

Exemplo 4.5: ex04-listas.tex

```
\begin{enumerate}
\item Este é o primeiro item da lista enumerada.
\item Agora, segundo item da lista enumerada
\end{enumerate}
\begin{itemize}
\item Este é o primeiro item da lista de itens.
\item Agora, segundo item da lista de itens.
\end{itemize}
```

```

\begin{itemize}
  \item Item com marcador padrão.
  \item [\$\ast\$] Item com marcador personalizado.
\end{itemize}
\begin{description}
  \item [enumerada] Cada item recebe uma enumeração.
  \item [itenizada] Em vez de enumeração, recebe um marcador.
  \item [descrição] Descrição das palavras.
\end{description}

```

1. Este é o primeiro item da lista enumerada.
2. Agora, segundo item da lista enumerada
  - Este é o primeiro item da lista de itens.
  - Agora, segundo item da lista de itens.
  - Item com marcador padrão.
  - \* Item com marcador personalizado.

**enumerada** Cada item recebe uma enumeração.

**itenizada** Em vez de enumeração, recebe um marcador.

**descrição** Descrição das palavras.

As listas podem conter outras listas como itens. Veja Exemplo 4.6.

Exemplo 4.6: ex04-sublistas.tex

```

\begin{enumerate}
  \item Este é o primeiro item da lista enumerada.
  \item Segundo item é lista de itens.
    \begin{itemize}
      \item Este é o primeiro item da lista de itens.
      \item Agora, segundo item da lista de itens.
    \end{itemize}
  \item Este é o terceiro item da lista enumerada.
\end{enumerate}

```

1. Este é o primeiro item da lista enumerada.
2. Segundo item é lista de itens.
  - Este é o primeiro item da lista de itens.
  - Agora, segundo item da lista de itens.
3. Este é o terceiro item da lista enumerada.

Para trabalhar com listas enumeradas, é aconselhável carregar o pacote `enumerate` que permite controlar as enumerações. Para isso, acrescente o comando `\usepackage{enumerate}` no `preamble` do documento.

No `enumerate`, passa-se um parâmetro opcional que seria o modelo de enumeração do primeiro item. Neste modelo, “1”, “i”, “I”, “a” e “A” serão considerados contadores e eles são incrementados a cada item. Se aparecer estas letras que não sejam contadores, delimite pelos chaves. Veja o Exemplo 4.7.

Exemplo 4.7: ex04-enumerate.tex

```
\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\usepackage{enumerate}
\begin{document}
\begin{enumerate}[Propriedade 1]
\item  $x + (y + z) = (x + y) + z$  (associativa)
\item  $x + y = y + x$  (comutativa)
\end{enumerate}
\end{document}
```

Propriedade 1)  $x + (y + z) = (x + y) + z$  (associativa)

Propriedade 2)  $x + y = y + x$  (comutativa)

Se precisar de ambientes extras de listas diferentes do padrão, podemos definir usando o ambiente `list`, o que não vamos entrar em detalhes.

## 4.4 Tabelas

Uma tabela no modo texto é produzido pelo ambiente `tabular` e a tabela no modo matemático é produzido pelo ambiente `array`, que apresentam o mesmo sintaxe e a mesma funcionalidade.

O argumento obrigatório destes ambientes é o “alinhamento” das colunas que devem ser especificados com “l” (left), “c” (center), “r” (right) ou “p{largura}” (texto justificado com largura fixa). Para traçar uma linha vertical entre colunas ou no bordo, usa-se o “|”, junto as especificações de alinhamento. Os elementos da tabela é indicado, separado pelo “&”, sendo que a mudança de linhas é feito pelo “\” que é `\newline`. Para traçar uma linha horizontal, usa-se o comando “`\hline`”.

O Exemplo 4.8 é um exemplo de tabelas no modo texto.

Exemplo 4.8: ex04-tabular.tex

```
\begin{tabular}{||l|c|r||} % linhas verticais duplas na borda e simples entre colunas
\hline % uma linha horizontal no começo da tabela
  & 2 & 3 \\ \hline % uma linha horizontal apos esta linha
abc & 3 & 4 \\
3 & 4 & $\frac{1+\sqrt{2}}{5}$ \\ \hline % uma linha horizontal pra finalizar
\end{tabular}
```

Tabela com uma coluna de largura fixa.

```
\begin{tabular}{|p{3.5cm}|l|} \hline
cenouras (500g) & R$0,50 \\ \hline
cogumelos (vidro de 500g) & R$5,00 \\ \hline
batata (1Kg) & R$1,20 \\ \hline \hline
total & R$7,20 \\ \hline
\end{tabular}
```

	2	3
abc	3	4
3	4	$\frac{1+\sqrt{2}}{5}$

Tabela com uma coluna de largura fixa.

cenouras (500g)	R\$0,50
cogumelos (vidro de 500g)	R\$5,00
batata (1Kg)	R\$1,20
total	R\$7,20

Para mesclar células (juntar mais de uma célula como sendo uma única célula), usa-se o comando `multicolumn` que tem como primeiro argumento, o número de células a serem juntados, o segundo especifica o alinhamento da coluna e terceiro, o que vai colocar nesta célula. Note que as formatações de colunas do `multicolumn` deve ser especificado um por um, independente de estar ou não estar especificado no começo da tabela.

No Exemplo 4.9 com o ambiente `array` (para ambiente matemático), foi usado o `multicolumn` para juntar duas primeiras colunas da primeira linha, centrando os dados e traçando linha vertical antes e depois da célula.

Exemplo 4.9: ex04-multicolumn.tex

```

\begin{array}{|c|c|c|} \hline
\multicolumn{2}{|c|}{2} & 3 \\ \hline
2^3 & 3 & 4 \\ \hline
3 & 4 & \frac{1+\sqrt{2}}{5} \\ \hline
\end{array}

```

2		3
2 <sup>3</sup>	3	4
3	4	$\frac{1+\sqrt{2}}{5}$

Para traçar linhas horizontais apenas em algumas células, usa-se o comando `\cline{i-j}` onde “i” e “j” são colunas iniciais e finais onde a linha é traçada. Veja o Exemplo 4.10.

Exemplo 4.10: ex04-cline.tex

```

\begin{tabular}{|l|l|} \hline
\textbf{Produto} & \multicolumn{2}{|c|}{\textbf{Preços}} \\ \hline
cenouras & R\$1.00 & (por Kg) \\ \cline{2-3}
& R\$0.20 & (por unidade) \\ \hline
cogumelos & R\$4.00 & (por vidro) \\ \cline{1-1} \cline{3-3}
pêssego & & (por Kg) \\ \hline
\end{tabular}

```

Produto	Preços	
cenouras	R\$1.00	(por Kg)
	R\$0.20	(por unidade)
cogumelos	R\$4.00	(por vidro)
pêssego		(por Kg)

No Exemplo 4.10, o comando `multicolumn` foi usado somente para formatar a coluna (colocar linha vertical) e não para juntar células.

Se precisar traçar linhas duplas ou similares na tabela, podemos usar o pacote `hhline` que não entraremos em detalhes.

Para alinhar os decimais, usamos o pacote `dcolumn` que define um novo especificador de colunas na tabela, especificado por `D{<decimal em TeX>}{<decimal em PDF>}{<casas decimais>}`.

`<decimal em TeX>` é a especificação de pontos decimais usado no arquivo TeX (como escreve no TeX), `<decimal em PDF>` é o decimal utilizado para documento de saída PDF, e o `<casas`

`decimais`> é o número de casas decimais a serem considerados (casas decimais excedentes serão truncados). Caso especificação do número de casas decimais for negativo, qualquer número de casas decimais é aceito sem ser truncado (alinhando nos pontos decimais).

Como especificar três parâmetros toda vez que precise alinhar os pontos decimais é trabalhoso, é sugerido que defina um especificador no `preamble`, usando o comando `newcolumntype` como abaixo:

```
\usepackage{dcolumn}
\newcolumntype{d}[1]{D{.}{\cdot}{#1}}
\newcolumntype{.}{D{.}{.}{-1}}
\newcolumntype{,}{D{,}{,}{2}}
```

Ele define especificador de colunas “`d<num>`” que é interpretado como `D{.}{.}{<num>}` e, “.” e “,” que serão interpretados como sendo `D{.}{.}{-1}` e `D{,}{,}{2}` respectivamente.

Note que podemos modificar os comandos anterior para que “.” seja convertida para “,” ou vice versa. Veja o Exemplo 4.11.

Exemplo 4.11: ex04-dcolumn.tex

```
\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\usepackage{dcolumn}
\newcolumntype{d}[1]{D{.}{\cdot}{#1}}
\newcolumntype{.}{D{.}{.}{-1}}
\newcolumntype{,}{D{,}{,}{2}}
\begin{document}
\begin{tabular}{|d{-1}|d{2}|.|,|}
\hline
1.2 & 1.2 & 1.2 & 1,2 \\
1.23 & 1.23 & 12.5 & 300,2 \\
1121.2 & 1121.2 & 861.20 & 674,29 \\
184 & 184 & 10 & 69 \\
.4 & .4 & & ,4 \\
& & & ,4 \\
\hline
\end{tabular}
\end{document}
```

1.2	1.2	1.2	1,2
1.23	1.23	12.5	300,2
1121.2	1121.2	861.20	674,29
184	184	10	69
.4	.4		,4
		.4	

Para ter as colunas igualadas automaticamente, usamos o pacote `tabularx` que define o ambiente `tabularx` onde primeiro argumento é largura da tabela e segundo é especificação de colunas, mas apresenta um especificador de coluna especial “X”. Todas as colunas especificadas por “X” terá mesma largura (largura destas colunas depende da largura da tabela). Veja o Exemplo 4.12.

Exemplo 4.12: ex04-tabularx.tex

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\usepackage{tabularx}
\begin{document}
\begin{tabularx}{\linewidth}{|c|X|c|X|}
\hline
\multicolumn{2}{|c|}{\texttt{\multicolumn} } & & \\
\scshape Terceira & \scshape Quarta & \hline
primeira & a largura desta coluna depende da largura da
tabela. & & \\
terceira & Quarta coluna funciona da forma similar a segunda e terá a mesma largura da
segunda & \hline
\end{tabularx}
\end{document}

```

multicolumn		TERCEIRA	QUARTA
primeira	a largura desta coluna depende da largura da tabela.	terceira	Quarta coluna funciona da forma similar a segunda e terá a mesma largura da segunda

No Exemplo 4.12, foi usado o comando `\linewidth` que é a medida da largura da linha atual. Este comando aparece frequentemente quando queremos ajustar largura de um elemento para largura da linha atual. Mais sobre medidas, veja a Seção 12.2 do Capítulo 12.

O comando `\footnote{texto}` que coloca o “texto” no rodapé do documento funciona também dentro do `tabularx`.

Note que `hhline` que produz linhas duplas e similares não funciona no `tabularx`.

## 4.5 Ambiente de tabulação

Ambiente de tabulação é bastante útil para escrever conteúdos alinhados em diversos pontos, tal como no caso de algoritmos. O ambiente é especificado pelo `tabbing`.

Para marcar a posição de tabulação, usa-se o comando `\=`. Quando uma linha for usada somente para marcar posição, coloca-se o comando `\kill` no final dele, para que o mesmo não seja impresso.

Para formatar conteúdos tabulados, é comum que queira tabular várias linhas sucessivas. O comando `\+` translada margem esquerda por uma tabulação e `\-` retira uma tabulação da margem esquerda. Com isso, podemos diagramar textos tabulados como no caso do algoritmo do Exemplo 4.13.

Exemplo 4.13: ex04-tabbing.tex

```
\begin{tabbing}
mm\=mm\=mm\=mm\=mm\= \kill % apenas para marcar tabulação
\{ Solução real da equação  $ax^2+bx+c=0$  pela fórmula de Baskara } \\\
 $\Delta = b^2-4ac$  \\\
se  $d < 0$  então \+ \\\ % aplicar uma tabulação em todas linhas de abaixo
  escreva “não há solução real” \\\
  pare \- \\\ % retirar uma tabulação de todas linhas de abaixo
senão \+ \\\
 $x_1 = \frac{-b-\sqrt{\Delta}}{2a}$  \\\
 $x_2 = \frac{-b+\sqrt{\Delta}}{2a}$  \- \\\
fim se
\end{tabbing}
```

```
{ Solução real da equação  $ax^2 + bx + c = 0$  pela fórmula de Baskara }
 $\Delta = b^2 - 4ac$ 
se  $d < 0$  então
  escreva “não há solução real”
  pare
senão
 $x_1 = \frac{-b-\sqrt{\Delta}}{2a}$ 
 $x_2 = \frac{-b+\sqrt{\Delta}}{2a}$ 
fim se
```

Se por algum motivo, pretende usar a acentuação no modo `TEX` dentro do ambiente `tabbing`, note que existe três acentos no modo `TEX` que não pode ser usado dentro do `tabbing` que são `\'`, `\`` e `\=`. Estes comandos são reservados para controle de tabulação. Assim, estes três comandos devem ser substituídos pelos comandos `\a'`, `\a`` e `\a=` respectivamente (eles só funcionam dentro do `tabbing`).

Existem vários outros comandos que podem ser explorados no ambiente `tabbing`, mas não vamos entrar em detalhes. Uma observação é que no caso de algoritmos ou código fonte de

programas, existem pacotes específicos que implementam ambientes bem mais fáceis e versáteis do que `tabbing`.

## 4.6 Textos de citações, versos e verbatim

O ambiente `quote` é usado para escrever citações (trecho de outra fonte), exemplos e frases importantes. O Exemplo 4.14 foi extraído do “The Not (So) Short Introduction to  $\text{\LaTeX}2\epsilon$ ” [OPHS18].

Exemplo 4.14: `ex04-quote.tex`

```

“Uma regra da tipografia sobre
o comprimento de uma linha é:
\begin{quote}
Em média, nenhuma linha deve
ser maior que 66 caracteres.
\end{quote}
Este é o motivo pelo qual as
páginas do  $\text{\LaTeX}$  possuem as
bordas tão grandes e também o
motivo pelo qual os jornais usam
impressão em colunas”.

```

```

“Uma regra da tipografia sobre o comprimento de uma linha é:

    Em média, nenhuma linha deve ser maior que 66 caracteres.

Este é o motivo pelo qual as páginas do  $\text{\LaTeX}$  possuem as bordas tão grandes e também o
motivo pelo qual os jornais usam impressão em colunas”.

```

No ambiente `quote` usado para colocar citações (trecho tirado da outra fonte), o texto inteiro ficará com margens maiores a esquerda e a direita, o que tornaria visível quando colocar texto maior.

Quando o texto da citação é formado por vários parágrafos, pode querer que os parágrafos sejam indentados (tabulados para direita). Neste caso, use o ambiente `quotation` em vez de `quote`. O Exemplo 4.15 também usa o trecho do [OPHS18] que explica os ambientes similares a `quote`, ficou com parágrafo indentado:

Exemplo 4.15: `ex04-quotation.tex`

```

Veja que a cópia do trecho do “The Not (So) Short Introduction to  $\text{\LaTeX}2\epsilon$ ” que explica
os ambientes similares a  $\text{\texttt{quote}}$ , ficou com parágrafo indentado:
\begin{quotation}
“Existem dois ambientes similares: os ambientes quotation e verse.

```

Como ambiente `\texttt{quotation}` faz a indentação dos parágrafos, ele é usado para citações longas que se estendem por vários parágrafos. O ambiente `\texttt{verse}` é usado em poemas onde as quebras de linhas são importantes. As linhas são separadas por `\verb+\` e por uma linha em branco no fim de cada verso”.  
`\end{quotation}`

Veja que a cópia do trecho do “The Not (So) Short Introduction to L<sup>A</sup>T<sub>E</sub>X2<sup>ε</sup>” que explica os ambientes similares a `quote`, ficou com parágrafo indentado:

“Existem dois ambientes similares: os ambientes `quotation` e `verse`. Como ambiente `quotation` faz a indentação dos parágrafos, ele é usado para citações longas que se estendem por vários parágrafos. O ambiente `verse` é usado em poemas onde as quebras de linhas são importantes. As linhas são separadas por `\` e por uma linha em branco no fim de cada verso”.

No caso de versos, a linha que for quebrado pela falta de espaços, mas que constitui a mesma linha de cima, deve estar tabulado para direita. O ambiente `verse` encarrega deste serviço.

O Exemplo 4.16 é um exemplo do “Humpty Dumpty” do [OPHS18], colocado na caixa com largura insuficiente. Para criar caixas mais estreitas, foi usado o ambiente `minipage` que será tratado na Seção 4.7 do Capítulo 4.

Exemplo 4.16: ex04-verse

```
\begin{flushleft}
\begin{minipage}{0.5\linewidth}
\begin{verse}
Humpty Dumpty sat on a wall:\
Humpty Dumpty had a great fall.\
All the King’s horses and all
the King’s men\
Couldn’t put Humpty together
again.
\end{verse}
\end{minipage}
\end{flushleft}
```

Humpty Dumpty sat on a wall:  
Humpty Dumpty had a great fall.  
All the King’s horses and all the  
King’s men  
Couldn’t put Humpty together  
again.

Para inserir código fonte de programas ou similar, usa-se o ambiente `verbatim`. Colocar texto de jeito que está (cópia `verbatim`) pode ser feito pelo ambiente `verbatim` e `verbatim*`, o

que é muito usado para colocar trecho do código de programas. Sem o “\*”, coloca espaço como espaço e com o “\*”, coloca caractere “\_” no lugar de espaço. Veja a diferença entre o código colocado pelo `verbatim` e `verbatim*` do Exemplo 4.17.

Exemplo 4.17: ex04-verbatim.tex

```
\begin{verbatim}
\begin{quote}
texto citado, texto importante ou exemplos.
\end{quote}
\end{verbatim}

\begin{verbatim*}
\begin{quote}
texto citado, texto importante ou exemplos.
\end{quote}
\end{verbatim*}
```

```
\begin{quote}
texto citado, texto importante ou exemplos.
\end{quote}

\begin{quote}
texto_citado, _texto_importante_ou_exemplos.
\end{quote}
```

O ambiente `verbatim` é usado para colocar conteúdos que devem ser lidos letra por letra, como o código fonte de programas ou trecho de documentos muito importantes.

Para nome das variáveis do programa, nome de arquivos, etc que são curtos, mas também precisam ser lidos letra por letra, existe a versão comando que é `\verb`. No `\verb`, o primeiro caractere é o delimitador do argumento que deve ser usado também para indicar o final do argumento. Este delimitador pode ser escolhido (só não pode ser “\*”), mas fique atento que o final do argumento deve usar mesmo caractere. A versão com “\*” exibe o espaço como “ ”. Veja o Exemplo 4.18.

Exemplo 4.18: ex04-verb.tex

O comando `\verb+\%+ produz “\%”`.

Nome de arquivos com espaço tal como `\verb*|meu arquivo.tex|` não é recomendado para `\LaTeX`, pois dificulta a sincronização de PDF com o código fonte. Use “`\verb+--+`” no lugar de espaço, como em `\verb*|meu—arquivo.tex|`.

O comando `\%` produz “%”.

Nome de arquivos com espaço tal como `meu_arquivo.tex` não é recomendado para  $\text{\LaTeX}$ , pois dificulta a sincronização de PDF com o código fonte. Use “-” no lugar de espaço, como em `meu-arquivo.tex`.

Para textos longos no ambiente `verbatim` deve carregar o pacote `verbatim`. Este pacote efetua algumas melhorias no ambiente `verbatim` e também implementa o comando `\verbatiminput` que insere o arquivo externo diretamente no ambiente `verbatim`.

Note que, para inserir trecho dos códigos fontes de programas, existem pacotes apropriados que é mais prático do que usar o ambiente `verbatim`.

## 4.7 Caixa minipage

O ambiente `minipage` cria uma “caixa” que não é exatamente uma estruturação de textos, mas é usado frequentemente quando não encontra ambiente pronto para estruturação desejada.

Por exemplo, podemos colocar um bloco de texto justificado no lado direito da página, como no Exemplo 4.19.

Exemplo 4.19: `ex04-minipage.tex`

```
\begin{flushright}
\begin{minipage}{0.5\linewidth}
Este texto ficará justificado, mas como fica dentro de \texttt{minipage},
podemos colocar no lado direito.
Para tanto, basta colocar \texttt{minipage} dentro do \texttt{flushright}.
Claro que podemos colocar equações, figuras e outros elementos sem problemas.
Por exemplo, o Teorema de Pitágoras foi posto abaixo:
\[ a^2=b^2+c^2 \]
\end{minipage}
\end{flushright}
```

Este texto ficará justificado, mas como fica dentro de `minipage`, podemos colocar no lado direito. Para tanto, basta colocar `minipage` dentro do `flushright`. Claro que podemos colocar equações, figuras e outros elementos sem problemas. Por exemplo, o Teorema de Pitágoras foi posto abaixo:

$$a^2 = b^2 + c^2$$

O argumento obrigatório para ambiente `minipage` é a largura da “caixa”. No Exemplo 4.19, foi usado `0.5\linewidth` que é metade da largura da linha.

Este ambiente é importante para colocar um parágrafo de texto dentro dos comandos que não aceitam parágrafos. Por exemplo, `\fbox{}` coloca moldura no elemento, mas não aceita o parágrafo de texto. Então, como colocar moldura no texto com várias linhas, ou nas figuras? É simples: colocar tudo no `minipage` e colocar dentro do `fbox`, como no Exemplo 4.20.

Exemplo 4.20: ex04-minipage-fbox.tex

```
\fbox{
\begin{minipage}{0.75\linewidth} % 3/4 da largura de linha
Usando o \texttt{minipage}, podemos colocar moldura no texto com parágrafo como este.

O \texttt{fbox} não acusa erros por ter parágrafo, pois o que está dentro de \texttt{
minipage} é apenas um “objeto”.

Este truque funciona também para outros comandos que impede de colocar parágrafos,
mudar linhas, etc.

Note que as notas de rodapé\footnote{rodapé é colocado pelo comando \texttt{\
\textbackslash footnote\{texto\} } }
colocado dentro do minipage ficará na parte de baixo do \texttt{minipage} e não da página.
O padrão dentro do \texttt{minipage} é enumerar eles como letra e não com números.
\end{minipage}
} % fbox
```

Usando o `minipage`, podemos colocar moldura no texto com parágrafo como este.

O `fbox` não acusa erros por ter parágrafo, pois o que está dentro de `minipage` é apenas um “objeto”.

Este truque funciona também para outros comandos que impede de colocar parágrafos, mudar linhas, etc.

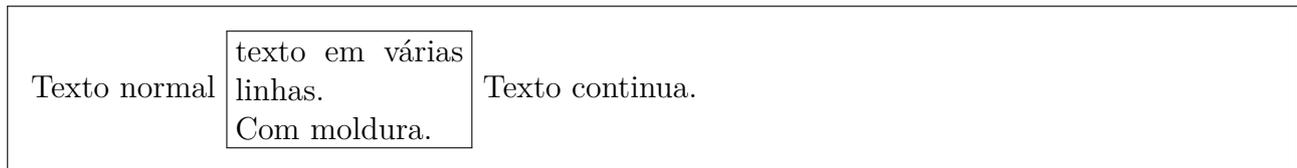
Note que as notas de rodapé<sup>a</sup> colocado dentro do `minipage` ficará na parte de baixo do `minipage` e não da página. O padrão dentro do `minipage` é enumerar eles como letra e não com números.

<sup>a</sup>rodapé é colocado pelo comando `\footnote{texto}`

Para elementos com poucas linhas de código, a versão comando `\parbox` é mais prático de ser usado. Mas `\parbox` é mais simplificado do que `minipage`, podendo perder alguns recursos. Por exemplo, alguns comandos e ambientes tais como `\verb` e `verbatim` não funcionam dentro do `\parbox`. Assim, prefira usar o `minipage` sobre `\parbox`. Veja o Exemplo 4.21.

Exemplo 4.21: ex04-parbox.tex

```
Texto normal
\fbox{\parbox{3cm}{texto em várias linhas. \\ Com moldura.}}
Texto continua.
```



O primeiro parâmetro do `\parbox` é a largura, como no caso de `minipage`. Tanto no `minipage` como o `parbox`, `\linewidth` dentro dele é a largura da caixa passada como parâmetro.

## 4.8 Colunas múltiplas de texto

Dependendo do documento, usa-se mais de uma coluna. Isto é o caso de alguns artigos que podem usar duas colunas, ou `poster` que costuma usar três colunas ou mais. Mesmo no documento de uma coluna, o índice remissivo costuma ficar em duas colunas. Apesar da maioria das classes de documentos tem implementado a opção de documento em duas colunas (opção `twocolumn`) e comandos para alternar entre uma e duas colunas (comandos `\onecolumn` e `\twocolumn`), a forma mais prático é usar o pacote `multicol` que implementa o ambiente `multicols` que permite criar qualquer número de colunas. O padrão é não traçar linhas separando colunas, mas isto pode ser alterado, como no Exemplo 4.22. Aqui foi usado o `\setlength` para alterar a medida da espessura de linha que separa as colunas. Medidas serão tratadas em Seção 12.2 do Capítulo 12. O Exemplo 4.22 ilustra o uso de múltiplas colunas.

Exemplo 4.22: ex04-multicols.tex

```

\documentclass[a4paper,12pt]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\usepackage{color}
\usepackage{color}
\usepackage{multicol} % permite usar multiplas linhas
% espessura da linha que separa as colunas (0pt para desabilitar)
\setlength{\columnseprule}{1pt}
\begin{document}

```

Este parágrafo está como uma única coluna, padrão para maioria das classes de documentos.

```
\begin{multicols}{2} % inicia duas colunas
```

Este parágrafo está em duas colunas.

O ambiente `\texttt{multicols}` balanceia os conteúdos em colunas para que todas colunas tenham mesmo tamanho.

Se preferir que passe para próxima coluna somente quando a coluna está cheia, use a versão “\*” que não balanceia seus conteúdos.

Se quer finalizar a coluna manualmente, poderá usar o `\verb+\columnbreak+` que finaliza a coluna atual e passa para próxima coluna.

```
\end{multicols} % finaliza o modo de duas colunas
```

Agora está no modo de uma única coluna novamente.

```
\end{document}
```

Este parágrafo está como uma única coluna, padrão para maioria das classes de documentos.

Este parágrafo está em duas colunas. O ambiente <code>multicols</code> balanceia os conteúdos em colunas para que todas colunas tenham mesmo tamanho. Se preferir que passe para próxima coluna somente quando a coluna	está cheia, use a versão “*” que não balanceia seus conteúdos. Se quer finalizar a coluna manualmente, poderá usar o <code>\columnbreak</code> que finaliza a coluna atual e passa para próxima coluna.
---	---

Agora está no modo de uma única coluna novamente.



---

## Capítulo 5

# Aprofundando nas Fórmulas Matemáticas

Aqui, vamos aprofundar mais nas fórmulas matemáticas. Para suporte à matemática, os pacotes `amssymb` e `amsmath` são usados. Portanto, coloque o comando `\usepackage{amssymb,amsmath}` no `preamble` do documento.

### 5.1 Usando algumas fontes matemáticas

Na matemática, além de símbolos especiais e letras gregas, também usam alfabetos romanos de formatos diferentes. Por exemplo, para o conjunto dos números, costuma usar “negrito do quadro negro” (por exemplo, letra de traço duplo do AMS). Tais fontes são selecionados pelos comandos apropriados. Aqui veremos alguns dos mais usados.

O conjunto dos números como real, racional, etc, usam a letra maiúscula em “negrito do quadro negro”. Para tanto, poderemos usar o comando `\mathbb`.

Para Conjunto de funções contínuas, costuma usar a letra “C” maiúscula na forma caligráfica. A fonte caligráfica é indicado pelo comando `\mathcal`.

Para algumas áreas de matemática, ainda usam a versão enfeitada de caracteres oferecido pelo comando `\mathfrak`.

Para nome das funções, devemos usar a letra romana reta. O  $\LaTeX$  dispõe de comandos prontos para maioria das funções comumente usadas, mas as vezes precisamos escrever o nome da função que não está pronto, como o `sen` que representa a função seno em português. Como  $\LaTeX$  implementa nome das funções em inglês, o nome disponível para `seno` é `sin` produzido pelo comando `\sin`. Também notemos que no Brasil costuma usar `tg` para `tangente` e não o `tan`.

Para estes e outros casos, podemos especificar os nomes das funções que  $\LaTeX$  não dispõe, colocando dentro do comando `\mathrm` que usa a fonte romana reta. Veja o Exemplo 5.1.

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\usepackage{amssymb,amsmath}
\begin{document}
Exemplo do uso de letra caligráfica e de “negrito do quadro negro”.
\[ f \in \mathcal{C}(\mathbb{R}, \mathbb{R}) \]

Exemplo da letra romana maiúscula enfeitada  $\mathfrak{R}$ .

Exemplo do nome das funções
\[ \forall \theta \in \mathbb{R}, \cos^2 \theta + \mathrm{sen}^2 \theta = 1 \]

Outro exemplo:  $\mathrm{tg} \theta = \frac{\mathrm{sen} \theta}{\cos \theta}$ .

\[
\mathop{\mathrm{argmin}}_x f(x) = \{x : f(x) = \min_{x'} f(x')\}
\]
\end{document}

```

Exemplo do uso de letra caligráfica e de “negrito do quadro negro”.

$$f \in \mathcal{C}(\mathbb{R}, \mathbb{R})$$

Exemplo da letra romana maiúscula enfeitada  $\mathfrak{R}$ .

Exemplo do nome das funções

$$\forall \theta \in \mathbb{R}, \cos^2 \theta + \mathrm{sen}^2 \theta = 1$$

Outro exemplo:  $\mathrm{tg} \theta = \frac{\mathrm{sen} \theta}{\cos \theta}$ .

$$\mathop{\mathrm{argmin}}_x f(x) = \{x : f(x) = \min_{x'} f(x')\}$$

Note no `argmin`, o  $x$  será colocado embaixo e não como índice. Para que o que foi colocado como índice ficar embaixo, usa-se o comando `\mathop`.

Em geral, costuma definir comando para nome das funções no `preamble` do documento para facilitar a digitação. A definição de comandos e ambientes serão estudados na Seção 6.1 do Capítulo 6.

Os comandos para nome das funções pré-definidas estão na Seção 10.4 do Capítulo 10.

## 5.2 Texto, função por partes e matrizes

Para inserir texto nas fórmulas, usa-se o comando `\text`. Evite de usar o `\mbox` para este propósito.

Para ilustrar, vamos usar o ambiente `cases` usado para definir funções por partes, no Exemplo 5.2.

Exemplo 5.2: ex05-mat-text.tex

```
\[
|x| = \begin{cases}
-x, & \text{se } x < 0 \\
x, & \text{caso contrário}
\end{cases}
\]
```

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

O caractere “&” indica o ponto de alinhamento e `\\` indica a mudança de linha.

O pacote `amsmath` dispõe de ambiente para produzir matrizes. O ambiente `matrix` produz matriz sem o delimitador, `pmatrix` é delimitado pelos parenteses, `bmatrix` é delimitado pelos colchetes, `Bmatrix` é delimitado pelas chaves, `vmatrix` é delimitado pelas retas verticais e `Vmatrix` é delimitado pelas retas verticais duplas. Eles estão ilustradas no Exemplo 5.3.

Exemplo 5.3: ex05-matriz.tex

```
\[
\begin{pmatrix}
1 & 2 & 3 \\
2 & 3 & 4 \\
3 & 4 & 5
\end{pmatrix},
\begin{bmatrix}
1 & 2 & 3 \\
2 & 3 & 4 \\
3 & 4 & 5
\end{bmatrix},
\begin{vmatrix}
1 & 2 & 3 \\
2 & 3 & 4 \\
3 & 4 & 5
\end{vmatrix},
\begin{Vmatrix}
1 & 2 & 3 \\
2 & 3 & 4 \\
3 & 4 & 5
\end{Vmatrix}
\]
```

```

2 & 3 & 4 \\
3 & 4 & 5
\end{Vmatrix}
\]

```

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{pmatrix}, \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{bmatrix}, \begin{vmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{vmatrix}, \begin{Vmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{Vmatrix}$$

### 5.3 Delimitadores auto ajustáveis, chaves embaixo e integrais

Existem delimitadores auto ajustáveis que aumenta conforme o seu conteúdo aumenta de altura. Para usar ele, use `\left` antes do delimitador esquerdo (“(”, “[”, “+”, “|”, “\|”, “\langle”, “\lfloor”, “\lceil”, “\ulcorner” e “.”) e `\right` antes do delimitador direito (“)”, “]”, “\|”, “|”, “\|”, “\rangle”, “\rfloor”, “\rceil”, “\urcorner” e “.”), onde “.” é usado para indicar delimitador vazio (quando tem somente em um dos lados, outro lado seria “.”). Cuidado para não usar `||` para barra vertical dupla. Veja o Exemplo 5.4.

Exemplo 5.4: ex05-delimitador.tex

```

\left\|\frac{\frac{1}{x}}{\sqrt[3]{x^2+1}}\right\|
\left\{x \in [0,1) : \frac{x+1}{x-1} < 0\right\}
2\left[1+\frac{\left(1+\frac{1}{5}\right)^2}{2}\right]+5
\left.\frac{x^2}{x+1}\right|_{x=1} = \frac{1}{2}
\left\langle f, g \right\rangle = \int_b^a f(x)g(x)dx

```

$$\left\|\frac{\frac{1}{x}}{\sqrt[3]{x^2+1}}\right\|$$

$$\left\{x \in [0,1) : \frac{x+1}{x-1} < 0\right\}$$

$$2\left[1+\frac{\left(1+\frac{1}{5}\right)^2}{2}\right]+5$$

$$\left.\frac{x^2}{x+1}\right|_{x=1} = \frac{1}{2}$$

$$\langle f, g \rangle = \int_b^a f(x)g(x)dx$$

Para casos em que delimitadores auto ajustáveis não funcionam como esperado, poderá indicar o tamanho manualmente, colocando a especificação do tamanho antes do delimitador. O especificador do delimitador grande na ordem crescente são: `\big`, `\Big`, `\bigg` e `\Bigg`. Veja Exemplo 5.5.

Exemplo 5.5: ex05-delimitador-grande.tex

```
\[ \bigg( \Big(x + (y - z) \Big) + w \bigg) \]
```

```
\[ \lambda \bigg( f(x+0) + \cdots f(x_n) \]
```

```
\[ + \sum_{i=0}^n x_i \bigg) \]
```

$$\left( \left( x + (y - z) \right) + w \right)$$

$$\lambda \left( f(x + 0) + \cdots f(x_n) \right)$$

$$+ \sum_{i=0}^n x_i$$

A “chave” sobre (ou abaixo) da fórmula é colocado pelo `\overbrace` e `\underbrace`. Veja Exemplo 5.6.

Exemplo 5.6: ex05-bracos.tex

```
\[ \overbrace{1+1+\cdots+1}^{n-\text{vezes}} = n \]
```

```
\[ \underbrace{k+k+\cdots+k}_n = nk \]
```

$$\overbrace{1 + 1 + \cdots + 1}^{n-\text{vezes}} = n$$

$$\underbrace{k + k + \cdots + k}_{n-\text{vezes}} = nk$$

Alguns casos de integrais múltiplas estão ilustrados no Exemplo 5.7.

Exemplo 5.7: ex05-integrais.tex

Integrais múltiplas

```
\[ \int_a^b f(x)dx \]; \iint_R f(x,y)dxdy \]; \iiint_B f(x,y,z)dxdydz \]
```

```
\[ \int \dots \int f(x_1, \dots, x_n) dx_1 \cdots dx_n \]
```

Integral de linha  $\oint f(x)dx$ .

Integrais múltiplas

$$\int_a^b f(x)dx \quad \iint_R f(x,y)dxdy \quad \iiint_B f(x,y,z)dxdydz$$

$$\int \cdots \int f(x_1, \dots, x_n)dx_1 \cdots dx_n$$

Integral de linha  $\oint f(x)dx$ .

Note que `cdots` produz três pontos no meio, enquanto que `\ldots` produz três pontos embaixo.

## 5.4 Quebrando fórmulas em várias linhas

Para quebrar uma equação em mais de uma linha, podemos usar o ambiente `split`.

Para criar alinhamento na equação quebrada, usa-se `aligned` em vez de `split` e insere o `&` nos pontos de alinhamento, como no Exemplo 5.8.

Exemplo 5.8: ex05-split.tex

```
\[
\begin{split}
2x+y=3 \\
x-y=1+a
\end{split}

\begin{equation}
\begin{aligned}
2x+y&=3 \\
x-y&=1+a
\end{aligned}
\end{equation}
```

$$\begin{aligned} 2x + y &= 3 \\ x - y &= 1 + a \end{aligned}$$

$$\begin{aligned} 2x + y &= 3 \\ x - y &= 1 + a \end{aligned} \tag{1}$$

Podemos tentar usar o ambiente `cases` para definir sistema de equações como no Exemplo 5.9, mas isto não efetua o alinhamento (na posição de igualdade). Não podemos usar o `&`,

pois o alinhamento seria efetuado de forma próprio para colocar condição da expressão e não a equação. Assim, é aconselhável usar o `aligned` em vez de `cases`.

Exemplo 5.9: ex05-cases-aligned.tex

```
\begin{equation}
  \begin{cases}
    2x+y=1 \\
    x-y=1+a
  \end{cases}
\end{equation}
\begin{equation}
  \left\{ \begin{aligned}
    2x+y&=1 \\
    x-y&=1+a
  \end{aligned} \right.
\end{equation}
```

$$\begin{cases} 2x + y = 1 \\ x - y = 1 + a \end{cases} \quad (2)$$

$$\left\{ \begin{array}{l} 2x + y = 1 \\ x - y = 1 + a \end{array} \right. \quad (3)$$

Existem diversos ambientes estilo `equations`. Vamos ver algumas delas:

O ambiente `gather` produz várias equações, todas centralizadas. Para inibir enumeração em algumas delas, coloque o `\nonumber` na equação que deseja remover a enumeração (antes da quebra de linhas). Veja o Exemplo 5.10

Exemplo 5.10: ex05-gather.tex

```
\begin{gather}
x+y+z=1 \\
x-y+z=2 \\
x+y=0
\end{gather}
\begin{gather}
x+y+z=1 \\
x-y+z=2 \nonumber \\
x+y=0
\end{gather}
```

$$x + y + z = 1 \tag{4}$$

$$x - y + z = 2 \tag{5}$$

$$x + y = 0 \tag{6}$$

$$x + y + z = 1 \tag{7}$$

$$x - y + z = 2 \tag{8}$$

$$x + y = 0 \tag{8}$$

Para equações em várias linhas, com pontos de alinhamento, usa-se o `align` que pode inibir enumeração de algumas equações como em `gather`.

Quando escreve uma expressão grande quebrado em várias linhas, as vezes é legal tabular cada linha para direita. Este efeito pode ser obtido pelo `multline`, como do Exemplo 5.11.

Exemplo 5.11: ex05-multiline.tex

```
\begin{align}
x+y+z&=1 \\
x-y+z&=2 \nonumber \\
x+y&=0
\end{align}
\begin{multline}
Ax+Ay+Az=
a_{11}x_1+\cdots+a_{1n}x_n+\cdots+\cdots+a_{n1}x_1+\cdots a_{nn}x_n
\\
a_{11}y_1+\cdots+a_{1n}y_n+\cdots+\cdots+a_{n1}y_1+\cdots a_{nn}y_n
\\
a_{11}z_1+\cdots+a_{1n}z_n+\cdots+\cdots+a_{n1}z_1+\cdots a_{nn}z_n
\end{multline}
```

$$x + y + z = 1 \tag{9}$$

$$x - y + z = 2 \tag{10}$$

$$x + y = 0 \tag{10}$$

$$\begin{aligned}
Ax + Ay + Az = & a_{11}x_1 + \cdots + a_{1n}x_n + \cdots + \cdots + a_{n1}x_1 + \cdots a_{nn}x_n \\
& a_{11}y_1 + \cdots + a_{1n}y_n + \cdots + \cdots + a_{n1}y_1 + \cdots a_{nn}y_n \\
& a_{11}z_1 + \cdots + a_{1n}z_n + \cdots + \cdots + a_{n1}z_1 + \cdots a_{nn}z_n \tag{11}
\end{aligned}$$

Para mais ambientes deste estilo, consulte o manual de AMS  $\LaTeX$ . [GMS04, Cap. 8] explica bem os ambientes e comandos matemáticos do AMS (que está sendo discutido parcialmente neste capítulo).

Note que, todos ambientes do estilo `equation` (`equation`, `gather`, `align`, `multline`, etc) apresentam a versão com `*` na qual remove toda enumeração (versão `*` equivale a colocar `\nonumber` em todas equações). Veja Exemmplo 5.12.

Exemplo 5.12: `ex05-align-star.tex`

```
\begin{align*}
x+y+z&=1 \\
x-y+z&=2 \\
x+y&=0
\end{align*}
```

$$\begin{aligned}x + y + z &= 1 \\x - y + z &= 2 \\x + y &= 0\end{aligned}$$

## 5.5 Nome sobre setas e delimitador empilhados

Para colocar nome nas setas, usa-se o `\stackrel`.

Para colocar limitante de soma, produto, etc em duas linhas, usa-se o `\substack`.

Para colocar embaixo, poderá usar o `\underset`.

Simplemmente empilhar um sobre outro, usa-se o `\atop`.

Veja Exemplo 5.13.

Exemplo 5.13: `ex05-stackrel.tex`

```
[ X \stackrel{f}{\to} Y ]
[ f : \underset{x}{X} \underset{\mapsto}{\to} \underset{f(x)}{Y} ]
[ L_i(x) = \prod_{\substack{j=0 \\ i \neq j}} \frac{x - x_j}{x_i - x_j} ]
[ X \atop Y ]
```

$$\begin{aligned}X &\stackrel{f}{\rightarrow} Y \\f : X &\underset{x}{\mapsto} \underset{f(x)}{Y} \\L_i(x) &= \prod_{\substack{j=0 \\ i \neq j}} \frac{x - x_j}{x_i - x_j} \\X & \\Y &\end{aligned}$$

Binomial de Newton pode ser produzido por `binom` como no Exemplo 5.14.

Exemplo 5.14: `ex05-binom.tex`, parte c

Regra de Pascal:

```
\[ \binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1} \]
```

Regra de Pascal:

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$$

## 5.6 Subequações

Podemos usar subequações no  $\text{\LaTeX}$  como no Exemplo 5.15.

Exemplo 5.15: `ex05-subequacao.tex`

```
\begin{subequations}
\begin{equation}\label{eq:regiao:a}
x^2+y^2 \le 1
\end{equation}
\begin{equation}\label{eq:regiao:b}
y \ge 0
\end{equation}
\end{subequations}
As equações \ref{eq:regiao:a} e \ref{eq:regiao:b} determinam a parte superior do disco.
```

$$x^2 + y^2 \leq 1 \tag{12a}$$

$$y \geq 0 \tag{12b}$$

As equações 12a e 12b determinam a parte superior do disco.

## 5.7 Acentuação no modo matemático

Na matemática, usam acentuação para produzir nomes relacionados ao original tais como  $\hat{f}$  e  $\bar{f}$  relacionados a  $f$ . Tais acentuações difere do modo texto. Alguns desses acentuações comumente encontrados na matemática são: acento circunflexo, til, ponto, seta e barra. Além das acentuações, tem também o caso de colorar expoentes como em  $f'$  e  $f^*$  que são relacionados com  $f$ . O  $\text{\LaTeX}$  produzem estes e outros acentos e símbolos usados como expoentes especiais na matemática.

comando	exemplo
<code>\hat</code>	$\hat{a}$
<code>\tilde</code>	$\tilde{a}$
<code>\dot</code>	$\dot{a}$
<code>\ddot</code>	$\ddot{a}$
<code>\ddd\dot</code>	$\dddot{a}$
<code>\dddd\dot</code>	$\dddd\dot{a}$
<code>\bar</code>	$\bar{a}$
<code>\vec</code>	$\vec{a}$
<code>\grave</code>	$\grave{a}$
<code>\acute</code>	$\acute{a}$
<code>\breve</code>	$\breve{a}$
<code>\check</code>	$\check{a}$
<code>\overset{\circ}{}</code>	$\overset{\circ}{a}$

Alguns deles, como barra, requer alongamento quando tiver mais de um símbolo (ou símbolo grande). Neste caso, deverá usar versão ajustáveis:

comando	exemplo
<code>\widehat</code>	$\widehat{abc}$
<code>\widetilde</code>	$\widetilde{abc}$
<code>\overline</code>	$\overline{ab}$
<code>\overrightarrow</code>	$\overrightarrow{AB}$
<code>\overleftarrow</code>	$\overleftarrow{ab}$
<code>\overleftrightarrow</code>	$\overleftrightarrow{AB}$

As vezes, precisamos colocar embaixo, como sublinhar.

comando	exemplo
<code>\underbar</code>	$\underbar{abc}$
<code>\underline</code>	$\underline{abc}$
<code>\underrightarrow</code>	$\underrightarrow{AB}$
<code>\underleftarrow</code>	$\underleftarrow{ab}$
<code>\underleftrightarrow</code>	$\underleftrightarrow{AB}$

Colocar a direita (ou como expoente)

comando	exemplo
<code>'</code>	$f'$
<code>\prime</code>	$f'$
<code>\backprime</code>	$f^\backprime$
<code>\ast</code>	$f^*$
<code>\circ</code>	$20^\circ$
<code>\llcorner</code>	$x^\llcorner$
<code>\ulcorner</code>	$x^\ulcorner$
<code>\perp</code>	$\vec{v}^\perp$

Não confundir apóstrofes (usado na derivada  $f'$  de  $f$ , por exemplo), com o acento agudo.

Note que  $\hat{i}$  e  $\hat{j}$  sem o ponto é produzido por `\imath` e `\jmath`, respectivamente como em  $\hat{i}$  e  $\hat{j}$ .

Veja o Exemplo 5.16.

Exemplo 5.16: ex05-acentos.tex

```
\[ \hat{f}, \tilde{f}, \dot{x}, \ddot{x}, \bar{y}, \overset{\circ}{X}, \vec{v} \]
\[ \widehat{ABC}, \overrightarrow{AB}, \]
\[ f', f^{\ast}, 20^{\circ}, \vec{v}^{\perp} \]
```

$$\hat{f}, \tilde{f}, \dot{x}, \ddot{x}, \bar{y}, \overset{\circ}{X}, \vec{v}$$

$$\widehat{ABC}, \overrightarrow{AB},$$

$$f', f^{\ast}, 20^{\circ}, \vec{v}^{\perp}$$

# Capítulo 6

## Definindo Comandos e Ambientes

Vamos ver o básico da definição de comandos e ambientes

### 6.1 Definindo comandos

Quando tem o comando longo a ser digitado, poderá definir um comando que funciona como atalho. Por exemplo, `\mathbb{R}` toda vez que queremos escrever o conjunto dos números reais é cansativo. Definir comando para `\mathbb{R}` permite também, trocar facilmente pelo outro comando para usar outras fontes no lugar de `\mathbb`.

Para definir um comando, usa-se o `\newcommand`. Por exemplo, `\newcommand{\Rset}{\mathbb{R}}` define o comando `\Rset` que será substituído pelo `\mathbb{R}` quando é compilado. Se quer que `\Rset` funcione tanto dentro como fora das fórmulas, poderá usar o `\ensuremath` como em `\newcommand{\Rset}{\ensuremath{\mathbb{R}}}`. Note que aqui foi evitado de definir como sendo `\real`, pois o comando `\natural` já está definido no L<sup>A</sup>T<sub>E</sub>X.

O nome do comando não deve ter números. Por exemplo, `\newcommand{\R2}{\mathbb{R}^2}` resulta em erros, pois o nome do comando `\R2` contém número.

Analogamente, `\newcommand{\sen}{\mathrm{sen}}` define o comando para função seno. Em geral, para definir nome da função, usa-se o `\DeclaremathOperator` do pacote `amsmath` tal como `\DeclaremathOperator{\sen}{sen}` em vez do `\newcommand`. Ele coloca automaticamente o `\mathrm` no segundo argumento.

O comando `\newcommand{\argmin}{\mathop{\mathrm{argmin}}}` define o comando `\argmin` para função `argmin`, mas por ter `\mathop`, o limitante será colocado embaixo em vez de ser colocado como índice. Para facilitar, o pacote `amsmath` define a versão com “\*” de `\DeclaremathOperator` que coloca o nome da função dentro do `\mathrm` e `\mathop`.

Quando já existe o comando, poderá redefinir usando o `\renewcommand`.

Se não sabe se existe o comando, poderá usar o `\providecommand` que define o comando caso não existir.

Estas definições de comandos costumam ficar no `preamble` do documento. Veja o Exemplo 6.1.

Exemplo 6.1: ex06-newcommand.tex

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\newcommand{\Rset}{\ensuremath{\mathbb{R}}}
% \DeclareMathOperator{sen}{sen} % modo amsmath
\newcommand{\sen}{\mathrm{sen}} % modo normal (sem amsmath)
% \DeclareMathOperator*{argmin}{argmin} % modo amsmath
\newcommand{\argmin}{\mathop{\mathrm{argmin}}} % modo sem amsmath
\begin{document}
\[ \forall \theta \in \Rset, \cos^2 \theta + \sen^2 \theta = 1 \]
\[ \argmin_x f(x) = \left\{ x : f(x) = \min_{x'} f(x') \right\} \]
\end{document}

```

$$\forall \theta \in \mathbb{R}, \cos^2 \theta + \sen^2 \theta = 1$$

$$\argmin_x f(x) = \left\{ x : f(x) = \min_{x'} f(x') \right\}$$

Os comandos podem ter até 9 parâmetros. Os comandos com parâmetros, é criado pelo comando `\newcommand{\nome-do-comando}[n]{definição}` onde `n` é o número de parâmetros. No corpo da definição do comando, `#1`, `#2`, `#3`, ...especificam os parâmetros na posição 1, 2, 3, etc.

Quando usa os parâmetros, o primeiro deles podem ser opcional. Neste caso, usa-se a sintaxe `\newcommand{\nome-do-comando}[n][v]{definição}` onde “v” é o valor padrão quando o parâmetro for omitido. O Exemplo 6.2 ilustra o caso.

Exemplo 6.2: ex06-parametro-opcional.tex

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\newcommand{\conj}[1]{\overline{\#1}} % com um parametro
\newcommand{\norma}[2][\left|\#2\right|_{\#1}] % com parametro opcional
\newcommand{\seq}[2][n]{\left\{\#2_0, \ldots, \#2_{\#1}\right\}}
\begin{document}
\[ \conj{z} \cdot z = \norma{z}^2 \neq \norma[\infty]{z}^2 \]
A sequência  $\seq{x}$  para  $n=5$  é  $\seq[5]{x}$ .

 $\{\conj{zw}\}$ 
\end{document}

```

$$\bar{z} \cdot z = \|z\|^2 \neq \|z\|_\infty^2$$

A sequência  $\{x_0, \dots, x_n\}$  para  $n = 5$  é  $\{x_0, \dots, x_5\}$ .  
 $\bar{z}w$

O comando `\let` cria uma cópia do comando existente. Por exemplo, `\let\comandocopia\comandoori` cria um comando `\comandocopia` que é uma cópia do `\comandooriginal`. Assim, podemos redefinir o `\comandooriginal` e se precisar do original, é só chamar o `\comandocopia`. Também poderá restaurar o comando original usando o `\let`. Por exemplo,

```
\let\tanoriginal\tan
\renwcommand{\tan}{\mathrm{tg}}
```

redefine o comando `\tan`, mas se precisar do `\tan` original, eh só chamar o `\tanoriginal`. Para restaurar o original, é só usar o `\let` novamente com em `\let\tan\tanoriginal`.

## 6.2 Criando ambientes

Para trecho maior de dados, o ambiente (o que tem `\begin` e `\end`) é mais adequado que os comandos. Para criar ambientes, usa-se o comando `\newenvironment` na qual o primeiro parâmetro é nome do ambiente, segundo é o que vai fazer antes e terceiro é o que vai fazer depois. Quando tem o parâmetro (que pode ter até nove), coloca o número de parâmetro como parâmetro opcional de `\newenvironment`, entre primeiro e segundo parâmetro. Assim como o comando, o ambiente também pode ter o primeiro parâmetro como opcional. Neste caso, passa o seu valor padrão após número de parâmetro como sendo parâmetro opcional ao `\newenvironment`.

No Exemplo 6.3, o ambiente `dem` foi definido somente como a ilustração. Em geral, usa o ambiente `proof` do pacote `amsthm` para demonstrações.

Exemplo 6.3: ex06-ambiente.tex

```
\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\newenvironment{myenv}{\begin{center}\em} {\end{center}}
\newenvironment{mypar}[1]{\par\noindent \hrulefill\fbbox{ #1 }\hrulefill \\\ } {\par\noindent \hrulefill \\\}
\newenvironment{dem}[1][Demonstração]{\textbf{#1:}\ } {\hfill\rule{1ex}{1ex}\par}
\begin{document}
\begin{myenv}
Texto enfatizado e centralizado.
\end{myenv}
\begin{mypar}{Teste}
```

Parágrafo com título simples.

```
\end{mypar}
\begin{dem}
\ldots
\end{dem}
\begin{dem}[Prova]
\ldots
\end{dem}
\end{document}
```

*Texto enfatizado e centralizado.*

Teste

Parágrafo com título simples.

**Demonstração:** ... ■

**Prova:** ... ■

Note que também pode definir os comandos no modo `TEX` em vez do modo `LATEX`. Para isso, basta usar o `\def` que é importante quando um comando define o outro. O `\def` que não vamos discutir aqui, deve ser usado com cuidado, pois ele não verifica se o comando existe ou não, redefinindo caso existir.

### 6.3 Quebrando o código em várias linhas

Em geral, a definição de comando ou ambiente deve ficar em uma única linha, mas quando isto é longo, precisamos dividir em mais de uma linha para boa organização. Neste caso, existe trechos que não podem ser quebrados, mas isto requer um conhecimento extra. Para contornar isto, note que o `LATEX` considera que a próxima linha é uma continuação da linha atual quando encontrar “%” grudado na última letra ativa (que não seja comentário). Assim, quando quebrar a definição de comandos ou ambiente em mais de uma linha, coloque o “%” grudado na última letra. Fique atento de não colocar espaço entre a última letra e “%”. Veja o Exemplo 6.4.

Exemplo 6.4: ex06-comando-multlinhas.tex

```
\newcommand{\norma}[2][1]{%
  \left|#2\right|_{#1}%
}
\newenvironment{mypar}[1]{%
  {\par\noindent \hrulefill\fbbox{ #1 }\hrulefill \\\}% o que faz antes
  {\par\noindent \hrulefill \\\}% o que faz depois
}\end{ltlisting}
%
```

# Capítulo 7

## Divisão Lógica de Documentos

Neste capítulo, vamos estudar a divisão lógica de documentos.

### 7.1 Capítulos, seções e similares

Nos livros e relatórios, os conteúdos são organizados em capítulos, indicado por `\chapter`. Os capítulos podem ser divididos em seções indicado pelo comando `\section`, se assim desejar. Neste caso, evite ter capítulos com seções e outro sem seções, para ter uniformidades.

Se a seção ficar grandes, podem subdividir em subseções com o comando `subsection`, mas tome cuidado para que todas seções tenham subseções.

Existem ainda, `\subsubsection` (subsubseções), `\paragraph` (parágrafos) e `\subparagraph` (subparágrafos) que são menos usados.

No caso de artigos, o conteúdo costuma ser dividido em seções e caso desejar, seções podem ser subdivididos em subseções (evitando que tenha seção com subseção e outro sem subseção).

O capítulo inicia-se com o comando `\chapter{Título}` e seções com o comando `\section{Título}`.

Estes comandos aceitam a opção de especificar os “títulos curtos” que são usados no sumário e cabeçalho, o que podem ser passados como parâmetro opcional colocado antes do título. Neste formato, usado quando título é longo, tem a forma `\chapter[Título Curto]{Título}` e `\section[Título Curto]{Título}`

Exemplo 7.1: ex07-capitulo.tex

```
\chapter[Titulo Curto]{Título Longo do Capítulo}
Apresentação bem rápida do capítulo.
\section[Título curto]{Título longo da seção}
Texto da seção
\section{Título da outra seção}
Texto da outra seção
\ldots
```

# Capítulo 1

## Título Longo do Capítulo

Apresentação bem rápida do capítulo.

### 1.1 Título longo da seção

Texto da seção

### 1.2 Título da outra seção

Texto da outra seção ...

Quando há os capítulos e seções, o sumário podem ser produzidos automaticamente pelo comando `\tableofcontents`.

Para comandos do tipo capítulos (`\chapter`, `\section`, etc), existem as versões com “\*” que não serão enumeradas e não serão colocadas no sumário, apesar de ter mesma formatação. Por exemplo, `\chapter*{Resumo}` não será enumerado, nem vai no sumário.

Note que, no caso de artigos, o resumo tem formatação diferente do restante das seções. Assim, existe o ambiente especial `abstract` para o resumo, o que não é mesmo que `\section*`.

Note a diferença e similaridade do Exemplo 7.2 que usa a versão “\*”, com o Exemplo 7.1 que usa a versão normal.

Exemplo 7.2: `ex07-capitulo-star.tex`

```
\chapter*{Resumo}
Resumo aqui.
```

# Resumo

Resumo aqui.

## 7.2 Capa, conteúdo frontal e principal

O Exemplo 7.3 ilustra a estrutura básica de documentos tipo livro na qual não vamos colocar a saída aqui, por ocupar várias páginas.

Exemplo 7.3: ex07-matter.tex

```

\documentclass[12pt,a4paper]{book}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\pagestyle{empty}
\begin{document}
% capa
\frontmatter
\chapter*{Resumo}\thispagestyle{empty}
Resumo aqui.
\pagestyle{headings}
\tableofcontents % Sumário
\chapter{Prefácio}
Apresentação do trabalho.
\mainmatter
\chapter{Título do Capítulo Aqui}
Apresentação bem rápida do capítulo.
\section{Títuo da seção aqui}
Texto da seção
\ldots
\appendix % se existir ãapndice
\chapter{Título do Apêndice 1} % se existir
Texto do apêndice 1
\ldots
\backmatter % opcional
% referencia bibliografica
% indice remissivo, se existir

```

```
\end{document}
```

O comando `\pagestyle{empty}` antes do `\begin{document}` remove as enumerações das páginas para que parte inicial do livro fiquem sem a enumeração.

A capa simples nos livros e relatórios podem ser produzidos pelo comando `\maketitle`, desde que seja fornecido algumas informações tais como título, autor e data pelos comandos `\title`, `\author` e `\date`, colocados antes do `\maketitle`. Veja Exemplo 7.4.

Exemplo 7.4: ex07-maketitle.tex

```
\documentclass[12pt,a4paper]{book}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\title{Exemplo}
\author{Sadao Massago}
\date{Fevereiro, 2018}
\begin{document}
\maketitle
\frontmatter
\ldots
\end{document}
```

Note que, `\date`, quando não especificado, será assumido como sendo `\today` que é data de compilação do documento. Se não quer que apareça data, deve colocar `\date{}`.

Para criar títulos personalizados nos livros e relatórios, use o ambiente `titlepage`. Note que nos livros e relatórios, a capa não será contada como páginas, mas todas outras páginas, mesmo totalmente em branco, serão contadas.

O comando `\frontmatter` disponível somente para livros, indica que será matéria frontal, tendo as páginas enumeradas em romano minúsculo e sem a contagem de capítulos (mesmo para versão sem “\*”).

O comando `\thispagestyle{empty}` no `\chapter*{Resumo}` remove a enumeração da página atual (primeira página do “Resumo”). Note que capítulos usa o estilo de página `plain` que enumera embaixo da página, mesmo que o estilo de página esteja em `empty`. Assim, colocar `\thispagestyle{empty}` é necessário para remover paginação desta primeira página do “capítulo”.

Antes do sumário colocado pelo comando `\tableofcontents`, foi colocado `\pagestyle{headings}` para que volte a colocar páginas e cabeçalho superior. Note que, no livro, o que vem antes do sumário não devem ser paginadas, mas o que vem depois do sumário, costuma ser paginados. Por ser `frontmatter`, a paginação será em romano minúsculo.

Depois do prefácio, tem o comando `\mainmatter` que também é disponível somente para livros. Este comando reinicia a paginação (começa a contar de 1 novamente) e usa a enumeração em arábico. Também faz começar a contar capítulos.

O comando `\appendix` indica que o que segue é apêndice que complementa o documento.

Ele é disponível para livros, relatórios e artigos. No caso de livros e relatórios, o comando `\chapter` começará a produzir título para apêndices em vez de capítulos (escreverá Apêndice A, Apêndice B, etc em vez de Capítulo 1, Capítulo 2, etc). No caso de artigos, o comando `\section` que produzirá os apêndices.

O comando `\backmatter` também é disponível somente para livros. O `\backmatter` faz com que `\chapter` funcione igual a `\chapter*` (versão “\*”). Como não é costume colocar `\chapter` no `backmatter`, o comando pode ser omitido.

Em geral, a parte final do documento é a referência bibliográfica. Se existir o índice remissivo, deverá ser colocado depois da referências bibliográficas.

Note que o relatório é um documento e não é livro. Assim, todas páginas exceto a capa, devem estar enumeradas em arábico e em sequencias. Assim, não há divisão de `frontmatter` e `mainmatter`, não existindo comandos correspondentes.

A configuração do documento no artigo é ilustrado no Exemplo 7.5.

Exemplo 7.5: ex07-article.tex

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\title{Exemplo}
\author{Sadao Massago}
\date{Fevereiro, 2018}
\begin{document}
\maketitle
\begin{abstract}
Resumo aqui.
\end{abstract}
% \tableofcontents % so se for artigo longo
\section{Título da seção 1}
Texto da seção 1
\section{Título da seção 2}
Texto da seção 2
\ldots
\appendix % se tiver apêndice
\section{Título do Apêndice 1}
Texto do apêndice 1
\ldots
% referência bibliográfica
\end{document}

```

# Exemplo

Sadao Massago

Fevereiro, 2018

## Resumo

Resumo aqui.

## 1 Título da seção 1

Texto da seção 1

## 2 Título da seção 2

Texto da seção 2 ...

## A Título do Apêndice 1

Texto do apêndice 1 ...

O `\maketitle` produz título do artigo com informações usando título, autor e data fornecidos como nas capas dos livros e relatórios.

Note que não costumamos usar o índice remissivo para artigos, apesar de ter comandos disponíveis para isso.

## 7.3 Limpando o verso das páginas

O livro costuma ser diagramado para impressão em dois lados (opção da classe de documento `twoside`), embora poderá optar em `oneside` que é um lado, se quiser. Também podemos usar `openright` que começa o capítulo somente no lado direito (`openany` começa o capítulo tanto no lado esquerdo como no lado direito).

Nestes casos, o verso do sumário, resumo, etc, assim como versos do final dos capítulos, se existir, ficará em branco em vez de ter enumeração de páginas ou cabeçalhos. Para que isso aconteça, é só inserir o código

```
\clearpage\thispagestyle{empty}\cleardoublepage
```

antes de cada capítulo ou comandos que produzem os capítulos ou similares, tais como sumário,

referências bibliográficas e índice remissivo. Também é necessário antes do `\mainmatter`. Veja o Exemplo 7.6 na qual a saída foi omitida.

Exemplo 7.6: ex07-clearpage.tex

```

\documentclass[12pt,a4paper,openright]{book}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\pagestyle{empty}
\begin{document}
% capa
\frontmatter
\chapter*{Resumo}\thispagestyle{empty}
Resumo aqui.
\cleardoublepage
\pagestyle{headings}
\tableofcontents % Sumário
\clearpage\thispagestyle{empty}\cleardoublepage
\chapter{Prefácio}
Apresentação do trabalho.
\clearpage\thispagestyle{empty}\cleardoublepage
\mainmatter
\chapter{Título do Capítulo 1}
Texto do primeiro capítulo.
\clearpage\thispagestyle{empty}\cleardoublepage
\chapter{Título do Capítulo 2}
Texto do segundo capítulo.
\ldots
\clearpage\thispagestyle{empty}\cleardoublepage
\appendix % se existir ãepndice
\chapter{Título do Apêndice 1} % se existir
Texto do apêndice 1
\ldots
\backmatter % opcional: sem efeito visual
\clearpage\thispagestyle{empty}\cleardoublepage
% referencia biliografica
\clearpage\thispagestyle{empty}\cleardoublepage
% indice remissivo, se existir
\end{document}

```

O comando `\clearpage` finaliza a página. Por exemplo, se tiver algo pendente como figuras a serem colocadas, será feito. Com `\thispagestyle{empty}`, desabilita o estilo da página atual e com `\cleardoublepage`, inserir página em branco, se necessário.

## 7.4 Efetuando pequenos ajustes

O primeiro parágrafo da seção ou capítulo não será indentado (empurrado para direita). A indentação é feita para distinguir um parágrafo do outro, mas como não tem parágrafo anterior, não há necessidades de indentação. No entanto, algumas pessoas podem querer indentar o primeiro parágrafo. Se for o caso, basta usar o pacote `indentfirst` no `preamble` do documento.

Por outro lado, se quer remover indentação de algum parágrafo como o que segue uma fórmula matemática no modo `displaystyle`, basta usar o comando `\noindent`.

Também pode querer que uma página fique um pouco maior para acomodar o conteúdo atual. Neste caso, use o comando `\enlargethispage`. Por exemplo, `\enlargethispage{\baselineskip}` aumenta a página atual por uma linha. Este tipo de ajustes requer cuidados para não comprometer a qualidade da diagramação.

Se quer que algum comando seja executado após o término da página atual, use o pacote `afterpage` que implementa o comando de mesmo nome. Por exemplo, `\afterpage{\clearpage}` colocará todas figuras e tabelas pendentes na próxima página.

# Capítulo 8

## Teoremas e Similares

Os ambientes para teoremas e similares podem ser criados de forma apropriada, facilmente no  $\text{\LaTeX}$ .

### 8.1 Criando ambiente para teoremas

Para definir o ambiente para escrever teoremas e similares, costuma usar o pacote `amsthm` que melhora a usabilidade do `\newtheorem` usados para este propósito.

O comando `\newtheorem{theorem}{Teorema}[chapter]` cria um ambiente chamado `theorem` que usa como título “Teorema” com contador vinculado a capítulo (`chapter`). Se for artigo, troque o `chapter` pelo `section` como em `\newtheorem{theorem}{Teorema}[section]`.

Para definir outros ambiente similares aos teoremas, tais como axioma, lema, corolário e proposição, poderá usar o comando tal como `\newtheorem{proposition}[theorem]{Proposição}` onde `theorem` no parâmetro opcional na segunda posição indica que será contado junto com o teorema. Assim ficará Teorema 1.1, Proposição 1.2, etc. O modelo para proposição serve para demais ambientes similares aos teoremas.

Para ambientes como definição, exemplos e exercícios, devemos usar o estilo definição, obtido pelo comando `\theoremstyle{definition}`.

A seguir, usa-se o `\newtheorem` novamente como `\newtheorem{definition}[theorem]{Definição}` e `\newtheorem{example}[theorem]{Exemplo}`.

Para observações e notas, deve mudar para estilo `remark` com o comando `\theoremstyle{remark}`.

Note que `amsthm` providencia o ambiente para demonstrações, denominado de `proof`. Veja o Exemplo 8.1.

Exemplo 8.1: `ex08-teorema.tex`

```
\documentclass[12pt,a4paper]{book}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\usepackage{amssymb,amsmath}
\usepackage{amsthm} % para configurar o teorema
```

```

% Definindo teoremas e similares. Contador unico, vinculado a capítulos.
\newtheorem{theorem}{Teorema}[chapter] % contador vinculado a capitulos
\newtheorem{corollary}[theorem]{Corolário}
\newtheorem{lemma}[theorem]{Lema}
\newtheorem{proposition}[theorem]{Proposição}
\newtheorem{axiom}[theorem]{Axioma}
\theoremstyle{definition}
\newtheorem{definition}[theorem]{Definição}
\newtheorem{example}[theorem]{Exemplo}
\newtheorem{exercise}{Exercício}[chapter] % contador próprio, vinculado a capítulo
\theoremstyle{remark}
\newtheorem{remark}[theorem]{Observação}
\begin{document}
\chapter{Triângulo Equilátero}
\begin{definition}
  Um triângulo é dito \emph{triângulo equilátero} quando todos os lados forem congruentes
  .
\end{definition}
\begin{theorem}
  Todo triângulo equilátero se, e somente se, todos ângulos forem congruentes.
\end{theorem}
\begin{proof}
\ldots
\end{proof}
\begin{remark}
  Um triângulo cuja todos ângulos são congruentes é dito \emph{equiângulo}.
\end{remark}
\end{document}

```

# Capítulo 1

## Triângulo Equilátero

**Definição 1.1.** Um triângulo é dito *triângulo equilátero* quando todos os lados forem congruentes.

**Teorema 1.2.** *Todo triângulo equilátero se, e somente se, todos ângulos forem congruentes.*

*Demonstração.* ... □

*Observação 1.3.* Um triângulo cuja todos ângulos são congruentes é dito *equiângulo*.

## 8.2 Parâmetros opcionais

Note que os ambientes tipo teoremas permite colocar o título como parâmetro opcional do ambiente. Isto também vale para o ambiente `proof` quando existem textos entre o enunciado do teorema e a sua prova. O comando `\proofname` armazena o “nome” do ambiente `proof` que o valor atual é “Demonstração”. Quando a demonstração é finalizada pela equação no modo `displaystyle`, a marca do final de demonstração fica na linha de baixo. Para corrigir isso, existe o comando `\qedhere` que indica onde a marca de final de demonstração deve ser colocada. Veja o Exemplo 8.2.

Exemplo 8.2: ex08-teorema-parametro.tex

```
\begin{definition}
Um triângulo é dito \emph{triângulo retângulo} quando tem um ângulo reto. O lado oposto
ao ângulo reto é denominado de \emph{hipotenusa} e outros dois lados são
denominados de \emph{catetos}.
\end{definition}
\begin{theorem}[Pitágoras]\label{thm:pitagoras}
Sejam  $\Delta ABC$ , um triângulo retângulo onde  $a$  é hipotenusa. Então

$$a^2=b^2+c^2$$

\end{theorem}
\begin{remark}
A recíproca do Teorema~\ref{thm:pitagoras} também é verdadeira.
\end{remark}
```

```

\begin{proof}[\proofname{} do Teorema~\ref{thm:pitagoras}]
\ldots
Assim, temos que
\[ a^2 + b^2 = c^2 \qedhere \]
\end{proof}

```

**Definição 1.4.** Um triângulo é dito *triângulo retângulo* quando tem um ângulo reto. O lado oposto ao ângulo reto é denominado de *hipotenusa* e outros dois lados são denominados de *catetos*.

**Teorema 1.5** (Pitágoras). *Sejam  $\Delta ABC$ , um triângulo retângulo onde  $a$  é hipotenusa. Então*

$$a^2 = b^2 + c^2$$

*Observação 1.6.* A recíproca do Teorema 1.5 também é verdadeira.

*Demonstração do Teorema 1.5.* ...Assim, temos que

$$a^2 + b^2 = c^2$$

□

## Capítulo 9

# Figuras e Tabelas Flutuantes, Tabelas Longas e Imagem Externa

Tratar também de caixas gráficas.

### 9.1 Figuras flutuantes

Elemento flutuante é aquele que não precisa ser colocado obrigatoriamente na posição “digitada”, mas que será colocado na melhor posição possível em termos de apresentabilidade.

A figura flutuante é especificado pelo ambiente `figure` e a tabela flutuante é especificado pelo ambiente `table`, respectivamente. Veja o Exemplo 9.1.

Exemplo 9.1: ex09-figura.tex

```
\begin{figure}[hbp]
\center

\

Aqui se coloca a primeira figura

\

\caption[Primeira figura]{Primeira figura como elemento flutuante}
\label{fig:simp}
\end{figure}
```

Aqui se coloca a primeira figura

Figura 1: Primeira figura como elemento flutuante

O Exemplo 9.1 ilustra o uso do ambiente flutuante `figure`. O parâmetro opcional é uma sequência de caracteres, especificando a ordem que tentará colocar a figura.

**h** Onde foi digitado

**b** na parte inferior da página

**t** na parte superior da página

**p** página separada.

**!** ignorar a restrição de espaçamento.

No ambiente flutuante, o  $\text{\LaTeX}$  tentará as posições seguindo a lista de especificação dos parâmetros e inserirá na primeira posição que satisfizer a exigência. Caso nenhuma posição for conveniente, criará uma página separada especialmente para ele.

Durante a tentativa de inserção, existem restrições estéticas tal como considerar inadequado quando sobra muito pouco espaço para texto. Restrições como estes podem ser ignorados quando utilizar a opção “!”

O comando `\caption` produz enumeração e título da figura, podendo esta no começo ou no final do ambiente `figure`. Ele aceita o nome curto como argumento opcional para ser usado na lista de figuras, caso título da figura for longa.

Note que, o rótulo colocado pelo `\label` deverá ficar dentro ou depois de `\caption`, pois ele que é responsável pela enumeração das figuras.

Para colocar moldura, poderá usar o `\fbox`, mas ele não pode ter parágrafo como argumento. Neste caso, podemos usar a combinação com `minipage` como no Exemplo 9.2.

Exemplo 9.2: ex09-figura-fbox.tex

```
\begin{figure}[hbp!]
  \center
  \fbox{\begin{minipage}{0.5\linewidth}
    \center
    Aqui insere a segunda figura
    \caption{Segunda figura} \label{fig:frame}
  \end{minipage}
  } % fbox
\end{figure}
```

Aqui insere a segunda figura

Figura 2: Segunda figura

## 9.2 Tabelas flutuantes

A tabela flutuante é criada da forma similar, usando o ambiente `table`. Os argumentos opcionais para controlar a posição de inserção é mesmo da figura. Veja o Exemplo 9.3 que ilustra a tabela flutuante, contendo texto como elemento.

Exemplo 9.3: ex09-tabela-flutuante.tex

```
\begin{table}[hbp]
\center

\

Aqui se coloca a primeira tabela

\

\caption{Primeira tabela}
\label{tab:simp}
\end{table}
```

Aqui se coloca a primeira tabela

Tabela 1: Primeira tabela

Normalmente, o conteúdo da tabela é criado pelo ambiente apropriado como `tabular` ou similar. Alguns casos, podem ser criados como arquivo separado e inserido como no caso de figuras.

Note que, podemos criar moldura, incluindo o título da tabela através de `fbox` combinado com o `minipage`, como feito na figura.

No caso do Exemplo 9.4 ilustra o uso de `tabular` dentro do ambiente `table` para criar tabela flutuante.

Exemplo 9.4: ex09-tabela-tabular.tex

```
\begin{table}[hbp]
\center
\begin{tabular}{|p{3.5cm}|l|} \hline
cenouras (500g) & R\$0,50 \\ \hline
cogumelos (vidro de 500g) & R\$5,00 \\ \hline
batata (1Kg) & R\$1,20 \\ \hline
total & R\$7,20 \\ \hline
\end{tabular}
\caption{Usando tabular} \label{tab:tab}
```

```
\end{table}
```

cenouras (500g)	R\$0,50
cogumelos (vidro de 500g)	R\$5,00
batata (1Kg)	R\$1,20
total	R\$7,20

Tabela 2: Usando tabular

Note que, o comando `supressfloats` podem ser usados para impedir que mais `floats` sejam inseridas na página específica, mas os “floats” com opção “!” não respeitam este comando.

Para que “floats” não processados (que ainda não foram colocados) sejam postos antes de mudar a página, basta usar `\clearpage` ou `\cleardoublepage` usado para limpar configurações de páginas (chamado pelo `\chapter`, por exemplo).

Note que, lista de figuras e de tabelas são geradas pelos comandos `\listoffigures` e `\listoftables` que costumam ser colocados depois do `\tableofcontents` (sumário).

### 9.3 Tabelas longas

Tabelas longas é ideal para tabelas pequenas. Mas as vezes precisamos de tabelas grandes que podem ocupar mais de uma página. Para isso, existe o ambiente `longtable` implementado no pacote do mesmo nome. A tabela longa é contado junto com as tabelas flutuantes e não devem ser colocados dentro do ambiente `table`. Ele também `\caption*` (a versão “\*”) que coloca o título, mas não acrescenta na lista de tabelas. Os parâmetros básicos são mesmo do ambiente `tabular`.

Veja o Exemplo 9.5. Para que este exemplo funcione, deverá ter `\usepackage{longtable}` no `preamble` do documento.

Exemplo 9.5: ex09-longtable.tex

```
\begin{longtable}{|c|r|}
\caption{Tabela Longa} \label{ltab:teste}
\\ % é necessário pular linha após definições preliminares: caption, label, etc.
\hline
\textbf{centrada} & \textbf{para direita} \\ \hline \hline
coluna 1 & coluna 2 \\ \hline
coluna 1 & coluna 2 \\ \hline
$\vdots$ & $\vdots$ \\ \hline
coluna 1 & coluna 2 \\ \hline
\end{longtable}
```

Tabela 3: Tabela Longa

<b>centrada</b>	<b>para direita</b>
coluna 1	coluna 2
coluna 1	coluna 2
⋮	⋮
coluna 1	coluna 2

Note que, ao iniciar o ambiente, coloca as configurações tais como títulos, o que fazer antes de mudar a página, etc. Depois pula a linha com “\” e resto segue normalmente. Como `longtable` usa o arquivo auxiliar para armazenar sua largura, pode precisar compilar duas vezes para ter o resultado desejado (assim como acontece com referencias cruzadas).

Para efetuar as configurações tais como o que fazer quando muda a página, etc, veja o manual correspondente.

## 9.4 Imagem externa

As figuras podem ser elaboradas usando o próprio  $\text{\LaTeX}$ , mas muitas vezes criamos usando um programa externo. As imagens externas para serem inseridas no documento  $\text{\LaTeX}$ , deverá estar no formato `pdf`, `jpg/jpeg` ou `png`. O formato `pdf` costuma ser usado para ilustrações científicas por ser pequeno e é de alta qualidade (exceto quando convertido de algum formato `bitmap`). Se o programa permite gerar `pdf`, prefira usar este formato. O formato `jpg/jpeg` é apropriado para fotos, mas não suporta transparências. O formato `png` suporta transparências.

O comando para incluir gráficos externos é `\includegraphics` implementado no pacote `graphicx` (não confundir com o pacote antigo `graphics` que tem menos recursos). Para exemplo desta seção, assumimos que tenha `\usepackage{graphicx}` no `preamble` do documento. Veja Exemplo 9.6.

Exemplo 9.6: ex09-imagem.tex

```
\begin{figure}[hbp]
\center
\includegraphics[width=0.5\linewidth]{latex-via-exemplos-fig}
\caption{Imagem PDF}
\end{figure}
```

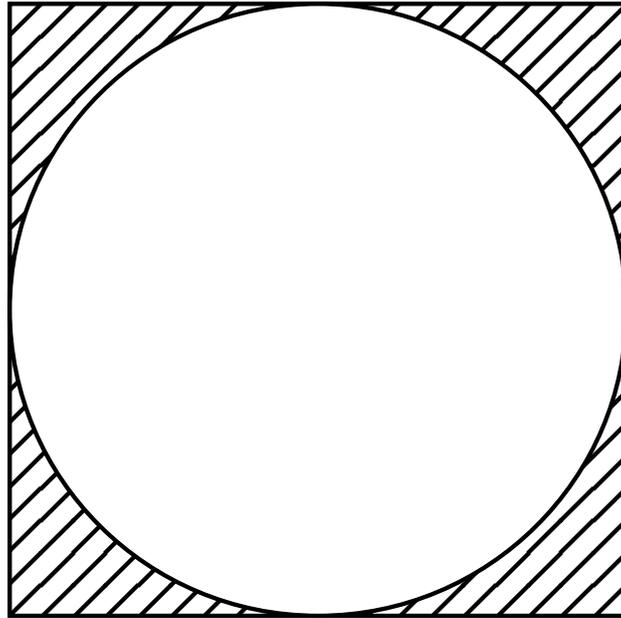


Figura 3: Imagem PDF

Note que não foi colocado a extensão (`.pdf`) no nome do arquivo. Em geral omitimos a extensão do arquivo para poder trocar de formatos, se desejar.

No parâmetro opcional, foi colocado `width=0.5\linewidth` para especificar que largura é metade da largura da linha atual. Os parâmetros opcionais controlam o tamanho e rotação das imagens.

**width** largura da imagem

**height** altura da imagem

**totalheight** altura total (quando é rotacionado. `height` só mede da linha de base para cima).

**scale** ampliação

**angle** rotação em graus

**origin** centro de rotação especificado pela combinação de “l” (left), “r” (right), “t” (top), “b” (bottom) e “c” (center).

**keepaspectratio** manter proporção quando `height` e `width` for especificado simultaneamente (usado sem o valor).

No caso de incluir o arquivo PDF com mais de uma página, poderá usar a opção `page` para indicar a página a ser incluída.

O Exemplo 9.7 ilustra a rotação por  $30^\circ$  em torno do centro.

Exemplo 9.7: ex09-imagem-rotacao.tex

```

\begin{figure}[hbp]
\center
\includegraphics[width=0.5\linewidth,angle=30,origin=c]{latex-via-exemplos-fig}
\caption{Imagem PDF}
\end{figure}

```

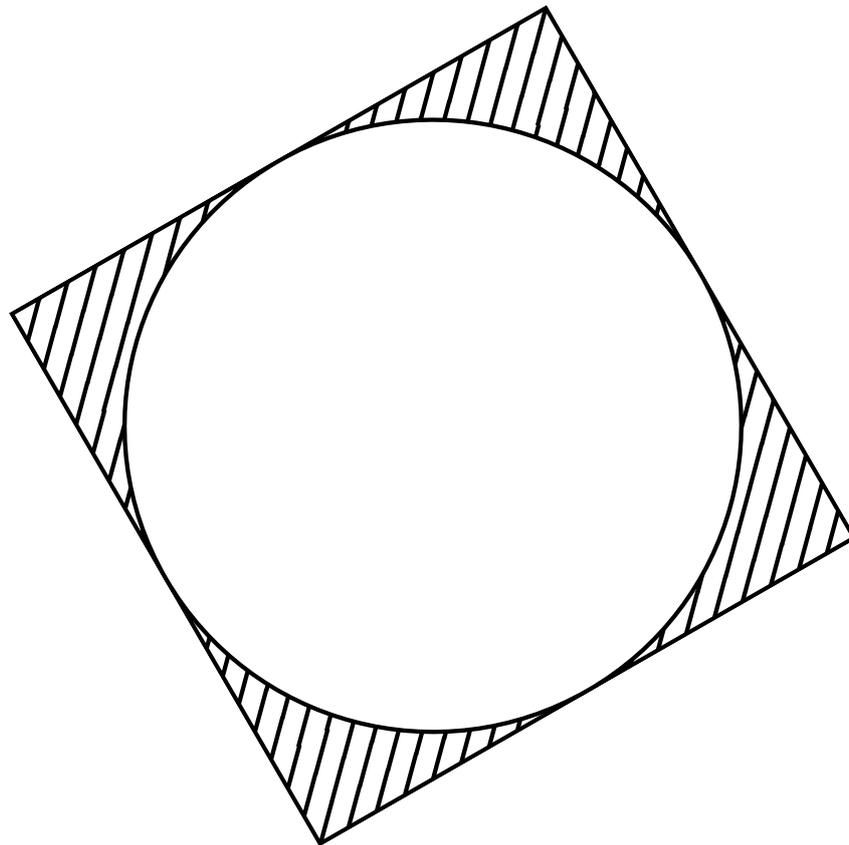


Figura 4: Imagem PDF

As vezes queremos colocar figuras lado a lado para comparação ou para economia de espaços. Para tanto, podemos usar o `minipage`, como no Exemplo 9.8.

Exemplo 9.8: ex09-imagem-lado.tex

```

\begin{figure}[hbp]
\center
\begin{minipage}{0.4\linewidth}
\center
\includegraphics[width=0.3\linewidth]{latex-via-exemplos-fig}
\caption{lado esquerdo} \label{fig:lado1}
\end{minipage}
\begin{minipage}{0.4\linewidth}
\center
\includegraphics[width=0.3\linewidth,angle=30,origin=c]{latex-via-exemplos-fig}

```

```
\caption{lado direito} \label{fig:lado2}
\end{minipage}
\end{figure}
```

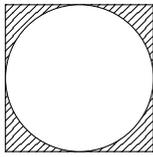


Figura 5: lado esquerdo

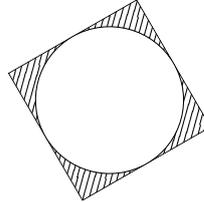


Figura 6: lado direito

## 9.5 Desenhando sobre a imagem externa

As vezes queremos desenhar sobre a imagem externa importada no documento tais como fórmulas ou mais alguns detalhes. Para isso, podemos usar o ambiente gráfico `picture`.

No Exemplo 9.9 ilustra a sobreposição para colocar fórmula sobre a imagem externa.

Note que a imagem foi inserida dentro do ambiente `picture`.

Observe como e onde foi usado o `\unitlength` para permitir ajuste de escala (mudando o valor de `\unitlength`) sem perder a posição de sobreposição já ajustada.

Exemplo 9.9: `ex09-imagem-sobreposicao.tex`

```
\begin{figure}[htbp!]
\center
%
\unitlength=0.45\linewidth % unidade
\begin{picture}(1.0,1.0) % caixa reservada
\put(0,0){\includegraphics[width=\unitlength]{latex-via-exemplos-fig}}
\put(0.35,0.45){\LARGE $e^{\pi i}+1=0$}
\end{picture}
\caption{Técnica de sobreposicao\label{fig:sobreposicao}}
\end{figure}
```

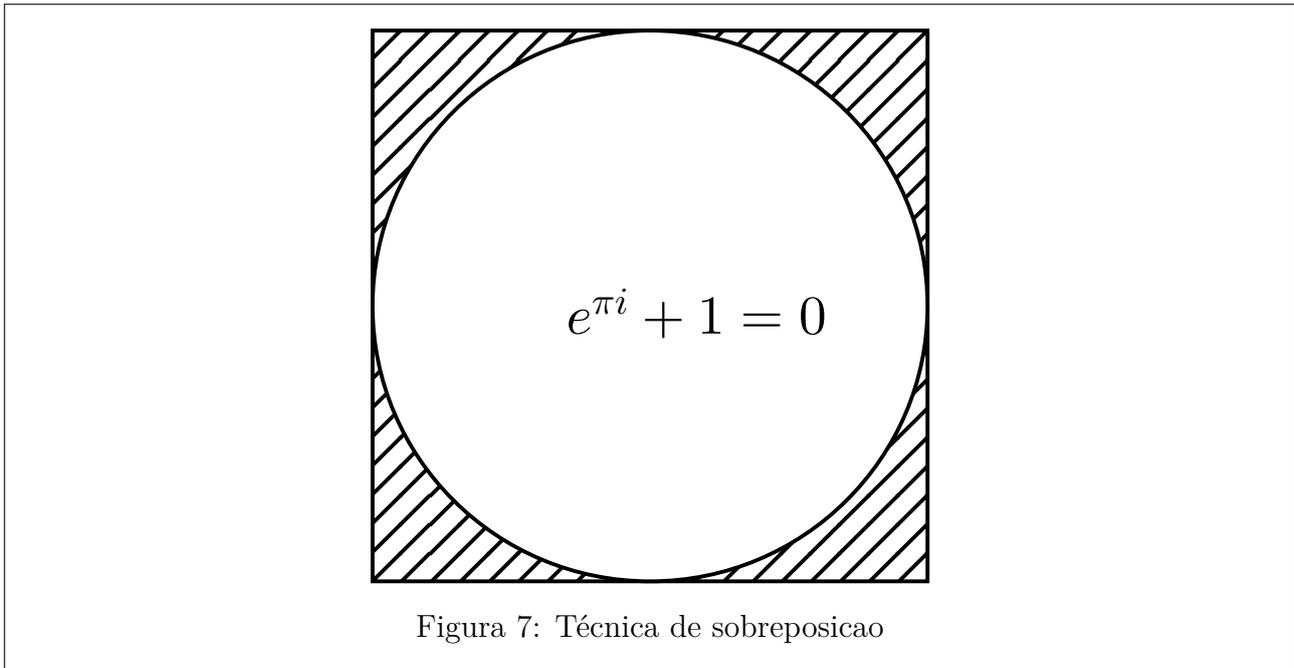


Figura 7: Técnica de sobreposicao

O ambiente `picture` é um ambiente gráfico padrão do  $\text{\LaTeX}$  que permite criar ilustrações. Note que, para criar ilustrações, existem vários pacotes gráficos apropriados, mas se quiser usar o ambiente `picture`, lembre-se de colocar o `\usepackage{pict2e}` no `preamble` do documento para eliminar limitações de alguns comandos gráficos.

O Exemplo 9.10 ilustra os comandos básicos do ambiente `picture` com o uso do pacote `pict2e`.

Exemplo 9.10: `ex09-picture.tex`

```

\begin{figure}[hbtpt!]
\center
\unitlength=1cm % medida da unidade
% \linethickness{1mm} % tome cuidado que e comando, nao a medida
\begin{picture}(5,4)(0,0)
% put coloca o objeto grafico na posição indicada
% line (segmento) e vector (vetor)
% Recebe vetor diretor e comprimento do segmento/vetor
\put(4,3.5){\line(0,1){0.5}} % segmento
\put(5,3){\vector(1,1){0.5}} % vetor
\put(2,2){\circle{1}} % circulo com centro e raio.
\put(1,1){\circle*{1.5}} % circle* é circulo solido
\put(3,2){\oval(3,2)} % oval
\put(3.5,2){\makebox(0,0)[cc]{ $A = \pi r^2$ }} % Caixa de texto (com formulas)
\put(3,0){\polyline(0,0)(1,0)(1,1)} % linhas poligonais
\put(5,0){\polygon(0,0)(1,0)(1,1)} % poligonos
\put(6.5,0){\polygon*(0,0)(1,0)(1,1)} % poligonos preenchidos
\thicklines % linha mais grossa
\put(4,1){\qbezier(0.0,0.0)(1.3,0.0)(2.0,2.8)} % curva bezier quadratica
\thinlines %linha normal (mais fina)
\put(6,1){\cbezier(0.0,0.0)(1.0,0.0)(1.0,1.0)(2.0,3.0)}

```

```
% curva bezier cubica
\end{picture}

\caption{Ilustração no formato \TeX{}}.\label{fig:tex}}
\end{figure}
```

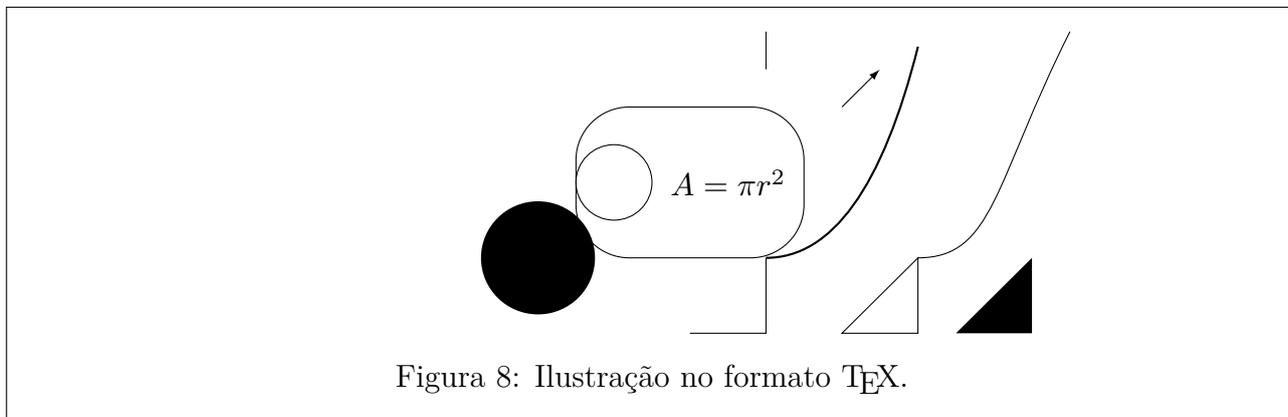


Figura 8: Ilustração no formato  $\TeX$ .

## 9.6 Caixas gráficas

O pacote `graphicx` dispõe de vários comandos relacionados com a mudança de tamanho e rotações. Eles são chamados de caixas gráficas. Os argumentos não devem conter quebra de parágrafos. Assim, se precisar, deverá usar juntamente com o `minipage`.

Para escalar, usa-se o `\scalebox`. Para indicar o tamanho fixo, usa-se o `\resizebox`. Veja o Exemplo 9.11.

Exemplo 9.11: `ex09-scale.tex`

Tamanho normal.

```
\scalebox{2.5}{Ampliado por 2.5 vezes}

\scalebox{2}{3}{Ampliado por 2x3 vezes} % 2 no horizontal e 3 em vertical (deforma)

\resizebox{10cm}{!}{Com 10 cm de comprimento}

\resizebox{!}{0.5cm}{Com 0.5 cm de altura}

\resizebox{5cm}{0.75cm}{Com 5cm$\times$0.7cm} % deforma
```

Tamanho normal.

Ampliado por 2.5 vezes

Ampliado por 2x3 vezes

Com 10 cm de comprimento

Com 0.5 cm de altura

Com 5cm×0.7cm

O “!” no `\resizebox` especifica que é calculado automaticamente com outra medida.

A rotação é feita pelo comando `\rotatebox`, onde parâmetro opcional `origin` permite configurar o centro de rotação, combinando “l” (left), “r” (right), “t” (top), “b” (bottom) e “c” (center). Veja o Exemplo 9.12.

Exemplo 9.12: ex09-rotate.tex

```
\rotatebox{30}{Rotacionado} por $30^\circ$.
```

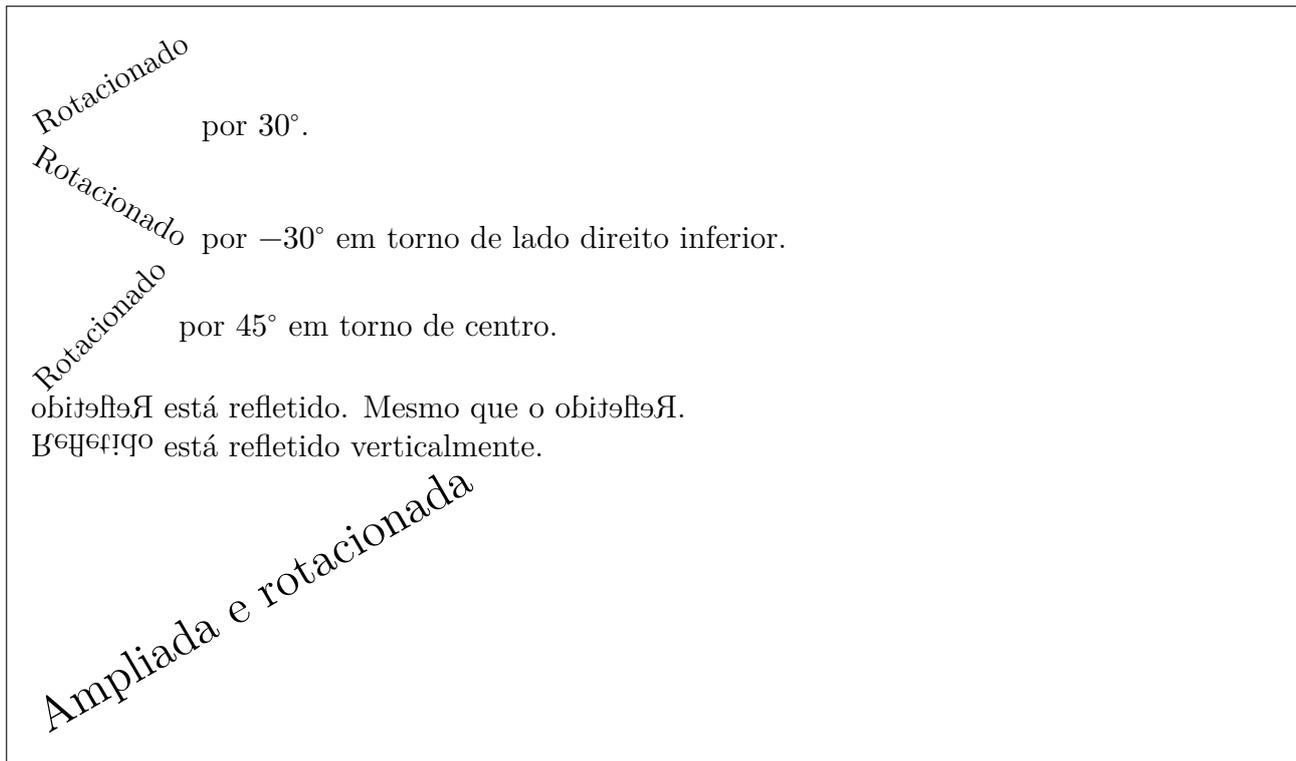
```
\rotatebox[origin=rb]{-30}{Rotacionado} por $-30^\circ$ em torno de lado direito inferior.
```

```
\rotatebox[origin=c]{45}{Rotacionado} por $45^\circ$ em torno de centro.
```

```
\reflectbox{Refletido} está refletido. Mesmo que o \scalebox{-1}[1]{Refletido}.
```

```
\raisebox{\depth}{\scalebox{1}[-1]{Refletido}} está refletido verticalmente.
```

```
\rotatebox[origin=c]{30}{\scalebox{1.5}{Ampliada e rotacionada}}
```



O `\reflectbox` reflete horizontalmente. Para refletir verticalmente, use o `\scalebox`, mas precisa ajustar a sua altura. Aqui foi usado o `\raisebox`, o que não vamos entrar em detalhes. Quem interessar, estude sobre caixas no L<sup>A</sup>T<sub>E</sub>X.

# Capítulo 10

## Ajuste das Fontes e de Espaçamento

Neste capítulo, veremos o controle das fontes e de espaçamento no  $\text{\LaTeX}$ .

### 10.1 Seleção da família de fontes

Num documento, usa-se vários tipos de fontes. Por exemplo, o corpo do documento nos livros e artigos costuma ser em romano (o que tem enfeite nas pontas, denominado de serifa). Os títulos dos artigos e livros podem ficar com fontes `sans serif` (sem serifa). O corpo de slide de apresentação ou poster costuma usar `sans serif` também. O código de programa, nome do arquivo, etc que deve ser lido no pé da letra em `typewriter` (mono espaçado).

Assim, nos documentos científicos, costumam usar três famílias de fontes básicos que são romano (com serifa), `sans serif` (sem serifa) e `typewriter` (mono espaçado) na qual devem estar em harmonia. O  $\text{\LaTeX}$  usa a fonte Computer Modern por padrão na qual tem essas três fontes que apresentam de forma harmoniosa. Note que podem trocar para outras fontes, se desejar, mas estas três fontes devem estar em harmonia.

A família de fontes são selecionadas pelo comando `\textrm` ou ambiente `rmfamily` para romano, o que é padrão para artigos e livros, pelo comando `\textsf` ou ambiente `sffamily` para `sans serif`, e pelo comando `\texttt` ou ambiente `ttfamily` para `typewriter`.

Quando aplica a mudança da fonte dentro de um ambiente, o ambiente de ajuste de fontes pode ser usado como comandos. Se não estiver dentro do ambiente e quer usar a versão comando do ambiente, poderá delimitar a sua atuação, delimitando pelos chaves. Veja Exemplo 10.1.

Exemplo 10.1: `ex10-family.tex`

```
\textsf{Fontes sem serifa}

\begin{sffamily}
Texto na fonte sem serifa.
\end{sffamily}

\textsf{Fontes mono espaçado}
```

```

\begin{ttfamily}
Texto na fonte mono espaçado.
\end{ttfamily}

\begin{center}
\sfamily
texto centralizado e em sans serif.
\end{center}

{\ttfamily Texto em nomo espaçado}

Texto normal.

```

```

Fontes sem serifa
Texto na fonte sem serifa.
Fontes mono espaçado
Texto na fonte mono espaçado.

                texto centralizado e em sans serif.

Texto em nomo espaçado
Texto normal.

```

## 10.2 Seleção de formas e peso das fontes

No  $\text{\LaTeX}$ , usa-se o comando para trocar forma de fontes, tais como negrito, itálico, etc. Usar comando adequado no  $\text{\LaTeX}$  é importante para automatizar o processo, não somente para formatação de textos. O texto enfatizado é produzido pelo comando `\emph` ou pelo ambiente `em`. O comando `\emph` altera entre fonte reto e itálico para destacar palavras tal como termo que está sendo definido.

O texto em negrito, por sua vez, é produzido pelo comando `\textbf` ou pelo ambiente `bfseries` para dar ênfase maior que destaca no meio do texto. Fonte não negrito que é padrão pode ser especificado pelo comando `\textmd` ou ambiente `mdseries`.

Ainda existem outras formas tal como comando `\textsc` e ambiente `scshape` para `small caps`, comandos `\textit` e ambiente `itshape` para itálicos.

Note que o formato normal é `\textup` ou ambiente `upshape`. Veja Exemplo 10.2.

Exemplo 10.2: `ex10-series.tex`

```

\textbf{Texto em negrito}

\begin{bfseries}
Negrito como ambiente
\end{bfseries}

```

```

\begin{em}
Parágrafo enfatizado. \emph{Texto enfatizado dentro dele}
\end{em}

\textsc{Small Caps}

\begin{center}
\bfseries
Negrito e centralizado.
\textmd{Normal}
\end{center}

{\scshape Small Caps dentro dos chaves}

Texto normal

```

```

Texto em negrito
Negrito como ambiente
Parágrafo enfatizado. Texto enfatizado dentro dele
SMALL CAPS

                Negrito e centralizado. Normal

SMALL CAPS DENTRO DOS CHAVES
Texto normal

```

Note que os comandos antigos `\rm`, `\tt`, `\bf`, `\sf`, `\it` e `\sc` para ajuste de fontes não devem ser usados, pois não podem ser combinados (por exemplo, itálico negrito), além ter de problemas de ajustes de espaçamentos (como a necessidade de correção de itálico).

Note que, para restaurar a fonte padrão, basta usar o ambiente `normalfont` (que tem versão comando dele também).

## 10.3 Tamanho das fontes

Tamanho das fontes são especificados também pelos comandos e o  $\text{\LaTeX}$  efetua ajuste automático em relação ao tamanho padrão.

O tamanho de menor para maior são definidos pelo ambientes

`tiny`: `tiny`

`scriptsize`: `scriptsize`

`footnotesize`: `footnotesize`

`normalsize`: `normalsize`

`large`: `large`

`Large`: `Large`

LARGE: LARGE  
 huge: huge  
 Huge: Huge

Lembrando que existem comandos correspondentes a cada um dos ambientes, similar a outros ambientes de fontes. Veja Exemplo 10.3.

Exemplo 10.3: ex10-size.tex

```
\begin{Large}
Letra grande (2 escalas acima)
\end{Large}

\begin{em}
\small Parágrafo enfatizado com letra pequena (uma escala abaixo)
\end{em}

{\footnotesize Fonte 2 escalas abaixo}

Texto normal
```

Letra grande (2 escalas acima)  
*Parágrafo enfatizado com letra pequena (uma escala abaixo)*  
 Fonte 2 escalas abaixo  
 Texto normal

## 10.4 Ajuste de fontes no modo matemático

No  $\text{\LaTeX}$ , as especificações das fontes no texto e na fórmula matemática usam os comandos diferentes. A especificação da fonte no modo matemático corresponde ao  $\text{\texttt{\textbackslashtext??}}$  são  $\text{\texttt{\textbackslashmath??}}$ :  $\text{\texttt{\textbackslashmathrm}}$ ,  $\text{\texttt{\textbackslashmathsf}}$ ,  $\text{\texttt{\textbackslashmathhtt}}$ ,  $\text{\texttt{\textbackslashmathbf}}$ ,  $\text{\texttt{\textbackslashmathit}}$ ,  $\text{\texttt{\textbackslashmathnormal}}$ ,  $\text{\texttt{\textbackslashmathcal}}$ .

O comando  $\text{\texttt{\textbackslashmathcal}}$  que não tem correspondente no modo texto, usa a fonte caligráfica, mas somente em letras maiúsculas.

Os comandos acima não funcionam para símbolos e eles não podem ser combinados. Por exemplo,  $\text{\texttt{\textbackslashmathbf}\{\text{\texttt{\textbackslashmathrm}}\{A\}}}$  é mesmo que  $\text{\texttt{\textbackslashmathrm}}\{A\}$ .

Assim,  $\text{\texttt{\textbackslashmathbf}}$  não é apropriado para escrever fórmulas em negrito. Portanto, use  $\text{\texttt{\textbackslashbm}}$  do pacote **bm** para converter parte da fórmula e negrito. Note que, quando usa em conjunto com o **amsmath**, pode precisar colocar chaves duplas para prevenir erros em  $\text{\texttt{\textbackslashbm}}$  (veja Exemplo 10.4).

O pacote **amssymb** dispõe de mais dois fontes bastante usados na matemática que são  $\text{\texttt{\textbackslashmathfrak}}$  e  $\text{\texttt{\textbackslashmathbb}}$ . Em geral, usa-se os pacotes **amsmath** e **amssymb** quando produz textos

matemáticos. Então acrescente `\usepackage{amssymb,amsmath}` no `preamble` do documento com no Exemplo 10.4.

Exemplo 10.4: `ex10-fontes-mat.tex`

```
\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\usepackage{amssymb,amsmath}
\usepackage{bm}
\begin{document}
\[ \forall x \in \mathbb{R}, \mathbf{sen}^2(x) + \mathbf{cos}^2(x) = 1 \]
{\bfseries Para  $\mathbf{0!} = 1$ .}
\end{document}
```

$$\forall x \in \mathbb{R}, \mathbf{sen}^2(x) + \mathbf{cos}^2(x) = 1$$

**Para**  $i = 0$ : Por definição,  $0! = 1$ .

Note que nomes (ou abreviações) das funções devem estar em fonte romano reto. Assim, `sen` foi produzido, usando `mathrm`.

Para não precisar especificar `\mathrm{}`, existem comandos para nomes de maioria das funções matemáticas mais conhecidas.

Nome das funções pré-definidas:

`arccos`, `arcsin`, `arctan`, `arg`, `mod`, `cos`, `cosh`, `cot`, `coth`, `csc`, `deg`, `det`, `dim`, `exp`, `gcd`, `hom`, `inf`, `ker`, `lg`, `lim`, `lim inf`, `lim sup`, `ln`, `log`, `max`, `min`, `sec`, `sin`, `sinh`, `sup`, `tan`, `tanh`, `Pr`.

Note que o *seno* em inglês é *sine*, abreviado para `sin` que não coincide com o termo usado no Brasil que é `sen`. Neste caso, costuma definir o comando para não ficar digitando `mathrm` toda hora. De forma análoga, quando o comando produz nome não usual, como no caso de tangente que é `tan` no `LATEX`, mas costuma usar `tg` no Brasil, podem ser redefinidos. A definição e redefinição de comandos estão na Seção 6.1 do Capítulo 6.



# Capítulo 11

## Referências Bibliográficas e Índice Remissivo

Veremos como produzir índice remissivo e referências bibliográficas no  $\LaTeX$ .

### 11.1 Referências bibliográficas

Uma referência bibliográfica é uma lista de referências externas usadas no trabalho. Esta lista é definido pelo ambiente `thebibliography` cuja argumento é o elemento com maior largura, para ajustar alinhamento de seus itens. Cada item da referência é especificado pelo comando `\bibitem` cuja argumento obrigatório é uma chave e argumento opcional é rótulo a ser impresso no item (se for omitido, será usado números). Depois segue os dados da referência bibliográficas como devem ser impressas. Em geral, esta lista será colocado no final do documento (se existir o índice remissivo, será antes do índice remissivo). No Exemplo 11.1, o argumento `OPHS15` do `thebibliography` será usado para medir o espaço deixado à esquerda dos itens. Além disso, cada item tem o rótulo (argumento opcional) a ser impresso como nome. Depois vem a chave que será usado no documento para citar o item específico.

Exemplo 11.1: ex11-bib.tex

```
\begin{thebibliography}{OPHS15}
\bibitem[GMS04]{Goossens:2004}
Michel Goossens and Frank Mittelbach
\emph{The  $\LaTeX$  companion (second edition)}, Addison—Wesley, 2004.
\bibitem[Tea00]{IndianTUG:00}
Tutorial Team, \emph{Online tutorials on  $\LaTeX$ },
Indian  $\TeX$  User Group, 2000.
\bibitem[LL94]{Lamport:1994}
Leslie Lamport,
\emph{ $\LaTeX$ : A Document Preparation System (2nd Edition)}, Addison—Wesley
Professional, 1994.
\bibitem[OPHS15]{Oetiker:2015}
Tobias Oetiker et. al.,
```

```
\emph{The Not So Short Introduction to \emph{\LaTeXe}}, URL: https://ctan.org/tex
  –archive/info/lshort/english/,
2015.
\end{thebibliography}
```

## Referências Bibliográficas

- [GMS04] Michel Goossens and Frank Mittelbach *The L<sup>A</sup>T<sub>E</sub>X companion (second edition)*, Addison–Wesley, 2004.
- [Tea00] Tutorial Team, *Online tutorials on L<sup>A</sup>T<sub>E</sub>X*, Indian T<sub>E</sub>X User Group, 2000.
- [LL94] Leslie Lamport, *L<sup>A</sup>T<sub>E</sub>X: A Document Preparation System (2nd Edition)*, Addison–Wesley Professional, 1994.
- [OPHS15] Tobias Oetiker et. al., *The Not So Short Introduction to L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>*, URL: <https://ctan.org/tex-archive/info/lshort/english/>, 2015.

No texto, uma citação da referência bibliográfica é inserida pelo comando `cite{chave}` onde `chave` é a chave colocado no `\bibitem`.

Se for citar mais de um item, coloque as chaves separadas pela virgula. Também poderá colocar informações adicionais como parâmetro opcional do `\cite`. Veja Exemplo 11.2.

Exemplo 11.2: ex11-cite.tex

Exemplo de referência bibliográfica (veja `\cite{Lamport:1994}` e `\cite{Oetiker:2015}`).

Para exemplos usando o BibTeX, poderá consultar `\cite{IndianTUG:00, Goossens:2004}`.

Para recursos avançados de BibTeX, veja o `\cite[Cap.~13]{Goossens:2004}`.

Exemplo de referência bibliográfica (veja [LL94] e [OPHS15]).

Para exemplos usando o BibTeX, poderá consultar [Tea00, GMS04].

Para recursos avançados de BibTeX, veja o [GMS04, Cap. 13].

## 11.2 Usando o BIBTEX

Checar qual das referências foram usadas e formatar uniformemente de acordo com as exigências do editor é uma tarefa difícil. Para automatizar este serviço, podemos usar o BIBTEX que é uma ferramenta especialmente desenvolvida para manipular referências bibliográficas. Os editores costumam deixar um arquivo de estilo próprio para BIBTEX e usando ele, o BIBTEX formatará automaticamente as referências bibliográficas de acordo. Outra coisa importante que o BIBTEX faz é checar quais dos itens foram usados e imprimir somente as referencias citadas no texto.

Para usar o BIBTEX, prepara o arquivo de referencia bibliográfica separado com extensão `.bib` que contém informações das referências. Em geral, costumamos usar os aplicativos gráficos tal como aplicativo gratuito e multi plataforma `jabref` (<http://www.jabref.org/>) para editar o arquivo de BIBTEX, mas também pode ser editado manualmente.

Aqui, vamos ver como preparar manualmente o arquivo `.bib`. Veja o Exemplo 11.3 para ver como deve ficar o arquivo `.bib`.

Exemplo 11.3: `ex11-bibtex.bib`

```
@string{AW="Adilson--Wesley"}
@string{TUG="{\TeX{}} User Group"}

@book{Lamport:1994,
author={Leslie Lamport},
title="{\LaTeX{}}, A Document Preparation System (2nd Edition)",
publisher=AW,
address = {Reading, MA},
year=1994
}

@book{Goossens:2004,
author={Michel Goossens and Frank Mittelbach},
title="The {\LaTeX} companion (second edition)",
publisher=AW,
address = {Reading, MA},
year=2004
}

@book{Oeticker:2015,
author={Tobians Oeticker and Hubert Partl and Irene Hyna and Elisabeth Schlegl},
title="The Not So Short Introduction to {\LaTeXe{}}",
publisher=CTAN,
note={https://ctan.org/tex-archive/info/lshort/english/},
year=2015
}

@book{IndianTUG:00,
author={Tutorial Team},
title="Online Tutorials on {\LaTeX{}}",
```

```

publisher="Indian " # TUG,
note = {http://www.tug.org/tutorials/tugindia/},
year=2000
}

@article{Mertz:2009,
author={Andrew Mertz and William Slough},
title={A TikZ tutorial: Generating graphics in the
spirit of {\TeX}},
journal=TUGboat,
note={http://www.tug.org/TUGboat/tb30-2/tb95mertz.pdf}
volume=30,
number=2,
year=2009
}

```

O comando de  $\text{BIB}_{\text{TEX}}$  inicia com “@” seguido de nome e seus dados entre chaves. Os dados são listas separados pela virgula na forma forma “chave=dado”, onde dado pode ser delimitado por aspas ou chaves.

`@string` permite definir abreviaturas. No começo do arquivo, define abreviatura `AW` para “Adilson–Wesley” e `TUG` para “ $\text{T}_{\text{E}}\text{X}$  User Group” para facilitar a digitação de dados. Note que, comando de  $\text{T}_{\text{E}}\text{X}$  devem ficar delimitados pelas chaves no arquivo de  $\text{Bib}_{\text{TEX}}$ .

Depois encontra o `@book` que especifica que é dado de um livro. Entre chaves, estão os dados do livro tais como título, autor, etc, onde cada campo é separado da outra pela virgula e dados do campo são delimitados pelos chaves ou aspas. Note que o primeiro elemento de `@book` é uma chave a ser usado dentro do documento pelo comando `\cite` e é único que não é da forma `campo = dado do campo` Note que, tanto aspas como chaves podem ser usados para delimitar dados do campo, mas se dado for uma única palavra como o ano, não precisa do delimitador.

Também lembrar novamente que, comandos de  $\text{T}_{\text{E}}\text{X}$  devem ficar delimitados por chaves, o que indica ao  $\text{BIB}_{\text{TEX}}$  para não alterar esta parte (como converter entre maiúsculo e minúsculo). Note o uso da abreviatura `AW` em alguns pontos do documento (sem colocar delimitadores). Esta abreviatura foi definido no começo do arquivo e será substituído automaticamente pelo seu valor “Adilson–Wesley” pelo  $\text{BIB}_{\text{TEX}}$ .

No `publisher="Indian "# TUG`, “#” faz a concatenação de dois strings `Indian` e o `stringTUG` que é `{\TeX}` User Group.

Tendo o arquivo `.bib` pronto, efetuamos as citações como no caso anterior.

Agora, no final do arquivo, onde colocamos o ambiente `thebibliography`, substituímos por

```

\bibliographystyle{[estilo bib]}
\bibliography{[arquivo bib]}

```

onde `[estilo bib]` é o estilo usado para formatar as referências bibliográficas e `[arquivo bib]` é o nome do arquivo de  $\text{BIB}_{\text{TEX}}$  (extensão `bib`) sem extensão. Quando tiver mais de um arquivo, separe os nomes pela vírgula.

Agora execute o  $\LaTeX$ , BibTeX e  $\LaTeX$  novamente. Em geral, os editores especializados para  $\LaTeX$  tem os botões ou menus para rodar o BibTeX, além dos botões de compilar (executar o  $\LaTeX$ ).

Ao executar BibTeX, será gerado a saída das referências bibliográficas somente com itens citados. Por exemplo, para as citações do Exemplo 11.2, o artigo com chave `Mertz:2009` será ignorado por não estar citados.

Para mudar a formatação das referências, é só alterar o estilo em `\bibliographystyle` e executar o BibTeX e  $\LaTeX$  novamente. Os estilos básicos são: `plain` (rótulo numérico), `unsrt` (similar a `plain`, mas sem ordenar – na ordem que foi citado), `alpha` (rótulo pelo sobrenome e ano), `abbrv` (similar a `plain`, mas mais compacto), `amsplain` (estilo `plain` do AMS), `amsalpha` (estilo `alpha` do AMS).

Além desses, existem vários outros estilos, dependendo da instalação de cada sistema TeX em uso.

Se o editor dispõe de arquivos de estilo para BibTeX (com extensão `bst`), deixe este arquivo junto com arquivo `tex` e no `\bibliographystyle`, passe o nome do arquivo `.bst` sem a extensão. Se arquivo é `meuart.bst`, o estilo é `meuart`.

Uma observação importante é que o BibTeX não suporta a adequadamente a acentuação direta ainda. Assim, deverá acentuar no modo TeX. Lembre-se que a acentuação no modo TeX precisa estar entre chaves para BibTeX não tentar processar esta parte.

Quando o editor exige um padrão diferente do disponível e não fornece o arquivo de estilo para BibTeX, podemos contornar o problema da seguinte forma. Primeiro termine o artigo ou livro com citações necessárias e usando um estilo mais próximo do exigido. Depois rode o  $\LaTeX$  e BibTeX. O BibTeX gerará um arquivo com extensão `bbl`, como mesmo nome do arquivo `tex`. Este arquivo `.bbl` contém o ambiente `thebibliography` com itens de bibliografia já formatada. Copie ele para documento original onde está o

```
\bibliographystyle{[estilo bib]}
\bibliography{[arquivo bib]}
```

e efetue alterações anualmente.

Não esqueça de comentar o

```
\bibliographystyle{[estilo bib]}
\bibliography{[arquivo bib]}
```

Além de `@book` (livro), existem muitas especificações de documentos tais como `@article` (artigo), `@manual` (documentação técnica), `@phdthesis` (tese de doutorado), etc, uma para cada categoria. Se não enquadrar em nenhuma das categorias disponíveis, use o `@misc` (diversos).

O texto sem estar com especificação que inicia com “@” será ignorado pelo BibTeX e funciona como comentário. Mas existe o especificador `@comment` especial para inserir comentários para fins de organização.

Como existem muitas categorias e cada categoria tem campos diferentes a ser preenchidos, editar o arquivo de BibTeX manualmente não é muito simples. Assim, costumamos usar os

aplicativos próprios para isso. Um desses aplicativos é o `jabref` (<http://www.jabref.org/>) que funciona em quase toda plataforma e é gratuito.

### 11.3 Índice remissivo

Para usar o índice remissivo, usa-se o pacote `makeidx` e o comando `\makeindex` no `preamble` do documento para ativar o seu uso.

Para acrescentar itens no índice remissivo, usa-se o comando `\index{}`, mas lembre-se de não colocar espaços entre palavra referida e o comando. Para colocar índice remissivo com sub-entrada, usa-se o “!” para separar entrada e sub-entrada, mas evite espaços nos lados de “!” (qualquer espaço extra pode causar confusões na organização do índice remissivo).

As vezes, é necessário usar uma “chave” para classificar os itens.

Isto ocorre quando usamos símbolos matemáticos, aspas, ou outros caracteres especiais, formatação de caracteres, etc. Sem a “chave” de classificação, `makeindex` pode classificar usando o que está escrito, o que nem sempre corresponde a posição correta. Para resolver este problema, usa-se o `\index{chave@item}`. Por exemplo

Ps itens em negrito `\index{item em negrito@\textbf{item em negrito}}`,  
itens com aspas `\index{item com aspas@"item com aspas"}` e símbolos `\index{símbolos}`, etc  
podem ser colocados com ordenação correta.

inserem entradas em negrito e item com aspas classificado corretamente. Lembre também que as letras acentuadas costumam vir na posição diferente das não acentuadas. Neste caso também poderá usar a chave para colocar na posição correta. Veja o Exemplo 11.4.

#### Exemplo 11.4: ex11-index.tex

Podemos usar índice remissivo `\index{índice remissivo}` com sub-entrada `\index{índice remissivo!sub-entrada}`.  
Também podemos usar “chave” para classificar os itens, como itens em negrito `\index{item em negrito@\textbf{item em negrito}}`, itens com aspas `\index{item com aspas@"item com aspas"}` e símbolos como  $\alpha$  `\index{alfa@$\alpha$}`.  
Também podemos usar item e enumeração em negrito `\index{item e enumeração em negrito@\textbf{item e enumeração em negrito}|textbf}`  
`\printindex % colocar indice remissivo (glossário) aqui.`

No corpo do documento, terá nenhuma diferença visual em ter o `\index{}` e logo, a sua saída foi omitida.

Podemos especificar sub-entrada com chaves como em

itens em negrito com sub-entrada `\index{item@\textbf{item}!\textbf{em negrito}!\textbf{com sub-entrada}}`.

Observe a ordem que usa o “@” e “!” na especificação de itens com chave e subentrada.

Para indicar a ocorrência num trecho, usa-se o `\index{item|}` ( para iniciar e `\index{item|}` para finalizar.

Quando um ítem referê-se ao outro, usa-se o `\index{ítem|see{referencias}}` com em `grossário\index{grossário|see{índice remissivo}}`.

Para mudar a fonte usada na enumeração da página, coloque a especificação após “|” ou “|)” dependendo de ser página ou trecho como em

Índice com paginação em negrito `\index{item e enumeração em negrito@\textbf{ítem e enumeração em negrito}|textbf}`.

Para colocar o índice remissivo gerado, usa-se o comando `\printindex`.

Finalmente, para produzir o índice remissivo, deverá executar `LATEX`, `makeindex` e `LATEX` novamente. Os editores especializados para `LATEX` costuma vir com botão ou menu para chamar o `makeindex`.

O índice remissivo do Exemplo 11.4 ficaria como segue

## Índice Remissivo

índice remissivo, 1  
 sub-entrada, 1

$\alpha$ , 1

“item com aspas”, 1

**item em negrito**, 1

**item e enumeração em negrito**, 1

Note que @, !, | e ”tem significados na entrada de índice remissivo. Para colocar estes caracteres na entrada de índice remissivo, coloque ”antes dele. Por exemplo, `\index{"@author}` insere “@author” na entrada de índice remissivo.



# Capítulo 12

## Medidas e Contadores

### 12.1 Unidade de medidas e espaçamentos

As vezes o espaçamento ajustado automaticamente não está bom e queremos fazer pequenos ajustes.

Para aumentar espaçamento entre parágrafos em um determinado ponto, como entre texto e equações altas, poderá usar o `\smallskip`, `\medskip` e `\bigskip` para ter espaçamentos maior que o padrão. Note que, se quer ter espaçamento maior entre texto e equações em todo o documento, deverá efetuar ajustes de parâmetros `\abovedisplayskip` e `\belowdisplayskip`.

Para espaço horizontal e vertical, usa-se o `\hspace{medida}` e `\vspace{medida}` respectivamente. Caso estiver inserindo o espaço vertical no começo das páginas, ou espaço horizontal no começo das linhas, use a versão com “\*” `\hspace*{medida}` e `\vspace*{medida}`.

Medidas comumente usados são:

**pt** (point - unidade gráfica) =  $\frac{1}{72.27}$ in (polegada) ou 0.351mm

**mm** (milímetro) = 2.845pt

**pc** (pica) = 12pt ou 4.218mm

**cm** (centímetro) = 2.371pc

**in** (polegada) = 25.4mm ou 72.27pt ou 6.022pc

**ex** altura da letra “x” minúsculo da fonte corrente

**em** largura da letra “M” maiúsculo da fonte corrente

**mu** (math unit) =  $\frac{1}{18}$ em

`\stretch{peso}` espaço esticável com peso especificado.

Existem ainda várias outras medidas menos usadas que foram omitidos aqui.

O `\stretch{peso}` produz medida que preenche o espaço. Existem alguns comandos deste tipo que são úteis: `\hfill` (equivale a `\hspace{stretch{1}}`), `\vfill` (equivale a

`\vspace{stretch{1}}`), `\hrulefill` (similar a `\hfill`, mas preenche com linha), `\dotfill` (similar a `\hfill`, mas preenche com pontos).

Ainda existem comandos para inserir pequenos espaços, que são `\quad` (insere espaço de  $1em = 18\mu$ ), `\qqquad` (insere espaço de  $2\quad=2em$ ), `\enspace` ou `\enskip` (insere espaço de  $\frac{1}{2}\quad$ ). Estes comandos costumam ser usados para inserir pequenos espaçamentos para melhora a aparência das fórmulas matemáticas. No modo matemático, ainda existem comandos de inserção de espaços menores que são

`\,` ( $\frac{3}{18}\quad = 3\mu$ ),

`\:` ( $\frac{4}{18}\quad = 4\mu$ ),

`\;` ( $\frac{5}{18}\quad = 5\mu$ ),

`\!` ( $\frac{-3}{18}\quad = -3\mu$ ).

Veja Exemplo 12.1.

Exemplo 12.1: ex12-espacos.tex

palavra a esquerda `\hfill` palavra a direita.

palavra a esquerda `\hrulefill` palavra a direita.

palavra a esquerda `\dotfill` palavra a direita.

primeiro`\hfill` segundo`\hrulefill\hrulefill` terceiro`\dotfill\dotfill\dotfill` quarto

`\bigskip`

`\$x \in \mathbf{I\!R}, \enspace x>0\$`.

palavra a esquerda	palavra a direita.
palavra a esquerda _____	palavra a direita.
palavra a esquerda .....	palavra a direita.
primeiro           segundo_____terceiro .....	quarto

$x \in \mathbf{R}, x > 0.$

Nota: o `\mathbf{R}` construído acima é apenas como ilustração. Deverá usar a fonte apropriada tal como `\mathbb{R}` do AMS.

Outra coisa que as vezes usamos para alinhamento dos elementos de fórmulas é o comando `\phantom` que, em vez de produzir elementos da fórmula, reserva o espaço usado por ele. Por exemplo,  $\Gamma_{ij}^k$  e  $\Gamma_{ij}^k$ .

## 12.2 Medidas pré-definidas ou definidos pelo usuário

Existem algumas medidas pré-definidas como comandos, relacionadas à configuração das páginas e similares que costuma ser usados com certa frequência. Aqui, vamos citar algumas delas. Para referenciar, coloque “\” antes dele, para ficar como comandos

`\textwidth` Largura de texto atual.

`\linewidth` largura da linha atual. Por exemplo, dentro da lista, será menor que o `textwidth`.

`\columnwidth` largura da coluna. Se for em uma coluna, coincide com `textwidth`.

`\columnsep` distância entre colunas no modo multi colunas.

`\columnseprule` largura da linha que separa colunas no modo multi colunas (`Opt` para desabilitar).

`\textheight` altura do texto atual

`\parindent` indentação (quanto deixa no lado esquerdo) do parágrafo.

`\parskip` Quanto deixa de espaço antes do parágrafo.

`\paperwidth` largura do papel.

`\paperheight` altura do papel.

`\unitlength` medida de unidade usado no ambiente `picture`.

Dentro do `minipage`, `\textwidth`, `\linewidth` e `\columnwidth` assumem a largura do `minipage`, mas no `\parbox`, somente `\linewidth` assumirá a largura da caixa.

Para imprimir estas medidas, coloque o comando `\the` antes do comando de medidas. Por exemplo, para imprimir o valor de `\textwidth`, use `\the\textwidth`.

Para definir uma nova medida, usa-se o comando `\newlength{\minhamedida}`. Este comando criará a medida `\minhamedida`. Para definir tamanho da medida, use `\setlength{\minhamedida}{<medida>}` onde `medida` é a medida da `\minhamedida`.

O valor da medida pode ser configurado a partir de `box` (caixa) existente (como texto, fórmulas, etc). Uma caixa (de elemento) tem altura (`height`), largura (`width`) e a distância da parte inferior até a linha base (`depth`).

`\settoheight{\minhamedida}{<conteúdo>}`, `\settowidth{\minhamedida}{<conteúdo>}` e `\settodepth{\minhamedida}{<conteúdo>}` configuram para essas medidas correspondentes ao seu parâmetro.

Note que, pela facilidade, as vezes usamos o modo `TEX` para configurar as medidas como em `\minhamedida=3.0cm` Para facilitar os cálculos das medidas, normalmente usa-se o pacote `calc`.

O pacote `calc`, além de permitir calcular medidas com facilidade, ainda acrescentam alguns novos comandos tais como `\setttotalheight{\algumamedida}{<conteúdo>}` que configura para `width+depth` e `\widthof{<conteúdo>}`, `\heightof{<conteúdo>}`, `\depthof{<conteúdo>}` e `\totalheightof{<conteúdo>}` que retornam `width`, `height`, `depth`, e `width+depth`, respectivamente.

## 12.3 Contadores

$\LaTeX$  usa diversos contadores para efetuar enumeração automática, tais como páginas, equações enumeradas, capítulos e seções, figuras, etc.

Estes contadores podem ser controladas, tanto na contagem como na sua aparências.

Em geral, o costume é nomear contador com mesmo nome do ambiente e o que precede `\the` será usado para imprimir o seu valor. Por exemplo, a impressão de enumeração da página usa o `\thepage` e enumeração do capítulo usa `\thechapter` e assim por diante.

Os principais contadores pré definidos no  $\LaTeX$  são: `part`, `chapter`, `section`, `subsection`, `subsubsection`, `paragraph`, `subparagraph`, `page`, `figure`, `table`, `footnote`, `mpfootnote` (rodapé dentro do minipage) e `equation`.

Então, alterando o `\the<contador>`, alterará como será impresso estes contadores.

Por exemplo, `\renewcommand{\thepage}{\roman{page}}` alterará a enumeração das páginas para `i`, `ii`, `iii`, etc.

O estilo de enumeração `roman` acima, podem ser

`\arabic` que é 1, 2, 3 ...

`\alph` que é a, b, c ...

`\Alph` que é A, B, C ... `\roman`] que é i, ii, iii ... `\Roman`] que é I, II, III ... `\fnsymbol`] que é sequencia de símbolos (pode ser usado no rodapé, se for pouco).

Veja Exemplo 12.2.

Exemplo 12.2: `ex12-contadores.tex`

A página atual é `\thepage`.

Seção atual em romano minúsculo é `\roman{section}`.

A página atual é 86.

Seção atual em romano minúsculo é iii.

As vezes, precisamos ajustar valores iniciais dos contadores, tais como das páginas, listas enumeradas, etc. Para tanto existem comandos `\setcounter`, `\stepcounter` e `\addtocounter`. Para decrementar, use o valor negativo no parâmetro de `\addtocounter`.

No caso de listas enumeradas, os contadores são `enumi`, `enumii` (sub lista), `enumiii` (subsub lista), `enumiv` (subsubsub lista).

Para que estes contadores sejam impressos na fonte reta, mesmo no ambiente em itálico (enunciado do teorema, por exemplo), acrescente

```
\renewcommand{\labelenumi}{\textup{\theenumi.}}
\renewcommand{\labelenumii}{\textup{(\theenumii)}}
\renewcommand{\labelenumiii}{\textup{\theenumiii.}}
\renewcommand{\labelenumiv}{\textup{\theenumiv.}}
```

no `preamble` do documento.

Além disso, o contador novo pode ser criado por `\newcounter`, útil para ser usado no ambiente/comando novo a ser criados.

Aqui, vamos criar um contador para testá-los. Veja o Exemplo 12.3.

#### Exemplo 12.3: ex12-newcounter.tex

```
\newcounter{teste} % cria e inicializa com o valor zero.
\setcounter{teste}{1} % novo valor
```

O contador é `\theteste`.

O contador em `\texttt{Alph}` é `\Alph{teste}`.

Adicionando 1 e usando.

```
\refstepcounter{teste}\label{count:teste}.
```

Novo valor é `\theteste`.

Adicionando por 1 de novo `\addtocounter{teste}{1}`

Novo valor é `\theteste`.

O valor atribuído no rótulo `\texttt{count:teste}` após o `\texttt{refstepcounter}` é `\ref{count:teste}`.

Agora, a lista enumerada começando de `$3$`

```
\begin{enumerate}\setcounter{enumi}{2}
```

```
\item Um item.
```

```
\item Outro item.
```

```
\end{enumerate}
```

O contador é 1.

O contador em Alph é A.

Adicionando 1 e usando. . Novo valor é 2.

Adicionando por 1 de novo Novo valor é 3. O valor atribuído no rótulo `count:teste` após o `refstepcounter` é 2.

Agora, a lista enumerada começando de 3

3. Um item.

4. Outro item.

Para que o `\label{}` pegue o valor atual do contador, use o `\refstepcounter` que incrementa o contador por um e atualiza o valor para `\label`.

Para pegar o valor de um contador (para ser usado como argumento para `\setcounter`, por exemplo), use o comando `\value`.

Muitas vezes, um contador está vinculado no outro e quando outro for incrementado, ele será reinicializado. Além disso, contadores vinculados aos outros imprimem dois valores (contador do outro mais dele).

Este vínculo pode ser criado pelo comando `\numberwithin{<contador>}{<pai>}`. Quando o `<pai>` foi incrementado, o `<contador>` é reinicializado. Por exemplo, para que a enumeração da equação seja da forma `<no. do capítulo>.<no da equação>`, basta colocar `\numberwithin{equation}{chapter}` no `preamble` do documento.

Mais sobre manipulações dos contadores, veja o [wik18, Capítulo 3, Seção 3].

# Capítulo 13

## Mais Alguns Cuidados e Ajustes

Neste capítulo, veremos mais alguns ajustes e incrementos no documentos.

### 13.1 Comandos frágeis

Existem comandos denominados frágeis por poder causar problemas quando é passado como parâmetros de alguns comandos ou ambientes. Por exemplo, O argumento do `\chapter` e `\section` costumam ser usado no sumário também. Se colocar comando que tem contadores como o `\footnote`, causará problemas, pois tentará colocar rodapé também no sumário.

Assim, quando colocar comandos no argumento da função que usam o seu parâmetro em mais de um lugar, requer cuidados. Além dos comandos de seccionamento (`chapter`, `section`, etc), `caption`, `thanks`, comandos que produz saída do cabeçalho como `markboth`, etc também usam o seu argumento em mais de um lugar.

Os comandos que causam problemas quando é passado para argumento destes comandos são chamados de comandos frágeis e deve ser precedido de `\protect` para prevenir problemas.

Alguns comandos frágeis são: comandos com argumento opcional, ambientes, fórmula no modo `displaystyle`, fórmula no modo `textstyle` delimitado por `\(` e `\)` (no entanto, delimitado por “\$” não é frágil), `\phantom`, “\”, `\item` e `\footnote`.

Veja Exemmplo 13.1.

Exemplo 13.1: ex13-fragil.tex

```
\section{Sublinhando texto\protect\footnote{Forma antiga de enfatizar o texto}} %
  footnote é frágil: usar \protect
\ldots
\section{Sobre a famosa fórmula  $a^2=b^2+c^2$ } % delimitado por "$" não é frágil
\ldots
\begin{figure}[hbp!]
\center
Figura aqui.
\caption{$\Gamma_{ij}^{\phantom{ij}2}$} % \phantom é frágil
\end{figure}
```

`\ldots`

## 1.1 Sublinhando texto<sup>1</sup>

...

## 1.2 Sobre a famosa fórmula $a^2 = b^2 + c^2$

...

Figura aqui.

Figura 1.1:  $\Gamma_{ij}^2$ 

...

---

<sup>0</sup>Forma antiga de enfatizar o texto

## 13.2 Babel e nomes

O pacote de internacionalização `babel` oferece regra de hifenização e nomes dos elementos tais como `capítulo`, `figura`, `tabela`, `sumário`, etc em vários idiomas. Para carregar mais de um idioma, coloque os idiomas separado pela vírgula no parâmetro do `babel`. O idioma do último será considerado padrão e outros são opcionais que podem ser ativados quando quiser.

Seleção de idiomas pode ser mudado por `\selectlanguage{<idioma>}`. Para demarcar somente um trecho no outro idioma, usa-se o ambiente `otherlanguage`. Para trecho bem curto, poderá usar também o comando `\foreignlanguage`. Veja Exemplo 13.2.

Exemplo 13.2: `ex13-babel.tex`

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
\usepackage[english,brazil]{babel}
\title{Exemplo}
\author{Sadao Massago}
\date{Fevereiro, 2018}
\begin{document}
\maketitle
\begin{abstract}
Resumo aqui.
\end{abstract}
% \tableofcontents % so se for artigo longo
\section{Mudando o idioma de um trecho}
O trecho curto pode ser no outro idioma como em “\foreignlanguage{english}{This is english}”.
Para trechos maiores, use o ambiente \texttt{otherlanguage} como em

```

```

\begin{quote}
\begin{otherlanguage}{english}
This environment switches all language–related definitions, like the language specific names
for figures, tables etc. to the other language.
\end{otherlanguage}
\end{quote}
\ldots
% referência bibliográfica
\end{document}

```

## Exemplo

Sadao Massago

Fevereiro, 2018

### Resumo

Resumo aqui.

## 1.1 Mudando o idioma de um trecho

O trecho curto pode ser no outro idioma como em “This is english”. Para trechos maiores, use o ambiente `otherlanguage` como em

This environment switches all language-related definitions, like the language specific names for figures, tables etc. to the other language.

...

Para escrever o nome tais como “Capítulo”, “Figura”, “Sumário”, etc, tem os comandos que produz estes nomes. Redefinindo apropriadamente estes comandos, podemos alterar a sua saída.

Alguns nomes pré definidos são: `\abstractname` (somente `article` e `report`), `\appendixname`, `\bibname` (somente `book` e `report`), `\chaptername` (somente `book` e `report`), `\contentsname`, `\figurename`, `\indexname`, `\listfigurename`, `\listtablename`, `\partname`, `\refname` (somente `article`), `\tablename`.

Note que o nome para referência bibliográfica no livro e relatório são `\bibname` enquanto que no artigo, será `\refname`.

Alguns pacotes definem o nome usado para seus pacotes. Por exemplo, `amsthm` que define o ambiente `proof`, define também `\proofname`.

Para alterar ou definir novos nomes quando usa o babel, requer usar o recurso do babel,

pois nomes pré definidos no babel são redefinidos dinamicamente quando altera os idiomas.

Assim, deverá definir/redefinir os nomes dentro do comando `\captions<idioma>` onde `<idioma>` é o nome do idioma. Mas, se redefinir o `captions`, perderá as definições anteriores feitas pelo `babel`. Logo, utiliza o comando `\addto` do `babel` que acrescenta porções de código no final do comando especificado.

Por exemplo,

```
\addto\captionsbrazil{%
  \renewcommand\refname{Referências Bibliográficas}%
}
```

no `preamble` altera o nome da referência bibliográfica no caso de artigos quando usa o idioma `brazil`.

Note que, no caso de português brasileiro, tanto pode usar o `brazil` como o `brazilian`. Assim, precisará redefinir em ambos idiomas, se estiver implementando um pacote na qual não sabe qual opção o usuário final vai usar. Neste caso, define um comando auxiliar e coloca em cada um dos idiomas.

```
\providecommand{\theoremname}{Theorem} % Providenciando novo nome com valor
  padrao
\addto\braziliannames{%
  \renewcommand\refname{Referências Bibliográficas}%
  \renewcommand{\theoremname}{Teorema}%
}
\newcommand{\englishnames}{%
  \renewcommand{\theoremname}{Theorem}%
}

\addto\captionsbrazil{\braziliannames}
\addto\captionsbrazilian{\braziliannames}

\addto\captionsestlish{\englishnames}
\addto\captionssmerican{\englishnames}

\newtheorem{theorem}{\theoremname}[section]
```

no `preamble` após carregar o pacote `babel` habilita suporte para português brasileiro e inglês. Note que o `\newtheorem` usa o comando `\theoremname` que será redefinido quando o idioma muda. Veja o exemplo 13.3.

Exemplo 13.3: ex13-babel-caption.tex

```
\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{amsmath,amssymb}
```

```

\usepackage{amsthm}
\usepackage[english,brazil]{babel}
\providecommand{\theoremname}{Theorem} % Providenciando novo nome com valor
    padrao
\addto\braziannames{%
  \renewcommand\refname{Referências Bibliográficas}%
  \renewcommand{\theoremname}{Teorema}%
}
\newcommand{\englishnames}{%
  \renewcommand{\theoremname}{Theorem}%
}

\addto\captionsbrazil{\braziannames}
\addto\captionsbrazilian{\braziannames}

\addto\captionenglish{\englishnames}
\addto\captionamerican{\englishnames}

\newtheorem{theorem}{\theoremname}[section]

\title{Exemplo}
\author{Sadao Massago}
\date{Fevereiro, 2018}
\begin{document}
\maketitle
\begin{abstract}
Resumo aqui.
\end{abstract}
% \tableofcontents % so se for artigo longo
\section{Teoremas multi lingue}
\begin{otherlanguage}{english}
\begin{theorem}
\ldots
\end{theorem}
\end{otherlanguage}
\begin{theorem}
\ldots
\end{theorem}
\begin{thebibliography}{99}
\bibitem[GMS04]{Goossens:2004}
Michel Goossens and Frank Mittelbach
\emph{The {\LaTeX} companion (second edition)}, Adilson—Wesley, 2004.
\end{thebibliography}
\end{document}

```

## Exemplo

Sadao Massago

Fevereiro, 2018

### Resumo

Resumo aqui.

## 1 Teoremas multi lingue

Theorem 1.1. ...

Teorema 1.2. ...

## Referências Bibliográficas

[GMS04 ] Michel Goossens and Frank Mittelbach *The L<sup>A</sup>T<sub>E</sub>X companion (second edition)*, Adilson–Wesley, 2004.

Note que, se usar a opção de idiomas como opção de documentos (no `documentclass`, o pacote `babel` reconhece este idioma e vai usar, mesmo que não tenha opções no `babel`).

### 13.3 Sobre espaçamentos entre linhas e “estouro de linhas”

O espaçamento entre linhas é definido pelo comando `\linespread` onde `\linespread{1.3}` é espaçamento um e meio e `\linespread{1.6}` é espaçamento duplo, o que atuará a partir do próximo parágrafo.

Note que o L<sup>A</sup>T<sub>E</sub>X deixa “estourar” as linhas (`Overfull \hbox`) quando não consegue acomodar devidamente o conteúdo nas linhas. Para que ele use o espaçamento grande entre elementos (`Underfull \hbox`) em vez de “estourar linhas”, coloque `\sloppy` (para voltar, use `\fussy`). Note que, independente de linha estar “estourando” ou “espaçando demais”, deverá checar cada um deles para ver se não está prejudicado visualmente e se for o caso, resolver.

## 13.4 Sobre hifenização

O pacote `babel` carrega a regra de hifenização para idioma selecionada, mas as vezes encontramos as exceções na qual a regra do `babel` não funciona devidamente. Neste caso, poderá indicar na palavra de texto, a posição que pode ser hifenizada por “-” como em `re\-\fe\-\rên-\cia`.

Este comando é ignorado quando não há necessidade de hifenização. Quando precisar, será hifenizada somente nestes locais.

Em geral, quando encontra uma palavra hifenizada indevidamente, poderemos querer acrescentar regra de hifenização para tal palavra em vez de indicar localmente. Para tanto, coloque o comando `\hyphenation` onde seu argumento é a lista de palavras separados pelo espaço, onde cada palavra contém “-” na posição de hifenização. Note que, para colocar regra de hifenização das letras acentuadas pelo comando `\hyphenation`, requer `\usepackage[T1]{fontenc}` ou similar para ativar fontes que suportam acentuações.

o seguinte trecho de código colocado depois do carregamento do pacote `babel` no `preamble` acrescenta hifenizações das palavras correspondentes.

```
% regra de hifenização das palavras nao acentuadas:
% nao requer \usepackage[T1]{fontenc}
\hyphenation{li-vro tes-te cha-ve bi-blio-te-ca}
% regra de hifenização das palavras acentuadas:
% requer \usepackage[T1]{fontenc}
\hyphenation{co-men-tá-rio re-fe-rên-cia}
```

Para que ele não hifenize, é só colocar dentro de uma caixa, o que pode ser feito pelo comando `\mbox`. O código do tipo `\mbox{não hifenizar}` não pode ser quebrado em linhas, mesmo na posição de espaços. Este recurso permite proibir hifenização das palavras localmente.

Para que não use a hifenização no documento inteiro, costuma aumentar o valor da penalidade de hifenização no `preamble` do documento. Com em

```
\hyphenpenalty=10000
\exhyphenpenalty=10000
```

## 13.5 Acessando os comandos com “@”, criando comandos versão “\*” e tipo chave=valor

As vezes precisamos usar os comandos ou ambientes que contém “@” no seu nome. O caso típico é redefinir comandos existentes tais como `\maketitle`, estilo de cabeçalho das páginas, etc. Note que os comandos e ambientes que usam “@” no seu nome são comandos sensíveis e devem ser usados com cuidado. Para acessar estes comandos e ambientes, coloque `\makeatletter` antes de usar e + depois.

Por exemplo, ajusta o espaçamento de linhas como simples no texto de rodapé, mesmo que

esteja usando espaçamento um e meio ou dupla no corpo do documento. Isto pode ser feito, acrescentando a seguinte código no `preamble` do documento.

```
\makeatletter % ativa uso de “@” no nome
\renewcommand\@makefnmark[1]{%
  \parindent 1em%
  % inicio da altera\c{c}\~ao
  \linespread{1} \@currsiz \noindent
  % \hb@xt@1.8em{\hss\@makefnmark}#1}
  \hb@xt@0.45em{\hss\@makefnmark}#1}
% fim da altera\c{c}\~ao
\makeatother % desativa o uso de “@” no nome
```

Outro exemplo e io caso de usar linhas pontilhadas no sumário.

```
\makeatletter
\renewcommand*\l@section{\@dottedtocline{1}{1.5em}{2.3em}}
\makeatother
```

no `preamble` do artigo habilita o uso de linhas pontilhadas no sumário, para seções. Para habilitar o pontilhado no capítulo do sumário é mais complexo. Veja o código do <https://tex.stackexchange.com/questions/62438/how-to-add-leaders-to-table-of-contents-without-toclo>

```
\renewcommand*\l@chapter[2]{%
  \ifnum \c@tocdepth > \m@ne
    \addpenalty{-\@highpenalty}%
    \vskip 1.0em \@plus\p@
    \setlength\@tempdima{1.5em}%
    \begingroup
      \parindent \z@ \rightskip \@pnumwidth
      \parfillskip -\@pnumwidth
      \leavevmode \bfseries
      \advance\leftskip\@tempdima
      \hskip -\leftskip
      #1\nobreak
      \xleaders\hbox{\$ \m@th
        \mkern \@dotsep mu\hbox{.}\mkern \@dotsep
        mu$}\hfill%
      \nobreak\hb@xt@\@pnumwidth{\hss #2}\par
      \penalty\@highpenalty
    \endgroup
  \fi}
```

Note que no  $\text{\LaTeX}$ , existem vários comandos e ambientes que tem versão normal e versão “\*” na qual a versão “\*” é uma variação da versão normal. Para criar os comandos e ambientes versão “\*”, usa-se o comando `\@ifstar` que determina se está ou não usando a versão “\*”, mas seu uso requer cuidados.

```

\makeatletter
\def\myemph{\@ifstar\@myemph\@@myemph} % selecionando os comandos
\newcommand{\@myemph}[1]{\underline{#1}} % versao “*”
\newcommand{\@@myemph}[1]{\emph{#1}} % versao normal
\makeatother

```

Define a versão normal e versão “\*” do `\myemph`. Note que na primeira linha foi usado o `\def` em vez de `\newcommand`, o que é necessário quando o comando tem parâmetros. Com o `\@ifstar`, seleciona o comando de acordo com a existência do “\*”. Note que o primeiro é versão com “\*” e o segundo é a versão sem “\*”.

Depois define os comandos com “\*” e comandos sem “\*” que foi usado anteriormente. Note o uso de “@” no nome destes comandos para proteger do uso indevido deles.

Assim, podemos usar a versão normal que enfatiza e versão “\*” que sublinha. Veja o Exemplo 13.4.

Exemplo 13.4: ex13-star.tex

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\makeatletter
\def\myemph{\@ifstar\@myemph\@@myemph} % selecionando os comandos
\newcommand{\@myemph}[1]{\underline{#1}} % versao “*”
\newcommand{\@@myemph}[1]{\emph{#1}} % versao normal
\makeatother
\begin{document}
Enfatizando palavras com \myemph{Teste} e \myemph*{Teste}.
\end{document}

```

Enfatizando palavras com *Teste* e Teste.

Note que no caso de ambientes, a versão com “\*” pode ser definido diretamente, sem a necessidade do artifício dos comandos como em

```

\newenvironment{mypar}[1]% 1 argumento eh titulo
  {\par \center -----< #1 >----- \\}% título e
  linha
  {\par \center -----< fim >----- \\}

\newenvironment{mypar*}[1]% 1 argumento que eh titulo
  {\par \center -----< #1 >----- \\}% título e
  linha
  {\par \center ----- \\}
\makeatother

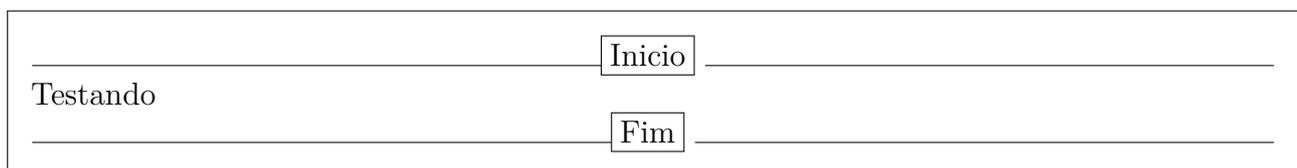
```

No caso de ambiente precisar usar o argumento na finalização, precisará associar a algum comando. Isto porque, o comando `\newenvironment{ambiente}` criará o par de comandos `\ambiente` (chamado pelo `\begin{ambiente}`) e `\endambiente` (chamado pelo `\end{ambiente}`) que funcionará em sincronismo. Os parâmetros do ambiente são passado para o comando de inicialização, mas não para a finalização.

Veja o Exemplo 13.5 que usa o segundo parâmetro na finalização.

Exemplo 13.5: ex13-parametro-finilizacao.tex

```
\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\newenvironment{mypartwo}[2]{
\par\noindent\hrulefill%
\ifx #1\empty
\else % se nao for vazio, coloca com fbox
  \fbox{#1}
\fi
\hrulefill\par
\def\mypartwoargtwo{#2}}{% agora eh finalizacao do ambiente
\par\noindent\hrulefill%
\ifx \mypartwoargtwo\empty
\else
  \fbox{\mypartwoargtwo}
\fi%
\hrulefill\par}
\begin{document}
\begin{mypartwo}{Inicio}{Fim}
Testando
\end{mypartwo}
\end{document}
```



Note que no exemplo acima, foi usado o comando `\ifx` para verificar se o argumento correspondente é ou não vazio. O comando `\else` é **senão** e `\fi` é **fim se**.

Outra coisa que pode querer é criar comandos e ambientes que recebem os parâmetros na forma `chave=valor`.

Para isso, usa-se o pacote `keyval`. para implementar o parâmetro do tipo `chave=valor`, deverá criar a regra e o que fazer quando tiver um determinado chave no parâmetro, usando o comando `\define@key`.

O primeiro parâmetro é o grupo que o parâmetro pertence. Em geral, coloca o nome do comando ou ambiente para evitar conflitos. Segundo parâmetro é nome da chave e terceiro é o que vai fazer. O #1 será o valor da chave.

Em geral, se tiver o nome da chave sem o valor, dará erro. Se quer aceitar a chave sem o valor, poderá passar o valor padrão como parâmetro opcional.

O parâmetro será avaliado pelo comando `\setkeys` onde primeiro parâmetro é o grupo que o parâmetro pertence.

Note que, se o parâmetro for definido pelo comando, precisará expandir antes, ou seja, retardar o `\setkeys`. Para isso, usa-se o comando `\expandafter`, mas retardar o comando para expandir o segundo argumento primeiro não é simples. Neste caso, define um comando auxiliar com um único argumento que chama o `\setkeys` com nome do grupo prefixado e aplicar `\expandafter` neste comando. Veja o Exemplo 13.6.

Exemplo 13.6: ex13-keyval

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\makeatletter
% O que vai fazer se encontrar "foo" no parametro do grupo "my".
\define@key{my}{foo}{foo vale #1\par} % se receber foo sem o seu valor, dara erro
\define@key{my}{bar}[99]{bar vale #1\par} % com valor padrao "99" se valor nao for
  repassado.
\makeatother

% Definindo o comando auxiliar com um argumento
\def\mysetkeys#1{\setkeys{my}{#1}}
% Comando associado ao argumento
\def\mykeyvalue{foo=5}
\begin{document}
Testando o parâmetro chave=valor.

\setkeys{my}{foo=3,bar}

O argumento que esta “armazenado” no comando.

\mykeyvalue

% expandindo o argumento primeiro para poder avaliar
\expandafter\mysetkeys\expandafter{\mykeyvalue}
\end{document}

```

```

Testando o parâmetro chave=valor.
foo vale 3
bar vale 99
O argumento que esta “armazenado” no comando.
foo=5
foo vale 5

```

Agora, veja o Exemplo 13.7 para exemplo de implementação do ambiente. Nele, foi usado o `\newif` que cria um condicional. O comando `\newif\if<nome>` cria o condicional `\if<nome>` que torna verdadeiro ou falso quando chamar `if<nome>true` ou `if<nome>false` respectivamente.

Exemplo 13.7: ex13-keyval-env

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\makeatletter
\newif\ifmyparkeyvaltitle % testar se tem titulo
% O titulo sera associado ao comando para poder ser usado posteriormente.
\define@key{myparkeyval}{title}{
\hrulefill\fbbox{#1}\hrulefill\myparkeyvaltitletrue
\def\myparkeyvaltitle{#1}}

\newenvironment{myparkeyval}[1][]{
\myparkeyvaltitlefalse
\par\noindent
\setkeys{myparkeyval}{#1}
\ifmyparkeyvaltitle
\else
\hrule
\fi
\par}{
\par\noindent
\ifmyparkeyvaltitle
\hrulefill\fbbox{fim do \myparkeyvaltitle}\hrulefill
\else
\hrule
\fi
\par}
\makeatother

\begin{document}
\begin{myparkeyval}
Sem titulo.
\end{myparkeyval}

```

```
\begin{myparkeyval}[title=Titulo]
Com titulo.
\end{myparkeyval}
\end{document}
```



## 13.6 Trocando fontes

Fontes padrão do  $\text{\LaTeX}$  é **Computer Modern** desenhado especialmente para ele, mas existem outras fontes incluídos no  $\text{\TeX}$  que podem ser usados.

Em geral, selecionar fontes manualmente requer cuidados e conhecimento sobre tipografia para não combinar fontes incompatíveis. Portanto, recorreremos aos pacotes desenvolvidos pelos **experts** no assunto.

Vamos ver como usar a fonte **Times** que é uma das fontes populares. Para selecionar a fonte **times**, usa os pacotes **mathptmx**. Basta colocar `\usepackage{mathptmx}` para que fontes fiquem como **Times**.

Para o Palatino, use o pacote **mathpazo** com o comando `\usepackage{mathpazo}`.

Ainda existem vários outros fontes que vem em qualquer distribuição  $\text{\TeX}$ , assim como fontes adicionais do  $\text{\TeX}$  mais completo.

## 13.7 Trocando marcador da lista itemizada

A marca de itens da lista `itemize` também pode ser redefinidos. Por exemplo, o código

```
\renewcommand{\labelitemi}{\bullet}
\renewcommand{\labelitemii}{\cdot}
\renewcommand{\labelitemiii}{\diamond}
\renewcommand{\labelitemiv}{\ast}
```

ajustado adequadamente no `preamble` faz isso.

## 13.8 Cores no $\text{\LaTeX}$

O pacote **color** (carregado automaticamente pelo pacote **graphicx**) oferece recursos básicos para trabalhar com cores no  $\text{\LaTeX}$ . Aqui, vamos supor que o `\usepackage{graphicx}` já está no `preamble`. Para poder referenciar cores pelos nomes, coloque `\usepackage[usenames]{color}` no `preamble` também.

Os comandos de cores usam o parâmetro opcional para indicar o modelo. Se for omitido, será assumido o que foi especificado na hora de carregar o pacote `color`. Como estamos supondo que vai usar a opção `usenames` na opção do pacote `color`, se o modelo for omitido nos comandos de cores, será assumido como `named` (por nomes).

A cor é especificado pelo comando `\color[modelo]{cor}`, mas lembre-se de proteger pelas chaves para que a cor retorne ao padrão fora delas. Para a cor do fundo (das páginas), usa-se o comando `\pagecolor[modelo]{cor}`. No caso do trecho de textos, podemos usar os comandos `\textcolor[modelo]{cor}{texto}` (texto com cor especificado), `\colorbox[modelo]{cor}{texto}` (caixa de texto com cor do fundo especificado) e `\fcolorbox[modelo]{fcor}{bcor}{texto}` (caixa de texto com cor `bcolor` e cor do contorno `fcolor`) são usados. Veja o Exemplo 13.8.

Exemplo 13.8: `ex13-color.tex`

Cor normal. `{\color{red}}` Somente neste trecho muda de cor.} Aqui é cor normal.

Combinando:

`\textcolor{white}{\colorbox{black}{texto colorido na caixa colorida}}`

Cor normal. **Somente neste trecho muda de cor.** Aqui é cor normal.

Combinando: **texto colorido na caixa colorida**

Segue uma tabela de cores básicos.

Tabela 13.1: cores aceitos em todos `drivers` no pacote `color`

black		white		red		green	
blue		cyan		magenta		yellow	

Para mais opções de cores, poderá usar opções `dvipsnames`, `usenames` na opção do pacote `color`, mas memorizar muitas cores não é produtivo. Assim, costuma usar o pacote mais avançado de cores que permite misturar cores.

# Capítulo 14

## Usando Pacotes Fora do base e required

Até agora, só estudamos os pacotes disponíveis em qualquer distribuição  $\text{T}_{\text{E}}\text{X}$ , usando somente os pacotes do conjunto denominado de **base** e **required**. No entanto, maioria das distribuições  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$  instala uma grande quantidade de pacotes adicionais para incrementar a sua funcionalidade. Assim, neste capítulo, vamos aventurar em alguns destes pacotes adicionais para facilitar o nosso trabalho.

### 14.1 Ajustando a configuração das páginas e similares

Para ajustar as margens, tamanho de papéis, espaçamento entre cabeçalho e texto, etc, usamos o pacote **geometry**. As configurações podem ser feitas, passando no parâmetro opcional do pacote, a lista dos itens na forma **chave=valor**, separado pela virgula. Também pode usar o comando `\geometry` após carregar o pacote.

Para ajustar margem superior (**tmargin**), margem inferior (**bmargin**), margem esquerda (**lmargin**, será de dentro se for frente/verso), margem direita (**rmargin**, será de fora se for frente/verso), basta colocar algo como

```
\usepackage[tmargin=2cm,bmargin=2cm,lmargin=2cm,rmargin=2cm]{geometry}
```

ou

```
\usepackage{geometry}
\geometry{tmargin=2cm,bmargin=2cm,lmargin=2cm,rmargin=2cm}
```

no **preamble** do documento.

Para ajuste de espaçamento entre linhas, usamos o pacote **setspace** que tem os comandos `\singlespacing` (espaçamento simples), `\onehalfspacing` (espaçamento um e meio) e `\doublespacing` (espaçamento duplo), além do ambiente **spacing** que permite qualquer espaçamento dentro dele.

Para que o documento fique em espaçamento um e meio, coloque

```
\usepackage{setspace}
\onehalfspacing
```

no `preamble` do documento.

Usuário da classe `memoir` devem colocar `\DisemulatePackage{setspace}` antes de carregar o `setspace`.

Para desabilitar a hifenização em todo documento ou ativar hifenização na fonte nono espaçado, poderá usar o pacote `hyphenat`.

```
\usepackage[none]{hyphenat}
```

no `preamble` desabilita hifenização em todo documento. Se não quer usar o pacote e quer desativar a hifenização, coloque

```
\hyphenpenalty=10000
\exhyphenpenalty=10000
```

no `preamble` do documento.

Para melhorar o ajuste de espaços, poderemos usar o pacote `microtype` que usa o recurso de micro tipografia. Por exemplo, reduzir ou ampliar levemente a fonte melhora o espaçamento. Por exemplo, podemos colocar

```
\usepackage[stretch=10]{microtype}
```

no `preamble`.

Cabeçalho no estilo de página `headings` coloca o título do capítulo e seção no cabeçalho em maiúsculo. Para tanto, usa-se o comando `\MakeUppercase` que não funciona para letras acentuadas diretamente (funciona para acentuação no modo `TEX`). Além disso, ele tenta converter inclusive as fórmulas, o que pode causar problemas. O pacote `textcase` permite sobrescrever o `\MakeUppercase` para evitar tais problemas, além de mais alguns recursos adicionais. Coloque

```
\usepackage[overload]{textcase}
```

no `preamble`.

Normalmente, a referência bibliográfica e índice remissivo não costumam ficar no sumário. Para que eles constem no sumário, use o pacote `tocbibind`. Usando a opção adicional, pode impedir que sumário, lista de figuras e de tabelas fiquem fora do sumário.

```
\usepackage[nottoc,notlof,notlot]{tocbibind}
```

no `preamble` efetua esta tarefa.

Note que no `LATEX`, o texto é justificado por padrão, mas não existe comando para justificar um trecho dentro do outro alinhamento. Uma saída é usar o `minipage`, mas `minipage` não permite quebrar entre páginas. Para resolver este problema, existe o pacote `ragged2e` que providencia o ambiente `justify` (e comando `\justify`) que pode ser usado como outros comandos de alinhamento de texto no parágrafo.

## 14.2 Estilo europeu

Em Alguns países da Europa, um parágrafo é separado pela outra com o espaçamento maior entre linhas em vez de indentação. Remover a indentação é feito simplesmente ajustando o valor de `\parindent` para zero, mas ajustar o espaçamento antes do parágrafo por `\parskip` e acertar alguns detalhes requer conhecimento extra. Para facilitar, existe o pacote `parskip` que, ao ser usado, ajustam de forma apropriada. Se não quer usar o pacote, coloque

```
\setlength{\parindent}{0pt}
\setlength{\parskip}{1ex plus 0.5ex minus 0.2ex}
```

no `preamble`, mas isto não ajustará tudo que o pacote faz.

Para produzir o estilo literário francês na qual a primeira letra do parágrafo ocupa mais de uma linha, tem o pacote `letrine`. O uso comum nos livros de literatura é usar o `letrine` no primeiro parágrafo do capítulo e manter o restante como normal. Veja o Exemplo 14.1 na qual assume que tem o `\usepackage{letrine}` no `preamble` do documento.

Exemplo 14.1: `ex14-letrine.tex`

```
\chapter{Um Capítulo}
\letrine{E}{xiste} um pacote chamado \texttt{letrine} que oferece o estilo literário francês
(estilo bíblico) na qual a primeira letra do parágrafo ocupa varias linhas.
Alguns livros literários usa este estilo somente para o parágrafo do primeiro capítulo,
mantendo outros parágrafos como sendo normal. O primeiro parâmetro é a letra (
primeira letra) que ocupará várias linhas, e o segundo parâmetro é a continuação dele até
finalizar a palavra, que será escrito em maiúsculo. O comando ainda aceita parâmetros
opcionais para ajustar quantas linhas vai ocupar, etc.
```

# Capítulo 1

## Um Capítulo

**E**XISTE um pacote chamado `letrine` que oferece o estilo literário francês (estilo bíblico) na qual a primeira letra do parágrafo ocupa varias linhas. Alguns livros literários usa este estilo somente para o parágrafo do primeiro capítulo, mantendo outros parágrafos como sendo normal. O primeiro parâmetro é a letra (primeira letra) que ocupará várias linhas, e o segundo parâmetro é a continuação dele até finalizar a palavra, que será escrito em maiúsculo. O comando ainda aceita parâmetros opcionais para ajustar quantas linhas vai ocupar, etc.

## 14.3 Ajustando o cabeçalho, títulos de capítulos e de figuras

Ajustar o cabeçalho diretamente não é simples. Assim, costumamos usar o pacote `fancyhdr` que permite personalizar o cabeçalho das páginas. Veja o Exemplo 14.2 usado para produzir estilo de páginas deste documento.

Exemplo 14.2: `ex14-fancyhdr.tex`

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}

\usepackage{calc} % para calculo de medidas
\usepackage{fancyhdr} % personalizar o cabeçalho

% \fancyheadoffset[LE,RO]{\marginparsep+\marginparwidth}

% redefine a saida do capitulo e da secao
% \pagestyle{fancy} redefine eles,
% entao deve colocar no final do ps@fancy
% para ter efeito. Usando \addto do babel
\makeatletter
\addto\ps@fancy{%
  \renewcommand{\chaptermark}[1]{%
    \markboth{\if@mainmatter\thechapter. \fi #1}%
    {\if@mainmatter\thechapter. \fi #1}}
  \renewcommand{\sectionmark}[1]{%
    \markright{\if@mainmatter\thesection. \fi #1}}
}
\makeatother

% redefinindo o estilo da pagina "fancy"
\fancyhf{} % limpa o cabecalho
% redefine o cabcalho. \nouppercase foi usado para eliminar conversao para maiusculo do
% sumario/bibliografia
\fancyhead[LE,RO]{\bfseries\thepage}
\fancyhead[LO]{\bfseries\nouppercase\rightmark}
\fancyhead[RE]{\bfseries\nouppercase\leftmark}

% redefine o "plain" (usado na primeira pagina do capitulo).
\fancypagestyle{plain}{%
  \fancyhead{} % get rid of headers
  \fancyhead[LE,RO]{\bfseries\thepage} % colocar enumeracao
  \renewcommand{\headrulewidth}{0.5pt} % com linha horizontal
}

```

```

% linha horizontal entre cabeçalho e corpo do documento
\renewcommand{\headrulewidth}{0.5pt}

% sem linha horizontal entre corpo de texto e rodape
\renewcommand{\footrulewidth}{0pt}

%%%%%%%%%% fim da redefinicao do cabeçalho %%%%%%%%%%%

% iniciar com estilo empty
\pagestyle{empty}
\begin{document}
...
\pagestyle{fancy} % inicia o cabeçalho personalizado
...
\end{document}

```

Note que o comando `\fancypagestyle` permite definir/redefinir estilo de páginas, exceto o `fancy` que é configurado diretamente, o que requer cuidados. Se usar o `\fancypagestyle{fancy}`, o  $\text{\LaTeX}$  pode parar no meio da compilação quando encontrar o `\pagestyle{fancy}`.

Para configurar o formato de título do capítulo e seções, usamos o pacote `titlesec`. Note que, para ajuste somente de capítulos, existe o pacote `fncychap` que permite escolher modelo entre alguns prontos com a opção do pacote (`Sonny`, `Lenny`, `Glenn`, `Conny`, `Rejne`, `Bjarne`, `Bjornstrup`) e ajustar as fontes com comandos tais como `\ChNumVar` e `\ChTitleVar`. Mas não consegue refazer o estilo como no `titlesec`.

Assim, para configuração mais refinado de títulos de capítulos e seções, costumam usar o `titlesec`. Este pacote, além de configuração refinado de títulos, também pode ser usado para configurar o cabeçalho. Note que, para configuração mais refinado de cabeçalhos, costuma usar o `fancyhdr`. Veja o Exemplo 14.3.

Exemplo 14.3: ex14-titlesec.tex

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}

\usepackage{calc} % para calculo de medidas

\usepackage[pagestyles]{titlesec} % para formatar titulo do capitulo e secao

% formatando o titulo do capítulo
\titleformat{\chapter}[display] % modo display
{\bfseries\Large} % fonte usado no titulo do capítulo
{\filleft\MakeUppercase{\chaptertitle} \Huge\thechapter} % Como colocar o nome
do capítulo
{4ex} % espaco entre nome do capítulo e titulo do capítulo
{\titlerule\vspace{2ex}\filright} % o que colocar antes do titulo do capitulo

```

```

[\vspace{2ex}\titlerule] % depois do titulo do capítulo

% \titlespacing*{\chapter}{0pt}{20pt}{16pt} % espacamento do titulo de capitulos

% Exemplo ajustado da documentação
% formatando o título da seção
\newcommand{\sectiontitlename}{Seção}

\titleformat{\section}[frame]% colocar moldura
{\normalfont}% fonte normal
{\filright\footnotesize
\enspace \MakeUppercase{\sectiontitlename}\enspace \thesection\enspace}% Como
colocar o nome/enumeração da seção
{8pt}% espaço antes do titulo da seção
{\Large\bfseries\filcenter}% antes do titulo da seção (ajustando fontes)

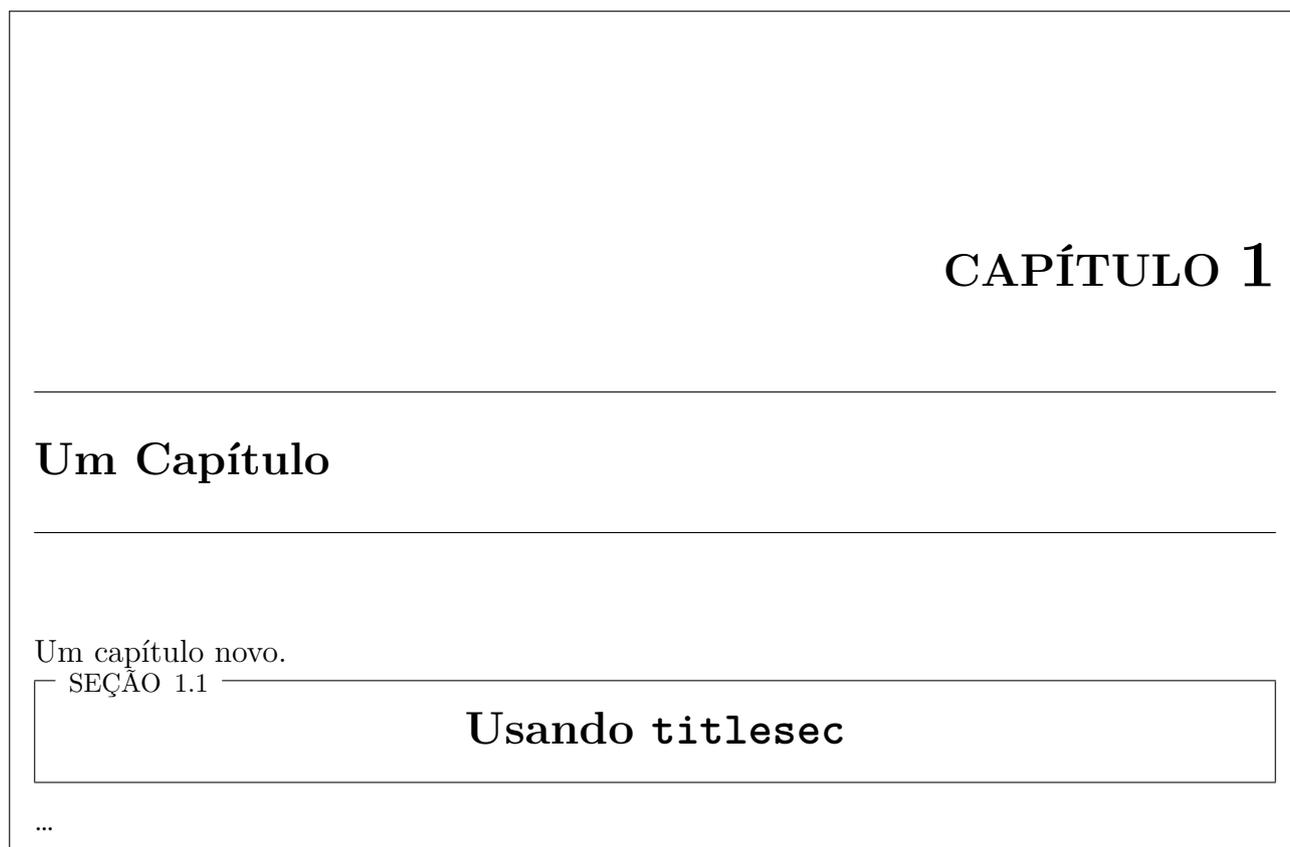
% \titlespacing{\section}{0pt}{*2}{*2}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% definindo o estilo do cabeçalho usando titlesec
% para configuração mais sofisticada, use fancyhdr

\newpagestyle{main}{ % Novo estilo de pagina
  \headrule
  \sethead[\thechapter. \scshape\chaptertitle]{}{}{\thechapter. \slshape\chaptertitle}
  \setfoot[\thepage]{}{\thepage}}
}

\pagestyle{empty}
\begin{document}
\pagestyle{main} % inicia o cabeçalho personalizado
\chapter{Um Capítulo}
Um capítulo novo.
\section{Usando \texttt{titlesec}}
\ldots
\end{document}

```



Note que a primeira página do capítulo será sempre do estilo `plain`. Então, se quer que a primeira página do capítulo seja diferente do `plain` padrão, deverá redefinir ele.

Para configurações mais complicadas de títulos de capítulos e seções, use a opção `explicit` na qual os textos de títulos só serão colocados se for referenciado diretamente.

Para usar o `titlesec` e `fancyhdr` ao mesmo tempo, deverá carregar o `titlesec` com opção `pagetypes` e antes do `fancyhdr`.

Para configurar a saída de títulos dos elementos flutuantes como figuras e tabelas que são produzidos pelo comando `\caption`, costuma usar o pacote `caption` que implementa o comando `\captionsetup` para configurar o formato de títulos das figuras e tabelas e também oferece versão “\*” do `\caption` que não será contabilizada (só coloca o título).

No Exemplo 14.4, será configurado que no `figure`, nome será `Imagem`, fonte do nome será em negrito, título em itálico, formato do nome é modo simples e nome com título será separado pelo ponto.

Exemplo 14.4: `ex14-caption.tex`

```
\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}

\usepackage{caption} % para formatar titulo do float (figura e tabela)
```

```

\captionsetup[figure]{name=Imagem,labelfont={bf}, textfont=it, labelformat=simple,
  labelsep=period}
\pagestyle{empty}
\begin{document}

\begin{figure}[hbp!]
\center
Figura aqui.
\caption{Novo título}\label{fig:título:novo}
\end{figure}

\clearcaptionsetup{figure}

\begin{figure}[hbp!]
\center
Figura aqui.
\caption{Título normal}\label{fig:título:normal}
\end{figure}
\end{document}

```

Figura aqui.  
**Imagem 1.** *Novo título*  
 Figura aqui.  
 Figura 2: Título normal

Para limpar as configurações, use o comando `\clearcaptionsetup{figure}`, onde `figure` pode ser `table`, se for da tabela.

Para criar configurações complexas que é difícil de ser ajustado com comandos acima, poderá usar o comando `\DeclareCaptionFormat`.

Por exemplo, se quer que nome da figura fique em maiúsculo negrito, separado por dois pontos e título em itálico, basta criar o estilo novo com

```

\DeclareCaptionFormat{meuestilo}{\MakeUppercase\bfseries #1}#2\textit{#3}\par}

```

e usar como `\captionsetup[figure]{format=meuestilo, labelsep=colon}`.

## 14.4 Links

Quando usa o pacote `hyperref`, sumário, índice remissivo, as referências cruzadas e citações ganham link automaticamente. Além disso, poderá configurar informações gerais do documento PDF gerado, tais como título, autor, palavras-chave, etc relacionado com o documento PDF.

A configuração do PDF pode ser feito por algo como segue.

...

```

\usepackage{hyperref}

% configurando o PDF
\hypersetup{
  bookmarks=true, % show bookmarks bar?
  unicode=true, % non-Latin characters in 'Acrobats bookmarks
% pdftoolbar=true, % show 'Acrobats toolbar?
% pdfmenubar=true, % show 'Acrobats menu?
% pdffitwindow=false, % window fit to page when opened
% pdfstartview={FitH}, % fits the width of the page to the window
  pdftitle={Pacote hyperref}, % title
  pdfauthor={Sadao Massago}, % author
  pdfsubject={Usado do pacote hyperref}, % subject of the document
% pdfcreator={Creator}, % creator of the document
% pdfproducer={Producer}, % producer of the document
% pdfkeywords={keyword1, key2, key3}, % list of keywords
% pdfnewwindow=true, % links in new PDF window
% colorlinks=false, % false: boxed links; true: colored links
% linkcolor=red, % color of internal links (change box color with linkbordercolor)
% citecolor=green, % color of links to bibliography
% filecolor=magenta, % color of file links
% urlcolor=cyan % color of external links
}
...

```

Note que, apenas carregar o `hyperref` já é suficiente para maioria dos casos, mas em geral, se estiver usando pacotes que ajustam os comandos de referências cruzadas ou citações, o pacote `hyperref` deve vir depois deles para não perder os links.

Além do link automático, poderá inserir links para sites, etc, usando o comando `\url`.

Veja `\url{https://en.wikibooks.org/wiki/LaTeX/Hyperlinks}` por exemplo.

Atualmente, o `url` é importante para ser colocado nas referências bibliográficas. O estilo moderno para `BIBTEX` costuma usar o campo `url` na qual o seu valor costuma ser colocado no arquivo de `BIBTEX` com o comando `\url` como em

```

...
url = \url{https://en.wikibooks.org/wiki/LaTeX/},
...

```

Obviamente, o arquivo `tex` correspondente deve usar o pacote `hyperref` ou o `url`.

Quando o texto onde clicar é diferente do endereço do link, poderá usar o `href` como em

```

\href{https://en.wikibooks.org/wiki/LaTeX/}{wikibooks, \LaTeX{}}

```

Também poderá criar os links manualmente, dentro do documento. Para isso, define a chegada do link com o comando `\hypertarget` e cria link para ele, com o comando `\hyperlink`. Veja o <https://en.wikibooks.org/wiki/LaTeX/Hyperlinks> para detalhes.

Quando usa as fórmulas nos títulos, aparece “lixo” no indicador (`bookmarks`) do PDF. Isto porque, no indicador, só o texto é aceito. Para evitar isso, existe o comando `\texorpdfstring` que coloca conteúdo de  $\LaTeX$  ou texto, dependendo de estar no documento ou no `bookmarks`.

Por exemplo,

```
\section{\texorpdfstring{$E=mc^2$}{E = mc ** 2}}
```

Usará  $E = mc^2$  para documento, mas `E = mc ** 2` para o indicador (`bookmarks`).

Quando adiciona um ítem manualmente no sumário através do comando `\addcontentsline`, o link do sumário pode não posicionar corretamente na página correta. Para o funcionamento correto (posicionamento correto) do link no sumário requer que o `\addcontentsline` deve ser chamado logo em seguida da mudança de capítulo ou seção. Caso não esteja, é só colocar `\phantomsection` oferecido pelo pacote `hyperref` antes do `\addcontentsline`.

Note que “|” na entrada do índice remissivo para inserir o símbolo “|” não funciona em conjunto com o pacote `hyperref`. A forma de contornar isso é definir um comando para “|” e usar na entrada do índice remissivo.

## 14.5 Trocando fontes

$\LaTeX$  dispõe de diversas fontes gratuitos para diagramar o documento. Em geral, usamos o `Latin Modern` que é extensão do `Computer Modern` para derivados de latim que tem letras acentuadas. Para isso, é só carregar o pacote `lmodern` com o comando

```
\usepackage{lmodern} % latin modern
```

no `preamble`.

Aqui, veremos algumas combinações que podem ser usadas, entre muitas opções.  
Concrete (texto e fórmula)

```
\usepackage{lmodern} % para complementar
\usepackage{ccfonts} % Concrete (serifa grossa?)
```

Fonte `Times` pode ser escolhido com

```
\verb+\usepackage{newtxtext,newtxmath}+
```

A fonte `Palatino` com

```
\verb+\usepackage{newpxtext,newpxmath}+.
```

A fonte `iwona` ideal para `poster` e `slides`.

```
\usepackage{lmodern} % para complementar
\usepackage[math]{iwona} % sans serif com transicao goosso fino,
```

Fonte Schoolbook L

```
\usepackage{lmodern} % para complementar
\usepackage{fouriernc} % schoobook L: Serifa grossa?
```

A fonte (estilo compatível com Times New Roman)

```
\usepackage{newtxtext} % para complementar
\usepackage{stix} % compativel com New Times Roman
```

Note que, para complementar a fonte como do `amssymb`, deve carregar antes do pacote de fontes desejadas. Quando vários pacotes de fontes são carregados e tiver conflitos, o que vale é do pacote carregado por último.

Também existem pacotes de fontes que acrescentam comandos para fontes adicionais. Por exemplo, o pacote `dsfont` adiciona o comando `\mathds` para fonte “negrito de quadro negro” alternativo usado dentro das fórmulas. Da forma análoga, o pacote `calligra` acrescenta o comando `\textcalligra` para textos na fonte caligráfica.

Para ver as fontes livres do L<sup>A</sup>T<sub>E</sub>X e os pacotes correspondentes, veja o site <http://www.tug.dk/FontCatalogue/>.

## 14.6 Figuras e similares

As vezes queremos que o elemento flutuante (figuras e tabelas) sejam inseridas exatamente no lugar onde foi colocado. Para isso, existe o pacote `float` que fornece a opção extra “H” nas figuras e tabelas na qual impede de auto posicionar (manter no lugar). Para usar, é só carregar o pacote no `preamble` e usar o “H” como opção de posicionamento das figuras e tabelas que querem que fiquem no lugar. Outro recurso do pacote `float` é alterar algumas configurações do elemento flutuante (como colocar moldura nas figuras), e criar um novo elemento flutuante. Veja o Exemplo 14.5

Exemplo 14.5: ex14-float.tex

```
\begin{figure}[H] % No lugar
\centering
\begin{tikzpicture}
\draw (0,0) -- (1,0) -- (0.5, 0.5) -- cycle;
\end{tikzpicture}
\caption{Figura em onde foi inserido}
\end{figure}
```



Figura 1: Figura em onde foi inserido

Para colocar subfiguras ou subtabelas, use o pacote `subcaption`. Tendo `\usepackage{subcaption}` no `preamble`, podemos produzir sub figuras e sub tabelas facilmente. Veja o Exemplo 14.6.

Exemplo 14.6: ex14-subcaption.tex

```
\begin{figure}[hbp!]
  \centering
  \begin{subfigure}[t]{0.45\linewidth}
    \centering
    Sub figura aqui.
    \caption{sub título 1}
  \end{subfigure}%
  ~
  \begin{subfigure}[t]{0.45\linewidth}
    \centering
    Segunda sub figura aqui.
    \caption{sub título 2}
  \end{subfigure}
  \caption{Uso de sub figura}
\end{figure}
```

Sub figura aqui.

(a) sub título 1

Segunda sub figura aqui.

(b) sub título 2

Figura 2: Uso de sub figura

Se a classe/pacote conflitar com `subcaption`, poderá optar pelo pacote `subfig`.

Para que o texto contorne as figuras ou tabelas, poderá usar o pacote `wrapfig` ou `floatflt`. O Exemplo 14.7 ilustra o uso de `wrapfig`.

Exemplo 14.7: ex14-wrapfig.tex

```
\begin{wrapfigure}{l}{0.3\linewidth}
\centering
\begin{tikzpicture}
\draw (0,0) rectangle (3,1);
\end{tikzpicture}
\caption{Figura com texto contornando}
\end{wrapfigure}
```

A figura deve ser colocado dentro do ambiente `wrapfigure`. No exemplo, “l” significa left ( esquerda) que posiciona a figura a esquerda (e texto a direita). Se colocar o “r” (right), a figura ficará a direita. O próximo parâmetro é a largura reservada para figura que foi `30\%` da largura da linha.

Note que são contadas junto as outras figuras.

Se pretende colocar tabela com texto em torno dele, use o ambiente `\texttt{wraptable}`, também do pacote `\texttt{wrapfig}`.



Figura 3: Figura com texto contornando

do pacote `wrapfig`.

A figura deve ser colocado dentro do ambiente `wrapfigure`. No exemplo, “l” significa left (esquerda) que posiciona a figura a esquerda (e texto a direita). Se colocar o “r” (right), a figura ficará a direita. O próximo parâmetro é a largura reservada para figura que foi 30% da largura da linha. Note que são contadas junto as outras figuras. Se pretende colocar tabela com texto em torno dele, use o ambiente `wraptable`, também

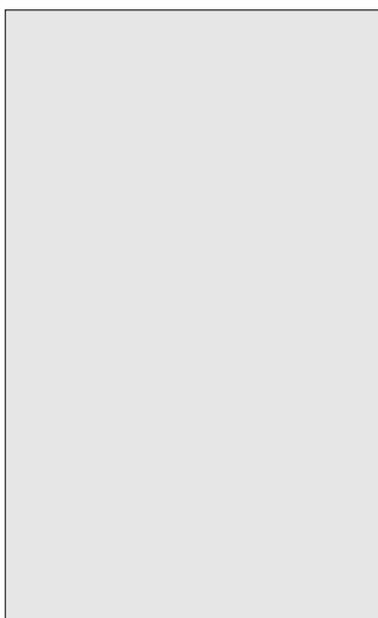
Quando tem figura ou table muito larga, precisará rotacionar a página. Isto pode ser feito pelo pacote `pdflscape`.

Para que uma página fique rotacionada, coloque dentro do ambiente `landscape`. Note que, para rotacionar tabelas longas que pode ocupar mais de uma página, deverá usar o pacote `rotating`. Veja o Exemplo 14.8.

#### Exemplo 14.8: ex14-pdflscape.tex

```
\begin{landscape}
\center
\begin{figure}[H]
\center
\begin{tikzpicture}
\draw[fill=gray!20] (0,0) -- (\linewidth,0) -- (\linewidth,5) -- (0, 5) -- cycle;
\end{tikzpicture}
\caption{Figura larga rotacionada}
\end{figure}
\end{landscape}
```

Figura: Figura larga rotacionada



## 14.7 Escrevendo medidas internacionais

Para escrever medidas, requer alguns cuidados tais como usar letra romana reta (por ser abreviatura da palavra), ter pequeno espaço entre valor e medida, entre outros. Por exemplo, 10 kg e não *10kg*, 5 l e não *5l* e assim por diante.

Para facilitar a escrever unidades de medidas corretamente no documento, o `siunitx` providencia comandos para escrever na unidade internacional.

Para produzir o número rapidamente, tem o comando `\num`. O comando `\ang` gera ângulos rapidamente. Para exemplos a seguir, será assumido que foi carregado o pacote `siunitx` no `preamble` Veja o Exemplo 14.9.

Exemplo 14.9: ex14-num.tex

### Números

```
\num{12345.67890} \\ % espacando de 3 em 3 casas
\num{12345,67890} \\ % “,” também pode ser usado para decimal
\num{1+-2i} \\ % “+-” vira $pm$
\num{.3e45} \\ % notacao cientifica
\num{1.654 x 2.34 x 3.430} % “x” vira $times$
```

### Ângulos

```
\ang{10} \\ % grau
\ang{5.3} \\ % grau com decimal
\ang{-1,5} \\ % “,” também pode ser usado como decimal
\ang{1;2;3} \\ % grau, minuto e segundo
\ang{;1} \\ % so o segundo
```

### Números

12 345.678 90

12 345.678 90

$1 \pm 2i$

$0.3 \times 10^{45}$

$1.654 \times 2.34 \times 3.430$

### Ângulos

$10^\circ$

$5.3^\circ$

$-1.5^\circ$

$1^\circ 2' 3''$

$1''$

O comando `\si` produz unidade de medida de acordo com o seu parâmetro. A configuração de saída pode ser efetuado pelo parâmetro opcional. O `\si` assim como `\num` e `\ang`, funciona

tanto no modo texto como modo matemático.

Em geral, coloca-se pequeno espaço entre valor e medida. O comando `\SI` automatiza isto, além de poder colocar símbolo antes do valor.

Ele tem a forma `\SI[opção]{valor}[símbolo anterior]{unidade}` Veja o Exemplo 14.10.

#### Exemplo 14.10: ex14-si.tex

Entrada literal (Neste modo, não há opção de formatação).

```
\si{kg.m/s^2} \\
\si{g_{polymer}~mol_{cat}.s^{-1}}
```

Entrada pelo macro (permite configurar a formatação usando opção do comando ou do parâmetro do pacote).

```
\si{\kilo\gram\per\square\second} \\
\si{\gram_{polymer}\mol_{cat}\per\second}
```

Mais exemplos\\

```
\si{\kilo\gram\metre\per\square\second} \\
\si{\gram\per\cubic\centi\metre} \\
\si{\square\volt\cubic\lumen\per\farad} \\
\si{\metre\squared\per\gray\cubic\lux} \\
\si{\henry\second}
```

Medidas com valor

```
\SI{1.23}{J.mol^{-1}.K^{-1}} \\ % modo textual
\SI{.23e7}{\candela} \\ % como macros
\SI{1.99}[R\$\]{\per\kilogram} \\ % com simbolo antes
\SI{1,345}{\coulomb\per\mole} % outro exemplo
```

Entrada literal (Neste modo, não há opção de formatação).

$\text{kg m/s}^2$

$g_{\text{polymer}} \text{mol}_{\text{cat}} \text{s}^{-1}$

Entrada pelo macro (permite configurar a formatação usando opção do comando ou do parâmetro do pacote).

$\text{kg s}^{-2}$

$g_{\text{polymer}} \text{mol}_{\text{cat}} / \text{s}$

Mais exemplos

$\text{kg m s}^{-2}$

$\text{g cm}^{-3}$

$\text{V}^2 \text{lm}^3 \text{F}^{-1}$

$\text{m}^2 \text{Gy}^{-1} \text{lx}^3$

H s

Medidas com valor

$1.23 \text{J mol}^{-1} \text{K}^{-1}$

$0.23 \times 10^7 \text{cd}$

$\text{R}\$1.99 \text{kg}^{-1}$

$1.345 \text{C mol}^{-1}$

Alguma das medidas padrão são: `\ampere`, `\candela`, `\kelvin`, `\kilogram`, `metre` ou `meter`, `\mole`, `\second`, `\litre`, `\celsius`, `\percent`, etc que são extensos. Para saber, consulte o manual.

Para conversão de medidas, `\giga`, `\mega`, `\kilo`, `\hecto`, `\deca`, `\deci`, `\centi`, `\milli`, `\micro`, `\nano`, etc. são disponíveis.

Exemplo: `\SI{20}{\kilo\gram}` e mesmo que `\SI{20}{\kilogram}`.

Para operar, tem o `per`, `square`, etc.

`\SI{30}{\kilo\meter\per\square\second}` é mesmo que `\SI{30}{\kilo\meter/\second^2}` no modo padrão, mas o segundo não consegue controlar a formatação por usar forma literal em “/” e “^2”.

A configuração geral de formatação pode ser feito pela opção do pacote ou pelo comando `\sisetup`, mas também pode especificar localmente como no Exemplo 14.11.

#### Exemplo 14.11: ex14-sisetup.tex

```
\SI[mode=text]{30}{\kilo\meter\per\square\second}
```

```
\SI[per-mode=power]{30}{\kilo\meter\per\square\second}
```

```
\SI[per-mode=symbol]{30}{\kilo\meter\per\square\second}
```

```
\SI[per-mode=fraction]{30}{\kilo\meter\per\square\second}
```

```

30 km s-2
30 km s-2
30 km/s2
30  $\frac{\text{km}}{\text{s}^2}$ 

```

## 14.8 Enfeites e código fontes

Para enfeitar o documento, como colocar moldura nos elementos, usamos pacotes adicionais, exceto molduras simples fornecidos pelo `\fbox`.

O pacote `fancybox` oferece caixas com molduras extras tais como `\shadowbox` (com sombra), `doublebox` (moldura dupla) e `\ovalbox` (moldura com quinas arredondadas).

Note que `\shadowbox` não permite controlar a cor da sombra. Se quer caixas personalizadas, deverá usar o pacote gráfico tal como o `tikz`. Veja o Exemplo 14.12.

Exemplo 14.12: `ex14-fancybox.tex`

```

\shadowbox{Caixa com sombra}

\doublebox{Caixa com moldura dupla}

\ovalbox{Caixa com quina arredondada}

\Ovalbox{Outra caixa com quina arredondada}

```

Caixa com sombra

Caixa com moldura dupla

Caixa com quina arredondada

Outra caixa com quina arredondada

Lembre-se que se o conteúdo tiver parágrafo, deverá usar em conjunto com o `minipage`. Também permite colocar conteúdo ou moldura na página com comandos tais como `\fancyput`, `\thisfancyput`, `\fancypage` e `\thisfancypage`.

Por exemplo, o código

```

\thisfancypage{%
\setlength{\fboxsep}{8pt}%
\setlength{\shadowsize}{8pt}%
\shadowbox}{%

```

acrescenta moldura com sombra na página atual.

Para colocar moldura nos conteúdos longos que pode ocupar mais de uma página, podemos usar o pacote `framed`. Para colocar cor de fundo, deverá definir a cor do fundo `shadecolor`. Para isso, é recomendado usar o pacote `xcolor` visto na Seção 15.1 do Capítulo 15 (não é o pacote `color`) que tem facilidade de manipulação de cores. Assim, para executar o código do Exemplo 14.13, deverá carregar tando o pacote `framed` como o `xcolor`.

Exemplo 14.13: ex14-framed.tex

```
\begin{framed}
O texto com moldura pode ser produzido facilmente com o pacote \texttt{framed}.
\end{framed}

\colorlet{shadecolor}{black!15} % cor do fundo usando xcolor
\begin{shaded*}
O texto com fundo pode ser produzido facilmente com o pacote \texttt{framed}.
\end{shaded*}

\begin{leftbar}
O ambiente \texttt{leftbar} coloca traço no lado esquerdo do texto.
\end{leftbar}
```

O texto com moldura pode ser produzido facilmente com o pacote `framed`.

O texto com fundo pode ser produzido facilmente com o pacote `framed`.

■ O ambiente `leftbar` coloca traço no lado esquerdo do texto.

O comando `\colorlet` do pacote `xcolor` permite definir cor a partir das cores existentes. `black!15` é 15% de preto.

Note que, diferente das caixas padrão do  $\text{\LaTeX}$  ou do pacote `graphicx`, os ambientes do `framed` podem quebrar em linhas e em páginas.

Para indicar as correções a serem efetuadas, é útil ter linhas enumeradas. Para enumerar as linhas do documento, usa-se o pacote `lineno`.

Para que as fórmulas também sejam enumeradas, use

```
\usepackage[mathlines]{lineno}
```

no `preamble`. Os comandos `\linenumbers` e `\nolinenumbers` são usados para ativar/desativar as enumerações. Se pretende enumerar todo documento, coloque `\linenumbers` no `preamble`. Para enumerar a cada “<n>” linhas, use `\modulolinenumbers[<n>]`. Veja o Exemplo 14.14.

Exemplo 14.14: ex14-lineno.tex

```

\linenumbers % ativa a enumeracao das linhas
\modulolinenumbers[3] % enumerar de 3 em 3

\lipsum[1] % preencher

\nolinenumbers % desativa a enumeração das linhas
\lipsum[1] % 1 paragrafos

% enumerando so um trecho.
\resetlinenumber % reset the line number
\modulolinenumbers[1] % enumerar todas
\begin{linenumbers}
  \lipsum[2] % 2 paragrafos
\end{linenumbers}

```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

1 Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem  
 2 non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet,  
 3 tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi.  
 4 Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque  
 5 a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus  
 6 mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus  
 7 luctus mauris.

No Exemplo 14.14 foi usado o comando `\lipsum` que gera textos aleatórios em grego para preencher as páginas para testes. Logo, precisa carregar o pacote `lipsum` no `preamble` para que o exemplo funcione.

Existem vários pacotes que permitem colocar conteúdos em todas páginas, como a marca d'água. Um deles é o `eso-pic`.

Por exemplo, o código

```
% marca d'agua
\AddToShipoutPicture{%
  \AtPageLowerLeft{%
    \unitlength=\paperwidth
    \begin{picture}(1.0,1.0)
      \put(0.3,0.5){\rotatebox{45}{%
        \scalebox{5}{\color{black!25}Rascunho}}}%
    }
  \end{picture}
}%
}%
```

após carregar o pacote `eso-pic`, coloca a palavra “Rascunho” rotacionado por 45° em todas as páginas como marca d'água. Note que, se quer colocar a imagem externa, basta usar o `\includegraphics`.

Para limpar o fundo, usa-se o `\ClearShipoutPicture`.

Para colocar código fonte de um programa, costuma usar o pacote `listings` que implementa o ambiente `lstlisting` que formata de acordo com a linguagem de programação escolhida.

Ele já vem com configurações para diversas idiomas, tais como Ada, Algol, Assembler, awk, bash, Basic, C#, C++, C, Cobol, Delphi, Fortran, Gnuplot, HTML, Java, Lisp, Logo, Lua, Make, Mathematica, Matlab, Metapost, Modula-2, Objective C, Octave, Pascal, Perl, PHP, Prolog, Python, R, Ruby, SAS, Scilab, sh, SQL, TeX, XML, entre vários outros. Se a linguagem pretendida não estiver na lista, poderá definir novos idiomas.

Para o seu uso, costuma efetuar configurações iniciais no `preamble` com o comando `\lstset` e no ambiente de `lstlisting`, efetuar mais alguns ajustes se necessários. Veja o Exemplo 14.15

Exemplo 14.15: `ex14-listings.tex`

```
\begin{lstlisting}[language=Pascal,caption={Alô pessoal}]
program teste;
{ Programa de teste }
begin
  writeln('Alo, pessoal!');
end.
\end{lstlisting}
```

Listing 1: Alô Pessoal

```
program teste;
{ Programa de teste }
begin
```

```
writeln('Alo, pessoal!');
end.
```

A formatação de saída depende da configuração do `lstlisting`. A saída do Exemplo 14.15 foi pela configuração

```
\lstset{
  numbers=none,
  breaklines=true,
  breakautoindent=true,
  aboveskip=2ex,
  columns=fullflexible, % para poder copiar código do PDF
  frame=lines, %none, single, line
}
```

Note o uso de `columns=fullflexible` para que o código fonte copiado do PDF não fique bagunçado.

Para o arquivo externo, podemos usar o comando `\lstinputlisting` que carrega o código diretamente do arquivo fonte.

Note que `listings` não tem suporte nativo a `utf8`. Portanto, quando usa os caracteres acentuadas, deverá tomar cuidados. Para o `\lstinputlisting` funcione no `utf8`, basta carregar o pacote `listingsutf8`, mas para o código colocado diretamente no ambiente `lstlisting`, deverá criar tabela de caracteres unicode correspondente. Para não ter tal trabalho, poderá executar no  $X_{\text{L}}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  ou  $\text{L}^{\text{u}}\text{a}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  em que tem total suporte a unicode.

Quando queremos colocar o trecho do código de  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  juntamente com a sua saída, o pacote `showexpl` é prático. Ele usa o pacote `listings` para listar o código fonte. A configuração inicial é feito com `\lstset{explpreset={...}}` no `preamble`. O ambiente `LTXexample` aceitam os parâmetros do `lstlisting` e também o `pos` e `preset` que especificam a posição onde colocar a saída, e a configuração inicial antes de executar o código. Veja o Exemplo 14.16.

Exemplo 14.16: ex14-showexpl.tex

```
\begin{LTXexample}[pos=b,caption={\LaTeX}]
% comentario
Teorema de Pitágoras afirma que
\[ a^2 = b^2+c^2 \]
\end{ltlisting}
```

Listing 1: L<sup>A</sup>T<sub>E</sub>X

<pre>% comentario Teorema de Pitágoras afirma que \[ a^2 = b^2 + c^2 \]</pre>	<div style="border: 1px solid black; padding: 10px; margin: 0 auto; width: 80%;"> <p>Teorema de Pitágoras afirma que</p> <math display="block">a^2 = b^2 + c^2</math> </div>
---	--

O valor de “pos” que determina onde ficará a saída do resultado do código em relação ao do código fonte, podem ser “t”, “b”, “l”, “r”, “o” ou “i”, que são abreviaturas de “top” (encima), “bottom” (embaixo), “left” (esquerda), “right” (direita), “outer” (lado de fora) e “inner” (lado de dentro).

Note que o `\usepackage` e similar estão desabilitados dentro do ambiente `LATEXexample` e se o código precisar, deverá estar carregado no `preamble` do documento principal. Agora outro exemplo, supondo que `amssymb` está carregado no documento principal (para usar `\mathbb`). Veja o Exemplo 14.1.

Exemplo 14.1: ex14-showexpl-completo

```
\begin{LATEXexample}[pos=b,caption={\LaTeX{ } completo}]
\documentclass{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\usepackage{amssymb,amsmath}
\newcommand{\sen}{\mathrm{sen}}
\begin{document}
% comentario
\[ \forall \theta \in \mathbb{R}, \cos^2\theta + \sen^2\theta = 1 \]
\end{document}
\end{LATEXexample}
```

Listing 2:  $\text{\LaTeX}$  completo

```

\documentclass{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\usepackage{amssymb,amsmath}
\newcommand{\sen}{\mathrm{sen}}
\begin{document}
% comentario
\[ \forall \theta \in \mathbb{R}, \cos^2 \theta + \sen^2 \theta = 1 \]
\end{document}

```

$$\forall \theta \in \mathbb{R}, \cos^2 \theta + \sen^2 \theta = 1$$

## 14.9 Mais sobre referências bibliográficas

Na citação original do  $\text{\LaTeX}$ , usa-se o mesmo padrão tanto para caso de fazer parte do texto (citação textual) ou não fazer parte do texto (citação dentro de parênteses). Para fazer esta distinção e controlar melhor a forma de citação, foi desenvolvido o pacote chamado `natbib`.

O pacote `natbib` possui comandos `\citet`, usado para citações textuais e `\citep` para citações entre parênteses (ele já coloca parênteses automaticamente).

Além disso, o `natbib` permite configurar a formatação de citações. Veja o Exemplo 14.2

Exemplo 14.2: ex14-natbib.tex

Para que a citação faça parte do texto (estilo textual), usa-se o comando `\verb+\citet+` como em “Por exemplo, `\citet{IndianTUG:2000, Goossens:94}` explica como usar o BibTeX”.

Para que ele não faça parte do texto, usa-se o `\verb+\citep+` Como em “`\ldots` recursos avançados de BibTeX `\citep[Cap.~13]{Goossens:94}`”.

Para que a citação faça parte do texto (estilo textual), usa-se o comando `\citet` como em “Por exemplo, Team (2000); Goossens et al. (1994) explica como usar o BibTeX”.  
Para que ele não faça parte do texto, usa-se o `\citep` Como em “...recursos avançados de BibTeX (Goossens et al., 1994, Cap. 13)”.

A saída da referência correspondente ao código do Exemplo 14.2 no caso do estilo `apalike` é

## Referências Bibliográficas

[Goossens et al., 1994] Goossens, M., Mittelbach, F., and Samarin, A. (1994). *The L<sup>A</sup>T<sub>E</sub>X Companion*. Adilson–Wesley, Reading, MA.

[Oeticker et al., 2002] Oeticker, T., Partl, H., Hyna, I., and Schlegl, E. (2002). *The Not So Short Introduction to L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>*. Comprehensive T<sub>E</sub>X Archive Network.

[Team, 2000] Team, T. (2000). *Online Tutorials on L<sup>A</sup>T<sub>E</sub>X*. Indian T<sub>E</sub>X User Group. <http://www.tug.org/tutorials/tugindia/>.

O `\cite` tradicional também estará disponível, funcionando como `\citet`.

Usando a opção do pacote ou comandos apropriados, poderemos configurar o estilo de citações.

Note que nos estilos antigos, os campos tais como “url” e “doi” não serão usadas (logo, não aparecem nas referencias). Para que URL apareça na referência desses estilos antigos, colocavam no campo “note”.

Para estilos mais modernos como do `natbib`, o URL deve ser colocado no campo “url” e “note” pode ser usado para outra finalidade tal como colocar a data de último acesso.

Para criar próprio arquivo de estilo `bst`, comece gerando um modelo com `makebst` do `custom-bib`.

Estilo padrão do `natbib` são `plainnat`, `abbrvnat`, `unsrtnat`. Também suporta vários `bst` de outros pacotes com estilo `autor-ano`. `apalike`, `newapa`, (`chicagochicago`, `named`, etc do `chicago`), (`agsm`, `dcu`, `kluwer`, etc do `harvard`), (`astron`, `apa`, `humanbio`, etc do `astron`), (`authordate<n>` e `aaai-named` do `authordate<n>`).

Note que `bst` não compatíveis com `autor-ano`, como o estilo padrão do `BIBTEX` será considerado como numérico.

O `biblatex` suporta o arquivo `bib` em `utf8`, permitindo que tenha citações dos livros em caracteres fora do `latin` e também suporta a localização da referência bibliográfica.

Outro ponto positivo é ter maior facilidade de elaborar arquivo de estilos.

O `biblatex` não é compatível com maioria dos pacotes específicos para `BIBTEX`. Também terá problemas com o pacote `titlesec` quando usar as opções `refsection`, `refsegment` ou `citereset` no `biblatex`.

Note que, caso não tenha estilo de referência bibliográfica desejado, o truque de copiar o conteúdo do arquivo `bb1` (extensão `bb1`) e ajustar, mas isto não funciona no `biblatex`, pois ele não usa o ambiente `thebibliography`. Isto vale mesmo que use `BIBTEX` como `backend`.

Como nem toda revista dispõe de estilos para `biblatex` (mesmo tendo estilo para `bibtex`), fique atento nesta parte para decidir se vai usar o `BIBTEX` direto, ou usar o `biblatex`.

Quando carrega o pacote `biblatex`, efetua algumas configurações.

```
\usepackage[
  backend=biber,
  bibencoding=utf8,
  % natbib=true, % natbib compatible mode (citep, citet, etc)
  style=authoryear,
]{biblatex}
```

Carrega o `biblatex` configurado para usar `biber` como processador de referências bibliográficas. Se usar o `bibtex` em vez do `biber`, perderá os recursos estendidos tais como suporte a `utf8` e localização da referência bibliográfica. Também foi indicado que o arquivo `bib` está em `utf8`, podendo conter caracteres unicode (fora do `latin`, por exemplo). A opção `natbib` está comentado, mas se ativar, terá disponibilidade para usar comandos `\citet` e `\citep` similar a do `natbib`. Finalmente, o estilo escolhido é `authoryear`.

Para indicar os arquivos `bib`, usa-se o comando `\addbibresource{<arquivo bib>}` que costuma estar no `preamble`, onde `<arquivo bib>` é nome do arquivo `bib`. Caso quer indicar vários arquivos `bib`, deverá colocar um `\addbibresource` para cada arquivo `bib`.

Após efetuar citações normalmente no documento, coloca-se o comando `\printbibliography` no lugar onde quer que apareça a referência bibliográfica.

Se quer que a referência bibliográfica conste no sumário, coloque

```
\phantomsection % se estiver usando o hyperref
\printbibliography[heading=bibintoc]
```

onde `\phantomsection` é necessário somente quando usa o pacote `hyperref` para que o link do sumário para a referência bibliográfica aponte para página correta.

O estilo padrão do `biblatex` são `numeric`, `alphanumeric`, `authoryear`, `authortitle`, `verbose`, `reading` e `draft`, mas existem muitos outros disponíveis em <https://ctan.org/tex-archive/macros/latex/exptl/biblatex-contrib>.

Para quem pretende criar o estilo novo, note que no `biblatex` deve implementar tanto o estilo de referências (como será impresso na referências) com o estilo de citações (como aparece dentro do texto).



# Capítulo 15

## Gráfico e Diagramas

Os recursos do ambiente `picture` do  $\text{\LaTeX}$  é bastante limitado, mesmo que use o pacote `pict2e`. Assim, costumamos usar os pacotes fora do `base` e `required` como o `tikz`. Neste capítulo, vamos estudar alguns pacotes relacionados aos gráficos, que não são da `base` e `required`.

### 15.1 Misturando cores

O pacote de cores mais recomendados é o `xcolor` que estende a funcionalidade do pacote `color`. Uma das extensões mais importantes do `xcolor` é capacidade de especificar a quantidade de cores e permitir misturar cores.

Para misturar cores, use o “!” seguido de percentual. Por exemplo, `\color{blue!30}` é 30% de azul.

No caso de `\color{red!30!yellow}` é 30% de vermelho e restante (70%) de amarelo.

Em `\color{blue!20!black!30!green}` é 20% de azul, 50% de preto do restante (do 80% que é 40%) e restante (40%) de verde.

Para definir nova cor a partir do existente, use o comando `\colorlet`.

Por exemplo, `\colorlet{mygreen}{green!80!yellow}` define `mygreen` como sendo a mistura de 80% de verde e 10% de amarelo. Veja o Exemplo 15.1.

Exemplo 15.1: `ex15-xcolor.tex`

```
\colorlet{mygreen}{green!80!yellow} % definindo cor a partir do existente

\colorbox{blue!30}{\hspace*{1cm}}

\colorbox{red!30!yellow}{\hspace*{1cm}}

\colorbox{blue!20!black!50!green}{\hspace*{1cm}}

\colorbox{mygreen}{\hspace*{1cm}}
```



A seguir, cores básicas do pacote `xcolor`.

Tabela 15.1: cores aceitos em todos drivers no pacote `xcolor`

black		blue		brown		cyan	
darkgray		gray		green		lightgray	
lime		magenta		olive		orange	
pink		purple		red		teal	
violet		white		yellow			

Dependendo da especificação do `driver`, terão muito mais cores, mas é mais prático trabalhar com cores básicas e suas misturas do que lidar com grande quantidade de nomes para cores.

## 15.2 Criando ilustrações gráficas

Para criar ilustrações gráficas, um dos mais indicados é o pacote `tikz`[Tan].

O pacote `tikz` possui muitas extensões que podem ser carregados pelo comando `\usetikzlibrary`.

Nos exemplos a seguir, será assumido que tem o código do Exemplo 15.2 no `preamble`.

Exemplo 15.2: `ex15-tikz-preamble.tex`

```
\usepackage{tikz} % pacote gráfico
\usetikzlibrary{babel} % para compatibilidade com o pacote babel, requerido por algumas
bibliotecas como o cd.

\usetikzlibrary{calc} % calc eh para efetuar calculos matematicos ou expressoes em
coordenadas

\usetikzlibrary{through} % circulo passando por ponto, por exemplo.

\usetikzlibrary{patterns} % preenchimentos

\usetikzlibrary{intersections} % intersecao entre caminhos

\usetikzlibrary{matrix} % matriz no tikz

\usetikzlibrary{cd} % diagrama comutativa
```

Os comandos gráficos sempre termina com ponto e vírgula. Ao compilar o código sem ter terminado com ponto e virgula no trecho de `tikz`, pode travar o  $\text{\LaTeX}$ . No caso de estar usando

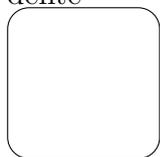
o LyX que não salva antes de compilar, é importante que salve manualmente antes de compilar. Note que o “instant preview inset” do LyX não trava. Veja o Exemplo 15.3 para começar.

Exemplo 15.3: ex15-tikz:basico.tex

Poderá desenhar no modo `\texttt{inline}` como em `\tikz \draw[fill] circle (2pt);` ou `\tikz{\draw (0,0) -- (1.5,0);}` e também como figura independente

```
\begin{tikzpicture}
\draw[rounded corners=8pt] (0,0) -- (0,2) -- (2,2) -- (2,0) -- cycle;
\end{tikzpicture}
```

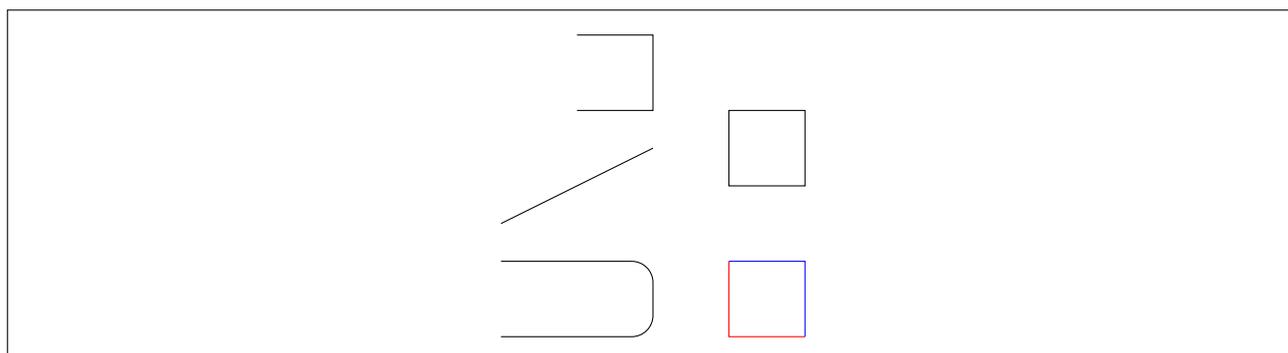
Poderá desenhar no modo `inline` como em `•` ou `_____` e também como figura independente



O comando `\draw` desenha o elemento. Como desenhar pode ser configurado com o parâmetro opcional. “--” indica que vai ligar os pontos a esquerda com da direita com uma linha. Cores podem ser passado no parâmetro opcional do `\draw`. Veja o Exemplo 15.4.

Exemplo 15.4: ex15-tikz-draw.tex

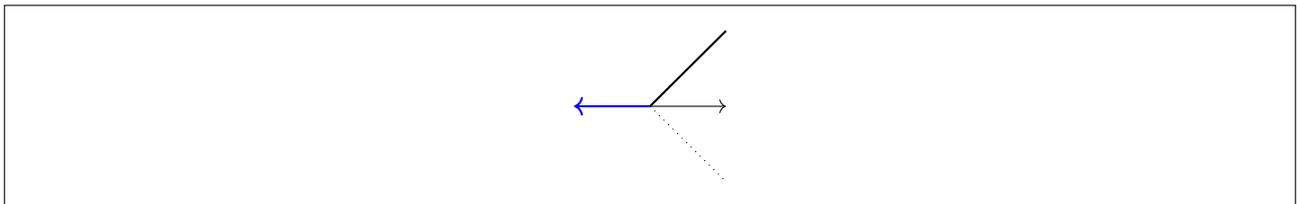
```
\begin{center}
\begin{tikzpicture}
\draw (-1,-0.5) -- (1,0.5); % segmentos
\draw (0,1) -- (1,1) -- (1,2) -- (0,2); % Linhas poligonais
\draw (2,0) -- (3,0) -- (3,1) -- (2,1) -- cycle; % Linhas poligonais fechadas
\draw[rounded corners=8pt] (-1,-1) -- (1,-1) -- (1,-2) -- (-1,-2); % Quinas arredondadas
\draw[red] (2,-1) |- (3,-2); % Conectando com linha vertical–horizontal
\draw[blue] (2,-1) -| (3,-2); % Conectando com linha horizontal–vertical
\end{tikzpicture}
\end{center}
```



Quando tem mais de um parâmetro opcional, coloque separado pela vírgula. Veja Exemplo 15.5.

Exemplo 15.5: ex15-tikz-parametro.tex

```
\begin{center}
\begin{tikzpicture}
% Espessuras: very thin, thin, thick, very thick
\draw[thick] (0,0) -- (1,1);
% Estilo: dashed, dotted
\draw[dotted] (0,0) -- (1,-1);
% Setas ->, <-, <->, etc
\draw[->] (0,0) -- (1,0);
% Combinadas
\draw[color=blue,thick,->] (0,0) -- (-1,0);
\end{tikzpicture}
\end{center}
```



Podemos definir e usar coordenadas, assim como efetuar alguns cálculos com coordenadas. Veja o Exemplo 15.6.

Exemplo 15.6: ex15-tikz-coordenadas.tex

```
\begin{center}
\begin{tikzpicture}
\coordinate (A) at (0,0);
\coordinate (B) at (1,0);
\coordinate (C) at (1,1);

\draw (A) -- (B) -- (C) -- cycle;

% interpolacao linear das coordenadas
% $(A)!t!(C)$ sera interpretado como sendo  $(1-t)*(A)+t*(C)$ 
\draw[dotted] (A) -- ($(B)!0.5!(C)$);

% Interpolação linear com rotação
% rotacionado por angulo de 15 em torno de (B)
\coordinate (D) at ($(B)!1!15:(C)$);

\draw[dashed] (A) -- (D);
\end{tikzpicture}
\end{center}
```

```

% projeção ortogonal
% $(A)!(B)!(C)$ sera pe do perpendicular abaixado de (B)
% para a reta determinado pelos pontos (A) e (C)
\coordinate (H) at ($(A)!(B)!(C)$);

\draw (A) --- (B) --- (C) --- cycle;

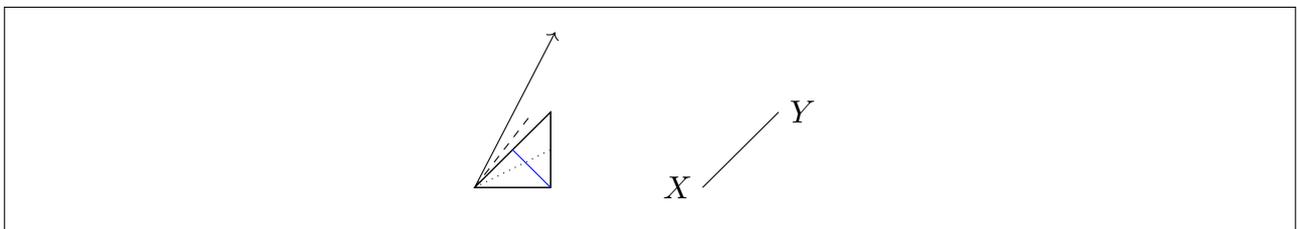
\draw[blue] (B) -- (H);

% Coordenada com rotulos (ja desenha o rotulo quando define)
\coordinate[label=left:$X$] (X) at (3,0);
\coordinate[label=right:$Y$] (Y) at (4,1);
\draw (X) -- (Y);

% Coordenada polar.
% As coordenada polar é dado por (angulo:raio).
% O ângulo é em graus.
\coordinate (u) at (45:1); % (angulo:raio)
\coordinate (v) at (90:2);

% combinação linear das coordenadas
% Note que devem ficar delimitados pelo $
\draw[->] (A) -- ($1.5*(u)+0.5*(v)$);
\end{tikzpicture}
\end{center}

```



Também podemos obter intersecções. Para pintar dentro, podemos usar o comando `\fill` (ou `\draw` com opção) Veja o Exemplo 15.7.

Exemplo 15.7: ex15-tikz-intersection.tex

```

\begin{center}
\begin{tikzpicture}
% desenhando e definindo coordenadas
\draw (0,0) coordinate (A) -- (2,3) coordinate (B)
      (0,2) coordinate (C) -- (3,1) coordinate (D);
\fill[red] (intersection of A--B and C--D) circle (2pt);

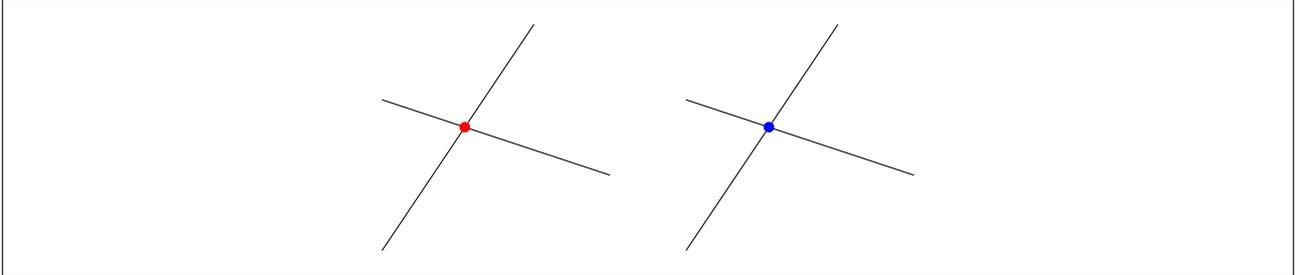
% Outra forma de obter intersecção.
\draw (4,0) coordinate (E) -- (6,3) coordinate (F)
      (4,2) coordinate (G) -- (7,1) coordinate (H);
\fill[blue] (intersection cs:

```

```

first line={ (E) -- (F) },
second line={ (G) -- (H) } circle (2pt);
\end{tikzpicture}
\end{center}

```



Círculos, elipses e arcos também podem ser desenhados. Veja o Exemplo 15.8.

Exemplo 15.8: ex15-tikz-circle.tex

```

\begin{center}
\begin{tikzpicture}
\draw[red] (-1.5,0) circle (0.5); % circolo

\coordinate [label=left:$A$] (A) at (0,0);
\coordinate [label=right:$B$] (B) at (1.25,0.25);
\draw (A) -- (B);
% Circulo passando por (B), com centro em (A)
\node [draw,circle through=(B),label=above:$c$] at (A) {};

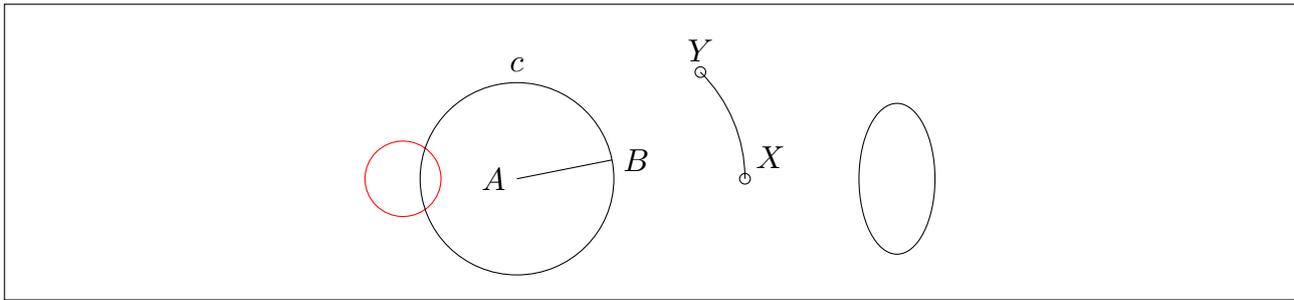
\coordinate (X) at (3,0);
\def\localRadius{2}
\def\localAngle{45}

% fazendo circulo na ponta inicial e rotulando como $X$
\draw (X) circle (2pt) node[anchor=south west] {$X$};

% faz arco e os circulos na ponta final e rotula como $Y$
\draw (X) arc (0:\localAngle:\localRadius) circle (2pt) node[anchor=south] {$Y$};

% Elipse: Atento pelo uso de "and" quando indica os raios da elipse
\draw (5,0) ellipse (0.5 and 1.0);
\end{tikzpicture}
\end{center}

```



Também podemos desenhar o gráfico das funções Veja Exemplo 15.9.

Exemplo 15.9: ex15-tikz-grafico.tex

```

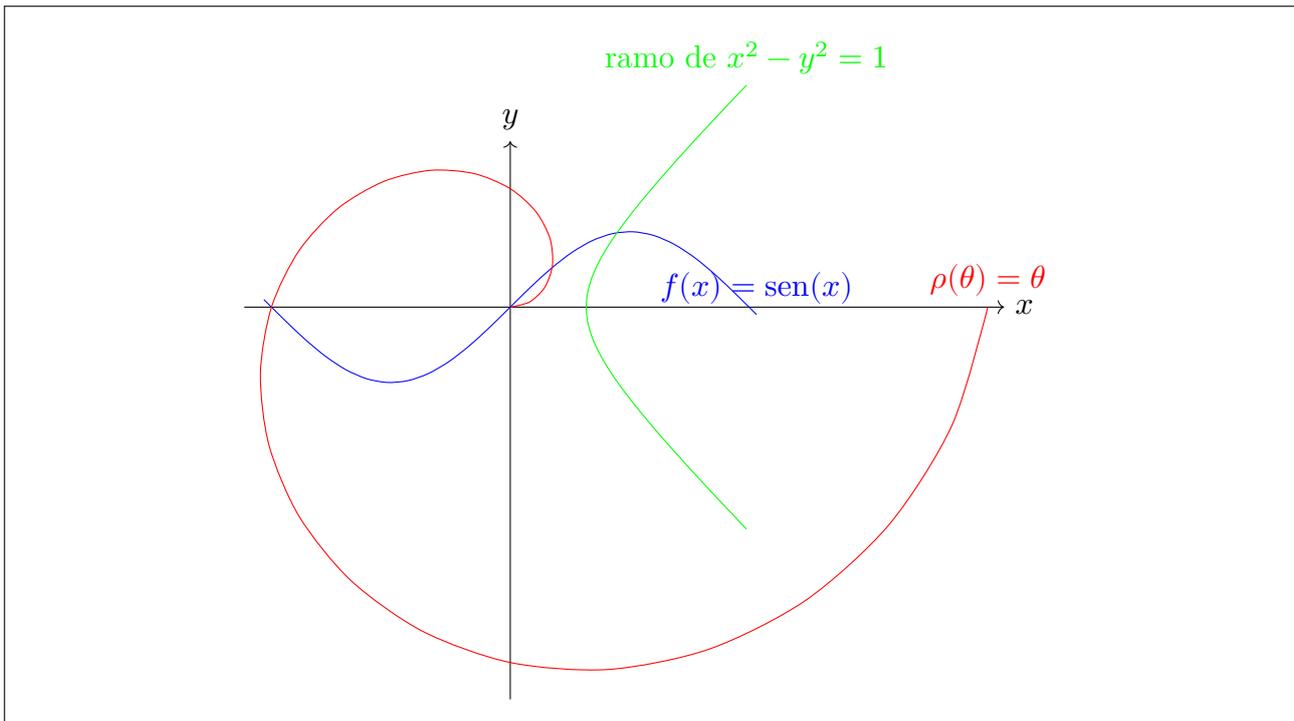
\begin{center}
\begin{tikzpicture}[domain=-3.2:3.2]
% \draw[dotted] (-3.2,-1.2) grid (3.2,1.2);
\draw[->] (-3.5,0) -- (6.5,0) node[right] {$x$};
\draw[->] (0,-5.2) -- (0,2.2) node[above] {$y$};

% O angulo padrao sera em graus. Para usar em radianos, acrescente " r" no parametro da
% funcao trigonometrica
\draw[smooth,color=blue,domain=-pi-0.1:pi+0.1] plot (\x,{sin(\x r)}) node[above] {$f(x) = \mathbf{sen}(x)$};

% Curvas em coordenada polar. O angulo padrao sera em graus. A funcao deg() converte
% radiano para grau (mesmo que sufixo r)
\draw[smooth,color=red,domain=0:2*pi] plot ({deg(\x)}:\x) node[above] {$\mathbf{\rho}(\mathbf{\theta}) = \mathbf{\theta}$};

% curvas parametricas.
% podera especificar o nome da variavel em vez de usar o padrao \x
\draw[smooth,color=green,domain=-1.8:1.8,variable=\t] plot ({cosh(\t)},{sinh(\t)}) node[above]{$\text{ramo de } x^2-y^2=1$};
\end{tikzpicture}
\end{center}

```



Também podemos desenhar gráficos da função tabelada. No Exemplo 15.10, está usando a função tabelada no arquivo de texto externo. Ele também faz o uso do laço `\foreach` para marcar valores sobre eixos.

Exemplo 15.10: ex15-tikz-funcao-tabelada.tex

```

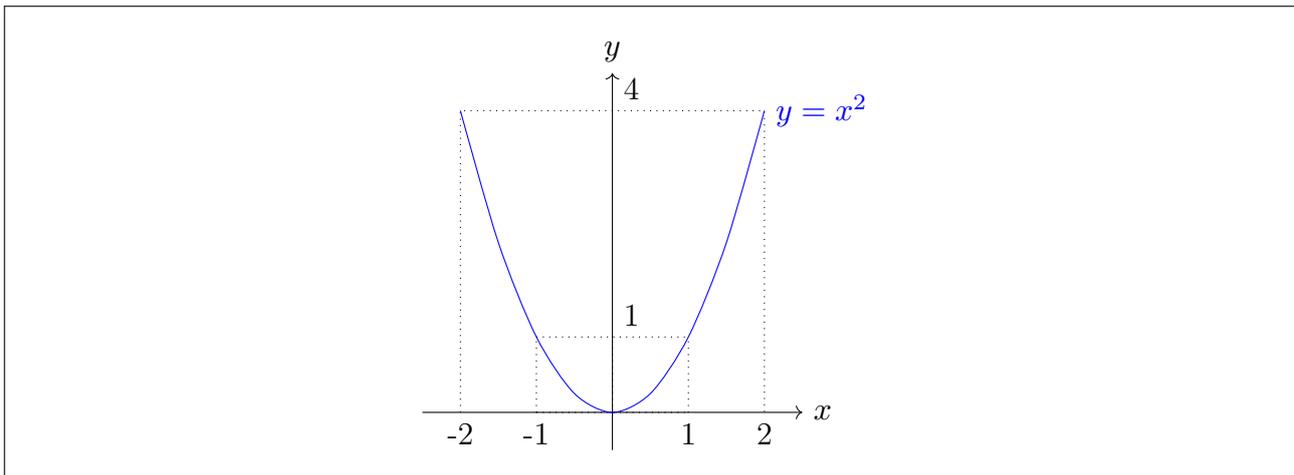
\begin{center}
\begin{tikzpicture}
  % eixos
  \draw[>-] (-2.5,0) -- (2.5,0) node[right] {$x$};
  \draw[>-] (0,-0.5) -- (0,4.5) node[above] {$y$};

  % grid
  \draw[dotted] (-1,0) grid (1,1);
  \draw[dotted] (-2,0) -- (-2,4) -- (2,4) -- (2,0);

  % efetuando laço nos parametros para marcar escala
  \foreach \x in {-2,-1,1,2} % sobre eixo x
    \node at (\x, 0) [below] {\x};
  \foreach \y in {1,4} % sobre eixo y
    \node at (0,\y) [anchor=south west] {\y};

  % grafico tabelado no arquivo externo.
  % Note que tikz não reconhece notação científica
  % no arquivo de entrada.
  \draw[color=blue] plot[smooth] file {latex-via-exemplos-tabela.txt} node [right] {$y=x^2$};
\end{tikzpicture}
\end{center}

```



Também podemos pintar, preencher com padrão e sombrear. Veja Exemplo 15.11.

Exemplo 15.11: ex15-tikz-pintar.tex

```

\begin{center}
\begin{tikzpicture}[scale=1.5] % podemos mudar a escala
% fill (pinta) e filldraw (pinta e desenha o contorno) podem ser usados para pintar. Outra
% forma é passar o parâmetro fill para draw.

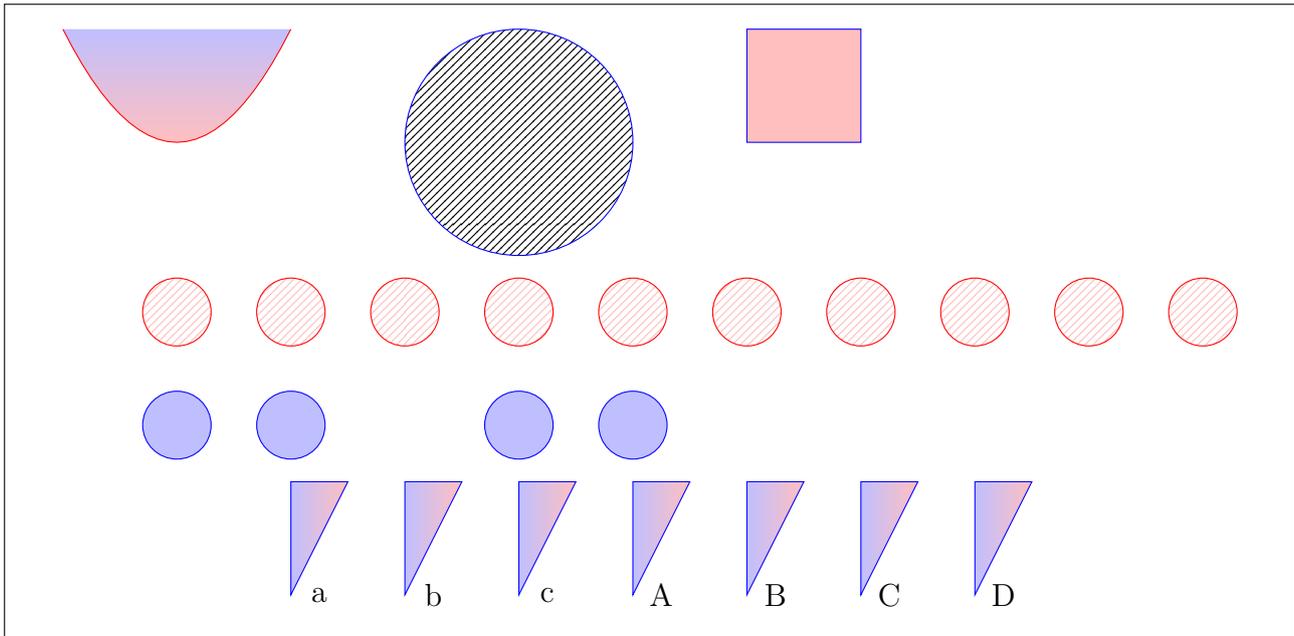
\filldraw[fill=red!25,draw=blue] (5,0) rectangle (6,1);

% pattern preenche com padrao de preenchimento.
% Parametro draw eh para desenhar o contorno
\pattern[pattern=north east lines, draw=blue] (3,0) circle (1);

% sombra eh feito pelo shade e sombra com contorno com shadewdraw (igual a shade com
% parametro draw)
% sombra pode ser degrade e aceita o sintaxe do xcolor (cor!percentual)
% Na funcao, \x^2 nao funciona. Deve ser (\x)^2.
\shade[domain=-1:1,top color=blue!25,bottom color=red!25,draw=red] plot (\x,{(\x)^2});

% fazendo tudo com draw
% Iterando num intervalo
\foreach \x in {0,...,9} \draw[red,pattern=north east lines,pattern color=red!25] (\x,-1.5)
circle (0.3);
% Iterando sobre listas
\foreach \x in {0,1,3,4} \draw[fill=blue!25,draw=blue] (\x,-2.5) circle (0.3);
% contador com iteracao em letras
\foreach \x [count = \xi] in {a,...,c,A,B,...,D}
{
\draw[shade, left color=blue!25, right color=red!25,draw=blue] (\xi,-3) --- (\xi+0.5,-3)
--- (\xi,-4) --- cycle;
\draw (\xi+0.25,-4) node{\x};
}
\end{tikzpicture}
\end{center}

```



Na matriz do `tikz`, podemos desenhar sobre ele. Veja Exemplo 15.12.

Exemplo 15.12: `ex15-tikz-matriz.tex`

```

\begin{center}
\begin{tikzpicture}
\matrix (A) [matrix of math nodes,left delimiter=[,right delimiter=}]
{ 1 & 2 & -1 & 5 \\
0 & |[draw=red,circle]|3 & 2 & 1 \\
0 & -2 & \phantom{-}2 & -4 \\
};
% separando a matriz da parte aumentada
\draw[thick,dashed,blue] (A-1-3.north east) -- (A-3-3.south east);
% limitando a parte escalonada
\draw[thick,dotted,blue] (A-2-1.north west) -- (A-2-1.north east) -- (A-3-1.south
east);
% onde vai colocar a seta para piv\hat{o}
\coordinate (P) at ($(A-2-1)!-2!(A-2-2)$);
\coordinate (Q) at ($(A-2-1)!-1.3!(A-2-2)$);
% colocando a seta na linha de piv\hat{o}
\draw[->] (P) -- (Q);
\node at ($(P)!-1.2!(Q)$) {piv\hat{o}};
\end{tikzpicture}
\end{center}

```

$$\text{piv\hat{o}} \rightarrow \left[ \begin{array}{ccc|c} 1 & 2 & -1 & 5 \\ 0 & 3 & 2 & 1 \\ 0 & -2 & 2 & -4 \end{array} \right]$$

O diagrama comutativo é uma espécie de tabela com comando `\ar` para desenhar setas. `\ar` é seta e o primeiro parâmetro opcional dele é para que lado vai a seta (pode ser combinado). “r” (direita), “l” (esquerda), “d” (para baixo), “u” (para cima). Também pode usar o “to=” para indicar a célula diretamente. O rótulo, caso exista, deve ser delimitado entre aspas.

No Exemplo 15.13, precisará também do pacote `amssymb` para o comando `\circlearrowleft`, mesmo com `unicode-math` no  $X_{\text{L}}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}/\text{L}^{\text{A}}\text{U}^{\text{A}}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  (Se estiver usando  $X_{\text{L}}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}/\text{L}^{\text{A}}\text{U}^{\text{A}}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ , carregue antes do `unicode-math`). No `unicode-math`, este símbolo é `\acwgapcirclearrow`, mas na versão 2016, ele não aparece (março de 2018, também não). Provavelmente a fonte padrão “Latin Modern Math” não tem este símbolo.

Exemplo 15.13: ex15-tikz-diagrama.tex

```
\begin{center}
\begin{tikzcd}
A \ar[r, "\phi"] \ar[d, red] \ar[to=2-2, phantom, "\circlearrowleft" description] & B \ar[d,
"\psi" red] \\
C \ar[r, red, "\varphi" red, "\eta" {swap, blue}] & D
\end{tikzcd}
\end{center}
```

$$\begin{array}{ccc}
 A & \xrightarrow{\phi} & B \\
 \downarrow & \circlearrowleft & \downarrow \psi \\
 C & \xrightarrow[\eta]{\varphi} & D
 \end{array}$$

As bibliotecas para `tikz` é extensa, cobrindo vários tipos de desenhos e diagramas. Se precisar de algo desse tipo, consulte o material para ver se já tem a biblioteca pronta.

Para desenhar gráficos de funções ou de dados em 2D ou em 3D, existe o pacote especializado `pgfplots` que não vamos discutir aqui.



# Capítulo 16

## Produzindo Poster e Slides

### 16.1 Poster

Para quem quer produzir o `poster` diretamente no  $\text{\LaTeX}$ , a maneira mais prática é diagramar com um quarto (metade de largura e altura) do tamanho desejado, usando letra 12pt. Após finalizar, amplie o poster para produzir no tamanho real. Com tal procedimento, os caracteres serão ampliados para equivalentes a 24pt. Use a fonte **Sans Serif** para poster. Caso de precisar usar a fonte romana, aumente o tamanho de caracteres em torno de 20%.

Por exemplo, para poster final de 100x120cm, dimensione para 50x60cm, para dimensão final de A0, dimensione para A2, etc.

Para criar “layout” de divisão em colunas e similares, use o ambiente `multicols` (do pacote `multicol`) e `minipage`. O enfeite das “caixas de texto” podem ser feitos usando diversas caixas, tais como `\fbox`.

No Exemplo 16.1, foi usado o pacote `lipsum` para preencher o espaço com texto pare ver como fica a aparência do poster.

Exemplo 16.1: `ex16-poster.tex`

```
\documentclass[12pt]{article}
% 12pt torna 24pt apos dobrar a dimensao do documento usando jPDFWeak ou poster
\usepackage[T1]{fontenc} % codificação da fonte em 8–bits
\usepackage[utf8]{inputenc} % acentuação direta
\usepackage[brazil]{babel} % em portugues brasileiro
\usepackage{lmodern} % Fonte Latin Modern (Computer Modern com extensao latin)
\usepackage{lipsum} % para preencher o espaco (para teste)
% acerto de margens usar metade da dimensao final
% para poster final de 100x120cm
% \usepackage[paperwidth=50cm,paperheight=60cm, margin=0.7cm,]{geometry}
% para poster final em A0
% \usepackage[a2paper, margin=0.7cm]{geometry}
% Poster em 90cmx120cm
\usepackage{geometry}
\geometry{paperwidth=45cm,paperheight=60cm,
```

```

lmargin=0.7cm,rmargin=0.7cm,tmargin=0.7cm,bmargin=0.7cm}
\usepackage{multicol} % usar varias colunas
\usepackage{graphicx} % usar graficos
\usepackage[usenames]{xcolor} % usar cores
\usepackage{fancybox} % para molduras adicionais nas caixas

% espacamento entre colunas
\setlength{\columnsep}{1cm}
% \setlength{\columnseprule}{1pt} % separador de colunas
% No poster, nao costuma usar a indentacao
\setlength{\parindent}{0pt} % sem indentação

% comando para colocar titulo customizado
%-----
\setlength{\fboxsep}{0pt} % bordas grudadas no conteudo
%% caixa de titulos: versao colorida.
\newcommand{\maintitlebox}[1]{\shadowbox{\colorbox{yellow}{\parbox{0.99\columnwidth}{#1}}}}
\newcommand{\titlebox}[1]{\fbox{\colorbox{yellow}{\parbox{1.0\columnwidth}{#1}}}}
%% caixa de titulos: versao monocromatica.
% \newcommand{\maintitlebox}[1]{\shadowbox{\parbox{0.99\columnwidth}{#1}}}
% \newcommand{\titlebox}[1]{\fbox{\parbox{1.0\columnwidth}{#1}}}

% \usepackage{cmbright} % computer modern compatible sans serif font for text and math

\renewcommand\familydefault{\sfdefault} % Usar sans serif por padrao
\pagestyle{empty} % sem enumeracao das paginas
\begin{document}
% \large % aumentar um pouco a letra (apesar de nao ser necessario)

\maintitlebox{
\begin{minipage}[t]{0.98\textwidth}
\begin{center}
\vspace{1pc}
\Huge
% podera acrescentar logo, usando includegraphics
% \includegraphics[height=4pc]{logo-esquerda}
% \hfill
Poster de Teste
% \hfill
% \includegraphics[height=4pc]{logo-direita}
\end{center}
\vspace{1pc}
\end{minipage}
} % maintitlebox
\vspace{1pc}

\begin{center}
{\huge Sadao Massago}

```

```

\
{\large DM–UFSCar} (Universidade Federal de São Carlos)

web: \texttt{http://www.dm.ufscar.br/\sim$adao} \\\
e–mail: \texttt{sadao@dm.ufscar.br}
\end{center}

\vspace{2pc}
% \hrule \vspace{1pc} \hrule

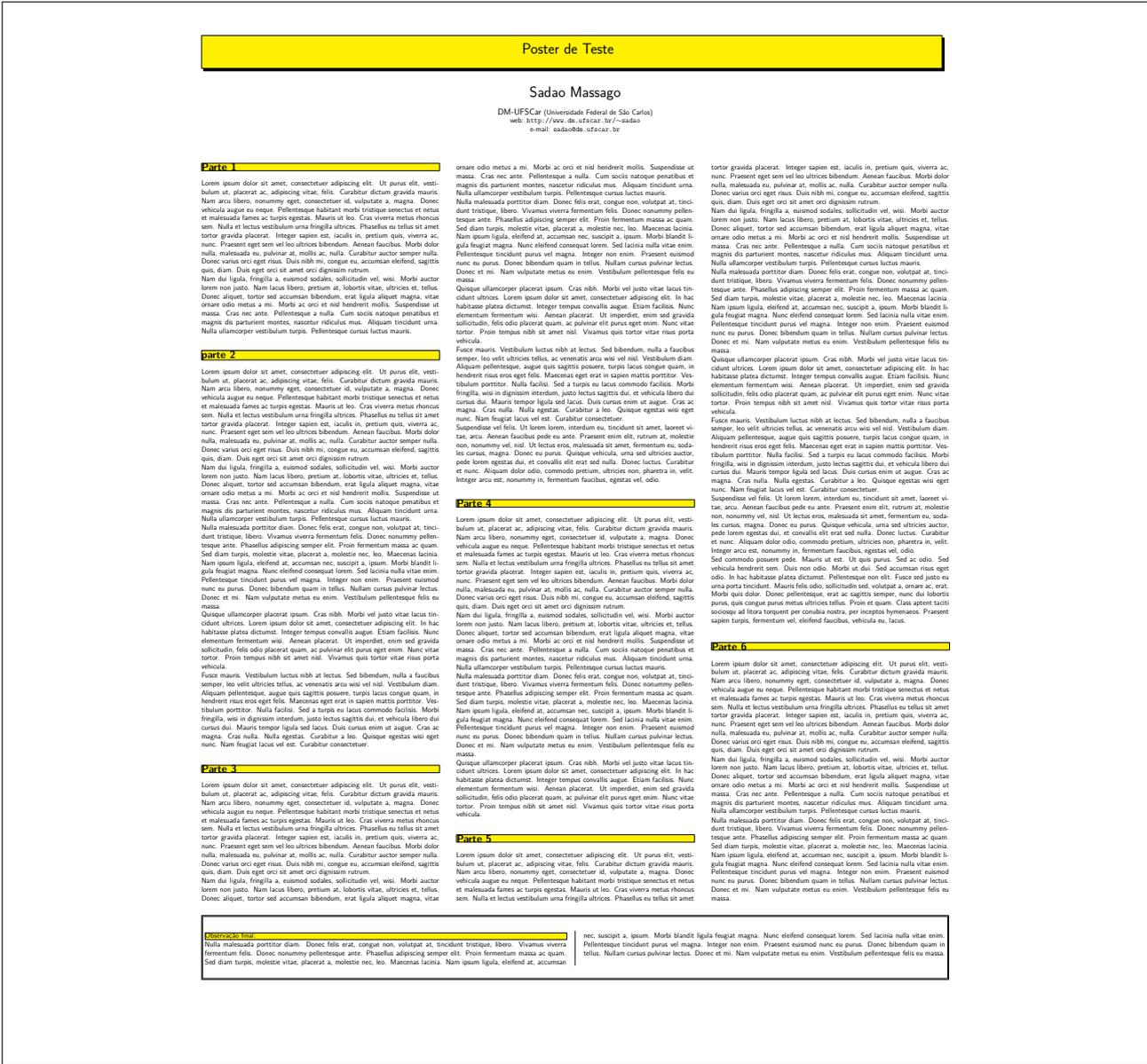
% \setlength\columnseprule{.4pt}
\begin{multicols}{3} % 2 colunas, por ter linha comprida
\section*{\titlebox{Parte 1}}
\lipsum[1–2]
\section*{\titlebox{parte 2}}
\lipsum[1–5]
\section*{\titlebox{Parte 3}}
\lipsum[1–6]
\section*{\titlebox{Parte 4}}
\lipsum[1–4]
\section*{\titlebox{Parte 5}}
\lipsum[1–7]
\section*{\titlebox{Parte 6}}
\lipsum[1–3]
\end{multicols} % 2 colunas, por ter linha comprida

\vspace{1pc}
\doublebox{
\begin{minipage}{0.99\textwidth}
\vspace{2pc}
\setlength\columnseprule{.5pt}

\begin{multicols}{2} % 2 colunas
\titlebox{Observação final:}
\lipsum[3]
\end{multicols} % 2 colunas

\vspace{1pc}
\end{minipage}
} % \doublebox{
\end{document}

```



Se quer usar  $\LaTeX$  para produzir material publicitário como posters, jornais, revistas e livros ilustrados com a qualidade profissional, o pacote `flowfram` é o pacote adequado. Além da sua qualidade final, a facilidade e a versatilidade costumam ser superior aos pacotes similares.

Para criar layout complexos que inclui caixas não retangulares, poderá usar o `flowframtk` disponível no site <http://www.dickimaw-books.com/latex/admin/html/flowfram.shtml>

Para quem estiver acostumado a usar o pacote `beamer` (pacote para produzir slides de apresentação), o pacote `beamerposter` é interessante para produzir poster com cara de apresentação em `beamer`.

Em vez de usar o  $\LaTeX$ , poderá usar aplicativos gráficos para criar poster ou similares. Para posters científicos, ideal que o aplicativo tenha suporte ao  $\LaTeX$  como no caso do editor gráfico `inkscape` e editor para publicações `scribus`, ambos livres e multi plataforma.

Para aumentar o poster dimensionado com fontes 12pt, como do Exemplo 16.1, podemos usar o aplicativo livre e multi plataforma `jpDFTweak` que é um aplicativo para pós produção de documentos PDF, disponível em <http://jpdftweak.sourceforge.net/>.

Entre várias funções importantes deste aplicativo, vamos usar o redimensionamento para ampliar o poster.

No jPDFtweak:

1. indique o arquivo de entrada em “input”.
2. indique o arquivo de saída em “output”.
3. em [page size], cheque o “[v] scale pages” e selecione o tamanho da página como “Page size” = “escala dobro do original”.
4. clique no [run]

**Observação:** Para dobrar a escala no papel ISO, escolha dois números para baixo. Por exemplo, se o original estiver em A2 landscape, escolha A0 landscape. Caso a medida do papel estiver em cm, dobre o valor e em seguida, multiplique por  $72/2.54$  para converter em postscript point que jPDFtweak usa na largura e na altura do papel. Por exemplo, se o original estiver com 45cmx60cm, vamos colocar  $90 \times 72/2.54 = 2551.18$  e  $120 \times 72/2.54 = 3401.57$  para largura e altura respectivamente.

## 16.2 Slides

Para criar apresentações, costuma usar a classe `beamer`.

Algumas opções da classe `beamer` são tamanho da fonte (normalmente usa 12pt, embora suporta o tamanho maior), `handout` (para impressão. Ignora o `overlay`), `notes` (incluir notas), `notesonly` (somente notas).

A classe `beamer` usa área reduzida com letra normal e será ampliado quando projetar na tela. Assim, costuma usar a fonte de tamanho 12pt.

Nesta classe, cada tela de slide será delimitado pelo ambiente `frame`. Tem a versão comando do `frame` também, para ser usado quando tem poucos comandos contidos nele, como no caso de título e sumário.

O tema (modelo) de slides é escolhido pelo comando `\usetheme`. Tema que vem como padrão são: `AnnArbor`, `Antibes`, `Bergen`, `Berkeley`, `Berlin`, `Copenhagen`, `Darmstadt`, `Dresden`, `Frankfurt`, `Goettingen`, `Hannover`, `Ilmenau`, `JuanLesPins`, `Luebeck`, `Madrid`, `Malmoe`, `Marburg`, `Montpellier`, `PaloAlto`, `Pittsburgh`, `Rochester`, `Singapore`, `Szeged`, `Warsaw`, `boxes`, `default` e `CambridgeUS`.

Para cada tema, ainda podemos escolher o tema de cores, ou combinação de cores a serem usados, escolhidos pelo comando `\usecolortheme`. Tema de cores padrão que vem são: `default`, `albatross`, `beaver`, `beetle`, `crane`, `dolphin`, `dove`, `fly`, `lily`, `orchid`, `rose`, `seagull`, `seahorse`, `whale` e `wolverine`.

Para ver como ficará a combinação do tema e tema de cores, poderá consultar o <https://hartwork.org/beamer-theme-matrix/>.

É pouco usado, mas ainda podemos alterar as combinações das fontes com o comando `\usefonttheme`. Tema de fontes padrão são: `default`, `serif`, `professionalfonts`, `structurebold`, `structureitalicserif`, `structuresmallcapsserif`. Em geral, usa-se o comando `\setbeamerfont` para configurar fontes de seus elementos do que escolher um tema para fontes (combinação de fontes).

A classe `beamer` carrega o pacote `hyperref` por padrão. Então podemos configurar alguns aspectos sobre informações do PDF com o `\hypersetup`, mas isto não é obrigatório.

Para criar slide de títulos ou quando estiver usando tema que usa títulos, autores, etc, precisará informar o título, autor, instituição e data, respectivamente pelos comandos `\title`, `\author`, `\institute` e `\date`. Quando tiver mais de um autor, separe com `\and`. O comando `\inst` dentro do `\author` e `\institute` faz a ligação de autor com o instituto correspondente.

Os elementos do slide pode ser alterado pelo comando `\setbeamertemplate`. Para criar degrade no fundo do slide, costuma usar o recurso do pacote `tikz`. Note que `xcolor` é carregado pelo `beamer` por padrão. O comando `\setbeamertemplate` também pode ser usado para desativar os botões de navegação.

Os comandos `\maketitle` e `\tableofcontents` funcionam normalmente. Note que o sumário é construído a partir de `section` e não pelo `frame`. Assim, para ter o sumário, deverá colocar `\section` entre os `frame's` no local desejado.

Cada `slide` (tela) é criado pelo ambiente `frame` ou comando `\frame`. O ambiente `frame` aceita o comando `\frametitle` para colocar títulos.

O comando `\note` serve para inserir notas adicionais que não são colocados no `slide`, mas pode ser gerado e impresso com a opção `notes` ou `notesonly` na opção da classe, para auxiliar na apresentação.

As notas adicionadas fora do `frame` será assumido que do `frame` anterior.

O Exemplo 16.2 ilustra um slide e uma das telas do slide gerado pelo código listado. Nele, foi usado o pacote `lipsum` para preencher alguns slides.

#### Exemplo 16.2: ex16-slides.tex

```
\documentclass[12pt]{beamer} % para apresentacao.

\usepackage[T1]{fontenc} % codificação da fonte em 8–bits
\usepackage[utf8]{inputenc} % acentuação direta
\usepackage[brazil]{babel} % em portugues brasileiro

\usetheme{Warsaw} % tema (modelo)
\usecolortheme{default} % tema de cores (Esquema de cores)
% para ver como fica as combinacoes de tema e esquema de cores,
% veja o site https://hartwork.org/beamer-theme-matrix/
% \usefonttheme[onlymath]{serif} % use serif for math
%\setbeamerfont{title}{family=\rm} % titulo em romano

\usepackage{amssymb,amsmath} % para incrementar fórmulas
\usepackage{tikz} % para cria degrade no fundo
```

```

%\usepackage[condensed,math]{iwona} % Mudando fontes
\usepackage{lipsum} % para gerar texto, para teste

% \usepackage{hyperref} % ja eh carregado pelo beamer
\hypersetup{% informacoes do PDF
  pdftitle={Slide beamer},%
  pdfauthor={Sadao Massago},
  pdfsubject={Exemplo de Slide},
  pdfkeywords={LaTeX, Slide}
} % \hypersetup

% Informacoes para criar titulo
\title[Exemplo de Slide]{Exemplo de Slide}
\author{%
  Sadao Massago\inst{1} % \and ???\inst{2}
}
\institute[Universidade Federal de São Carlos]{
  \inst{1}%
  Departamento de Matemática
  % \and
  % \inst{2} ???
}
\date[Março 2018]{\LaTeX{} Via Exemplos, 2018}

% fundo em degrade
\setbeamertemplate{background canvas}{%
\begin{tikzpicture}[remember picture,overlay]
%\shade[top color=red!10,bottom color=blue!10, middle color=white!10]
\shade[top color=red!10,bottom color=blue!10]
  (current page.north west) rectangle (current page.south east);
\end{tikzpicture}%
}

\setbeamertemplate{navigation symbols}{} % desativa botao de navegacao

\begin{document}
\frame{\titlepage} % slide de titulos
\frame{\transdissolve\tableofcontents} % slide de sumario

\note{Em torno de 1 minuto para tópicos.} % notas

\section{Slide e Transparências} % section será usado no sumário e similares

\begin{frame} % slide
  \frametitle{Slides de apresentação}
  \begin{itemize}
\item <1->\alert<1>{Usar letras grandes}
\item <2->\alert<2>{Cor do fundo deve criar contraste com texto}
\item <3->\alert<3>{Para apresentação, contraste pode ser pela cor}
\item <4->\alert<4>{Para imprimir, contraste deve ser claro/escuro}

```

```

\end{itemize}
\only<5->{\alert<5>{Escrever pouco e falar muito}}
\end{frame}

\begin{frame}
\frametitle{blocos}
\begin{block}{}<1->
Este é um bloco sem título. Bloco aceita overlay.
\end{block}

\begin{block}{Segundo bloco}<2->
Este é um bloco com título.
\end{block}

“bloco” é uma “caixa” com ou sem título e aceita o parâmetro de \texttt{overlay}.

\end{frame}

\section{Ambiente \texttt{verbatim} no slide} % outra entrada de sumário

\begin{frame}[fragile] % outro slide
\frametitle{Verbatim}
Ambiente \texttt{verbatim} e similar requer opção \texttt{fragile}.
%\only<2>{
\begin{verbatim}
\begin{frame}[fragile]
\frametitle{...}
...
\end{frame}
\end{verbatim}
}%
\end{frame}

%\begin{note} % outra nota: ambiente note so funciona fora do frame Não funciona na
    versão 2016. Testar na versão mais recente.
\note{Sem a opção \texttt{fragile}, não pode usar o ambiente \texttt{verbatim} dentro do \
    \texttt{frame}.
    Não pode usar “\texttt{overlay}” no ambiente \texttt{verbatim}
}
%\end{note}

\section{Quebra automática em frames} % mais uma entrada par asumario

\begin{frame}[allowframebreaks] % e mais um slide
\lipsum[1-2]
\end{frame}
\note{
Para que mude o frame automaticamente quando tornar cheio
coloque a opção \texttt{allowframebreaks} no frame
}

```

```

\section{Opção plain}
\begin{frame}[plain]
\frametitle{Opção \texttt{plain}}
\lipsum[1]
\end{frame}
\note{

```

Opção `\texttt{plain}` desativa cabeçalho e rodapé do frame para ter mais espaço. Útil para colocar figura maior, por exemplo.

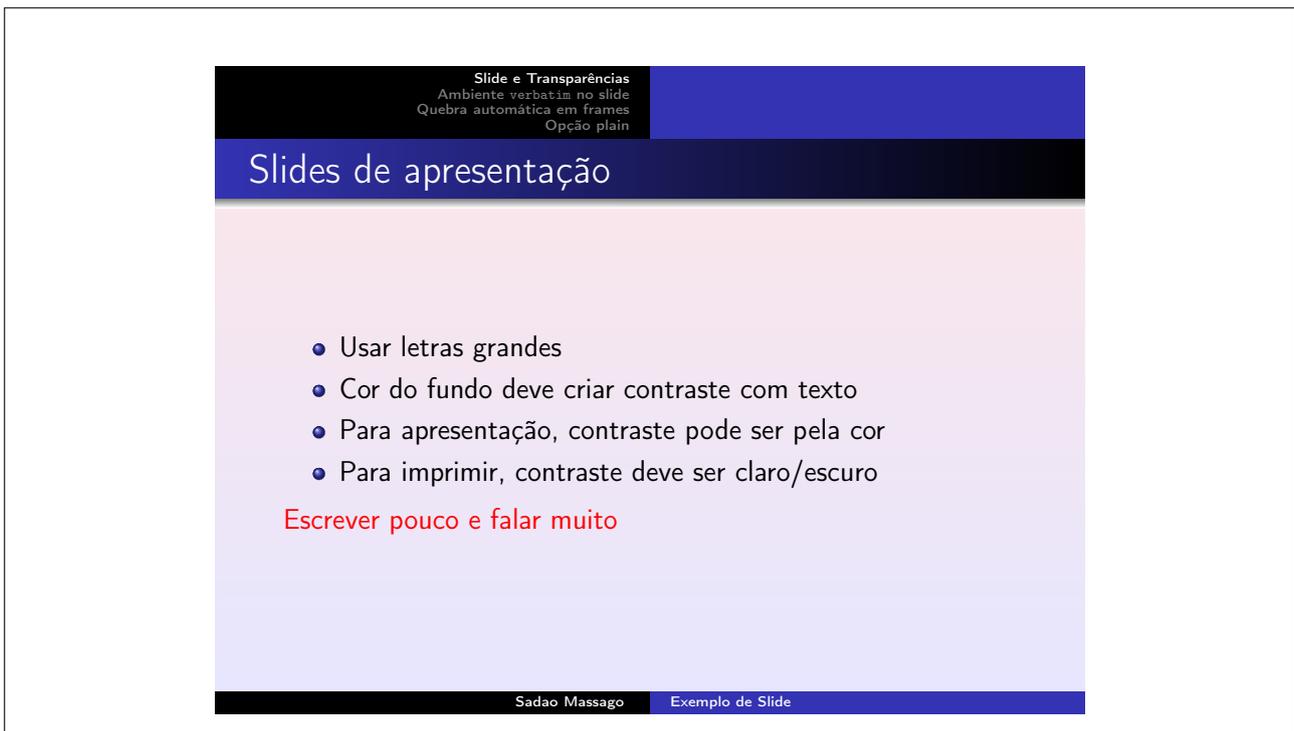
```

}
```

```

\end{document}

```



Note que, nos itens de listas aceitam o parâmetro opcional de “`overlay`” delimitado por “`<`” e “`>`”. Com o uso de `overlay`, o `slide` será apresentado por etapas. Em cada etapa do `overlay`, será mostrado uma parte do `slide`. O mais usado é ir mostrando cada vez mais elemento na medida que etapa avança (andar no `slide`).

A especificação do `overlay` são:

`<n>` indica que será mostrado somente na etapa `n`.

`<m-n>` é mostrado somente nas etapas `m` até `n`.

`<n->` é mostrado nas etapas `n` em diante.

`<-n>` é mostrado até as etapas `n`.

Os comandos de formatação de textos do tipo `\textsf{}`, `\textbf{}`, `\emph`, etc, também aceitam a opção de `overlay`. Também foi acrescentado o comando `\alert` neste conjunto para deixar o texto em vermelho.

Para criar **overlay** nas partes desejadas que não é necessariamente itens da lista ou formatação de texto, existe o comando `\only` que mostra o conteúdo somente no **overlay** indicado.

Também tem o ambiente chamado de **block** que produz “bloco” com ou sem títulos e aceita “overlay”.

No **frame**, alguns parâmetros opcionais podem ser usados. Alguns dos mais importantes são

**fragile** Necessário quando usa o ambiente `verbatim` ou similar.

**allowframebreaks** Ativa a quebra automática de **frame**'s quando o conteúdo não cabe no **frame** atual.

**plain** Limpa as configurações (cabçalhos e rodapés) do **frame** para caber mais conteúdos. Útil para figuras e tabelas grandes.

**shrink** Reduz o tamanho do conteúdo para caber no **frame** atual, se necessário. Isto pode reduzir o tamanho da letra e por isso, deve ser usado com muita cautela. Use em conjunto com **plain** para ampliar a área de **slide**.

Note que `BIBTEX` não funciona no **beamer**. Assim, deverá criar manualmente a referência bibliográfica com o ambiente `thebibliography`.

# Capítulo 17

## Usando X<sub>Ǝ</sub>LaTeX e LuaLaTeX

Aqui será tratado sobre X<sub>Ǝ</sub>LaTeX e LuaLaTeX, considerado as próximas gerações de LaTeX.

### 17.1 LuaLaTeX e X<sub>Ǝ</sub>LaTeX

O LuaLaTeX foi designado para ser o sucessor do PDFLaTeX e por isso, deve ser o LaTeX padrão no futuro, mas pode ser um pouco mais lento do que o X<sub>Ǝ</sub>LaTeX. Até o momento (2018), o X<sub>Ǝ</sub>LaTeX é considerado mais estável do que LuaLaTeX. Lua(La)TeX permite estender a funcionalidade do (La)TeX usando a linguagem script Lua (Lua foi desenvolvido no PUB-Rio, aqui no Brasil).

Já existem pacotes especiais para LuaLaTeX como o módulo de diagramação automática de grafos no `pgf/tikz`. O suporte ao recurso de microtipografia (efetuar pequenos ajustes no tamanho das letras e espaçamento para que texto acomode melhor) pelo pacote `microtype` também funciona melhor com LuaLaTeX do que em X<sub>Ǝ</sub>LaTeX.

A maioria dos documentos de LaTeX devem funcionar sem modificações no X<sub>Ǝ</sub>LaTeX/LuaLaTeX, mas para usar as fontes do sistema, devem usar a forma própria de selecionar fontes. No caso em que a fonte selecionada contiver símbolos matemáticos suficientes, costuma usar o `unicode-math` em vez dos `amssymb` e `amsmath`. Caso precisar de alguns símbolos de `amssymb` ou comandos de `amsmath`, carregue antes do `unicode-math`.

A forma de ajustar a codificação no LaTeX, X<sub>Ǝ</sub>LaTeX e LuaLaTeX diferem, mas como X<sub>Ǝ</sub>LaTeX e LuaLaTeX usam `uft8` como padrão, não vamos preocupar com isso. Só no LaTeX que precisará especificar, usando o `inputenc`.

Embora `babel` funcione no X<sub>Ǝ</sub>LaTeX/LuaLaTeX também, no Exemplo 17.1 foi usado o `polyglossia` feito especialmente para X<sub>Ǝ</sub>LaTeX/LuaLaTeX. Ainda (2018) é cedo para saber se `babel` ou `polyglossia` vai ficar como pacote recomendável para X<sub>Ǝ</sub>LaTeX/LuaLaTeX.

Embora maioria dos pacotes de fontes do LaTeX funcione no X<sub>Ǝ</sub>LaTeX/LuaLaTeX, no Exemplo 17.1, foi usado o método próprio deles que permite selecionar, inclusive as fontes do sistema.

Se a fonte usada contiver símbolos matemáticos, use `unicode-math` em vez do `amssymb/amsmath`. Se precisar dos símbolos ou comandos de `amssymb` e `amsmath`, carregue eles antes do `unicode-math`.

Note que os pacotes e comandos do LaTeX padrão, exceto o `inputenc` costumam funcionar

no Xe<sub>Λ</sub>TeX e Lua<sub>Λ</sub>TeX, mas não o contrário. Assim, para que o documento possa ser compilado tanto em PDF<sub>Λ</sub>TeX como em Xe<sub>Λ</sub>TeX/Lua<sub>Λ</sub>TeX, poderá usar o pacote `iftex` para detectar a engenharia do TeX em uso.

Exemplo 17.1: ex17-lualatex.tex

```

\documentclass[a4paper,12pt]{article}

\usepackage{iftex} % Para detectar engenharia de TeX

\ifPDFTeX % Se (PDF)LaTeX
  \usepackage[T1]{fontenc} % codificação da fonte em 8–bits
  \usepackage[utf8]{inputenc} % acentuação direta
  \usepackage[english,brazil]{babel} % em português brasileiro (ingles como secundario)

  \usepackage{lmodern} % latin Modern (Computer modern com extensao latin)

  % \usepackage{noto} % fonte Noto patrocinado pelo Goggle
  % \usepackage[italic]{mathastext} % A fonte Noto nao possui fontes matematicos ainda

  \usepackage{textcomp}
  \usepackage{amssymb,amsmath}

\else % Se XeLaTeX/LuaLaTeX
  \usepackage{polyglossia}
  \setdefaultlanguage{brazil}
  \setotherlanguage{english} % secundario

  % Estas fontes são padrões e não precisavam especificar. Foram colocados somente para
  % ilustrar
  \setmainfont{Latin Modern Roman}
  \setsansfont{Latin Modern Sans}
  \setmonofont{Latin Modern Mono}
  % Se precisar do amssymb,amsmath, deverao carregar antes do unicode–math
  % \usepackage{amssymb,amsmath}

  % \usepackage{xunicode} % textcomp equiv. nao funciona no lualatex recente (marco de
  % 2018)
  \usepackage{unicode–math} % amsmath, amssymb equiv.
  \ifLuaTeX % Somente LuaLaTeX
    \usepackage{lualatex–math} % alguns fix do amsmath/mathtools equiv. para
    % LuaLaTeX.
  \fi
  \setmathfont{Latin Modern Math} % Padrao e não precisava. Foi colocado somente para
  % ilustracao

  % Fonte Noto nao tem fontes matematicos ainda
  %\setmainfont[Ligatures=TeX]{NotoSerif}
  %\setsansfont{NotoSans}

```

```

% \setmonofont{NotoMono}
% Asana math (Palationo like)
% \setmainfont{Asana–Math}
% STIX (Times like font)
% \setmainfont[Ligatures=TeX]{STIX}
% \setmathfont{STIX Math}
%% XITS (Times like font)
% \setmainfont[Ligatures=TeX]{XITS}
% \setmathfont{XITS Math}
\fi

% Mais alguns pacotes
\usepackage[margin=2.5cm]{geometry}
\usepackage{hyperref}
\hypersetup{
  bookmarks=true
}
\usepackage[stretch=10]{microtype} % microtipografia: funciona no PDF $\text{\LaTeX}$  e Lua $\text{\LaTeX}$ 
  (parcialmente em Xe $\text{\LaTeX}$ )

\begin{document}
O Lua $\text{\LaTeX}$  foi designado para ser o sucessor do PDF $\text{\LaTeX}$  e por isso, deve ser o  $\text{\LaTeX}$ 
  padrão no futuro. Por outrolado, Xe $\text{\LaTeX}$  e mais rápido e estável (2018).

\ldots

Se  $F'(x)=f(x)$  for contínua,
\begin{mathdisplay}
\int_a^b f(x)dx = F(b) - F(a)
\end{mathdisplay}
\end{document}

```

O Lua $\text{\LaTeX}$  foi designado para ser o sucessor do PDF $\text{\LaTeX}$  e por isso, deve ser o  $\text{\LaTeX}$  padrão no futuro. Por outro lado, Xe $\text{\LaTeX}$  e mais rápido e estável (2018).

...

Se  $F'(x) = f(x)$  for contínua,

$$\int_a^b f(x)dx = F(b) - F(a)$$

Existem algumas fontes **Open Type** com suporte a matemática que podem ser usados no Xe $\text{\LaTeX}$ /Lua $\text{\LaTeX}$ . O site <https://www.overleaf.com/help/193-what-otf-slash-ttf-fonts-are-su> lista algumas delas.

- **Cambria Math** (Microsoft, somente Windows).
- **Latin Modern Math** (Bogusław Jackowski, Janusz M. Nowacki). Padrão do Xe $\text{\LaTeX}$ /Lua $\text{\LaTeX}$ .

- TeX Gyre Pagella Math (Bogusław Jackowski, Janusz M. Nowacki).
- TeX Gyre Termes Math.
- Asana Math (Apostolos Syropoulos), Estilo Times.
- Neo Euler (Khaled Hosny).
- STIX (STI Pub).
- XITS Math (Khaled Hosny).

Note que a fonte `\mathcal` e `\mathbb` do `unicode-math` é diferente do `amssymb`. O `\mathbb` é mais próximo do `\mathds` do `dsfont` e `\mathcal` é mais parecido com do pacote `eucal` do que de `amssymb`.

Quem quer a versão do `amssymb`, poderá colocar

```
\let\mathcal\relax % remove the definition by unicode-math
\DeclareMathAlphabet{\mathcal}{OMS}{cmsy}{m}{n}
\let\mathbb\relax % remove the definition by unicode-math
\DeclareMathAlphabet{\mathbb}{U}{msb}{m}{n}
```

no `preamble`, após carregar o `unicode-math`.

## 17.2 Mais sobre básicos das fontes no X<sub>Y</sub>LaTeX/LuaLaTeX

No X<sub>Y</sub>LaTeX/LuaLaTeX, poderá redigir em qualquer idioma desde que esteja utilizando as fontes que suportam tais idiomas. Assim, quando redige o documento multi lingue, basta definir qual fontes será usado em qual trecho. Para facilitar a especificação de fontes em cada trecho de textos, costuma definir família de fontes usando `\newfontfamily` do pacote `fontspec`. Ele define comandos para usar a fonte especificada com as configurações também especificadas, que funcionam como outros comandos de seleção de famílias de fontes pré definidos tais como `\rmfamily`, `\sffamily`, etc.

Por exemplo,

```
\newfontfamily{\notorm}{Noto Serif}
\newcommand{\textnotorm[1]{{\notorm #1}}
\newfontfamily{\notosf}{Noto Sans}
\newcommand{\textnotosf[1]{{\notosf #1}}
```

Definem comandos `\notorm` e `\notosf` para usar fontes `Noto Serif` e `Noto Sans`. O `\newcommand` foi usado para criar a versão `\textxx` que são `\textnotorm` e `\textnotosf` respectivamente. Note o uso de chave dupla para que a configuração da fonte não “vaze” para fora.

Para usar fontes não definidos anteriormente, poderá usar o comando `\fontspec` para selecionar fonte especificada.

Apesar de poder configurar a especificação da fonte com o parâmetro opcional dos comandos de seleção de fontes e definições de família de fontes, isto também pode ser efetuado separadamente pelo comando `\defaultfontfeatures` para especificar recursos das fontes. O parâmetro opcional será usado para indicar o nome da fonte que está especificando. Por exemplo,

```
\defaultfontfeatures[Noto Serif]{Ligatures=TeX}
```

Especifica que no **Noto Serif**, será usado a ligadura no modo TeX. Além de indicar nome das fontes, poderá indicar pelo comando de família das fontes como em

```
\defaultfontfeatures[\rmfamily,\sffamily]{Ligatures=TeX}
```

especifica que na fonte **romana** e **sans serif**, será usado a ligadura padrão do TeX.

Para limpar a especificação das fontes, basta usar como

```
\defaultfontfeatures[\rmfamily,\sffamily]{}
```

O comando `\defaultfontfeatures` tem a versão “+” que acrescenta a especificação em vez de substituir. O comando `\addfontfeature` pode ser usado para ajustar localmente a especificação. Coloque entre chaves para especificação adicional seja aplicado somente no trecho.

Quando lida com várias fontes no X<sub>Y</sub>LaTeX/LuaLaTeX, as vezes é importante checar se a fonte existe. `\IfFontExistsTF` do pacote `fontspec` checa a fonte do primeiro parâmetro existe. Se existir, executará o segundo parâmetro e se não existir, executa o terceiro parâmetro. O comando a seguir configura a fonte principal para **Noto Serif** caso ele existir.

```
\IfFontExistsTF{Noto Serif}{\setmainfont{Noto Serif}}{%
  \PackageWarn{\jobname.tex}{Noto Sans not found. Using default fonts}}
```

Ir testando várias fontes com este comando até achar uma disponível na lista é trabalhoso. Assim, para procurar fontes na lista, costuma usar o pacote `iffont`. Ele implementa o comando `\settofirstfound{\nomefonte}{<fontes>}` onde `<fontes>` é list de nome das fontes, separado pela vírgula. Ele associa o nome da primeira fonte encontrada no comando `\nomefonte`, o que pode ser usado para selecionar ou definir família de fontes. Caso nenhuma fonte existir, retorna o valor de `\@iffont@firstfont` que está como a fonte **Fira Sans**. Em geral, coloca-se o nome de uma fonte existente para evitar de cair no **Fira Sans**.

No entanto, `iffont` só funciona no X<sub>Y</sub>LaTeX (testados nas versões 2016 e março de 2018). Para que funções do `iffont` funcione no LuaLaTeX, reescreva a função usando o `\IfFontExistsTF` do pacote `fontspec` como em

```
...
\usepackage{iftex}
...
\usepackage{iffont}
\ifLuaTeX % patch
  % \usepackage{etoolbox} % iffont ja carregou
  \let\iffontexists\IfFontExistsTF
  \makeatletter
```

```

\renewcommand{\@iffont@checkfont}[1]{
  \IfFontExistsTF{#1}{
    \nottoggle{\@iffont@fontfound}{%
      \renewcommand{\@iffont@firstfont}{#1}
      \toggletrue{\@iffont@fontfound}
    }{}
  }{\stepcounter{\@iffont@fontsnofound} }
}
\makeatother
\fi

```

Com este comando, podemos selecionar fontes da lista, e usar a primeira fonte encontrada.

```

\settofirstfound{\mainfont}{Noto Serif, Latin Modern Serif}
\setmainfont{\mainfont}

```

Se o que quer é saber se fonte existe, o `iffont` também tem os comandos `\iffontsexist` e `\iffontexists` na qual executa o segundo parâmetro quando fonte existe e terceiro parâmetro quando a fonte não existe. A diferença é que no `\iffontsexist`, o parâmetro é lista de nome das fontes separado pela vírgula e no `\iffontexists`, o parâmetro é nome de uma única fonte (igual a `\IfFontExistsTF` do `fontspec`). Por exemplo,

```

\iffontexists{Noto Serif}{\setmainfont{Noto Serif}}{%
  \PackageWarn{\jobname.tex}{Noto Sans not found. Using default fonts}
}

```

Configura para `Noto Serif` só quando ele for acessível.

Se os comandos do pacote `fontspec` não conseguirem encontrar fontes pelo nome, tente usar nome de arquivo. Normalmente, para fontes instalados dentro da árvore de diretório do `TeX`, espera-se que o nome da fonte seja dado pelo nome de arquivo (com extensão) em vez de nome da fonte.

### 17.3 Usando em conjunto com `biblatex`

O documento em vários idiomas pode precisar também de referências bibliográficas em vários idiomas. O `biblatex` permite internacionalizar a referência bibliográfica (mesmo sem usar `XeLaTeX`/`LuaLaTeX`).

Para especificar o idioma da referência bibliográfica no arquivo `bib` para `biblatex`, coloque o idioma no campo `LANGID` de cada item. O “\*\_romanized” é a escrita em alfabeto romano caso o campo esteja em caracter que não seja romano. Ele será usado para a ordenação e similar. O campo “\*\_translated\_<idioma>” é a tradução para o <idioma> se referências forem formatados para <idioma>. Veja o Exemplo 17.2.

Exemplo 17.2: `ex17-biblatex.bib`

```

@book{wikibooks:latex,
  LANGID = {english},

```

```

author={wikibooks},
title="{\LaTeX{}}",
publisher="wikibooks",
url = "\url{https://en.wikibooks.org/wiki/LaTeX}",
date = {2018},
urldate={2018-03-05}
}

@Book{Kawasaki,
  LANGID = {japanese},
  TITLE = {バラと折り紙と数学と},
  TITLE_romanised = {bara to origami to sugaku to},
  TITLE_translated_english = {Roses, Origami \& Math},
  AUTHOR = {川崎敏和},
  AUTHOR_romanised = {Kawasaki, Toshikazu},
  PUBLISHER = {森北出版株式会社, Japan},
  PUBLISHER_translated_english = {Morikita Syuppan Co. Ltd, Japan},
  year={1998},
  ISBN = {4-627-01671-9}
}

```

O comando de configuração para `biblatex` é algo como

```

\usepackage[
  backend=biber,
  language=auto,
  % autolang=other, % for <otherlanguage> environment from babel and polyglossia
  % autolang=langname, % only for polyglossia <language> env.
  % bibencoding=utf8,
  style=authoryear,
]{biblatex}
\addbibresource{ex17-biblatex.bib} % arquivo bib

```

A opção `language=auto` indica que é para selecionar automaticamente o idioma (pelo campo `LANGID`) Obviamente, onde quer que apareça a referência bibliográfica, coloca-se o

```

\phantomsection % se estiver usando hyperref
\printbibliography[heading=bibintoc]

```

Note que, até agora, não usamos nada que não funcione no  $\text{\LaTeX}$  normal. Mas como a referência bibliográfica contém caracteres em japonês, precisará definir o uso de japonês em algum trecho. Isto será facilitado se estiver usando  $\text{\XeLaTeX}$  ou  $\text{\LuaLaTeX}$  em vez do  $\text{\PDFLaTeX}$ . Assim, vamos supor que está usando o  $\text{\XeLaTeX}$ / $\text{\LuaLaTeX}$  para prosseguir com a explicação. Para definir comandos para texto em japonês, coloque o comando

```

\usepackage[Ligatures=TeX]{fontspec}
\usepackage{iffont}
\settofirstfound{\mainjfont}{Noto Serif CJK JP, MS Mincho, TakaoMincho, IPAexMincho, Hiragino Mincho Pro}
\settofirstfound{\gothicjfont}{Noto Sans CJK JP, MS Gothic, TakaoGothic, IPAexGothic, Hiragino Maru Gothic Pro}

```

```
% If Gothic not found, use Mincho instead.
\IfFontExistsTF{\gothicfont}{}{\let\gothicfont\mainfont}
% If Mincho not found, error.
\IfFontExistsTF{\mainfont}{\newfontfamily{\jafamily}[BoldFont=\gothicfont]{\mainfont}}{}%
\PackageError{\jobname}{Japanese Mincho font not found}{Install appropriate ones}
}%
\newcommand{\textja}[1]{\{\jafamily #1}}
```

O comando `\jafamily` será definido quando encontrar algumas fontes da lista. Também será definido o comando `\textja` para trecho pequeno. Note que `\settofirstfound` do `iffont` não funciona no Lua<sub>L</sub>AT<sub>E</sub>X (testado na versão 2016 e março de 2018). Então, se quer usar o Lua<sub>L</sub>AT<sub>E</sub>X, deverá usar a sequência de `\IfFontExistsTF` em vez de `\settofirstfound`.

Agora, basta usar o comando para trocar de fontes padrão para fonte japonês onde aparece o texto em japonês. Veja o Exemplo 17.3.

### Exemplo 17.3: ex17-biblatex.tex

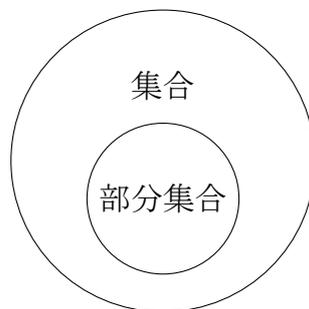
Figura contendo texto em japonês.

```
\begin{figure}[hbt]
\center
\begin{tikzpicture}
\draw (0,0) circle(2) (0,1) node{\textja{集合}};
\draw (0,-0.5) circle(1) (0,-0.5) node{\textja{部分集合}};
\end{tikzpicture}
\caption{Figura com texto em japonês}
\end{figure}
```

Sobre uso básico do `\texttt{biblatex}`, veja o `\cite{wikibooks:latex}`.

Agora, citando o livro em japonês `\textja{\cite{Takami,Kawasaki}}`.

Figura contendo texto em japonês.



Figuraq 1: Figura com texto em japonês

Sobre uso básico do `biblatex`, veja o `wikibooks` 2018. Agora, citando o livro em japonês 川崎敏和 1998.

Como a referência bibliográfica contém texto em japonês, colocamos o comando de seleção de fonte japonês.

```
\phantomsection % se estiver usando hyperref
{\jafamily \printbibliography[heading=bibintoc]}
```

e a saída será

## Referências Bibliográficas

- [1] wikibooks (2018).  $\text{\TeX}$ . wikibooks. URL: <https://en.wikibooks.org/wiki/LaTeX> (acesso em 15/03/2018).
- [2] 川崎敏和 (1998). バラと折り紙と数学と. 森北出版株式会社, Japan. ISBN: 4-627-01671-9.

Note que, no exemplo acima, a fonte de todo trecho da referência bibliográfica foi trocada. Assim, se tiver letras acentuadas na referências bibliográficas que conflitam com letras japonesas, começará aparecer coisas indesejáveis. Para contornar o problema, ou acentuar no modo  $\text{\TeX}$ , ou especificar o uso de japonês dentro do arquivo bib a cada trecho, em vez de aplicar no  $\text{\printbibliography}$ .

No caso de escrever o documento em japonês ou similar, deverá carregar o pacote apropriado que configurará para ambiente do idioma correspondente, ajustando os parâmetros para particularidade daquele país. Além disso, com o uso do tal pacote, a troca de fontes para japonês e vice versa será automática, sem a necessidade de estar especificando a família de fontes a ser usadas a cada trecho.

Como observação final sobre  $\text{X}_{\text{\TeX}}/\text{Lua}\text{\TeX}$ , os comandos protegidos dos pacotes de  $\text{X}_{\text{\TeX}}/\text{Lua}\text{\TeX}$  costumam usar “\_” em vez de “@” ou em conjunto com “@”. Isto foi introduzido no  $\text{\LaTeX}$  pelo equipe de desenvolvimento  $\text{\LaTeX}$  3 e aparecem nos pacotes mais recentes. Assim, para usar estes comandos e ambientes no `preamble`, deverá colocar entre  $\text{\ExplSyntaxOn}$  e  $\text{\ExplSyntaxOff}$ .



# Apêndice A

## Desenvolvendo Pacotes

Aqui veremos como escrever seu próprio pacote, também chamado de arquivos de estilos.

### A.1 Criando pacotes

Pacote é conjunto de configurações de documentos e definições de comandos ou similares para ser aplicado no documento desejado. Pacote também é chamado de arquivo de estilos e tem a extensão `sty`.

Um pacote começa com o cabeçalho. O comando `\NeedsTeXFormat{LaTeX2e}` colocado no começo do pacote indica que ele precisa do  $\text{\LaTeX} 2_{\epsilon}$ . Em seguida, costuma colocar o comando `\ProvidesPackage{<nome do pacote>}[<mensagem>]`, onde `<nome do arquivo>` é nome do arquivo, incluindo a extensão e o argumento opcional `<mensagem>` é a mensagem a ser emitido quando carrega o pacote. As informações adicionais tais como versões, mensagens de advertência ou erros, etc pode ser emitido pelo comando `\typeout{<mensagem>}` a qualquer momento.

Note que não precisa colocar `\makeatletter` e `\makeatother` (nem deve) no arquivo de estilo para acessar comandos que usam “@”, pois já está com uso de “@” ativado por padrão. No caso de  $\text{\XeLaTeX}$ / $\text{\LuaLaTeX}$  que usam pacotes mais modernos, usa “\_” no nome dos comandos. Para usar estes comandos no `pramble`, use entre `\ExplSyntaxOn` e `\ExplSyntaxOff`. Também não é necessário no arquivo de estilos.

É recomendado que use a acentuação n modo  $\text{\TeX}$ , pois não saberemos a codificação que o usuário final vai usar.

Dentro do arquivo de estilos, o comando para carregar os pacotes é `\RequirePackage` em vez de `\usepackage`. Para testar se o pacote existe, cheque se tem o arquivo correspondente com o `\IfFileExists`. Não esqueça da extensão do arquivo no nome. Se arquivo do primeiro parâmetro existir, executará o segundo parâmetro. Caso não existir, executará o terceiro parâmetro.

Também é bom checar se não está executando no modo de compatibilidade (modo antigo). Para isso, usa-se o comando `\if@compatibility` que é da forma

```
\if@compatibility
```

```

<comando 1>
\else
<comando 2>
\fi

```

A mensagem de erro é emitido por `\PackageError{<nome do pacote>}{<mensagem curta>}{<mensagem 1>}`. Se for só a advertência, usa-se o comando `\PackageWarning{<nome do pacote>}{<mensagem>}`.

Para saber se foi carregado a classe de documento compatível, usa-se o comando `\@ifclassloaded{}` que tem a forma `\@ifclassloaded{<classe>}{<cmd1>}{<cmd2>}`. Quando `<class>` for usado, executa o `<cmd1>` e caso não for a `<classe>`, executa o `<cmd2>`. Com isso, podemos elaborar o cabeçalho do arquivo de estilo. O Exemplo A.1 é uma listagem do exemplo de arquivos de estilos.

#### Exemplo A.1: ex-a-estilo.sty

```

% ex-a-estilo.sty

\NeedsTeXFormat{LaTeX2e}
% \ProvidesFile{ex-a-estilo.sty}[2018/02/10 v0.5 ex-a-estilo style (require book, amsbook
, or report as document class)]
\ProvidesPackage{ex-a-estilo.sty}[2018/02/10 v0.5 ex-a-estilo style (require book,
amsbook, or report as document class)]

\typeout{ex-a-estilo style 0.5 <February/2018>.}

% Pacotes necessarios

% Da base e required
\RequirePackage{amsthm, amssymb, amsmath}

% \RequirePackage[brazil]{babel}

% pacotes nao obrigatorios do LaTeX
% \RequirePackage{geometry} % carregar sempre

\IfFileExists{dsfont.sty}% carregar, se existir
{\RequirePackage{dsfont}}{}%

% testando se tem o pacote instalado e providenciando o
% comando, caso pacote nao exista.
\IfFileExists{hyperref.sty}%
{\RequirePackage{hyperref}}%
{\PackageWarning{ex-a-estilo}{hyperref.sty not found. Using draft mode.}%
\providecommand\url[1]{\texttt{##1}}%
}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Para evitar a tentativa de usar no modo de
% compatibilidade com LaTeX antigo (documentstyle)

```

```

%-----
\if@compatibility % modo de compatibilidade com LaTeX antigo?
  \PackageError{ex-a-estilo}{Not support older compatible mode (documentstyle)}
    {Use documentclass instead of documentstyle}
  \endinput
\else
  % OK
\fi

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% requer book, amsbook, ou report como classe de
% documento (enumeracao do teorema eh vinculado
% ao chapter)
%-----

\@ifclassloaded{book}% Using book?
  {}% YES (OK)
  {\@ifclassloaded{amsbook}% No, but are using AMS book?
    {}% YES (OK)
    {\@ifclassloaded{report}% No, but are using report?
      {}% YES (OK)
      { % No! Then can't apply this package
        \PackageError{ex-a-estilo}{Require book, amsbook, or report as document class}
          {Use the required document class}
        %\endinput
      }
    }
  }
}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% definicao dos macros para ambiente de teoremas
% teoremas, lemas, proposi\c{c}\~oes, etc

\theoremstyle{plain}

\newtheorem{theorem}{Teorema}[chapter]
\newtheorem{axiom}[theorem]{Axioma}
\newtheorem{corollary}[theorem]{Corol'ario}
\newtheorem{lemma}[theorem]{lema}
\newtheorem{proposition}[theorem]{Proposi\c{c}\~ao}
\newtheorem{conjecture}[theorem]{Conjectura}

% defini\c{c}\~ao de defini\c{c}\~oes, exemplos, etc.
\theoremstyle{definition}
\newtheorem{definition}[theorem]{Defini\c{c}\~ao}

\theoremstyle{remark}

```

```

\newtheorem{remark}[theorem]{Obserca\c{c}\~{a}o}

\newtheorem{note}[theorem]{Nota}
\newtheorem{example}[theorem]{Exemplo}
%
\newtheorem{question}[theorem]{Pergunta}

% alterando as penalidades (para corte de linhas em...)
\hyphenpenalty=5000 % hifeniza\c{c}\~{a}o
\exhyphenpenalty=500 % palavras com hifem
\binoppenalty=3000 % operador binario (+, -, etc)
\relpenalty=2000 % operador relacional ( = \cong \ne, ...)
\clubpenalty=1000 % ???
\brokenpenalty=1000 % ???

\sloppy % prefere underfull do que overfull

% conjunto numerico (evitando o real, complexo, etc,

\newcommand{\Rset}{\mathbb{R}}
\newcommand{\Cset}{\mathbb{C}}
\newcommand{\Zset}{\mathbb{Z}}
\newcommand{\Iset}{\mathbb{I}}
\newcommand{\Qset}{\mathbb{Q}}
\newcommand{\Nset}{\mathbb{N}}

% Funcoes em portugues (nome da funcao deve ser em romano)
\DeclareMathOperator{\sen}{sen}
\DeclareMathOperator{\arcsen}{arcsen}
\DeclareMathOperator{\senh}{senh}
\DeclareMathOperator{\arcsenh}{arcsenh}

% Opcoes
\DeclareOption{usedsfont}{ % se usedsfont for usado como opcao
  \IfFileExists{dsfont.sty}{%
    {%
      \renewcommand{\Rset}{\mathds{R}}
      \renewcommand{\Cset}{\mathds{C}}
      \renewcommand{\Zset}{\mathds{Z}}
      \renewcommand{\Iset}{\mathds{I}}
      \renewcommand{\Qset}{\mathds{Q}}
      \renewcommand{\Nset}{\mathds{N}}
    }{%
      \PackageError{ex-a-estilo}{dsfont.sty not found.}%
      {Install dsfont package.}%
    }
  }%
}% \DeclareOption

%% Fallback (opcoes nao declaradas)
\DeclareOption*{

```

```

\PackageWarning{ex-a-estilo}{Unknown option '\CurrentOption'}
}

\ProcessOptions\relax
% fim: ex12-07.sty

```

Quando o arquivo de estilo usa opções, coloca o que vai fazer quando usar a opção, com o comando `\DeclareOption{<opção>}{<comando>}`. lembre-se que não é permitido carregar o pacote dentro da declaração de opções, mas poderemos testar se o pacote existe.

Quando tem a declaração de opções, deve colocar `\ProcessOptions` no final do arquivo para que opções fiquem prontos para o uso.

O Exemplo A.2 é um exemplo do uso do pacote do Exemplo A.1.

#### Exemplo A.2: ex-a-estilo.tex

```

\documentclass[a4paper,12pt]{book}
\usepackage[T1]{fontenc} % codificação da fonte em 8-bits
\usepackage[utf8]{inputenc} % acentuação direta
\usepackage[brazil]{babel} % em portugues brasileiro

\usepackage{ex-a-estilo} % arquivo de estilo definido pelo usuário.

% \usepackage[usedsfont]{ex12-06} % com opcao (se usar este, conjuntos numericos ficam
% como dsfont, se dsfont estiver instalado)
\begin{document}
\chapter{Teste}

\begin{definition}\label{def:triangulo:retangulo}
Um triângulo é dito retângulo se tiver um ângulo reto.
\end{definition}

\begin{theorem}[Pitágoras]\label{thm:pitagoras}
Dado um triângulo retângulo  $ABC$  com ângulo reto em  $A$ , temos que
\begin{equation}\label{eq:pitagoras}
a^2=b^2+c^2
\end{equation}
\end{theorem}

\begin{proof}
Demonstração aqui.
\end{proof}

Usando a definição do  $\sen$  e  $\cos$  juntamente com o Teorema~\ref{thm:pitagoras}, temos
que

\begin{proposition}
\l[
\forall t \in \Rset, \sen^2 t + \cos^2 t = 1
\r]

```

```
\end{proposition}
\end{document}
```

# Capítulo 1

## Teste

**Definição 1.1.** Um triângulo é dito retângulo se tiver um ângulo reto.

**Teorema 1.2** (Pitágoras). *Dado um triângulo retângulo  $ABC$  com ângulo reto em  $A$ , temos que*

$$a^2 = b^2 + c^2 \quad (1.1)$$

*Demonstração.* Demonstração aqui. □

Usando a definição do sen e cos juntamente com o Teorema 1.2, temos que

**Proposição 1.3.**

$$\forall t \in \mathbb{R}, \text{sen}^2 t + \text{cos}^2 t = 1$$

## A.2 Preenchendo o documento para teste

Para testar estilo de formatação, as vezes queremos preencher rapidamente o espaço.

Para isso existe o pacote `lipsum` que permite gerar parágrafos de texto em grego sem sentido.

O commando `\lipsum` gera em torno de uma página de texto. `\lipsum[<n>]` gera <n>  $n$ -ésimo parágrafo de texto. O comando `\lipsum[<a>-<b>]` gera o  $a$ -ésimo até  $b$ -ésimo parágrafo de texto. Veja o Exemplo A.3.

Exemplo A.3: `ex-a-lipsum.tex`

```
% \lipsum % em torno de uma página
\lipsum[5] % quinto parágrafo
% \lipsum[2-3] % de parágrafo 2 até parágrafo 3
```

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Note que `lipsum` gera somente parágrafos de textos. Para testar fórmulas e listas também, poderá usar o pacote `blindtext` que tem o comando de mesmo nome que gera um texto.

O comando `\blindtext[<n>]` gera o texto  $n$  vezes. O comando `\Blindtext` gera um parágrafo. O parâmetro opcional pode ser usado para repetir  $n$  vezes.

Tem o comando `\blindmathtrue` que ativa as fórmulas no texto gerado, mas ele só funciona em inglês (definido pelo `babel`). Assim, se não estiver em inglês, use o comando `\blindmathpaper` gera parágrafos de textos e fórmulas. Veja o Exemplo A.4.

#### Exemplo A.4: ex-a-blindtext.tex

```
% \blindmathtrue % funciona somente em ingles
% \blindtext % texto
% \Blindtext % Gera vários parágrafos
\Blindtext[1] % um parágrafo de textos
% \blinditemize % lista de itens
% \blindenumerate % lista de enumeracao
% \blinddescription % lista de descricao
% \blindmathpaper % texto e formulas.
% \blinddocument % documento com secao, subsecao, listas, etc. Nao inclui formulas
% \Blinddocument % documento longo, com texto com secao, subsecao, listas, etc. Este inclui
  formulas
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nul- lam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcor- per, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.



# Apêndice B

## Usando o Editor $\text{L}\text{Y}\text{X}$

Neste apêndice, será explicado o básico sobre o editor  $\text{L}\text{Y}\text{X}$ .  $\text{L}\text{Y}\text{X}$  é um editor do tipo WYSIWYG (o que você vê é o que você obtém) e poderá editar o documento de forma similar ao aplicativo de escritório, mas processando o documento final com o  $\text{L}\text{A}\text{T}\text{E}\text{X}$ .

### B.1 Visualização do documento final

Para compilar/ver o documento, poderá usar os formatos comumente suportados pelo  $\text{L}\text{A}\text{T}\text{E}\text{X}$ . O botão de “visualizar” (botão de olho, que fica no lado esquerdo) é usado para visualizar o documento em PDF.

Se estiver usando o pacote que precisa ser compilado para dvi como o pacote gráfico `pstrick` e `psfrag`, poderá escolher outro formato de visualização em “visualizar outros formatos” (botão que fica mais à direita, com olho e um pequeno triângulo).

O uso do pacote `hyperref` é recomendado para que o  $\text{L}\text{A}\text{T}\text{E}\text{X}$  crie links automáticos, assim como o `bookmarks` (indicadores). Os usuários de Windows, deve considerar em evitar o uso do Adobe Reader na fase de elaboração, por ser pesado e apresentar dificuldade de interagir com os editores de  $\text{L}\text{A}\text{T}\text{E}\text{X}$ . Uma boa opção é o freeware `SumatraPDF`.

Lembre-se que o  $\text{L}\text{Y}\text{X}$  permite exportar/importar arquivos fonte em  $\text{L}\text{A}\text{T}\text{E}\text{X}$ , além de vários outros formatos na qual existem conversores gratuitos do/para  $\text{L}\text{A}\text{T}\text{E}\text{X}$ .

**Dicas:** Para facilitar a navegação do documento em elaboração, é bom ativar o “Exibir->Painel de estrutura de tópicos”. Se usar o código  $\text{L}\text{A}\text{T}\text{E}\text{X}$  inserido no documento com frequência, como o do pacote `tikz`, seria bom ativar a pré-visualização na “Ferramentas->Preferências” e em “Aparências & Comportamento”, ítem “Exibição”, na “Pré-visualização Instantanea”, escolher “sem matemática”, e colocar pré-visualização no bloco que contém a caixa de  $\text{L}\text{A}\text{T}\text{E}\text{X}$  que julgar conveniente, com “Inserir->Pré-visualização”. Se quer que toda fórmula matemática tenha pré-visualização automática, poderá escolher “ligado” em vez de “sem matemática”. No entanto, as equações no modo “Instant Preview” ficará com a mesma cor do texto, tornando mais difícil de identificar rapidamente, o que não é recomendável. Para ver como ficaria o código  $\text{L}\text{A}\text{T}\text{E}\text{X}$  em edição, poderá ativar o “View->View source”. Se o texto todo ficar sublinhado em vermelho, ou não sublinha a palavra errada, pode ser que o corretor ortográfico não está

selecionado corretamente. Vá em “Ferramentas->Preferências” e em “Configurações de Idiomas”, item “Verificador Ortográfico”, escolha um verificador em “verificador ortográfico”. Também certifique de que botão de “verificar ortografia continuamente” (Botão ABC sublinhado em vermelho).

Depois de tudo pronto, “file->export” permite criar arquivo final no formato desejado.

Note que maioria dos editores de L<sup>A</sup>T<sub>E</sub>X salva o documento antes de compilar, mas o LyX não faz isso. É necessário salvar manualmente sempre que convém.

## B.2 ABC do LyX

O tutorial e guias do usuário do LyX pode ser acessado pelo “Help->Tutorial” e “Help->User’s Guide” respectivamente.

Também é importante notar que copiar/colar padrão funciona somente entre trechos de documentos do LyX.

Caso de colar trecho de código do outro aplicativo como navegador de internet ou editor de texto para o campo de comando L<sup>A</sup>T<sub>E</sub>X (botão T<sub>E</sub>X ou <ctrl>L), use o colar especial (editar->colar especial->texto simples) ou ctrl+shift+v, para não perde a quebra de linhas.

No lado esquerdo da barra de ferramentas, poderá escolher o tipo de texto (parágrafo). Alguns dos mais importantes são:

**Standard** texto normal.

**Itemize** lista não enumerada.

**Enumerate** lista enumerada.

**Chapter** Título do capítulo

**Section** Título da seção.

**Subsection** Título da subseção.

**Lyx-code** O texto como foi escrito, inclusive quebra de linhas (para colocar algoritmo, código fonte do programa, etc).

Outra forma de colocar comentário no documento é através da nota inserido por “insert->note” que é adequado para notas mais longas por poder “fechar” a “caixa de notas”.

As “caixas” como de notas, T<sub>E</sub>X, Figure, etc tem um botão junto a “caixa” e clicando nele, poderá abrir ou fechar, para facilitar a editoração. O fato de caixa estar fechada ou aberta não influencia no processamento do documento final.

Como notas não serão impressas, também serve para desativar temporariamente o trecho do documento, selecionando e clicando em “insert->note”. Para desfazer a caixa de notas ou similares, use o “dissolve->inset” acessível pelo botão direito sobre caixa de notas ou similares.

Apesar de LyX efetuar conversão automática de imagens, quando pretende exportar o código para L<sup>A</sup>T<sub>E</sub>X, é bom deixar convertido para pdf/jpg/png (para PDFL<sup>A</sup>T<sub>E</sub>X usado atualmente) ou eps (para L<sup>A</sup>T<sub>E</sub>X antigo) a fim de evitar problema de compatibilidade de imagem com o código fonte em (PDF)L<sup>A</sup>T<sub>E</sub>X.

### Acertando a configuração do documento

No “Documento”, escolha a classe de documento, margens e tamanho das fontes.

Quando compila um documento, o LyX costuma colocar uma opção de usar o pacote `amsmath`, mas no caso de inserir o comando de T<sub>E</sub>X diretamente, pode não ser detectado a necessidade de `amsmath`.

Para resolver este caso, entre em “Document->settings” e no “math Package”, tire o check do “use amsmath package automatically” e cheque em “use ams package”. O problema similar pode acontecer quanto inclui figuras, usar cores, etc dentro da caixa de L<sup>A</sup>T<sub>E</sub>X, sem estar usando fora dela. Neste caso, carregue os pacotes necessários, colocando o comando L<sup>A</sup>T<sub>E</sub>X `\usepackage` no `preamble` (Document->Settings, [LaTeX Preamble]).

Configuração do documento como estilo, margens, etc podem ser alterado pelo “Document->Settings” em qualquer momento.

Quando colar trecho de um documento para outro, pode ficar com a especificação da linguagem diferente e começar a ser sublinhado. Neste caso, selecione este trecho e no “text style” (botão com letras “ab”), escolha o “reset” na opção de [linguagem].

### Inserindo o comando de L<sup>A</sup>T<sub>E</sub>X

Para colocar pacotes adicionais, definições, configurações adicionais, etc que ainda não está disponível no LyX (ou que não tem paciência de descobrir como fazer), deverá ser colocado no campo de “preamble” em “Document->settings, [LaTeX Preamble]”, usando o comando de L<sup>A</sup>T<sub>E</sub>X. No preamble, deverá usar acentuação no modo T<sub>E</sub>X para evitar problemas de portabilidade.

Quando precisar colocar um comando extra de L<sup>A</sup>T<sub>E</sub>X, clique no botão “T<sub>E</sub>X” ou pressione `<ctrl>L` na qual insere uma caixa de comando L<sup>A</sup>T<sub>E</sub>X, podendo colocar qualquer comando válido de L<sup>A</sup>T<sub>E</sub>X. Para evitar problemas, coloque um espaço (ou par de chaves) no final do campo de comandos de L<sup>A</sup>T<sub>E</sub>X, caso finalizar com o comando L<sup>A</sup>T<sub>E</sub>X.

O comando L<sup>A</sup>T<sub>E</sub>X pode ficar divididos. Por exemplo, uma caixa contém `\begin{ambiente}` e outra caixa contém `\end{ambiente}` para colocar trecho do código no ambiente desejado.

Note que o campo de fórmulas também aceitam diretamente os comandos de L<sup>A</sup>T<sub>E</sub>X.

### Para letras acentuadas fora do corpo do documento

O campo exibido na janela separada como no caso do preamble (Document->Settings, [LaTeX preamble]), entrada de índice remissivo (insert->index entry), etc usam a exibição nativa do sistema. Assim poderá apresentar problemas na exibição ou compilação quando muda de sistema operacional. Para evitar problemas de portabilidade, use a acentuação pelo comandos T<sub>E</sub>X nestes casos. Caixa de código L<sup>A</sup>T<sub>E</sub>X (inserido com botão T<sub>E</sub>X) não apresenta este tipo de problemas.

## B.3 Formatando textos

Para alterar o tipo de fontes do texto, selecione o trecho e clique no botão “*abc*”. Nele tem a opção de escolher a família, forma e peso.

O **emph** fica na opção **misc** e **language** permite escolher idioma neste trecho. Note que `<ctrl>b` é atalho para negrito e `<ctrl>e` é atalho para enfatizar.

A opção “reset” de cada item restaura o item correspondente como padrão do documento.

O botão “A com seta redonda” aplicará a última configuração. Para saber o que cada botão faz, posicione o mouse sobre o botão e deixe parado para aparecer o texto informativo.

Para criar um novo parágrafo ou linhas:

`<ENTER>` cria novo parágrafo

`<control>+<ENTER>` quebra de linha (sem parágrafo)

`<control>+<space>` espaço forçado (~ de  $\LaTeX$ ).

Note que, para inserir linhas vazias, deve alterar (quebras de linhas ou parágrafo) com o espaço forçado.

## B.4 Matemática

Fórmulas matemáticas dentro do texto denominado de ```inlinestyle''` (`textstyle`) pode ser inserido, clicando no ícone de somatório  $\sum$  na barra de ferramentas, ou com menu “insert->math->inline formula” ou com a tecla `<control>M`.

Para inserir símbolos e fórmulas, podemos usar o painel matemático ativado quando o cursor está dentro da caixa de fórmulas. Para criar fórmulas de maneira rápida, é recomendado memorizar alguns comandos básicos de  $\LaTeX$  que podem ser digitados (em vez de clicar no mouse).

Note que o LyX tem dificuldade em lidar com o parâmetro opcional no modo matemático, dificultando o uso da forma `\sqrt[n]{x}`. Para contornar, foi definido o comando `\root` no LyX que não é definido por padrão no  $\LaTeX$ .

Para descobrir nomes de comandos  $\LaTeX$  que estão no “Painel Matemático”, posicione o mouse sobre símbolos no “Math Panel” (mostrado quando o cursor está na fórmula) que exibirá o nome de comando.

Uma fórmula numa linha independente denominado de “displaystyle” pode ser criado com `Control+Shift+M` ou pelo menu “insert->math->Display formula”.

Para sair rapidamente da fórmula, pressione o “ESC” que posicionará o cursor logo após a fórmula em edição.

Para converter tipo de fórmulas já digitadas, clique o botão direito sobre a fórmula e escolha o novo tipo de fórmula. Também pode posicionar o cursor na fórmula e use o “edit->math->change formula type”.

A função por partes pode ser criado com “cases”, inserido com botão direito do mouse sobre a fórmula, ou pelo menu “insert->math->cases”.

Nas fórmulas multi linhas, <control>+ENTER abre uma nova linha.

As fórmulas que ocupa uma linha independente tais como “Display fórmula” (fórmula no modo “displaystyle”), “AMS align environment”, etc podem ser enumerados automaticamente.

Para ativar/desativar enumeração na equação com linha independente, clique no botão direito do mouse sobre a fórmula, ou use o menu “edit->math->toggle numbering”.

Para referenciar equações ou elementos enumerados, use o botão de etiqueta para inserir rótulo e referência cruzada (que fica no lado) para inserir referências.



# Apêndice C

## Para Organizadores do Evento

Aqui, veremos alguns pacotes úteis para organizadores de eventos. Um deles é a mala direta no  $\LaTeX$ , útil para emitir certificados e outro é para criar caderno de trabalhos apresentados.

### C.1 Mala direta no $\LaTeX$

Certificados de congressos científicos pode conter fórmulas nos títulos, o que complica a mala direta nos aplicativos de escritórios. Assim, costumamos efetuar mala direta no  $\LaTeX$ .

Mala direta pode ser efetuado pelo `datatool` que lê os dados de um arquivo CSV. Após carregar o `datatool`, comece configurando o separador de colunas e delimitador de campos.

```
\DTLsetseparator{,} % Separador de campos
\DTLsetdelimiter{"} % delimitador de celulas
```

Para acessar o arquivo CSV, usa-se o comando `\DTLloaddb{lista}{ex-c-lista-nomes.csv}` para associar o arquivo `ex-c-lista-nomes.csv` na `lista`. Assim, ao referenciar `lista`, estará referenciando o arquivo `ex-c-lista-nomes.csv`

A primeira linha do arquivo `ex-c-lista-nomes.csv` é assumido que é título, isto é., contém nome das colunas. Caso não tenha linha de título, poderá providenciar como em `\DTLloaddb[noheader,keys]`. Neste caso, está indicando que não há linha de título no arquivo e está colocando nomes `Nome` e `Trabalho` na primeira e segunda coluna do arquivo de dados.

Agora podemos efetuar um laço e repetição por algo como

```
\DTLforeach{lista}{%
\personname=Nome,\worktitle=Trabalho}{%
...
} % \DTLforeach
```

O primeiro parâmetro do comando `\DTLforeach` é a lista de associação de comandos com valor do campo. `\personname=Nome` e `\worktitle=Trabalho` associa o campo (dado da coluna) `Nome` para o comando `\personname` e o campo `Trabalho` para o comando `\worktitle`. O segundo parâmetro são comandos a serem executados. O arquivo de entrada e como do Exemplo C.1

## Exemplo C.1: ex-c-lista-nomes.csv

```
"Nome","Trabalho","Observacao"
"Nome 1","Trabalho 1",
"Nome 2","Trabalho 2",
"Nome 3","Trabalho 3",
...
```

Agora, o certificado pode ter enfeites como molduras, logotipos, etc.

Ele será diagramado para papel a5, para ser impresso no papel a4 ampliando-o. Fonte também será alterado para `bookman` que é uma serifa grossa, ideal para letras maiores como este. Veja o Exemplo C.2.

## Exemplo C.2: ex-c-certificado.tex

```
\documentclass[12pt,a5paper,landscape]{article}
\usepackage[utf8]{inputenc}
\usepackage{graphicx} % para incluir logo
\usepackage{xcolor} % para cor
\usepackage{bookman} % Fonte bookman

\usepackage{datatool} % para mala direta

\usepackage[lmargin=1cm,tmargin=1cm,bmargin=1.8cm,rmargin=1.8cm]{geometry} %
  margins
\usepackage{fancybox} % para colocar moldura na pagina

\usepackage{tikz} % to produce custom shadow box
% \usetikzlibrary{shadows}
% \usetikzlibrary{backgrounds}
\usetikzlibrary{calc} % para calculo de medidas
\usepackage[none]{hyphenat} % sem hifenizacao

\sloppy % prefere underfull do que overfull

% Parametros para sombra
\newcommand{\shadowwidth}{2pt}
\newcommand{\shadowheight}{2pt}
\newcommand{\shadowcolor}{black!30}

\newcommand{\borderlinecolor}{black}
\newcommand{\borderlinewidth}{2pt}

% reimplementando "shadow box" usando tikz
% pois do fancybox nao suporta cores
% Como \settowidth nao funciona bem para pagina,
% largura da caixa sera passado como parametro opcional
% (valor padrao eh \textwidth)
% \settowidth{certificatetitle}{#1} causa perda e conteudos.
```

```

% Assim, configurando pelo \textwidth
\newlength{\certificateboxwidth}
\newcommand{\certificatebox}[2][\textwidth]{%
%\settowidth{\certificateboxwidth}{#2}%
\setlength{\certificateboxwidth}{#1}%
\addtolength{\certificateboxwidth}{2\fbboxsep}
\begin{tikzpicture}
  % framed box
  \node[draw=\borderlinecolor,line width=\borderlinewidth, rectangle] (certificate)
    {\parbox{\certificateboxwidth}
      {\vspace*\fbboxsep}\hspace*\fbboxsep#2\hspace*\fbboxsep\vspace*\fbboxsep}}
  };
  % shadow
  \draw[fill=\shadowcolor, draw=none]
    ($(certificate.south west) + (\shadowwidth, -\shadowheight)$)
      rectangle ($(certificate.south east)+(\shadowwidth,0)$) ;
  \draw [fill=\shadowcolor,draw=none]
    ($(certificate.south east)+(0,-\shadowheight)$)
      rectangle ($(certificate.north east)+(\shadowwidth,-\shadowheight)$) ;
\end{tikzpicture}
} % \newcommand

\fancypage{\setlength{\fbboxsep}{8pt}\certificatebox{}}

% titulo do certificado com sombra
\newlength{\certtitlewidth}
\newcommand{\certtitle}[1]{%
\settowidth{\certtitlewidth}{#1}%
\boxput*{#1}{\parbox{\certtitlewidth}%
{\vspace*\shadowheight}\hspace*\shadowwidth}%
\color{\shadowcolor}{#1}}%
}

\pagestyle{empty}

\begin{document}
\DTLsetseparator{,} % Separador de campos
\DTLsetdelimiter{”} % delimitador de celulas

% associa lista para o arquivo CSV
% \DTLloaddb[noheader,keys={Nome,Trabalho}]{lista}{lista.csv}
\DTLloaddb{lista}{latex-via-exemplos-lista-nomes.csv}

\DTLforeach{lista}{% processa cada item da lista
\personname=Nome,\worktitle=Trabalho}{%
% certificado

% incluindo a imagem de fundo (marca d’agua)
%\begin{flushleft}
%\noindent

```

```

% \unitlength 0.04\textwidth
% \begin{picture}(0,0)(0,15)
% \includegraphics[width=1.0\textwidth]{fundo}
% \end{picture}
% \end{flushleft}

% \sffamily

% timbre
\begin{center}
%\includegraphics[width=0.1\textwidth]{logo-esquerda}
\hfill
\begin{minipage}[b]{0.6\textwidth}
\center
Universidade Federal de São Carlos \\
Centro de Ciências Exatas e de Tecnologia \\
Departamento de Matemática
\end{minipage}
%\hfill
%\includegraphics[width=0.1\textwidth]{logo-direita}
\hfill~
\vfill
\end{center}

% titulo do certificado
\begin{center}
\certtitle{\scalebox{2}{\LARGE Certificado}}
\end{center}
\vfill

% corpo do certificado
\noindent
{\Large
Certificamos que \textit{\MakeUppercase\personname} apresentou o trabalho intitulado \
\textit{"\worktitle"} no \textsc{Nome do Congresso}, realizado no período de DATA, em
LOCAL.}
\vfill

\begin{flushright}
{\Large LOCAL E DATA.}
\end{flushright}
\vfill

% campo de assinatura, etc
\begin{minipage}{0.45\textwidth}
Realiza\c{c}\~ao: ORGANIZADORES
\end{minipage}
%
\begin{minipage}{0.45\textwidth}
\centering \noindent

```

```

\hrule{\hspace*{0.95\textwidth}} \\
{\Large Comiss\~ao Organizadora}
\end{minipage}

\newpage
} % csvreader
\end{document}

```

Um dos certificados gerados é como segue.



Note que, o que deixou o código complexo foi o sombreamento.

## C.2 Caderno de resumos

Quando recebe vários artigos e quer “encadernar”, poderá usar a classe `combine` em vez de gerar PDF de cada um e grudar. A classe `combine` agrupa os artigos automaticamente em um único documento, gerando “caderno” de artigos.

Para usar o `combine`, todos documentos que serão incluídos nele devem estar usando a mesma classe que podem ser `memoir`, `book`, `report`, `letter` ou `article`. As classes não suportadas oficialmente como o `amsart` pode precisar de ajustes manuais para funcionar perfeitamente.

A classe `combine` requer mais recursos que o documento normal. Assim, para aumentar memória reservada para isso, coloque

```

\usepackage{etex} % to use etex extensions
\reserveinserts{28} % using etex extensions

```

no preamble se ocorrer o erro do tipo “No room for ...”. Se ainda ocorrer erro, aumente o valor 28 para 50, por exemplo.

Note que `\maketitle` ou `titlepage` é necessário. Se não existir, causa erros.

Para gerar sumário de artigos incluídos, carregue o pacote `combine`. Para usar recursos do `natbib`, carregue o pacote `combnat` no lugar de `natbib`.

O modo padrão é ignorar os pacotes carregados pelos artigos. Assim, todos pacotes necessários devem estar carregados no arquivo mestre (arquivo do `combine` que vai incluir artigos).

Vamos supor que artigos a serem incluídos tem a forma como no Exemplo C.3.

#### Exemplo C.3: ex-c-conferencia1.tex

```
documentclass[10pt,a4paper]{article}
% Este arquivo sera processado automaticamente por um programa.
% Por favor nao altere nada no preamble (ate \begin{document} ).
\usepackage[utf8]{inputenc}
\usepackage[english,brazil]{babel}
\usepackage{amsmath}
\usepackage{amssymb} % Ele carrega amsfonts tambem
\usepackage{enumerate}
% Por favor nao inclua outros "packages".
\begin{document}
\title{Título da Conferência 1}
\author{Autor da conferência 1}
\maketitle

Texto da conferência 1
% Incluir bibliografia e opcional. Se decidir inclui-la use o formato abaixo

%\begin{thebibliography}{99}
%\bibitem{key1} Lamport, L. \emph{LaTeX: A Document Preparation System}, Addison-
Wesley, \textbf{1986}.
%\end{thebibliography}
\end{document}
```

Então o caderno, pode ser criado como no Exemplo C.4.

#### Exemplo C.4: ex-c-caderno.tex

```
\documentclass[12pt,a5paper]{combine}
% combine supports: memoir, book, report, and article

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% If the error "No room for .." occur, increase the reserved insets
\usepackage{etex} % to use etex extensions
\reserveinsets{28} % using etex extensions

% To create table of cotents automatically
\usepackage{combine}
```

```

% Todos pacotes usados nos artigos a ser importados
\usepackage[utf8]{inputenc}

\usepackage{amsmath}
\usepackage{amssymb} % it's include amsfonts too
\usepackage{enumerate}
\usepackage[english,brazil]{babel}

% packages usados somente no corpo de anais do congresso
\usepackage{graphicx}
\usepackage{geometry}
\geometry{lmargin=2.5cm,rmargin=2cm,tmargin=2cm,bmargin=2cm}

\sloppy % prefer underfull
% \fussy % prefer overfull

% No combine, chamada de \maketitle é obrigatorio.
% Assim, quando faz a capa manualmente, deverá desabilitar isto
\makeatletter\c@lmtitleempty\makeatother

\begin{document}
% capa
\begin{titlepage}
\begin{center}
\thispagestyle{empty}
\begin{flushleft}
\noindent
\unitlength=0.04\textwidth
\begin{picture}(0,0)(2,33)
% \includegraphics[width=1.16\textwidth]{fundo}
\end{picture}
\end{flushleft}
{\Large T\`ITULO DO CONGRESSO} \\
HOMENAGEM, ETC.

\vfill
{\huge Resumos de apresenta\c{c}oes \\
(Abstract of talks and posters)}
\vfill
DATA
\vfill
LOCAL
\vfill
\end{center}
\end{titlepage}

% contra capa
\clearpage
\thispagestyle{empty}

```

```

\noindent
{\bfseries Comitê Científico (Scientific Committee)} \\
Membro 1 (Instituição 1) \\
Membro 2 (Instituição 2)

\

\noindent
{\bfseries Comissão Organizadora (Organizing Committee)} \\
Membro 1 (Instituição 1) \\
Membro 2 (Instituição 2)

\

\begin{flushright}
  \begin{minipage}{0.9\textwidth}
    Realização (Hosted by): \\
    INSTITUIÇÕES
  \end{minipage}
\end{flushright}

\clearpage
\newpage

% Table of contents
% \pagestyle{combine}
\pagestyle{plain}

\tableofcontents
\clearpage
~ % um espaço para gerar página não vazia
\newpage % pulando uma página

% Importa artigos de conferências
% Usar o comando \import em vez de \input ou \include
\begin{papers}
\import{conferencia1}
\import{conferencia2}
...
\end{papers}

\end{document}

```

Note que foi diagramado no papel A5 para poder efetuar **booklet** (colocar duas páginas em uma, reordenar para grampear no meio) no papel A4. O processo de **booklet** costuma ser feito no aplicativo como o **jPDFtweak** (<http://jpdftweak.sourceforge.net/>).

# Apêndice D

## Para Professores

Será tratado algumas questões úteis aos professores.

### D.1 Lista de exercícios e provas

Nas provas ou listas de exercícios, os itens são curtas. Neste caso, é aconselhável que coloque mais de um item por linha, mas mantendo alinhamento.

Existem vários pacotes para elaboração de lista de exercícios e provas. Aqui será apresentado uma delas. O pacote `exsheets` implementa o ambiente `question` para produzir questionário enumerado e o pacote `tasks` permite colocar vários itens em uma única linha.

Veja o Exemplo D.1.

Exemplo D.1: `ex-d-exsheets.tex`

```
\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\usepackage{tasks} % cria lista curta
\usepackage{exsheets} % cria questoes
\SetupExSheets[question]{type=exam} % configurando

\begin{document}
% O ambiente question produz questões.

\begin{question}
Qual item não é metal?
% itens da questão será produzido pelo ambiente tasks
\begin{tasks}(4) % 4 itens por linha
\task ferro
\task carbono
\task cobre
\task mercúrio
\end{tasks}
\end{question}
```

```

\end{question}

% Alterando a enumeração de tasks
\settasks{
  counter-format=(tsk[r]), % entre parenteses, em romano minuscuro
  label-width=4ex % largura reservada para rótulo
}

\begin{question}
  O que é mamífero?
  \begin{tasks}(2) % 2 itens por linha
    \task Animais que botam ovos.
    \task Animais que amamentam.
    \task Animais que voam.
    \task Animais com 4 patas.
  \end{tasks}
\end{question}
\end{document}

```

**Exercício 1.**

Qual item não é metal?

- a) ferro                      b) carbono                      c) cobre                      d) mercúrio

**Exercício 2.**

O que é mamífero?

- (i) Animais que botam ovos.                      (ii) Animais que amamentam.  
 (iii) Animais que voam.                      (iv) Animais com 4 patas.

## D.2 Com e sem respostas

As vezes, queremos gerar versões com resposta e outro sem, ou que resposta fique no final do documento, mas que seja diagramado junto as questões. Neste caso, podemos usar o pacote `answers` (existem vários pacotes para tal propósito). Veja o Exemplo D.2.

Exemplo D.2: `ex-d-answers.tex`

```

\documentclass[12pt,a4paper]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}

\usepackage{answers} % resposta no arquivo separado

```

```

%\usepackage[nosolutionfiles]{answers} % resposta no lugar

\usepackage{amsthm}

% "sol" é ambiente a ser usado. "Solution" não deve ser modificada. A resposta sera gravada
% no arquivo referenciado internamente por "ans"
\Newassociation{sol}{Solution}{ans}
% \renewcommand{\solutionextension}{ans}

\newtheorem{ex}{Exercício}[section] % ambiente de exercicios

\begin{document}
% inicia a gravacao da resposta no arquivo "ans" cuja nome externo e \jobname-ans (\
% jobname e nome do arquivo atual)
\Opensolutionfile{ans}[\jobname-ans]

\section{Problemas}

\begin{ex}
Primeiro exercício.
\begin{sol}
Solução do primeiro exercicio.
\end{sol}
\end{ex}

\begin{ex}
Segundo exercício.
\begin{sol}
Solução do segundo exercício.
\end{sol}
\end{ex}
\Closesolutionfile{ans} % finaliza a gravação das respostas

\section{Soluções}
\Readsolutionfile{ans} % colocar solucoes do "ans" aqui.
\end{document}

```

## 1 Problemas

**Exercício 1.1.** *Primeiro exercício.*

**Exercício 1.2.** *Segundo exercício.*

## 2 Soluções

**1.1** Solução do primeiro exercício.

**1.2** Solução do segundo exercício.

# Apêndice E

## Para Projetos

Aqui será tratado de pacotes úteis ao desenvolvimento de documentos grandes como livros.

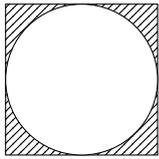
### E.1 Pacote `standalone`

Quando cria muitas ilustrações e diagramas, as vezes deixamos estas ilustrações como arquivo separado para compilar e ajustar mais rapidamente. Por exemplo, cada ilustração em `tikz` pode estar em um único arquivo e o documento principal inclui eles. Neste caso, não podemos usar `\input`, nem `\include` pois a ilustração tem sua própria classe de documento. Neste caso, podemos usar a classe `standalone`. Cada ilustração de `tikz` usará a classe `standalone` em vez do `article` ou similar. Assim, podemos compilar e ajustar a ilustração quando bem entender. No documento principal, use o pacote `standalone` e demais pacotes que arquivos inclusos podem precisar. Para incluir a ilustração feito pela classe `standalone`, basta usar `\input` ou similar.

A figura diagramado com a classe `standalone` é como do Exemplo E.1.

Exemplo E.1: `ex-e-standalone-fig.tex`

```
\documentclass[12pt,a4paper]{standalone}
\usepackage[utf8]{inputenc}
\usepackage{tikz}
\usetikzlibrary{patterns}
\begin{document}
\begin{tikzpicture}
\draw[pattern=north east lines] (-1,-1) rectangle(1,1);
\draw[fill=white] (0,0) circle(1);
\end{tikzpicture}
\end{document}
```



O documento que usa a figura, usa o pacote `standalone` e usa o `\input` para incluir o arquivo de figura, como no Exemplo E.2.

#### Exemplo E.2: `ex-e-standalone.tex`

```
\documentclass[12pt,a4paper]{article}

\usepackage{standalone} % para incluir arquivos diagramado com standalone

\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}

\usepackage{amsmath}
\usepackage{amssymb}

% pacotes usados pelo arquivo a ser incluído
\usepackage{tikz}
\usetikzlibrary{patterns}

\begin{document}
Para incluir o arquivo diagramado com classe standalone, basta usar o \verb+\input+.

\begin{figure}[hbp!]
\center
\input{ex-e-standalone-fig}
\caption{Inclusão de figuras}
\end{figure}

\end{document}
```

Para incluir o arquivo diagramado com classe standalone, basta usar o `\input`.

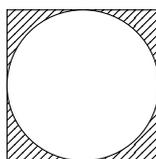


Figura 1: Inclusão de figuras

## E.2 Dividindo o documento em vários arquivos

Para dividir o documento em vários arquivos, poderá usar o `\include`, mas ele não é eficiente, pois os pedaços de documentos não podem ser compilados separadamente. O pacote `subfiles` resolve este problema. o uso de `subfiles` é similar ao do `standalone`, mas em vez de ignorar o `preamble` dos arquivos incluídos, os arquivos a serem incluídos usam o `preamble` do documento principal. Assim, como o `standalone`, o documento principal deve conter todos os pacotes e definições que suas partes vão precisar, mas não há necessidade de colocar tais pacotes e definições no arquivo de partes. A estrutura do documento principal com `subfiles` é algo como do Exemplo E.3.

Exemplo E.3: `ex-e-subfiles-principal.tex`

```
\documentclass[12pt,a4paper,oneside]{book}
\usepackage{subfiles} % para incluir partes do documento

\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[brazil]{babel}
\usepackage{amssymb,amsmath}
\usepackage{tikz}
\usetikzlibrary{patterns}
...
\begin{document}
...
\subfiles{ex-e-subfiles-prefacio}
\subfiles{ex-e-subfiles-capitulo1}
\subfiles{ex-e-subfiles-capitulo2}
\subfiles{ex-e-subfiles-capitulo3}
...
\end{document}
```

Exceto pelo uso do pacote `subfiles` e o comando `\subfiles` em vez do `\include`, tem nada de especial. Cada pedaço do documento tem a forma como do Exemplo E.4.

Exemplo E.4: `ex-e-subfiles-capitulo1.tex`

```
\documentclass[doc-principal]{subfiles}
\begin{document}
...
\end{document}
```

Note que a classe de documento é `subfiles` e sua opção é nome do arquivo principal. Como o `preamble` será obtido do documento principal, não há `preambles` e já começa a escrever o documento no ambiente `document` normalmente.

Assim, podemos compilar partes do documento, assim como todo documento.



---

## Referências Bibliográficas

- [GMS04] Michel Goossens, Frank Mittelbach, and Alexander Samarin, *The L<sup>A</sup>T<sub>E</sub>X companion (second edition)*, Adilson–Wesley, Reading, MA, 2004.
- [Knu86] Donald E. Knuth, *The T<sub>E</sub>X book*, third ed., Adilson–Wesley, Reading, MA, 1986.
- [Lam86] Leslie Lamport, *L<sup>A</sup>T<sub>E</sub>X, a document preparation system*, Adilson–Wesley, Reading, MA, 1986.
- [Lam94] ———, *L<sup>A</sup>T<sub>E</sub>X, a document preparation system, second edition*, Adilson–Wesley, Reading, MA, 1994.
- [OPHS18] Tobians Oeticker, Hubert Partl, Irene Hyna, and Elisabeth Schlegl, *The not so short introduction to L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>*, Comprehensive T<sub>E</sub>X Archive Network, 2018, URL: <https://ctan.org/tex-archive/info/lshort/>.
- [Tan] Till Tantau, *TikZ&pgf manual*, Download from <http://www.ctan.org/tex-archive/graphics/pgf/base/doc/generic/pgf/pgfmanual.pdf>.
- [Tea00] Tutorial Team, *Online tutorials on L<sup>A</sup>T<sub>E</sub>X*, 2000, URL: <http://www.tug.org/tutorials/tugindia/>.
- [wik18] wikibook, *L<sup>A</sup>T<sub>E</sub>X*, wikibook, 2018, URL: <https://en.wikibooks.org/wiki/LaTeX>.



# Índice Remissivo

- \!, 84
- \+, 20
- ±, 8
- \,, 84
- \-, 20, 95
- \:, 84
- \;, 84
- |, 16
- [, 86
- \$, 7
- \$\$, 7
- %, 6
- &, 16, 31
- ~, 8
- \_ , 8
- \, 4, 12
- \(, 7
- \), 7
- \[, 7
- \\, 12, 16, 31
- \], 7
- ~, 6, 10
- 12pt, 4
  
- \a', 20
- a4paper, 4
- \a=, 20
- \a`, 20
- \abovedisplayskip, 83
- abstract, 46
- \abstractname, 91
- acentuação
  - modo matemático, 38
  - modo TeX, 6
- acentuação direta, 6
- \acwgapcirclearrow, 139
- \addbibresource, 127
- \addcontentsline, 112
- \addfontfeature, 155
- \addto, 92
- \addtocounter, 86
  
- afterpage, 52
- align, 36
- align\*, 37
- aligned, 34
- alinhamento
  - centralizado, 11
  - direita, 11
  - esquerda, 11
  - justificado, 11
- \Alph, 86
- \alph, 86
- \smallskip, 83
- ambiente, 4, 43
  - múltiplas linhas, 44
  - parâmetro opcional, 43
  - parâmetro, 43
- amsmath, 29
- amssymb, 29
- amsthm, 53
- \ang, 116
- answers, 184
- \appendix, 48
- \appendixname, 91
- apêndice, 48
- apóstrofos, 6
- \ar, 139
- \arabic, 86
- \arccos, 73
- \arcsin, 73
- \arctan, 73
- \arg, 73
- arquivo
  - bbl, 79
  - bst, 79
  - referências bibliográficas, 77
- arquivos
  - sty, 161
- array, 15
- article, 4
- aspas, 6
- atop

- `\atop`, 37
- `\author`, 48
- babel, 4, 90
- `\backmatter`, 49
- base, 103
- beamer, 145
- beamerposter, 144
- `\begin`, 4
- `\belowdisplayskip`, 83
- bfseries, 70
- biber, 127
- `\bibitem`, 75
- biblatex, 126
  - alphabetic, 127
  - authortitle, 127
  - authoryear, 127
  - draft, 127
  - LANGID, 156
  - numeric, 127
  - reading, 127
  - romanized, 156
  - translated, 156
  - verbose, 127
- `\bibliographystyle`, 79
- `\bibname`, 91
- BibTeX, 77
  - `#`, 78
  - abbrv, 79
  - alpha, 79
  - amsalpha, 79
  - amsplain, 79
  - `@`, 78
  - `@article`, 79
  - `@book`, 78
  - `@comment`, 79
  - `@manual`, 79
  - `@misc`, 79
  - `@phdthesis`, 79
  - `@string`, 78
  - comentário, 79
  - plain, 79
  - unsrt, 79
  - url, 111
- `\Big`, 33
- `\big`, 33
- `\Bigg`, 33
- `\bigg`, 33
- `\bigskip`, 83
- `\binom`, 38
- `\blindmathpaper`, 167
- `\blindmathtrue`, 167
- `\Blindtext`, 167
- blindtext, 167
- `\blindtext`, 167
- block, 150
- bm, 72
- `\bm`, 72
- Bmatrix, 31
- bmatrix, 31
- `\bmod`, 73
- booklet, 182
- brazil, 92
- brazilian, 92
- calligra, 113
- caption, 109
- `\caption`, 58, 109
- `\caption*`, 60, 109
- `\captionn<idioma>`, 92
- `\captionsetup`, 109
- capítulo, 45
- caracteres especiais, 5
- cases, 31
- `\cdots`, 34
- center, 11
- `\center`, 11
- `\chapter`, 45
- `\chapter*`, 46
- `\chaptername`, 91
- `\circlearrowleft`, 139
- citações, 21
- `\cite`, 76
- `\citep`, 125
- `\citet`, 125
- `\clearcaptionsetup`, 110
- `\cleardoublepage`, 51
- `\clearpage`, 51
- `\ClearShipoutPicture`, 122
- `\cline`, 17
- color, 101
  - named, 102
  - usenames, 101
- `\color`, 102
- `\colorbox`, 102
- `\colorlet`, 120, 129
- `\columnsep`, 85
- `\columnseprule`, 85
- `\columnwidth`, 85
- com serifa, 69
- comando
  - múltiplas linhas, 44

- comandos
  - com parâmetros, 42
  - nome das funções, 73
  - parâmetro opcional, 42
- comandos frágil, 89
- combinat, 180
- combine, 179
- combinet, 180
- comentário, 6
- Computer Modern, 112
- \contentsname, 91
- \cos, 73
- \cosh, 73
- \cot, 73
- \coth, 73
- \csc, 73
- custom-bib, 126
- código fonte
  - programa, 22
- datatool, 175
- \date, 48
- dcolum, 17
- \DeclareCaptionFormat, 110
- \DeclareMathOperator, 41
- \DeclareMathOperator\*, 41
- \DeclareOption, 165
- \defaultfontfeatures, 155
- \defaultfontfeatures\*, 155
- \define@key, 98
- definir comandos, 41
- \deg, 73
- delimitador
  - ., 32
  - \{, 32
  - \}, 32
  - [, 32
  - ], 32
  - \lanagle, 32
  - \lceil, 32
  - \lfloor, 32
  - |, 32
  - \|, 32
  - (, 32
  - ), 32
  - \rangle, 32
  - \rceil, 32
  - \right, 32
  - \ulcorner, 32
  - \urcorner, 32
- \depthof, 86
- \det, 73
- \dim, 73
- \DisemulatePackage{setspace}, 104
- displaymath, 7
- displaystyle, 7
- documentclass, 3
- \documentclass, 4
- Donald Kunuth, 1
- \dotfill, 84
- \doublebox, 119
- \doublespacing, 103
- \draw, 131
- dscruption, 13
- \DTLforeach, 175
- \DTLloaddb, 175
- elemento
  - flutuante, 57
- \else, 98
- em, 13, 70
- \em, 13
- \emph, 13, 70
- \end, 4
- ênfase de texto, 13
- \enlargethispage, 52
- \enskip, 84
- \enspace, 84
- \ensuremath, 41
- enumerate, 13, 15
- enumi, 87
- enumii, 87
- enumiii, 87
- enumiv, 87
- equation, 9
- equation\*, 37
- eso-pic, 122
- espaço
  - depois do comando, 6
  - não quebrável, 6
- estilo da página
  - plain, 48
- estilo literário
  - francês, 105
- \exp, 73
- \expandafter, 99
- \ExplSyntaxOff, 159, 161
- \ExplSyntaxOn, 159, 161
- exsheets, 183
- fancy, 107
- fancybox, 119

- fancyhdr, 106
- \fancypage, 119
- \fancytext, 107
- \fancytext, 107
- \fbox, 25, 58, 119
- \fcolorbox, 102
- \fi, 98
- figura
  - flutuante, 57
- figure, 57
- \figurename, 91
- \fill, 133
- float, 113
  - b, 58
  - !, 58
  - H, 113
  - h, 58
  - p, 58
  - t, 58
- floatflt, 114
- flowfram, 144
- flowframtk, 144
- flushleft, 11
- \flushleft, 11
- flushright, 11
- \flushright, 11
- fncychap, 107
  - Bjarne, 107
  - Bjornstrup, 107
  - \ChNumVar, 107
  - \ChTitleVar, 107
  - Conny, 107
  - Glenn, 107
  - Lenny, 107
  - Rejne, 107
  - Sonny, 107
- \fnsymbol, 86
- fonte
  - comandos antigos, 71
  - Computer Modern, 69
  - ênfatisado, 70
  - formato normal, 70
  - itálico, 70
  - negrito, 70
  - nome espaçado, 69
  - não negrito, 70
  - restaurar padrão, 71
  - romano, 69
  - saml caps, 70
  - sem serifa, 69
  - tamanho, 71
- fontenc, 4
- fontspec, 154
- \fontspec, 154
- footnote, 12
- \foreach, 136
- \foreignlanguage, 90
- \frac, 8
- fração, 8
- frame, 145, 146
- \frame, 145, 146
- framed, 120
- \frametitle, 146
- \frontmatter, 48
- \fussy, 94
- fórmula
  - displaystyle
  - meio do texto, 9
  - scriptstyle, 9
  - textstyle, 9
- gather, 35
- gather\*, 37
- \gcd, 73
- geometry, 103
  - bmargin, 103
  - lmargin, 103
  - rmargin, 103
  - tmargin, 103
- \geometry, 103
- graphicx, 61
- headings, 104
- \heightof, 86
- \hfill, 83
- hhline, 17
- hifenização, 90
- \hline, 16
- \hom, 73
- \hrulefill, 84
- \haspace, 83
- \hspace\*, 83
- Huge, 72
- huge, 72
- \hyperlink, 111
- hyperref, 110
- \hypertarget, 111
- hyphenat, 104
- \hyphenation, 95
- i sem pingão, 6

- idioma, 90
- \@ifclassloaded, 162
- \if@compatibility, 161
- \IfFileExists, 161
- iffont, 155
- \iffontexists, 156
- \IfFontExistsTF, 155
- \iffontsexist, 156
- \@ifstar, 96
- iftex, 152
- \ifx, 98
- imagem externa, 61
- imagem PDF
  - múltiplas páginas, 62
- \imath, 40
- \includegraphics, 61
  - parâmetro opcional, 62
- indentfirst, 52
- \index, 80
- \indexname, 91
- índice remissivo, 80
- índice, 8
- índice remissivo
  - com chave, 80
  - sub-entrada, 80
- índice remissivo
  - trecho, 81
- \inf, 73
- inkscape, 144
- inlinestyle, 7
- \input, 187
- inputenc, 4
- \item, 13
- itemize, 13
- itshape, 70
  
- jabref, 77, 80
- \jmath, 40
- jPDFtweak, 144
- justify, 104
- \justify, 104
  
- \ker, 73
- \kill, 20
  
- \label, 10, 58
- landscape, 115
- LARGE, 72
- Large, 71
- large, 71
- LaTeX, 1
- Latin Modern, 112
- \ldots, 34
- \left, 32
- \left., 32
- Leslie Lamport, 1
- \let, 43
- letra
  - grega, 8
- lettrine, 105
- \lg, 73
- \lim, 73
- \liminf, 73
- limitante
  - duas linhas, 37
- \limsup, 73
- \linebreak, 12
- lineno, 120
- \linenumbers, 120
- \linespread, 94
- \linewidth, 19, 26, 85
- linhas em branco, 12
- lipsum, 121, 166
- \lipsum, 121, 166
- list, 15
- lista, 13
  - descrição, 13
  - enumerada, 13
    - controle, 15
  - item, 13
  - itemizada, 13
    - marca de itens, 101
- lista de figuras, 60
- lista de tabelas, 60
- \listfigurenname, 91
- listings, 122
- listingsutf8, 123
- \listoffigures, 60
- \listoftables, 60
- \listtablename, 91
- lmodern, 112
- \ln, 73
- \log, 73
- longtable, 60
- \lstinputlisting, 123
- lstlisting, 122
- \lstset, 122
- LTXexample, 123
- Lua(La)TeX, 151
- LyX, 2
- LYX, 169

- MacTeX, 2
- `\mainmatter`, 48
- `\makeatletter`, 95, 161
- `\makeatlther`, 95
- `\makeatother`, 161
- makebst, 126
- makeidx, 80
- makeindex, 81
- `\makeindex`, 80
- `\maketitle`, 48, 50
- `\MakeUppercase`, 104
- math, 7
- `\mathbb`, 29, 72
- `\mathbf`, 72
- `\mathcal`, 29, 72
- `\mathfrak`, 29, 72
- `\mathit`, 72
- `\mathnormal`, 72
- `\mathop`, 30, 41
- mathpazo, 101
- mathptmx, 101
- `\mathrm`, 29, 72
- `\mathsf`, 72
- `\mathtt`, 72
- matrix, 31
- matriz, 31
  - delimitador
  - chaves, 31
  - cochetes, 31
  - parenteses, 31
  - verticais duplas, 31
  - vertical, 31
- matéria
  - frontal, 48
- `\max`, 73
- `\mbox`, 31, 95
- mdseries, 70
- `\medskip`, 83
- memoir, 104
- microtype, 104
- MikTeX, 2
- `\min`, 73
- minipage, 24
- misturar cores, 129
- `\modulolinenumbers`, 120
- multicol, 26
- multicols, 26
- multicolumn, 16
- multiline, 36
- multiline\*, 37
- natbib, 125, 126
  - aaai-named, 126
  - abbrvnat, 126
  - agsm, 126
  - apa, 126
  - apalike, 126
  - astron, 126
  - authordate<n>, 126
  - dcu, 126
  - humanbio, 126
  - kluwer, 126
  - named, 126
  - newapa, 126
  - plainnat, 126
  - unsrtnat, 126
- `\NeedsTeXFormat`, 161
- newcolumntype, 18
- `\newcommand`, 41
- `\newcounter`, 87
- `\newenvironment`, 43
- `\newfontfamily`, 154
- `\newif`, 100
- `\newlength`, 85
- `\newline`, 12
- `\newtheorem`, 53
- `\noindent`, 52
- `\nolinenumbers`, 120
- `\nonumber`, 35
- normalfont, 71
- normalsize, 71
- nota de rodapé, 12
- `\note`, 146
- nova linha, 12
- `\num`, 116
- `\numberwithin`, 88
- `\onecolumn`, 26
- `\onehalfspacing`, 103
- oneside, 50
- `\only`, 150
- openany, 50
- openright, 50
- otherlanguage, 90
- `\ovalbox`, 119
- `\overbrace`, 33
- Overfull `\hbox`, 94
- overlay, 149
- `\PackageError`, 162
- `\PackageWarning`, 162
- pacote, 4

- \pagecolor, 102
- \pageref, 10
- \pagestyle
  - empty, 48
- \paperheight, 85
- \paperwidth, 85
- \paragraph, 45
- \parbox, 25
- \parindent, 85, 105
- parskip, 105
- \parskip, 85, 105
- \partname, 91
- parágrafo, 4, 45
- parâmetro
  - obrigatório, 4
  - opcional, 4
- parâmetros
  - chave=valor, 98
- \pagestyle
  - empty, 48
- pdfscape, 115
- pgfplots, 139
- \phantom, 84
- \phantomsection, 112, 127
- pict2e, 65
- picture, 64
- pmatrix, 31
- polyglossia, 151
- potência, 8
- \Pr, 73
- preamble, 3
- \printbibliography, 127
- printindex
  - \printindex, 81
- \ProcessOptions, 165
- proof, 43, 53
- \proofname, 55, 91
- \protect, 89
- \providecommand, 41
- \ProvidesPackage, 161
  
- \qedhere, 55
- \qqquad, 84
- \quad, 84
- quebra de linha, 12
- question, 183
- quotation, 21
- quote, 21
  
- ragged2e, 104
- \raisebox, 68
  
- raiz, 8
- \ref, 10
- referência cruzada, 10
- referência bibliográfica, 75
- \reflectbox, 68
- \refname, 91
- \refstepcounter, 88
- \renewcommand, 41
- required, 103
- \RequirePackage, 161
- \resizebox, 66
- \rfloor, 32
- \right., 32
- rmfamily, 69
- \Roman, 86
- \roman, 86
- \rotatebox, 67
  - origin, 67
- rotating, 115
- rótulo, 10
  
- sans serif, 69
- \scalebox, 66
- scribus, 144
- scriptsize, 71
- scshape, 70
- \sec, 73
- \section, 45
- \section\*, 46
- \selectlanguage, 90
- serifa, 69
- setas
  - com nome, 37
- \setbeamerfont, 146
- \setbeamertemplate, 146
- \setcounter, 86
- \setkeys, 99
- \setlength, 85
- setspace, 103
- \settodepth, 85
- \settofirstfound, 155
- \settoheight, 85
- \settototalheight, 86
- \settowidth, 85
- seção, 45
- sffamily, 69
- \shadowbox, 119
- shadowcolor, 120
- showexpl, 123
- \SI, 117
- si

- \ampere, 118
- \candela, 118
- \celsius, 118
- \centi, 118
- \deca, 118
- \deci, 118
- \giga, 118
- \hecto, 118
- \kelvin, 118
- \kilo, 118
- \kilogram, 118
- \litre, 118
- \mega, 118
- \meter, 118
- \metre, 118
- \micro, 118
- \milli, 118
- \mole, 118
- \nano, 118
- \percent, 118
- \second, 118
- \si, 116
- \sin, 73
- \singlespacing, 103
- \sinh, 73
- \sisetup, 118
- siunitx, 116
- \sloppy, 94
- spacing, 103
- split, 34
- \sqrt, 8
- \stackrel, 37
- standalone, 187
- \stepcounter, 86
- \stretch, 83
- sub listas, 14
- subcaption, 114
- subequação, 38
- subfig, 114
- subfiles, 189
- \subfiles, 189
- \subparagraph, 45
- subparágrafo, 45
- \subsection, 45
- subseção, 45
- \substack, 37
- \subsubsection, 45
- SumatraPDF, 169
- sumário, 46, 48, 60
  - linhas pontilhadas, 96
- \sup, 73
- \supressfloats, 60
- tabbing, 20
- tabela, 15
  - flutuante, 59
  - mesclar células, 16
- tabela longa, 60
- table, 57, 59
- \tablename, 91
- \tableofcontents, 46, 48, 60
- tabular, 15
- tabularx, 19
- tabulação, 20
- tamanho
  - letra, 4
  - papel, 4
- \tan, 73
- \tanh, 73
- tasks, 183
- teoremas, 53
- TeX, 1
- TeXLive, 2
- TeXMaker, 1
- TeXStudio, 1
- \text, 31
- textbf
  - \textbf, 70
- \textcalligra, 113
- textcase, 104
- \textcolor, 102
- \textheight, 85
- \textit, 70
- \textmd, 70
- \textorpdfstring, 112
- \textrm, 69
- \textsc, 70
- \textsf, 69
- textstyle, 7
- \texttt, 69
- \textup, 70
- \textwidht, 85
- \the, 85
- thebibliography, 75
- \thechapter, 86
- \the<contador>, 86
- \theoremstyle
  - definition, 53
  - remark, 53
- \thepage, 86
- \thisfancy page, 119

`\thisfancyput`, 119  
`\thispagestyle`  
  empty, 48, 51  
tikz, 130  
Times, 101  
tiny, 71  
`\title`, 48  
titlepage, 48  
titlesec, 107  
  explicit, 109  
tocbibind, 104  
`\today`, 48  
`\totalheightof`, 86  
ttfamily, 69  
twocolumn, 26  
`\twocolumn`, 26  
twoside, 50  
`\typeout`, 161  
typewriter, 69  
título curto, 45

`\underbrace`, 33  
Underfull `\hbox`, 94  
`\underset`, 37  
unicode-math, 151  
`\unitlength`, 64, 85  
upshape, 70  
`\url`, 111  
`\usecolortheme`, 145  
`\usefonttheme`, 146  
`\usepackage`, 4  
`\usetheorem`, 145  
`\usetikzlibrary`, 130

`\value`, 88  
`\verb`, 23  
`\verb*`, 23  
verbatim, 24  
verbatim\*, 22  
`\verbatiminput`, 24  
verbatimm, 22  
verse, 22  
verso, 22  
`\vfill`, 83  
Vmatrix, 31  
vmatrix, 31  
`\vspace`, 83  
`\vspace*`, 83

`\widthof`, 86  
wrapfig, 114  
xcolor, 129  
Xe(La)T<sub>E</sub>X, 151