

Minicurso de L^AT_EX

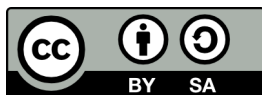
Encontro Científico dos Pós-Graduandos do IMECC 2013

Raniere Silva

Este trabalho é baseado em:

- “LaTeX com Vim (e Git)” de Raniere Silva, licenciado com a Licença Creative Commons Atribuição - CompartilhaIgual 3.0 Não Adaptada (<http://creativecommons.org/licenses/by-sa/3.0/>) e disponível em https://github.com/r-gaia-cs/latex_with_vim/;
- “TikZ para professores” de Raniere Silva, licenciado com a Licença Creative Commons Atribuição - CompartilhaIgual 3.0 Não Adaptada (<http://creativecommons.org/licenses/by-sa/3.0/>) e disponível em https://github.com/r-gaia-cs/latex_with_vim/.

Salvo indicação em contrário, este trabalho foi licenciado com a Licença Creative Commons Atribuição - CompartilhaIgual 3.0 Não Adaptada. Para ver uma cópia desta licença, visite <http://creativecommons.org/licenses/by-sa/3.0/> ou envie um pedido por carta para Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.



Prefácio

Esse matéria foi desenvolvido para o minicurso do Encontro Científico dos Pós-graduandos do IMECC 2013 da Universidade Estadual de Campinas (UNICAMP).

O minicurso foi preparado para ser ministrado em três aulas com duração de uma hora e vinte minutos cada com a seguinte distribuição didática:

Aula 0 find / -name `'*tex*`

Na primeira aula fala-se sobre a história do TeX e LaTeX, o significado de alguns nomes, alguns programas úteis.

São escritos os primeiros arquivos `.tex` que não utilizam nenhum pacote. Algumas classes são apresentadas e dependendo do tempo é apresentado o `beamer`.

Alguns ambientes são apresentados, dentre eles as listas e tabelas.

Aula 1 O preâmbulo, onde a mágica começa

Na segunda aula é construído um preâmbulo. Esse preâmbulo deve conter dentre outros pacotes aqueles voltados para internacionalização, codificação, formatação de página, inclusão de figuras.

Aula 2 AMSMATH, TikZ e BibTeX

A terceira e última aula destina-se aos pacotes `amsmath` (e família), `tikz` e `biblatex`. Esses são três pacotes muito utilizados.

Conteúdo

Prefácio	i
1 Introdução	1
1.1 História	1
1.2 Glossário	2
2 Utilitários	3
2.1 Compilação	3
2.2 Bibliografia	3
2.3 Conversores	4
2.4 Gerenciador de pacotes	4
2.5 Outras funcionalidades	4
2.6 Relacionados com PDF	4
3 Olá L^AT_EX	7
3.1 Instalação	7
3.2 Arquivo <code>.tex</code>	7
3.3 <i>Preâmbulo</i>	8
3.4 Hello world	9
3.4.1 Espaços, linhas, parágrafos e páginas	9
3.4.2 Hifenização	9
3.4.3 Acentos	10
3.5 Caracteres especiais	10
3.5.1 Aspas	10
3.5.2 Traço	10
3.5.3 Pontos sucessivos	10
3.5.4 Pontuação e demais símbolos	10
3.5.5 Comentários	10
3.6 Apresentações	11
3.6.1 Primeiro <i>slide</i>	11
3.6.2 Título do <i>slide</i>	12
3.6.3 Comandos e ambientes do L ^A T _E X	12
3.6.4 <i>Overlays</i>	12
3.6.5 Temas	13
4 Além do texto puro	15
4.1 Citações	15
4.2 Edição direta	15
4.3 Nota de rodapé	15
4.4 Listas	15
4.4.1 <code>itemize</code>	16
4.4.2 <code>enumerate</code>	16
4.5 Tabelas	16
4.5.1 <code>tabular</code>	16
4.5.2 <code>table</code>	17

4.5.3	Extensão Calc2LaTeX	17
4.6	Referência cruzada	17
5	Um pouco de layout	19
5.1	Fonte	19
5.1.1	Tamanho	19
5.2	Espaçamento	19
5.2.1	Espaçamento horizontal	20
5.2.2	Linha horizontal	20
5.2.3	Espaçamento vertical	20
5.2.4	Linha verticais	20
5.3	Alinhamento	20
6	O preâmbulo	23
6.1	Teclado e Idioma	23
6.2	Internacionalização	24
6.3	Parágrafos	24
6.4	Margens	24
6.4.1	<code>geometry</code>	24
6.4.2	Estilo de página	25
7	Alguns pacotes úteis	27
7.1	Cor	27
7.2	Endereços da internet	27
7.3	Hiperligação e metadados	27
7.4	Figuras	27
7.4.1	Arquivos de imagem	28
7.4.2	<code>figure</code>	28
8	Matemática no \LaTeX, amsmath	31
8.1	Modo matemático	31
8.1.1	<i>Inline</i>	31
8.1.2	<i>Displayed</i>	31
8.1.3	Uso de <i>inline</i> e <i>displayed</i>	31
8.2	Primeiros comandos no modo matemático	32
8.2.1	Operações aritméticas básicas	32
8.2.2	Índices e expoentes	32
8.2.3	Acentos	32
8.2.4	Delimitadores	32
8.2.5	Textos e espaçamentos	33
8.2.6	Matrizes	33
8.3	Comandos avançados no modo matemático	34
8.3.1	Equações, numeração e referenciação	34
8.3.2	<i>Tags</i>	34
8.3.3	Teorema	34
8.3.4	Demonstração	35
8.3.5	Alinhamento	35
8.3.6	Fórmulas longas	35
8.3.7	Ocultando termos	36
8.3.8	Funções definidas por partes	36
8.3.9	Fonte e Símbolos	36
8.4	Símbolos e operadores	37
8.4.1	Raiz quadrada	37
8.4.2	Binomial	37
8.4.3	Congruências	37

9	Desenhos utilizando o \LaTeX	41
9.1	TikZ	41
9.1.1	Ambiente <code>tikzpicture</code>	41
9.1.2	Sistema de coordenadas	41
9.1.3	Linhas	42
9.1.4	Operadores	44
9.1.5	Nó e texto	46
9.1.6	Preenchimento	47
10	Referência bibliográfica	49
10.1	BibTeX	49
10.2	<code>biblatex</code>	50
	Referência Bibliográfica	51
	Índice Remissivo	53

Lista de Tabelas

3.1	Parâmetros disponíveis para <code>opcoes</code>	8
3.2	Parâmetros disponíveis para <code>classe</code>	9
3.3	Acentuação (utilizando a vogal “o” para exemplo).	10
3.4	Para pontuação e símbolos especiais.	11
4.1	Opções disponíveis para <code>colunas</code>	16
4.2	Relação entre corrente e tensão para determinado circuito.	17
4.3	Opções disponíveis para <code>posicao</code>	17
5.1	Opções disponíveis para <code>XX</code> da fonte.	19
5.2	Opções disponíveis para o tamanho da fonte, em ordem crescente.	20
6.1	Opções disponíveis para <code>parametro</code> , referente ao pacote <code>geometry</code>	24
6.2	Opções disponíveis para <code>style</code>	25
7.1	Opções disponíveis para <code>parametro</code>	28
7.2	Opções disponíveis para <code>place</code>	29
8.1	Acentos disponíveis no modo matemático.	32
8.2	Delimitadores disponíveis no LaTeX.	33
8.3	Espaçamento no modo matemático.	33
8.4	Opções disponíveis para <code>XX</code> da fonte para o alfabeto matemático.	36
8.5	Opções disponíveis para <code>XX</code> da fonte para o alfabeto matemático interpretado como símbolo.	36
8.6	Setas	37
8.7	Relações binárias	38
8.8	Operadores binários	39
8.9	Operadores puros.	39
8.10	Operadores com intervalos.	39
8.11	Operadores similares ao limites.	39
8.12	Outros símbolos matemáticos	40
8.13	Alfabeto Grego, letras minúsculas	40
8.14	Alfabeto Grego, letras maiúsculo	40
10.1	tipo’s disponíveis no BibTeX padrão.	49
10.2	campo’s disponíveis no BibTeX padrão.	50

Capítulo 1

Introdução

Nesse capítulo será apresentado uma pouco da história a computação moderna e do contexto histórico no qual o TeX e o LaTeX surgiram. Posteriormente encontra-se um glossário de termos relacionados com o LaTeX.

1.1 História

Podemos dizer que a história da computação moderna tem início com a criação do ENIAC (Electronic Numerical Integrator and Computer), o primeiro computador digital eletrônico de grande escala, criado em fevereiro de 1946 pelos cientistas norte-americanos John Eckert e John Mauchly, da Electronic Control Company.

Por muitos anos o uso de computadores ficou restrito a grandes empresas e universidades como AT&T Bell Labs, General Electric, Massachusetts Institute of Technology entre outros. Em 1969 foi lançado o sistema operacional UNIX que rapidamente passou a ser utilizado pela maioria dos usuários da época.

Nos anos 70 ocorreu uma grande mudança nas técnicas de produção de livros e similares. Em 1977, Donald Knuth lançou a segunda edição do segundo volume de sua obra “The Art of Computer Programming” e não gostou do resultado (na primeira edição havia sido utilizada uma técnica de impressão diferente). Por volta desse ano, Knuth viu pela primeira vez o resultado de um sistema tipográfico digital de alta qualidade e ficou interessado pelo mesmo. Motivado pelo “problema” com o seu livro ele acabou desenvolvendo o seu próprio sistema tipográfico, o TeX¹, que foi lançado em 1978.

Usar o TeX não era fácil. Em 1985, Leslie Lamport lança o LaTeX, uma linguagem de marcação e preparativo do sistema para o TeX, facilitando a utilização do TeX.

Os primeiros computadores pessoais, como o Apple I, surgem nos anos 70. E nos anos 80 os computadores começam a invadir escritórios e depois lares, sendo que nessa década são lançados o IBM Personal Computer (IBM PC), Lisa, Macintosh e vários clones (principalmente do IBM PC).

Em 1985, uma pequena *start-up* chamada Microsoft lança seu sistema operacional, Windows, e seu processador de texto, Word, que possuía uma versão para Macintosh e foi um dos primeiros a possuir funcionalidades verdadeiramente WYSIWYG². Por ser WYSIWYG, utilizar o Word ou algum de seus concorrentes não exigia nenhum conhecimento prévio e isso acabou ofuscando o LaTeX.³

Com os computadores pessoais a Microsoft acabou adquirindo grande parte do mercado de sistemas operacionais para o seu produto, o Windows, por este ser compatível com os clones do IBM PC e possuir interface gráfica.⁴ Desde que o Windows passou a ser o sistema operacional dominante⁵ a Microsoft violou várias leis antitruste para promover outros de seus produtos como seu pacote de escritório, Microsoft Office, que inclui o Word, seu navegador de internet, Internet Explorer, e outros.

¹A pronúncia correta é semelhante a da palavra inglesa “tech”. Maiores informações em <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=TeXpronounce>

²Acrônimo da expressão em inglês “What You See Is What You Get”, cuja tradução remete a algo como “O que você vê é o que você obtém”.

³É importante destacar que, tipicamente, os usuários do LaTeX (ou TeX) e do Word (ou concorrentes) possuem necessidades bastante diferentes.

⁴Nessa época a Apple ainda era uma *start-up* quando comparada a seus concorrentes como, por exemplo, a IBM e ocorria a *UNIX wars* (ver detalhes em http://en.wikipedia.org/wiki/Unix_wars).

⁵Ao menos no ramo de computadores pessoais.

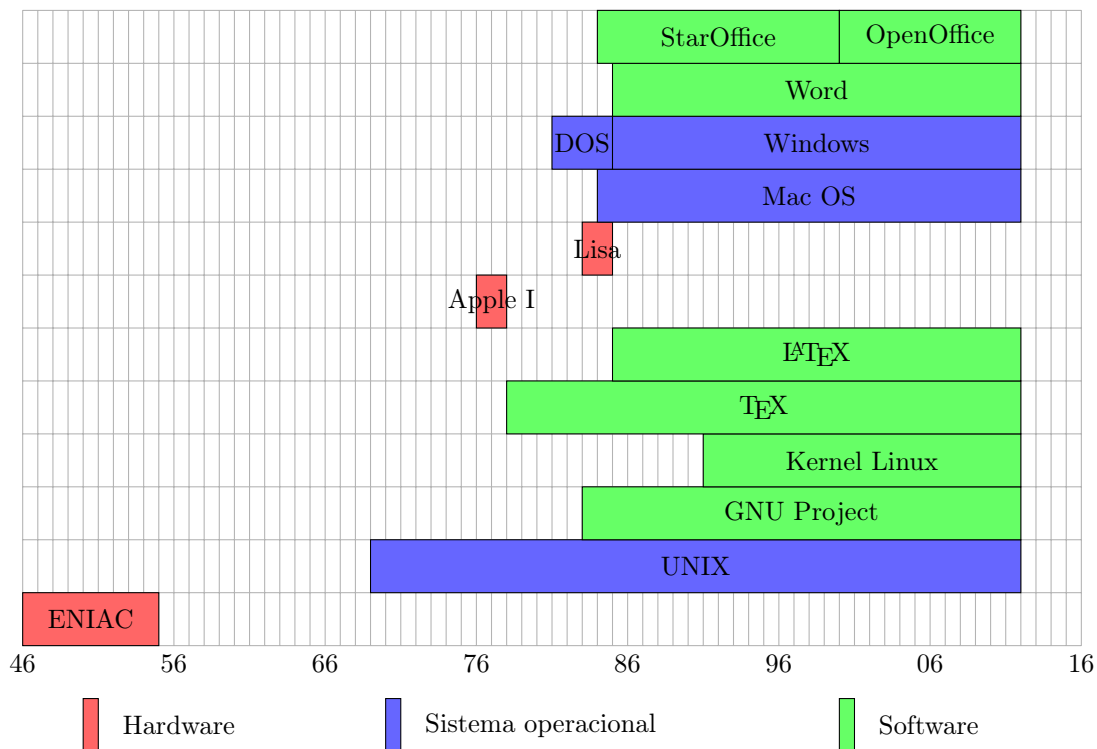


Figura 1.1: Linha do tempo de alguns softwares.

1.2 Glossário

Ao procurar ajuda é fundamental utilizar a palavra correta para o que deseja-se e como existem várias palavras que incluem TeX espera-se ajudar o leitor com algumas explicações (em ordem alfabética):

compilador é o arquivo binário responsável por ler o arquivo `.tex` e criar o arquivo para impressão.

distribuição uma coleção estruturada de software relacionados. Alguns exemplos de distribuições (La)TeX são: TeX Live e MiKTeX.

dvi acrônimo para DeVice-Independent.

LaTeX é o conjunto de macros escrita por Lamport para o TeX.

pdf acrônimo para Portable Document Format.

ps ou PostScript é linguagem para criação de desenhos vetoriais.

TeX é o sistema tipográfico criado por Knuth.

Capítulo 2

Utilitários

Devido ao LaTeX ser modular, é interessante conhecer alguns dos executáveis que costumam compor uma distribuição. Neste capítulo apresentaremos alguns destes executáveis.

2.1 Compilação

Relacionado com a compilação e manipulação do arquivo `.tex` temos:

latex gera um arquivo `dvi` a partir de um arquivo LaTeX.

latexmk automação completa do processo de compilação de documentos LaTeX.

luatex extensão do **pdftex** utilizando Lua como linguagem de script.

pdftex gera um `pdf` a partir de uma arquivo TeX.

pdflatex versão do **pdftex** para arquivo LaTeX.

tex gera um `div` a partir de um arquivo TeX.

Algumas das opções para alguns dos comandos anteriores são:

-interaction mode Configura o modo de iteração com o usuário. O modo deve ser uma das opções:

- `batchmode`,
- `nonstopmode`,
- `scrollmode`, e
- `errorstopmode`.

-shell-escape Habilita o uso de `\write18{comando}`. `comando` pode ser qualquer instrução válida para a linha de comando. Esse comando é normalmente desabilitado por razões de segurança mas necessários ao utilizar alguns pacotes para criar gráficos.

2.2 Bibliografia

Para o processamento de referências bibliográficas temos:

bibtex utiliza uma arquivo auxiliar gerado durante a compilação do arquivo `.tex` para criar o arquivo de bibliografia (`.bbl`) que será posteriormente incorporado.

biber é um substituto para o **bibtex** escrito para ser utilizado em conjunto com o pacote **biblatex**.

2.3 Conversores

Muitas vezes é preciso converter imagens que são incluídas durante a compilação para outro formato. Para essa tarefa temos:

a2ping utilitário que converte imagens rasterizadas e vetoriais para EPS e PDF.

e2pall procura no arquivo `.tex` pelo comando `\includegraphics` para encontrar os arquivos EPS utilizados e convertê-los para PDF.

2.4 Gerenciador de pacotes

Para o gerenciamento da distribuição LaTeX instalada, incluindo pacotes e configurações, temos o **tmlgt**.

2.5 Outras funcionalidades

Para remover todos os comentários e instruções do TeX e LaTeX de um arquivo pode-se utiliza o **detex**. O índice remissivo é construído pelo comando **makeindex**.

Para localizar e visualizar a documentação da distribuição, de classes ou de pacotes temos:

texdoc é um utilitário de linha de comando.

texdoctk é uma interface gráfica.

Para verificar o arquivo `.tex` por erros temos o **lacheck** lê o documento LaTeX e mostra mensagens caso encontre erros no documento.

Para comparar dois arquivos `.tex` temos:

latexdiff compara dois arquivos ignorando características da sintaxe do LaTeX.

texdiff compara dois arquivos para criar uma versão mostrando as diferenças.

Para navegar do código (La)TeX para o resultado após a compilação e fazer o caminho contrário de maneira sincronizada temos o **synctex**.

2.6 Relacionados com PDF

Atualmente, o formato de saída dos documentos escritos utilizando (La)TeX é o PDF. Poppler (ou libpoppler) é uma biblioteca para acessar arquivos no formato PDF que disponibiliza alguns binários eventualmente úteis:

pdfimages extrator de imagens.

pdfinfo informações do documento.

pdfseparate ferramenta de extração de página.

pdftoppm conversor de PDF para imagens PPM/PNG/JPEG.

pdftotext extrator de texto.

pdfunite ferramenta de mesclagem de documentos.

Além da biblioteca Poppler, outra biblioteca bastante útil é a Ghostscript que processa os arquivos PostScript. Para converter um arquivo `ps` para `pdf` pode-se utilizar o **ps2pdf** presente no Ghostscript e para a compressão do PDF:

```
$ gs -sDEVICE=pdfwrite -dCompatibilityLevel=1.4 -dPDFSETTINGS=/resolucao \  
> -dNOPAUSE -dQUIET -dBATCH -sOutputFile=saida.pdf entrada.pdf
```

onde `resolucao` deve ser substituído por um dos valores da lista abaixo:

- **screen**: para resolução baixa,
- **ebook**: para resolução média,
- **printer**: para qualidade de impressão (alta),
- **prepress**: para qualidade de pré-impressão,
- **default**: padrão.

Capítulo 3

Olá L^AT_EX

Neste primeiro capítulo apresentamos os conhecimentos mínimos de todo usuário do LaTeX.

3.1 Instalação

Para utilizar o LaTeX você precisa das macros que compõem o mesmo. A forma mais fácil de conseguir isso é instalando uma distribuição da lista abaixo:

- Linux: TeX Live (<http://www.tug.org/texlive>),
- Mac OS X: TeX Live (<http://www.tug.org/texlive>), MacTeX (<http://www.tug.org/mactex/>),
- Windows: TeX Live (<http://www.tug.org/texlive>), proTeXt (<http://www.tug.org/protext/>) ou MiKTeX (<http://www.miktex.org/>).

Além das macros também é necessário um editor de texto ou uma IDE (*Integrated Development Environment*) própria para o LaTeX, como

- GNU Emacs (<http://www.gnu.org/software/emacs/>) com o AUCTeX (<http://www.gnu.org/software/auctex/>),
- TeXworks (<http://www.leliseron.org/texworks/>),
- Kile (<http://kile.sourceforge.net/>),
- Texmaker (<http://www.xm1math.net/texmaker/>).

Uma lista com várias IDE's encontra-se disponível em http://en.wikipedia.org/wiki/Comparison_of_TeX_editors.

3.2 Arquivo .tex

O LaTeX utiliza `.tex` como extensão padrão. O arquivo `main.tex`, onde `main` representa o nome do arquivo `.tex`, é um arquivo de texto, estruturado em duas partes:

1. *preâmbulo*
2. *informação*

sendo que a segunda parte deve ser delimitada pelo ambiente `document`, i.e., ser incluída no lugar de `XXX` do código abaixo:

```
\begin{document}
XXX
\end{document}
```

É permitido incluir um ou mais arquivos dentro de `main.tex`, isto é, trabalhar com múltiplos arquivos. Os arquivos a serem incluídos também possuem a extensão `.tex` mas devem conter apenas a *informação*.¹

Uma das formas de incluir um arquivo é com o comando `\input`, como ilustrado a seguir:

```
\input{aux.tex}
```

onde `aux.tex` é o nome do arquivo a ser incluído.²

Quando `main.tex` for compilado o arquivo `aux.tex` será lido e processado exatamente como se tivesse sido inserido na posição que o comando `\input` ocupa.

3.3 *Preâmbulo*

O *preâmbulo* deve ser iniciado por

```
\documentclass[opcoes]{classe}
```

onde `classe` indica o tipo de documento a ser criado e `opcoes` é uma lista de palavras-chaves separadas por vírgula que personaliza o comportamento de `classe` (na Tabela 3.1 encontra-se algumas das palavras-chaves disponíveis).

Tabela 3.1: Parâmetros disponíveis para `opcoes`.

Função	Código	Descrição
Tamanho	10pt	Utiliza, por padrão, o tamanho 10.
	11pt	Tamanho 11.
	12pt	Tamanho 12.
Papel	letterpaper	Utiliza, por padrão, o tamanho da folha correspondente carta.
	a4paper	Tamanho da folha correspondente a A4.
	a5paper	Tamanho da folha correspondente a A5.
	b5paper	Tamanho da folha correspondente a B5.
	executivepaper	Tamanho da folha correspondente a folha executiva.
	legalpaper	Tamanho da folha correspondente a folha legal.
Al. equação		Por padrão centra as equações.
	fleqn	Alinha as equações à esquerda.
Nº equação		Por padrão enumera as equações à direita.
	leqno	Enumera as equações à esquerda.
Título		Por padrão a classe <code>article</code> não começa uma nova página após o título, enquanto que <code>report</code> e <code>book</code> o fazem.
	titlepage	Começa uma nova página após o título.
	leqno	Não começa uma nova página após o título.
Faces		Por padrão a classe <code>article</code> e <code>report</code> são a uma face e a classe <code>book</code> é a duas.
	oneside	Gera o documento a uma face.
	twoside	Gera o documento a duas faces.
Começo		Não funciona com a classe <code>article</code> por nesta não existirem capítulos e por padrão a classe <code>report</code> começa os capítulos na próxima página disponível e a classe <code>book</code> sempre nas páginas à direita.
	openright	Começa os capítulos sempre nas páginas à direita.
	openany	Começa os capítulos na próxima página disponível.
Colunas	twocolumn	Gera o arquivo utilizando-se de duas colunas.

`class` corresponde ao nome de um arquivo `.cls`, os principais são apresentados na Tabela 3.2 e outros são indicados em <http://aprendolatex.wordpress.com/2007/07/15/mais-classes-de-documentos/>.

¹Ao trabalhar com múltiplos arquivos deve-se apenas compilar o arquivo `main.tex`.

²Caso a extensão do arquivo seja suprimida será utilizada `.tex`.

Existe ainda alguns arquivos `.cls` personalizados disponíveis na internet, destacando-se o `abnt.cls`, disponível em <http://abntex.codigolivres.org.br/>, indicado para documentos que devem seguir as normas da ABNT e o usuário também pode escrever sua própria classe.

Tabela 3.2: Parâmetros disponíveis para classe.

Código	Descrição
<code>article</code>	Para artigos em revistas especializadas, palestras, trabalhos de disciplinas ...
<code>report</code>	Para informes maiores que constam de mais de um capítulo, projetos de fim de curso, dissertações, teses e similares.
<code>book</code>	Para livros.
<code>slide</code>	Para transparências.
<code>beamer</code>	Para apresentações.
<code>exam</code>	Para lista de exercícios.

3.4 Hello world

Anteriormente foi apresentado os aplicativos necessários para trabalhar com LaTeX e as duas partes principais do arquivo `.tex`. A seguir apresentaremos como construir a *informação*.

O documento mais simples que podemos criar é apresentado abaixo.

<pre>\documentclass[10pt,a4paper]{ article} \begin{document} Hello world. \end{document}</pre>	<pre>Hello world.</pre>
--	-------------------------

Os exemplos que serão apresentados aparecerão seguindo o modelo acima, isto é, em duas colunas sendo a coluna da esquerda contendo o código LaTeX e a coluna da direita contendo a saída obtida. Por simplicidade, nos demais exemplos iremos apresentar apenas a *informação*.

3.4.1 Espaços, linhas, parágrafos e páginas

No LaTeX o espaço entre palavras apresenta uma particularidade: ele é ignorado se houver dois ou mais espaços seguidos, como podemos observar a seguir.

<pre>Hello world.(2 spaces) Hello world.(3 spaces)</pre>	<pre>Hello world.(2 spaces) Hello world.(3 spaces)</pre>
--	--

Quando for necessário gerar dois ou mais espaços seguidos deve-se utilizar a barra invertida entre os espaços como ilustrado a seguir.

<pre>Hello \ world.(2 spaces) Hello \ world.(3 spaces)</pre>	<pre>Hello world.(2 spaces) Hello world.(3 spaces)</pre>
--	--

Nos dois exemplos anteriores é possível verificar que a mudança de linha no código não produz uma nova linha no documento gerado. A quebra de linha no LaTeX é representada por `\\` ou pelo comando `\newline`, como ilustrada a seguir.

<pre>Hello world.[1] \\ Hello world.[2] \newline Hello world.[3]</pre>	<pre>Hello world.[1] Hello world.[2] Hello world.[3]</pre>
--	--

Já a mudança de parágrafo é indicada por uma linha em branco.

Quando for necessário forçar uma mudança de página utiliza-se o comando `\newpage`. Assim como o LaTeX ignora dois ou mais espaços seguidos a mudança de linha e de página também é ignorada.

3.4.2 Hifenização

O LaTeX tenta balancear o tamanho das linhas a serem geradas e para isso utiliza-se de um banco de dados para hifenizar, quando necessário, alguma palavra.

Algumas vezes a hifenização ocorre de maneira inadequada e para corrigir devemos utilizar o comando `\hyphenation` cujo parâmetro é uma lista de palavras, separadas por espaço, onde o comando `-` é utilizado para indicar onde a palavra pode ser separada.

3.4.3 Acentos

Para inserir os acentos deve-se utilizar a codificação presente na Tabela 3.3.

Tabela 3.3: Acentuação (utilizando a vogal “o” para exemplo).

Com.	Res.	Com.	Res.	Com.	Res.	Com.	Res.
<code>\’{o}</code>	ó	<code>\={o}</code>	ō	<code>\u{o}</code>	ö	<code>\. {o}</code>	ò
<code>\v{o}</code>	õ	<code>\r{o}</code>	ô	<code>\c{c}</code>	ç	<code>\t{oo}</code>	ôo
<code>\~{o}</code>	ô	<code>\~{o}</code>	õ	<code>\" {o}</code>	ö	<code>\d{o}</code>	ø
<code>\H{o}</code>	ő	<code>\b{o}</code>	ö	<code>\‘{o}</code>	ò	<code>\i</code>	ı

3.5 Caracteres especiais

No LaTeX alguns caracteres apresentam forma própria de representação. A seguir enunciaremos alguns.

3.5.1 Aspas

Para as aspas não deve-se usar o caractere de aspas. Para abrir as aspas deve-se utilizar o acento simples e para fechar a aspa simples.

<code>‘Hello world.’</code> (aspas simples) <code>\\</code>		‘Hello world.’ (aspas simples)
<code>“Hello world.”</code> (aspas dupla) <code>\\</code>		“Hello world.” (aspas dupla)
<code>"Hello world."</code> (errado)		"Hello world." (errado)

3.5.2 Traço

LaTeX admite três tipos de traço.

<code>sem-terra</code> <code>\\</code>		sem-terra
<code>08--10 hours</code> <code>\\</code>		08–10 hours
<code>Campinas --- SP</code>		Campinas — SP

3.5.3 Pontos sucessivos

Utiliza-se o comando `\dots` ou `\ldots` para pontos sucessivos.

<code>patatoes, carrots \ldots</code> (correta) <code>\\</code>		patatoes, carrots ... (correta)
<code>patatoes, carrots \dots</code> (correta) <code>\\</code>		patatoes, carrots ... (correta)
<code>patatoes, carrots ...</code> (errada)		patatoes, carrots ... (errada)

3.5.4 Pontuação e demais símbolos

Para pontuação e demais símbolos especiais deve-se proceder como na Tabela 3.4.

3.5.5 Comentários

Também é possível inserir comentários no arquivo `.tex`, utilizando-se para isso do caractere `%` de forma que todo o texto posterior ao mesmo e na mesma linha é considerado comentário e conseqüentemente ignorado pelo compilador.

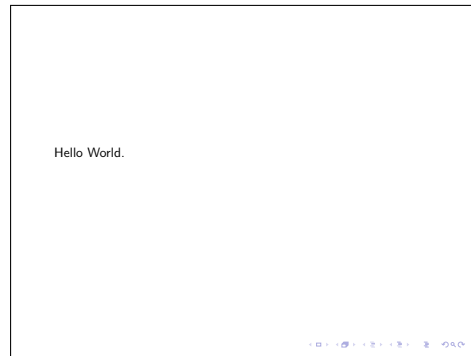
Tabela 3.4: Para pontuação e símbolos especiais.

Com.	Res.	Com.	Res.
<code>\&</code>	&	<code>\textasteriskcentered</code>	*
<code>\textbackslash</code>	\	<code>\textbar</code>	
<code>\{</code>	{	<code>\}</code>	}
<code>\texbullet</code>	•	<code>\textasciitilde</code>	~
<code>\textasciicircum</code>	^	<code>\copyright</code>	©
<code>\textregistered</code>	®	<code>\texttrademark</code>	™
<code>\textperiodcentered</code>	·	<code>\textexclamdown</code>	¡
<code>\textquestiondown</code>	¿	<code>\%</code>	%
<code>\textgreater</code>	>	<code>\textless</code>	<
<code>\#</code>	#	<code>\S</code>	§
<code>\P</code>	¶	<code>_</code>	—
<code>\dag</code>	†	<code>\ddag</code>	‡
<code>\pounds</code>	£	<code>a</code>	^a
<code>\textcircled{a}</code>	Ⓐ	<code>\textvisiblespace</code>	
<code>\\$</code>	\$	<code>\euro</code>	€

3.6 Apresentações

Apresentações podem ser criadas com a classe `beamer` e organizadas pelo ambiente `frame` que delimita onde começa e termina cada um dos *slides* da apresentação. A seguir apresentamos uma apresentação bem simples para exemplificar a utilização do ambiente `frame`.

```
\documentclass{beamer}
\begin{document}
\begin{frame}
  Hello World.
\end{frame}
\end{document}
```



3.6.1 Primeiro *slide*

Para a criação do primeiro *slide* com o título e autor pode utilizar os comandos `\title` e `\author` e, delimitado pelo ambiente `frame`, o comando `\titlepage`.

Além dos comandos `\title` e `\author` estão disponíveis os comandos `\subtitle`, `\date` e `\institute` que correspondem, respectivamente, ao subtítulo, data e local em que a apresentação irá ocorrer. Exceto pelo comando `\date` todos os demais comandos aceitam como opção uma abreviação do parâmetro.

```
\documentclass{beamer}
\begin{document}
\title[T\’{i}tulo]{T\’{i}tulo Completo}
\author[Autor]{Nome dos autores}
\institute[Escola]{Nome da Escola}
\begin{frame}
  \titlepage
\end{frame}
\end{document}
```



3.6.2 Título do *slide*

Para cada *slide* é possível atribuir um título com o comando `\frametitle` que normalmente será apresentado no topo do *slide*.

```
\documentclass{beamer}
\begin{document}
\begin{frame}
  \frametitle{T\''{i}tulo}
  Hello World.
\end{frame}
\end{document}
```



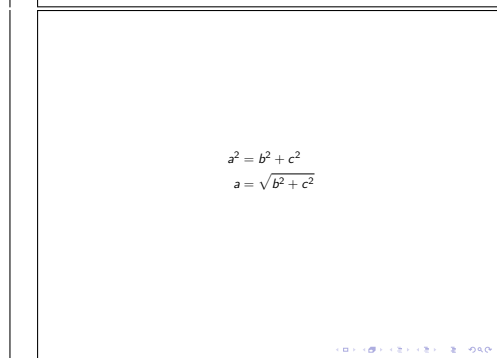
3.6.3 Comandos e ambientes do LaTeX

A classe `beamer` é compatível com grande parte dos comandos e ambientes do LaTeX sejam estes nativos ou presentes em algum pacote, i.e., para incluir listas, figuras, tabelas, expressões matemáticas, ... utiliza-se os mesmos comandos e ambientes.

```
\documentclass{beamer}
\begin{document}
\begin{frame}
  \begin{enumerate}
    \item Primeiro;
      \begin{enumerate}
        \item Subitem;
      \end{enumerate}
    \item Segundo.
  \end{enumerate}
\end{frame}
\end{document}
```



```
\documentclass{beamer}
\begin{document}
\begin{frame}
  \begin{align*}
    a^2 &= b^2 + c^2 \\
    a &= \sqrt{b^2 + c^2}
  \end{align*}
\end{frame}
\end{document}
```



3.6.4 Overlays

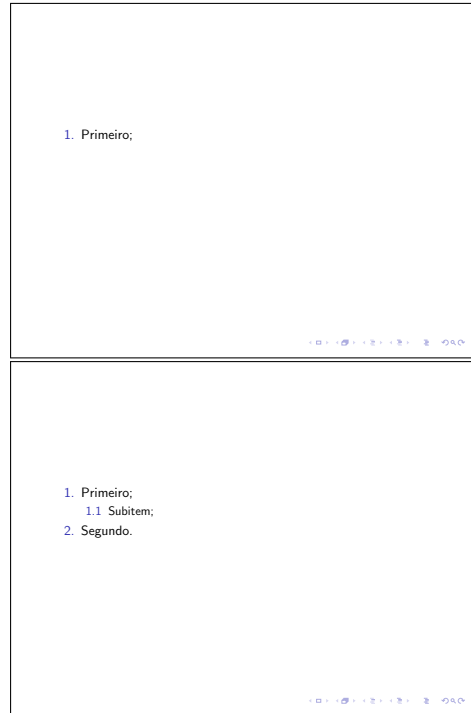
Até o momento todos os *slides* que construímos tinha sua informação apresentada em um único momento. Infelizmente não é isso que deseja-se na grande maioria das apresentações, i.e., deseja-se que fragmentos dos *slides* sejam apresentados em momentos distintos para que seja possível construir a informação desejada.

Para fragmentar o conteúdo dos *slides* podemos utilizar o comando `\pause` na posição que deseja-se fragmentar os *slides*.

```

\documentclass{beamer}
\begin{document}
\begin{frame}
  \begin{enumerate}
    \item Primeiro;
      \pause
      \begin{enumerate}
        \item Subitem;
      \end{enumerate}
    \item Segundo.
  \end{enumerate}
\end{frame}
\end{document}

```

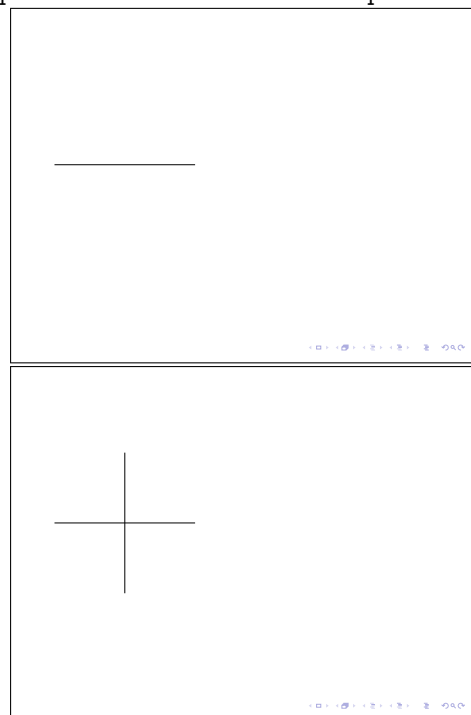


O comando `\pause` funciona dentro de vários ambientes do LaTeX sejam estes nativos ou presentes em algum pacote. No exemplo a seguir utilizamos o comando `\pause` dentro do ambiente `tikzpicture`.

```

\documentclass{beamer}
\usepackage{tikz}
\begin{document}
\begin{frame}
  \begin{tikzpicture}
    \draw (0,0) -- (4,0);
    \pause
    \draw (2,2) -- (2,-2);
  \end{tikzpicture}
\end{frame}
\end{document}

```



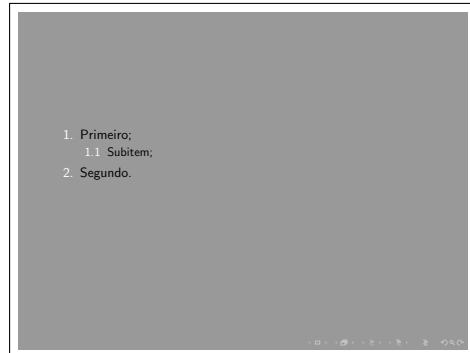
3.6.5 Temas

Até o momento, os *slides* apresentados possuíam fundo e bordas muito simples. É possível mudar isso utilizando os comandos `\usecolortheme`, muda apenas o esquema de cores, e `\usetheme`, mais genérico.

```

\documentclass{beamer}
\usecolortheme{beetle}
\begin{document}
\begin{frame}
  \begin{enumerate}
    \item Primeiro;
      \begin{enumerate}
        \item Subitem;
      \end{enumerate}
    \item Segundo.
  \end{enumerate}
\end{frame}
\end{document}

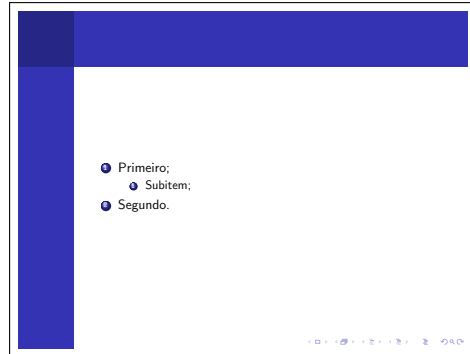
```



```

\documentclass{beamer}
\setheme{PaloAlto}
\begin{document}
\begin{frame}
  \begin{enumerate}
    \item Primeiro;
      \begin{enumerate}
        \item Subitem;
      \end{enumerate}
    \item Segundo.
  \end{enumerate}
\end{frame}
\end{document}

```



Para conhecer algumas dos parâmetros disponíveis para os comandos `usecolortheme` e `\setheme` sugere-se <http://www.hartwork.org/beamer-theme-matrix/>. Outros temas estão disponíveis na internet e alguns deles reunidos em <http://latex.simon04.net/>.

Capítulo 4

Além do texto puro

No capítulo anterior introduzimos os comandos mais básicos do LaTeX que possibilitam o usuário escrever um texto simples. Neste capítulo apresentamos alguns comandos do LaTeX que são seu diferencial ao escrever textos longos.

4.1 Citações

No LaTeX encontramos dois ambientes dedicados a citações. O primeiro deles é o `quote` próprio para citações de uma única linha e o segundo é o `quotation` adequado para citações de vários parágrafos.

4.2 Edição direta

Algumas vezes deseja-se inserir um texto que não deve ser interpretado. Isso é possível pelo ambiente `verbatim`, coloca o texto em uma nova linha, e pelo comando `\verb`, coloca o texto na mesma linha.

Tanto o ambiente `verbatim` como o comando `\verb` apresentam uma fonte própria.

<code>\textsc{texto~interpretado.} \\</code>		TEXTO INTERPRETADO.
<code>\verb+Texto~nao~interpretado.+</code>		Texto~nao~interpretado.

Vale destacar que o comando `\verb` é “flexível” quando ao delimitador, os caracteres `!`, `+` e `:` normalmente exercem satisfatoriamente esta função.

4.3 Nota de rodapé

Para produzir notas de rodapé deve-se utilizar o comando `\footnote` que deve ocorrer imediatamente depois da palavra ou texto a que se refere a nota de rodapé e como parâmetro do comando o texto a ser inserido na nota de rodapé.

4.4 Listas

Para a construção de listas podemos utilizar um dos quatro ambientes: `itemize`, `enumerate`, `description`¹ ou `list`². E para a criação de sublistas basta adicionar um dos ambientes dentro de um já existente.

Cada item de uma lista é identificado, no LaTeX, pelo comando `\item` que deve preceder o texto.

¹Não será tratado neste curso

²Não será tratado neste curso

4.4.1 itemize

O ambiente `itemize` utiliza um símbolo para indicar cada item da lista.

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

- Primeiro;
- Subitem;
- Segundo.

4.4.2 enumerate

O ambiente `enumerate` numera cada um dos itens da lista.

```
\begin{enumerate}
  \item Primeiro;
    \begin{enumerate}
      \item Subitem;
    \end{enumerate}
  \item Segundo.
\end{enumerate}
```

1. Primeiro;
- (a) Subitem;
2. Segundo.

Ao utilizar o ambiente `enumerate` é permitido para cada item adicionar um comando `\label` e posteriormente fazer referência a este pelo comando `\ref`.

4.5 Tabelas

O LaTeX permite construir tabelas e adicionar legendas à estas.

4.5.1 tabular

O ambiente `tabular` é utilizado para a construção de tabelas no LaTeX e sua sintaxe é

```
\begin{tabular}[colunas]
  informacao
\end{tabular}
```

onde `colunas` é uma sequência de caracteres, onde cada caractere corresponde a uma coluna e o respectivo alinhamento que são apresentados na Tabela 4.1, e `informacao` é o conteúdo de cada célula da tabela.

Tabela 4.1: Opções disponíveis para `colunas`.

Código	Descrição
l	Alinha com margem esquerda.
r	Alinha com a margem direita.
c	Centralizado.
p	Requer como parâmetro a largura da coluna.
	Imprime uma linha separando as colunas.

Cada célula da tabela deve ser separadas pelo comando `&` e a mudança de linha ocorre pelo comando `\\` ou `\tabularnewline`. Para imprimir uma linha horizontal separando duas linhas da tabela deve-se utilizar o comando `\hline`.

```
\begin{tabular}{|c|c|c|c|}
\hline Corrente (A) & Tensao (V) \\
\hline 0,0260 & 14,8 \\
\hline 0,0246 & 14,0 \\
\hline 0,0240 & 13,0 \\
\hline 0,0214 & 12,0 \\
\hline
\end{tabular}
```

Corrente (A)	Tensao (V)
0,0260	14,8
0,0246	14,0
0,0240	13,0
0,0214	12,0

Outros comandos também são importantes para a construção mas não trataremos deles aqui, para conhecê-los visitar <http://en.wikibooks.org/wiki/LaTeX/Tables>.

4.5.2 table

O ambiente `table` possibilita a inclusão de uma legenda para a tabela e trabalha a mesma como um objeto flutuante. A sintaxe deste ambiente é

```
\begin{table}[posicao]
  tabela
  \caption{legenda}
  \label{P:tebela}
\end{table}
```

onde `posicao` é o parâmetro que indica onde a tabela deve ser preferencialmente inserida (as opções disponíveis são apresentadas na Tabela 4.3 e a opção padrão é `tbp`), `tabela` corresponde ao código da tabela a ser inserida, `\caption` é o comando correspondente a legenda e `legenda` é o texto a ser apresentado como legenda, `\label` é o comando para referência cruzada como já apresentado.

```
\begin{table}[H] \label{T:tab_exemp} \centering
  \caption{Relacao entre corrente e tensao
  para determinado circuito.}
  \begin{tabular}{|c|c|c|c|}
    \hline Corrente (A) & Tensao (V) \\
    \hline 0,0260 & 14,8 \\
    \hline 0,0246 & 14,0 \\
    \hline 0,0240 & 13,0 \\
    \hline 0,0214 & 12,0 \\
    \hline
  \end{tabular}
\end{table}
```

Tabela 4.2: Relacao entre corrente e tensao para determinado circuito.

Corrente (A)	Tensao (V)
0,0260	14,8
0,0246	14,0
0,0240	13,0
0,0214	12,0

Tabela 4.3: Opções disponíveis para `posicao`.

Código	Descrição
<code>h</code>	Na posição onde o código se encontra.
<code>t</code>	No topo de uma página.
<code>b</code>	No fim de uma página.
<code>p</code>	Em uma página separada.
<code>!</code>	Modifica algumas configurações a respeito de boa posição para objeto flutuante.

Uma dica útil é que o comando `\clearpage` força as tabelas pendentes a serem inseridas.

4.5.3 Extensão Calc2LaTeX

Muitas vezes temos uma tabela no Calc³ e desejamos transportá-la para o LaTeX. Para essa tarefa a extensão/macro `Calc2LaTeX`, disponível gratuitamente em <http://extensions.services.openoffice.org/en/project/Calc2LaTeX>, é bastante eficiente.

4.6 Referência cruzada

Existem dois tipos de referência cruzada, a primeira para alguma parte do documento e a segunda para um outro documento. Nesta seção abordaremos o primeiro tipo e o segundo será tratado quando formos falar sobre o BibTeX.

Para alguns comandos e ambientes o LaTeX atribui um número, ou conjunto de caracteres, que pode ser vinculado a um nome pelo comando `\label` e referenciado pelo comando `\ref` e `\pageref`, este último quando deseja-se o número da página onde encontra-se o item referenciado.

O argumento do comando `\label` é uma sequencia de caracteres⁴, *case sensitive*, que será utilizada como argumento do comando `\ref` ao efetuar a referência.

Ao utilizar os comandos `\ref` ou `\pageref` é aconselhável precedê-los por um `~` para evitar uma quebra de linha antes da referência.

³O Calc é um dos aplicativos do pacote LibreOffice e corresponde ao popular Excel do pacote Microsoft Office.

⁴Recomenda-se escolher uma sequencia "amigável".

Capítulo 5

Um pouco de layout

Enquanto que no capítulo anterior foi apresentado algumas ferramentas para escrever textos mais complexos, por exemplo, contendo listas e tabelas, nesse capítulo iremos tratar um pouco do layout do texto.

5.1 Fonte

No LaTeX estão disponíveis algumas fontes opcionais. Comandos da forma `\textXX` são responsáveis por alterar a fonte sendo que `XX` corresponde ao código da fonte a serem utilizados. A Tabela 5.1 apresenta alguns das opções disponíveis.

Tabela 5.1: Opções disponíveis para `XX` da fonte.

Código	Descrição
<code>it</code>	Texto em itálico.
<code>bf</code>	Texto em negrito.
<code>rm</code>	Texto em romano.
<code>sf</code>	Texto em sans serif.
<code>tt</code>	Texto na tipografia de uma máquina de escrever.
<code>sc</code>	Texto em caixa alta.

A seguir é ilustrado as opções apresentadas na Tabela 5.1.

```
Italico: \textit{novo texto}. \\
Negrito: \textbf{novo texto}. \\
Romano: \textrm{novo texto}. \\
Sans serif: \textsf{novo texto}. \\
Maquina de escrever: \texttt{novo texto}. \\
Caixa alta: \textsc{novo texto}.
```

```
Italico: novo texto.
Negrito: novo texto.
Romano: novo texto.
Sans serif: novo texto.
Maquina de escrever: novo texto.
Caixa alta: NOVO TEXTO.
```

5.1.1 Tamanho

Uma das maneiras de mudar o tamanho da fonte em uma parte do texto é utilizando um dos ambiente ou comando de tamanho (a Tabela 5.2 apresenta algumas opções disponíveis).

Destaca-se que os tamanhos são baseados no tamanho padrão. A seguir um exemplo.

```
{\tiny muito pequeno} \\
{\small pequeno} \\
fonte padrao \\
{\Large grande} \\
{\Huge enorme}
```

```
muito pequeno
pequeno
fonte padrao
grande
enorme
```

5.2 Espaçamento

Nesta seção abordaremos como inserir espaços ao longo do texto no LaTeX, mas antes é importante destacar que podemos suprimir espaços ao utilizar medidas negativas.

Tabela 5.2: Opções disponíveis para o tamanho da fonte, em ordem crescente.

Código	Descrição
<code>\tiny</code>	O menor tamanho possível.
<code>\SMALL</code> ou <code>\scriptsize</code>	
<code>\Small</code> ou <code>\footnotesize</code>	Tamanho utilizado em notas de rodapé.
<code>\small</code>	
<code>\normalsize</code>	Tamanho padrão.
<code>\large</code>	
<code>\Large</code>	
<code>\LARGE</code>	
<code>\huge</code>	
<code>\Huge</code>	O maior tamanho disponível.

5.2.1 Espaçamento horizontal

Para produzir um espaço horizontal utiliza-se o comando `\hspace` que tem como parâmetro o tamanho do espaço a ser inserido. Se o comando ocorrer entre duas linhas ou no início de uma linha o LaTeX não produz o espaço e para este caso devemos utilizar `\hspace*`.

Para modificar a indentação característica de um novo parágrafo deve-se utilizar o comando

```
\setlength{\parindent}{tam}
```

onde `tam` é o novo tamanho para a indentação dos parágrafos. No caso de desejar-se suprimir a indentação deve-se utilizar o comando `\noindent`.

O comando `\hfill` cria um espaço suficiente para dividir o texto de modo que o que estiver antes do comando é alinhado a esquerda e o que estiver depois é alinhado a direita. É permitido utilizar o comando mais de uma vez em uma linha. O comando é ignorado quando ocorrer entre duas linhas ou no início de uma linha, neste caso devemos utilizar `\hfill*`.

5.2.2 Linha horizontal

Os comandos `\dotfill` e `\hrulefill` funcionam de maneira semelhante ao comando `\hfill`, mas ao invés de inserir um espaço em branco é introduzido, respectivamente uma linha pontilhada e uma linha contínua.

5.2.3 Espaçamento vertical

O comando `\baselineskip[tam]` estabelece o tamanho do espaçamento entre linhas para o texto posterior ao comando. Para modificar o tamanho entre duas linhas específicas pode-se utilizar o comando `\\[tam]` inicia uma nova linha de maneira que `tam` é o espaçamento entre as linhas.

Para aumentar o espaço entre parágrafos pode-se utilizar um dos comandos `\smallskip`, `\medskip` ou `\bigskip`, sendo que o tamanho do espaço está relacionado com o tamanho da fonte padrão do documento.

Os comandos `\vspace` e `\vfill` funcionam, respectivamente, de modo muito semelhante aos comandos `\hspace` e `\hfill` só que na vertical.

5.2.4 Linha verticais

O comando `\vrule` produz uma linha vertical.

5.3 Alinhamento

Por padrão, o alinhamento ocorre com a margem esquerda e para alterá-lo pode-se utilizar um dos seguintes ambientes: `center` (para texto centralizado), `flushleft` (alinhamento a esquerda) e `flushright` (alinhamento

```
a direita).
\begin{flushleft}esquerda
\end{flushleft}
\begin{center}centralizado
\end{center}
\begin{flushright}direita
\end{flushright}
```

esquerda

centralizado

direita

Também é permitido utilizar os comandos: `\centering` (para texto centralizado), `\raggedleft` (alinhamento a esquerda) e `\raggedright` (alinhamento a direita).

Capítulo 6

O preâmbulo

No capítulo 3 vimos que o *preâmbulo* é iniciado por

```
\documentclass[opcoes]{classe}
```

O *preâmbulo* é completado com a inclusão de pacotes que serão utilizados na *informação*. O comando para inclusão de um pacote segue a seguinte sintaxe:

```
\usepackage[opcoes]{pacote}
```

onde *pacote* é o nome do pacote e *opcoes* é uma lista de palavras chaves correspondente a opções do pacote. Nesse e nos próximos capítulos será apresentado alguns dos pacotes existentes.

No *preâmbulo* o usuário também pode definir seus próprios comandos e ambientes¹.

6.1 Teclado e Idioma

Na época que o TeX foi desenvolvido utilizava-se a codificação ASCII (American Standard Code for Information Interchange) e, conseqüentemente, o LaTeX foi desenvolvido para utilizar apenas os caracteres presentes na codificação ASCII.

As 52 letras (26 letras minúsculas + 26 letras maiúsculas) do alfabeto americano, os dez dígitos indo-arábicos, seis sinais de pontuação (, ; . ? ! :) e quatro parenteses (() []). Todos estas teclas são interpretadas como elas mesmas pelo LaTeX.

Na seção 3.4.1 abordamos como o LaTeX interpreta o espaço e enter (mudança de linha).

As teclas correspondentes a ‘, acento grave, ’, apóstrofe, e -, hífen, são interpretadas pelo LaTeX de acordo com os caracteres adjacentes.

Os seis símbolos matemáticos (* + = < > /) são interpretados de maneira diferentes quando no modo texto e no modo matemático².

Existem, também, 13 símbolos especiais (# \$ % & ~ _ ^ \ { } @ " /) que são interpretados pelo LaTeX de acordo com os caracteres adjacentes.

Os demais caracteres disponíveis no teclado, quando utilizados, costumam produzir erro.

Para facilitar o uso do LaTeX em outros idiomas que não o inglês pode-se utilizar alguma codificação diferente da ASCII para o arquivo *.tex*. Ao utilizar uma codificação diferente da ASCII fazendo uso de caracteres não presentes na ASCII é necessário utilizar o pacote *inputenc* e informar a codificação³. As codificações mais comuns são UTF-8 e Latin1 sendo que para arquivos codificados com UTF-8 deve-se adicionar a seguinte linha no preâmbulo

```
\usepackage[utf8]{inputenc}
```

enquanto que para arquivos codificados com Latin1

```
\usepackage[latin1]{inputenc}
```

¹Não será abordado neste curso, uma ótima fonte é http://en.wikibooks.org/wiki/LaTeX/Customizing_LaTeX

²O modo matemático é apresentado no capítulo 8.

³A maioria das codificações são compatíveis com a ASCII e por esse motivo se for utilizado apenas caracteres ASCII não é necessário a inclusão do pacote *inputenc*.

Recomenda-se utilizar a codificação UTF-8 (Unicode) pois a Latin1 não possui mais suporte desde 2004 (ver http://pt.wikipedia.org/wiki/ISO_8859-1) ou apenas os caracteres definidos na codificação ASCII pois estes possuem a mesma representação na maioria das codificações existentes.

É importante que o editor que esteja sendo usado também esteja configurado para trabalhar com a codificação especificada. Quando uma codificação errada estiver sendo usada, o editor pode trocar ou omitir alguns caracteres.

Ao gerar um arquivo pdf utilizando o LaTeX ocorre que copiar e colar um fragmento de texto no pdf com caracteres que não esteja presentes na codificação ASCII será preciso corrigir o fragmento. Para atenuar esse trabalho deve-se utilizar o pacote `fontenc`.

6.2 Internacionalização

Uma vez que parte considerável de uma obra produzida utilizando o LaTeX é feita de maneira automática a internacionalização é importantíssima. No desenvolvimento de software, internacionalização é o nome dado a capacidade de um programa adequar-se aos padrões de diferentes países como, por exemplo, a língua.

No LaTeX, a internacionalização é feita pelo pacote `babel` de Johannes L. Braams que ajusta algumas macros de acordo com o idioma desejado, como a traduções de alguns termos e uso de caixa alta. O pacote `babel` possui as seguintes opções para o idioma português: `portuges`, `portuguese`, `brazil`, `brazilian`. Maiores detalhes podem ser encontrados na documentação do pacote[1].

6.3 Parágrafos

Por padrão, o primeiro parágrafo de capítulo, seções, ..., não é indented. Quando desejar-se indentar o primeiro parágrafo uma solução é utilizar o pacote `indentfirst`.

6.4 Margens

A configuração de margens no LaTeX pode ser feita nativamente, utilizando o pacote `geometry` ou o pacote `fancyhdr`. A seguir abordaremos o pacote `geometry` e o estilo de página.

6.4.1 geometry

O uso deste pacote é bastante simples, precisa-se apenas fazer a chamada do pacote e atribuir valores para os parâmetros disponíveis. A seguir apresentamos um exemplo:

```
\usepackage{geometry}
\geometry{parametro = comprimento, ...}
```

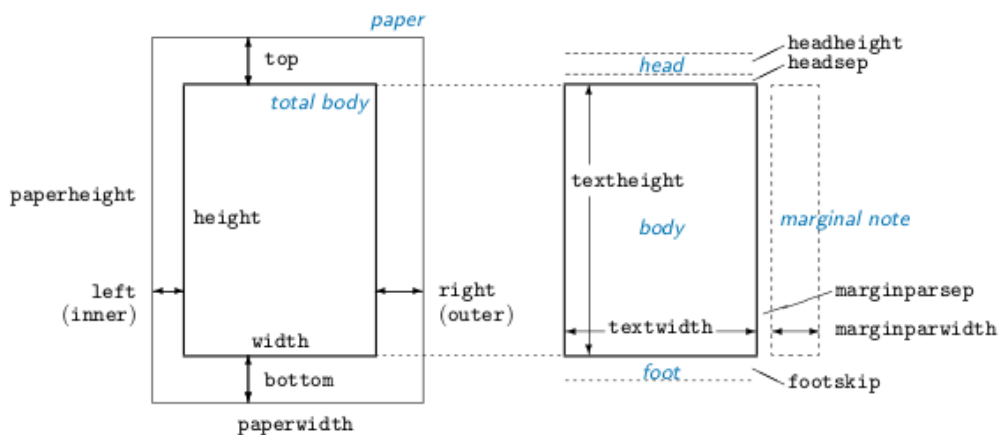
ou

```
\usepackage[parametro = comprimento, ...]{geometry}
```

Podemos utilizar `comprimento` em qualquer unidade disponível no LaTeX, mm, cm e outras. Já as opções para `parametro` mais utilizadas são apresentadas na Tabela 6.1 e ilustradas na Figura 6.1.

Tabela 6.1: Opções disponíveis para `parametro`, referente ao pacote `geometry`.

Código	Descrição
<code>paperwidth</code>	Largura do papel.
<code>paperheight</code>	Altura do papel.
<code>textwidth</code>	Largura da caixa de texto.
<code>textheight</code>	Altura da caixa de texto.
<code>top</code>	Margem superior.
<code>bottom</code>	Margem inferior.
<code>lefth</code>	Margem esquerda.
<code>right</code>	Margem direita.



Fonte: [13]

Figura 6.1: Ilustração da opções disponíveis para `parametro` apresentadas na Tabela 6.1.

6.4.2 Estilo de página

Existe um estilo de página definido como padrão⁴, quando deseja-se mudar o estilo em todo o documento pode-se utilizar o comando

```
\pagestyle{style}
```

e quando for necessário mudá-lo apenas na página atual utiliza-se o comando

```
\thispagestyle{style}
```

As opções para `style` são apresentadas na Tabela 6.2.

Tabela 6.2: Opções disponíveis para `style`.

Código	Descrição
<code>plain</code>	Imprime os números de página no centro do pé da página.
<code>headings</code>	No cabeçalho de cada página imprime o capítulo que está sendo processado e o número da página. O pé da página fica vazio.
<code>empty</code>	Coloca tanto o cabeçalho como o pé da página vazios.

Aos interessados em criar um estilo próprio, sugere-se utilizar o pacote `fancyhdr`.

⁴Corresponde ao estilo `plain` apresentado na Tabela 6.2.

Capítulo 7

Alguns pacotes úteis

No capítulo anterior foi apresentado três pacotes (`inputenc`, `babel` e `geometry`) que costuma estar presentes em todo documento LaTeX. Nesse capítulo vamos apresentar alguns outros pacotes mais alguns pacotes.

7.1 Cor

Para alterar a cor do texto é necessário os pacotes `graphicx` e `color` e pode-se utilizar um dos comandos: `\textcolor` ou `\color`.

A seguir apresentamos um exemplo.
`\textcolor{blue}{azul}` \\
`{\color{blue}azul}`

| azul
| azul

7.2 Endereços da internet

Nos endereços da internet é muito comum a presença de caracteres especiais para o LaTeX. Para inserir um endereço da internet facilmente pode-se utilizar o comando `\verb` que foi apresentado anteriormente ou utilizar o comando `\url` disponível no pacote `url`.

7.3 Hiperligação e metadados

Uma das capacidades do pdf é possuir metadados (informações para serem lidas por máquinas) e hiperligação internos e/ou externos (marcações ao longo do texto que possibilita ao usuário uma leitura não linear do documento).

Hiperligações são muito úteis ao leitor para que esse localize facilmente o texto a que uma referência cruzada refere-se. A criação das hiperligações é feita ao incluir o pacote `hyperref`.

A inclusão de alguns metadados também é feita pelo pacote `hyperref`. Para a inclusão nos metadados do pdf do título e autor da obra pode-se utilizar

```
\hypersetup{
  pdfinfo={
    Title={Titulo da obra},
    Author={Nome do autor},
  }
}
```

7.4 Figuras

No LaTeX é possível inserir figuras contidas em um arquivo de imagem ou desenhar uma¹. Também podemos adicionar uma legenda para a figura.

¹Ver a Seção 9.1

7.4.1 Arquivos de imagem

Para inserir arquivos de imagem é necessário o pacote `graphicx`. A imagem a ser inserida pode encontrar-se em um dos seguintes formatos: `jpg`, `png`, `pdf` ou `eps`².

O comando `\includegraphics` é o responsável por indicar a figura que será inserida, sendo a figura inserida ao longo do texto. A sintaxe deste comando é

```
\includegraphics [parametro=comprimento]{arquivo}
```

em que `parametro` é um comando disponíveis (algumas opções disponíveis são apresentadas na Tabela 7.1), `comprimento` é uma medida para `parametro` e `arquivo` é o nome do arquivo que contém a imagem.

Tabela 7.1: Opções disponíveis para `parametro`.

Código	Descrição
<code>width</code>	Corresponde a largura da figura.
<code>height</code>	Corresponde a altura da figura.
<code>scale</code>	Corresponde a escala da figura.
<code>angle</code>	Corresponde a uma rotação no sentido horário.
<code>page</code>	Apenas para PDF's, indica a página a ser utilizada.

Uma dica é que para `comprimento` podemos utilizar medidas correspondente a folha escolhida como por exemplo `\textwidth` ou `\textheight`.

```
\includegraphics [height=2cm]{figures/  
  anemonenfisch.png} \\  
Imagem de Andreas Preuss / marauder, dispon\''i  
  }vel em \url{http://openclipart.org/detail  
  /171242/anemonenfisch-by-marauder-171242} e  
  licenciada sobre CC0 PD Dedication.
```



Imagem de Andreas Preuss /
marauder, disponível em <http://openclipart.org/detail/171242/anemonenfisch-by-marauder-171242>
e licenciada sobre CC0 PD Dedicacion.

Maiores informações podem ser encontradas em http://en.wikibooks.org/wiki/LaTeX/Importing_Graphics.

7.4.2 figure

O ambiente `figure` possibilita a inclusão de uma legenda para a figura e trabalha a mesma como um objeto flutuante. A sintaxe deste ambiente é

```
\begin{figure}[place]  
  imagem  
  \caption{legenda}  
  \label{P:imagem}  
\end{figure}
```

onde `place` é o parâmetro que indica onde a figura deve ser preferencialmente inserida (as opções disponíveis são apresentadas na Tabela 7.2 e a opção padrão é `tbp`), `imagem` corresponde ao código da figura a ser inserida, `\caption` é o comando correspondente a legenda e `legenda` é o texto a ser apresentado como legenda, `\label` é o comando para referência cruzada como já apresentado.

²Este formato requer instalada o TeX Live 2011 ou superior pois a partir dessa versão o pacote para conversão do arquivo `eps` para um formato suportado é nativa.

```

\begin{figure}[H]
\centering
\includegraphics[height=2cm]{figures/
anemonenfisch.png} \\
Imagem de Andreas Preuss / marauder, dispon
\`{i}vel em \url{http://openclipart.org
/detail/171242/anemonenfisch-by-
marauder-171242} e licenciada sobre CC0
PD Dedication.

O par\`{a}metro \lstinline!H! neste
ambiente \`{e} necess\`{a}rio para
evitar error relacionado ao ambiente
flutuante.
\caption{Um peixe.}
\label{fig:example}
\end{figure}

```

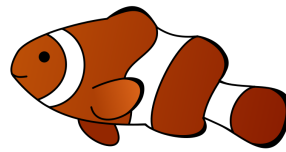


Imagem de Andreas Preuss / marauder, disponível em <http://openclipart.org/detail/171242/anemonenfisch-by-marauder-171242> e licenciada sobre CC0 PD Dedication.

O parâmetro H neste ambiente é necessário para evitar error relacionado ao ambiente flutuante.

Figura 7.1: Um peixe.

Tabela 7.2: Opções disponíveis para `place`.

Código	Descrição
h	Na posição onde o código se encontra.
t	No topo de uma página.
b	No fim de uma página.
p	Em uma página separada.
!	Modifica algumas configurações a respeito de boa posição para objeto flutuante.

Uma dica útil é que o comando `\clearpage` que força as figuras pendentes a serem inseridas.

Outras informações podem ser encontradas em http://en.wikibooks.org/wiki/LaTeX/Floats,_Figures_and_Captions.

Capítulo 8

Matemática no L^AT_EX, amsmath

Neste capítulo abordaremos o modo matemático do LaTeX, com uma ênfase nos pacotes amsmath, amssymb, amsthm e amssymb.

8.1 Modo matemático

Para que expressões matemáticas seja processadas corretamente, deve-se mudar do modo texto para o modo matemático, o que pode ser feito de várias maneiras.

A apresentação de expressões matemáticas pode ocorrer de duas maneiras: *inline*, quando aparecem na mesma linha do texto, e *displayed*, quando aparecem em uma linha própria e centralizada (podendo ou não ser numerada¹).

A seguir, informaremos como proceder para produzir expressões matemáticas *inline* ou *displayed*. Ao final, apresentaremos algumas dicas sobre o uso de expressões *inline* e *displayed*.

8.1.1 *Inline*

Expressões matemáticas *inline* devem ser iniciadas por `$` e fechadas por `$` ou iniciadas por `\` e fechadas por `\`.

<code>\$1 + 1 = 2\$</code> <code>\</code>		$1 + 1 = 2$
<code>\(1 + 1 = 2\)</code>		$1 + 1 = 2$

8.1.2 *Displayed*

Expressões matemáticas *displayed* devem ser iniciadas por `$$` e fechadas por `$$` ou iniciadas por `\[` e fechadas por `\]`.

<code>\$\$1 + 1 = 2\$\$</code>		$1 + 1 = 2$
<code>\[1 + 1 = 2\]</code>		$1 + 1 = 2$

Alguns ambientes, como `equation`, `eqnarray` e `align`, também produzem expressões matemáticas *displayed*.

8.1.3 Uso de *inline* e *displayed*

Um ótimo resumo sobre quando usar expressões *inline* e *displayed* encontra-se em <http://www.math.uiuc.edu/~hildebr/tex/displays.html> e a seguir apresentaremos tradução de alguns trechos. Para maiores detalhes recomenda-se uma leitura na obra “Mathematics Into Type”.

Expressões *inline* são “feias” quando apresentam frações, somatórios, integrais, ... e algumas vezes precisam de um cuidado especial para respeitarem as margens. Entretanto, deve-se preferir utilizar expressões *displayed* apenas nas seguintes ocasiões:

- a expressão é longa (ocupa mais da metade de uma linha);
- a expressão requer bastante espaço vertical, i.e., possui várias frações, somatórios, integrais, ...;

¹Deve-se numerar apenas equações as quais serão feita referências posteriormente.

- a equação será numerada;
- a expressão que você deseja destacar/enfatizar.

8.2 Primeiros comandos no modo matemático

A seguir enunciaremos como proceder para produzir as primeiras equações, mas antes é importante saber que o modo matemático ignora qualquer espaço (para inserir um espaço em branco no modo matemático veja a seção 8.2.5).

8.2.1 Operações aritméticas básicas

As operações aritméticas básicas são escritas normalmente, exceto pela multiplicação que utiliza-se dos comandos `\times` ou `\cdot`² e das frações representada pelo comando `\frac`³.

<pre>\$a a = a^2\$ \\ \$a_1, a_2, \dots, a_{11}, a_{12}\$ \\ \$f'(x)\$</pre>		$aa = a^2$ $a_1, a_2, \dots, a_{11}, a_{12}$ $f'(x)$
--	--	--

8.2.2 Índices e expoentes

Índices e expoentes são indicados pelos respectivos comandos: *underscore*, `_`, e *caret*, `^`. Por padrão apenas o primeiro símbolo depois do comando é alterado, quando for necessário mais de um símbolo deve-se utilizar chaves.

O símbolo *prime*, muito utilizado para derivadas, já vem posicionado corretamente.⁴

<pre>\$a a = a^2\$ \\ \$a_1, a_2, \dots, a_{11}, a_{12}\$ \\ \$f'(x)\$</pre>		$aa = a^2$ $a_1, a_2, \dots, a_{11}, a_{12}$ $f'(x)$
--	--	--

8.2.3 Acentos

Os acentos disponíveis no modo matemático são apresentados na Tabela 8.1.

Tabela 8.1: Acentos disponíveis no modo matemático.

Com.	Res.	Com.	Res.	Com.	Res.
<code>\acute{a}</code>	á	<code>\bar{a}</code>	\bar{a}	<code>\breve{a}</code>	\breve{a}
<code>\check{a}</code>	ǎ	<code>\dot{a}</code>	\dot{a}	<code>\ddot{a}</code>	\ddot{a}
<code>\dddota</code>	ǻ	<code>\ddddot{a}</code>	\ddddot{a}	<code>\grave{a}</code>	\grave{a}
<code>\hat{a}</code>	â	<code>\widehat{a}</code>	\widehat{a}	<code>\mathring{a}</code>	\mathring{a}
<code>\tilde{a}</code>	ã	<code>\widetilde{a}</code>	\widetilde{a}	<code>\vec{a}</code>	\vec{a}

8.2.4 Delimitadores

Parênteses, colchetes e chaves são exemplos de delimitadores. Uma lista completa dos delimitadores disponíveis no LaTeX encontra-se na Tabela 8.2.

Para expressões matemáticas no modo *displayed* ou longas é aconselhável utilizar os comandos `\left` e `\right` anteriormente ao limitador para ajustá-lo verticalmente.

<pre>\$\$\left(\frac{a}{b} \right) = a \left(\frac{1}{b} \right)\$\$</pre>		$\left(\frac{a}{b}\right) = a \left(\frac{1}{b}\right)$
--	--	---

²O uso do comando mais adequado depende muito do campo de estudo.

³Deve-se ponderar o uso deste comando por questão de legibilidade.

⁴Algumas vezes deve-se preferir utilizar o comando `prime` em conjunto com *underscore* e/ou *caret*.

Tabela 8.2: Delimitadores disponíveis no LaTeX.

Com.	Res.	Com.	Res.	Com.	Res.	Com.	Res.
(())	[[]]
\{	{	\}	}	\backslash	\	/	/
\langle	\langle	\rangle	\rangle			\	\
\lfloor	\lfloor	\rfloor	\rfloor	\lceil	\lceil	\rceil	\rceil
\ulcorner	\ulcorner	\urcorner	\urcorner	\llcorner	\llcorner	\lrcorner	\lrcorner

Nota: Enquanto que `|` é um limitador `\mid` é um operador lógico.

8.2.5 Textos e espaçamentos

Existem três ocasiões em que é preciso inserir um texto dentro de uma expressão matemática:

- um operador matemático é representado pelas primeiras letras de seu nome, e.g., `max`, `min`, `lim`, ...;
- uma variável é representada por mais de uma letra;
- incluir uma explicação/justificativa.

O LaTeX já possui vários operadores matemáticos definidos (são apresentados mais a frente) e quando o operador desejado não estiver definido deve-se utilizar o comando `\operatorname` ou `\DeclareMathOperator`, este último quando o operador for ser utilizado várias vezes no documento.

Em relação ao nome de variáveis, deve-se evitar ao máximo nomeá-las com mais de uma letra (utilizar o alfabeto grego para isso). Quando não for possível evitar, deve-se utilizar o comando `\mathrm` para evitar confusões.

<pre><code>\lanche = salgado + suco, \text{ errado.} \\$ \\ \\$ \mathrm{lanche} = \mathrm{salgado} + \mathrm{suco}, \text{ correto.} \\$ \\ suco}, \text{ correto.} \\$ \\ </code></pre>	$\text{lanche} = \text{salgado} + \text{suco, errado.}$ $\text{lanche} = \text{salgado} + \text{suco, correto.}$
---	--

Já para a inclusão de textos explicativos deve-se utilizar o comando `\text` e `\intertext`, este último reservado apenas para expressões *displayed*.

<pre><code>\a = b, \text{ por hipotese.} \\$</code></pre>	$a = b, \text{ por hipotese.}$
---	--------------------------------

Quanto ao espaçamento, normalmente não é preciso se preocupar com este pois o LaTeX inclui o espaçamento adequado. Em raras ocasiões deve-se incluir algum espaço apresentado na Tabela 8.3. Uma dessas ocasiões é

Tabela 8.3: Espaçamento no modo matemático.

Abrev.	Comando	Exemplo	Abrev.	Comando	Exemplo
	sem espaço	$\Rightarrow \Leftarrow$	<code>\,</code>	<code>\thinspace</code>	$\Rightarrow \Leftarrow$
<code>\:</code>	<code>\medspace</code>	$\Rightarrow \Leftarrow$	<code>\;</code>	<code>\thickspace</code>	$\Rightarrow \Leftarrow$
	<code>\quad</code>	$\Rightarrow \Leftarrow$		<code>\quad</code>	$\Rightarrow \Leftarrow$

em integrais.

<pre><code>\int_0^1 x \, \, \mathrm{d}x\$ (correto) \\ \int_0^1 x dx\$ (errado) \\ \int_0^1 \int_0^1 x y \, \, \mathrm{d}y \, \, \mathrm{d}x\$ (correto) \\ \int_0^1 \int_0^1 x y dy dx\$ (errado)</code></pre>	$\int_0^1 x \, dx \text{ (correto)}$ $\int_0^1 x dx \text{ (errado)}$ $\int_0^1 \int_0^1 xy \, dy \, dx \text{ (correto)}$ $\int_0^1 \int_0^1 xy dy dx \text{ (errado)}$
--	--

8.2.6 Matrizes

Para a construção de matrizes (e vetores) utiliza-se o ambiente `matrix` onde as colunas são separadas por `&` e as linhas por `\\`.

<pre><code>\begin{matrix} 2 & a+b \\ a/b & a^2 \end{matrix}</code></pre>	$\begin{matrix} 2 & a+b \\ a/b & a^2 \end{matrix}$
--	--

Destaca-se que o ambiente `matrix` só pode ser utilizado dentro do ambiente matemático e que na última linha não utiliza-se o comando `\\`.

Pode-se utilizar limitadores envolvendo o ambiente `matrix` ou utilizar uma variante: `pmatrix`, `bmatrix`, `Bmatrix`, `vmatrix` ou `Vmatrix` que corresponde, respectivamente, aos delimitadores `()`, `[]`, `{}`, `||` e `|||`.

8.3 Comandos avançados no modo matemático

8.3.1 Equações, numeração e referência

Para o uso de expressões matemáticas a serem referenciadas posteriormente, recomenda-se o ambiente `equation` em conjunto com o comando `\label`.

```
\begin{equation}\label{E:TeoPit}
a^2 = b^2 + c^2
\end{equation}
```

$$a^2 = b^2 + c^2 \quad (8.1)$$

No exemplo acima, `E:TeoPit` correspondente ao parâmetro do comando `\label`, como apresentado na Seção 4.6. A referência a equação ocorre pelo comando `\eqref`.

```
Na equacao (\ref{E:TeoPit}) $$a$$ corresponde a
hipotenusa de um triangulo e os catetos sao
$$b$$ e $$c$$. \\
```

Na equacao (8.1) a corresponde a hipotenusa de um triângulo e os catetos são b e c .

```
A equacao \eqref{E:TeoPit} e conhecida como
Teorema de Pitagoras.
```

A equacao (8.1) e conhecida como Teorema de Pitagoras.

8.3.2 Tags

O comando `\tag` do LaTeX nomeia uma equação e a referência passa a ser feito por este.

```
Sem tag: \begin{equation}\label{E:TeoPit_st}
a^2 + b^2 = c^2
\end{equation} \\
Com tag: \begin{equation}\label{E:TeoPit_ct}
\tag{Teorema de Pitagoras}
a^2 + b^2 = c^2
\end{equation} \\
\eqref{E:TeoPit_st} e \eqref{E:TeoPit_ct} sao
equivalentes.
```

Sem tag:

$$a^2 + b^2 = c^2 \quad (8.2)$$

Com tag:

$$a^2 + b^2 = c^2 \quad (\text{Teorema de Pitagoras})$$

(8.2) e (Teorema de Pitagoras) são equivalentes.

Vale destacar que podemos utilizar o comando `\label` como parâmetro do comando `\tag`.

8.3.3 Teorema

O comando `\newtheorem` deve ser inserido no *preâmbulo* e é responsável por criar um ambiente numerado para informações. Sua sintaxe é

```
\newtheorem{nome}{texto}
```

onde `nome` é o nome do ambiente a ser criado e `texto` é a sequência de caracteres que precede a numeração. Caso deseje-se não numerar deve-se utilizar a sintaxe

```
\newtheorem*{nome}{texto}
```

Para fazer uso do novo ambiente deve-se utilizar a sintaxe padrão para um ambiente

```
\begin{nome}
...
\end{nome}
```

ou ainda

```

\begin{nome}[XXX]
...
\end{nome}

```

onde XXX é uma sequência de caracteres que aparece entre parênteses logo após a numeração.

8.3.4 Demonstração

O ambiente `proof` é destinada a demonstrações e caracterizado por terminar com o comando `\qed`.

```

\begin{proof}
$a^2 + b^2 = c^2$
\end{proof}

```

Demonstração. $a^2 + b^2 = c^2$ \square

O ambiente `proof`, como podemos observar no exemplo abaixo, não trabalha adequadamente quando é finalizado com uma expressão matemática *displayed* e para corrigir isso devemos informar onde onde será inserido o símbolo *qed*.

```

\begin{proof}
Correto:
$$a^2 + b^2 = c^2 \qedhere$$
\end{proof}
\begin{proof}
Errado:
$$a^2 + b^2 = c^2$$
\end{proof}

```

Demonstração. Correto:

$$a^2 + b^2 = c^2 \quad \square$$

Demonstração. Errado:

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

\square

8.3.5 Alinhamento

O ambiente `equation` foi projetado para trabalhar apenas com equações de uma única linha, nesta seção vamos apresentar algumas formas de trabalhar com equações com várias linhas.

Para múltiplas equações alinhadas utilizamos o ambiente `align`, sendo cada linha separada pelo comando `\\` e o alinhamento por `&`.

```

\begin{align}
a^2 &= b^2 + c^2 \\
a &= \sqrt{b^2 + c^2}
\end{align}

```

$$a^2 = b^2 + c^2 \quad (8.3)$$

$$a = \sqrt{b^2 + c^2} \quad (8.4)$$

Quando o alinhamento ocorrer adjacente a um sinal de `=`, `+`, `...` devemos utilizar o comando `&` antes do sinal.

O ambiente `align` numera todas as equações. Caso não queira numerar uma ou mais equações deve-se utilizar o comando `\notag` em cada linha correspondente.

O comando `\label` deve estar presente em cada linha.

Quando desejar adicionar a alguma linha alguma anotação utiliza-se o comando `&&` entre a equação e a anotação.

```

\begin{align*}
a^2 &= b^2 + c^2 && \text{\textit{Teorema de Pitágoras}} \\
a &= \sqrt{b^2 + c^2}
\end{align*}

```

$$a^2 = b^2 + c^2 \quad \text{Teorema de Pitágoras}$$

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

8.3.6 Fórmulas longas

Fórmulas muito longas é fonte de vários problemas ao utilizar o LaTeX. Se existir fórmulas muito longas na obra que estiver trabalhando sugere-se inserir o pacote `breqn` por este quebrá-las automaticamente ao utilizar o ambiente `dmath` no lugar de `equation`.

Infelizmente o pacote `breqn` nem sempre funciona como desejado e nesses casos a solução é fazer a quebra da equação manualmente. Para isso, deve-se utilizar o ambiente `multline`, para uma única equação, ou `split`, este último deve ser utilizado dentro de um outro ambiente matemático. Se for quebrar as equações manualmente, recomenda-se ler a seção “Split equations without alignment” de “User’s Guide for the amsmath Package”.

8.3.7 Ocultando termos

Ao trabalhar com fórmulas muito longas tenta-se diminuir o tamanho utilizando sequências e muitas vezes é aconselhável indicar o número de termos. Para isso podemos utilizar os comandos `\overbrace` ou `\underbrace`.

$$\underbrace{x_1 + \dots + x_n}_n$$

8.3.8 Funções definidas por partes

É relativamente comum definirmos uma equações por partes e o ambiente adequado para representar esta construção é o `cases`.

$$|x - 1| = \begin{cases} x - 1, & \text{se } x \geq 1; \\ -x + 1, & \text{se } x < 1. \end{cases}$$

O ambiente `cases` também pode ser utilizado para sistemas de equações.

8.3.9 Fonte e Símbolos

No modo matemático, o LaTeX classifica os caracteres em alfabeto matemático e símbolos matemáticos. Baseado nessa classificação escolhe uma fonte a ser usada.

Para alterar a fonte de caracteres do alfabeto matemático utiliza-se o comando `\mathXX` sendo que `XX` corresponde ao código da fonte a ser utilizada. A Tabela 8.4 apresenta alguns das opções disponíveis.

Tabela 8.4: Opções disponíveis para `XX` da fonte para o alfabeto matemático.

Código	Descrição
<code>it</code>	Texto em itálico.
<code>bf</code>	Texto em negrito.
<code>rm</code>	Texto em romano.
<code>sf</code>	Texto em sans serif.
<code>tt</code>	Texto na tipografia de uma máquina de escrever.

A seguir é ilustrado as opções apresentadas na Tabela 8.4.

Normal: <code>\matha</code> .	Normal: a .
Itálico: <code>\mathit{a}</code> .	Itálico: a .
Negrito: <code>\mathbf{a}</code> .	Negrito: \mathbf{a} .
Romano: <code>\mathrm{a}</code> .	Romano: a .
Sans serif: <code>\mathsf{a}</code> .	Sans serif: a .
Máquina de escrever: <code>\mathtt{a}</code> .	Máquina de escrever: a .

Para símbolos matemáticos apenas é possível apresentá-los em negrito e, para isso, utiliza-se o comando `\boldsymbol`.

Normal: <code>\alpha</code> .	Normal: α .
Negrito: <code>\boldsymbol{\alpha}</code> .	Negrito: $\boldsymbol{\alpha}$.

No LaTeX também existe quatro alfabetos que são interpretados como símbolos. Um deles é o alfabeto grego, apresentado no capítulo anterior e os outros três são acessados com o comando `\mathXX`, sendo que `XX` corresponde ao código da fonte a ser utilizada. A Tabela 8.5 apresenta as opções disponíveis.

Tabela 8.5: Opções disponíveis para `XX` da fonte para o alfabeto matemático interpretado como símbolo.

Código	Descrição
<code>cal</code>	Texto em caligráfico, apenas para caixa alta.
<code>frak</code>	Texto em Euler Fraktur.
<code>bb</code>	Texto em blackboard bold, apenas para caixa alta.

A seguir é ilustrado as opções apresentadas na Tabela 8.5.

Normal: <code>\mathR</code> .	Normal: R .
Caligráfico: <code>\mathcal{R}</code> .	Caligráfico: \mathcal{R} .
Euler Fraktur: <code>\mathfrak{R}</code> .	Euler Fraktur: \mathfrak{R} .
Blackboard bold: <code>\mathbb{R}</code> .	Blackboard bold: \mathbb{R} .

Destaca-se que a fonte blackboard bold é normalmente utilizada para representar os conjuntos dos números naturais (\mathbb{N}), inteiros (\mathbb{Z}), reais (\mathbb{R}) e complexos (\mathbb{C}).

8.4 Símbolos e operadores

A seguir apresentaremos vários dos símbolos e operadores disponíveis no LaTeX. Para uma lista completa recomenda-se “The Comprehensive LaTeX Symbol List”. Ao final, abordamos os comandos para raiz quadrada, binomial e congruências.

Tabela 8.6: Setas

Com.	Res.	Com.	Res.	Com.	Res.
<code>\leftarrow</code>	\leftarrow	<code>\rightarrow</code>	\rightarrow	<code>\longleftarrow</code>	\longleftarrow
<code>\longrightarrow</code>	\longrightarrow	<code>\Leftarrow</code>	\Leftarrow	<code>\Rightarrow</code>	\Rightarrow
<code>\Longleftarrow</code>	\Longleftarrow	<code>\Longrightarrow</code>	\Longrightarrow	<code>\nleftarrow</code>	\nleftarrow
<code>\rightarrowtail</code>	\rightarrowtail	<code>\nLeftarrow</code>	\nLeftarrow	<code>\nrightarrow</code>	\nrightarrow
<code>\leftrightharpoonon</code>	\leftrightharpoonon	<code>\longleftarrowrightarrow</code>	\longleftrightarrow	<code>\Leftrightarrow</code>	\Leftrightarrow
<code>\Longleftarrowrightarrow</code>	\Longleftrightarrow	<code>\nletrightarrow</code>	\nleftrightarrow	<code>\nLeftrightarrow</code>	\nLeftrightarrow
<code>\dashleftarrow</code>	\dashleftarrow	<code>\dashrightarrow</code>	\dashrightarrow	<code>\leftrightharpoons</code>	\leftrightharpoons
<code>\rightleftharpoons</code>	\rightleftharpoons	<code>\leftrightarrows</code>	\leftrightarrows	<code>\rightleftarrows</code>	\rightleftarrows
<code>\mapsto</code>	\mapsto	<code>\longmapsto</code>	\longmapsto	<code>\iff</code>	\iff
<code>\uparrow</code>	\uparrow	<code>\downarrow</code>	\downarrow	<code>\Uparrow</code>	\Uparrow
<code>\Downarrow</code>	\Downarrow	<code>\updownarrow</code>	\updownarrow	<code>\Updownarrow</code>	\Updownarrow
<code>\Lsh</code>	\curvearrowleft	<code>\Rsh</code>	\curvearrowright	<code>\curvearrowleft</code>	\curvearrowleft
<code>\curvearrowright</code>	\curvearrowright	<code>\circlearrowleft</code>	\circlearrowleft	<code>\circlearrowright</code>	\circlearrowright

8.4.1 Raiz quadrada

Utiliza-se o comando `\sqrt` para raiz quadrada.

$$\begin{array}{l|l}
 \text{\$}\sqrt{4} = 2\text{\$} \backslash\ & \sqrt{4} = 2 \\
 \text{\$}\sqrt[3]{8} = 2\text{\$} & \sqrt[3]{8} = 2
 \end{array}$$

8.4.2 Binomial

Utiliza-se o comando `\binom` para os binômios.

$$\text{\$a} \equiv b \pmod{v} \text{\$} \quad | \quad a \equiv b \pmod{v}$$

8.4.3 Congruências

A forma mais comum para congruências corresponde ao uso dos comandos `\equiv` e `\pmod`.

$$\text{\$a} \equiv b \pmod{v} \text{\$} \quad | \quad a \equiv b \pmod{v}$$

Tabela 8.7: Relações binárias

Com.	Res.	Com.	Res.	Com.	Res.
<	<	\nless	⋈	>	>
\ngtr	⋈	\ll	⋈⋈	\lll	⋈⋈⋈
\gg	⋈⋈	\ggg	⋈⋈⋈	=	=
\neq	≠	:	:	\doteq	⋈⋈
\sim	≈	\nsim	≈	\cong	≅
\ncong	≇	\simeq	≈	\approx	≈
\equiv	≡	\leq ou \le	≤	\nleq	⋈
\geq ou \ge	≥	\ngeq	⋈	\leqslant	≤
\nleqslant	⋈	\geqslant	≥	\ngeqslant	⋈
\eqslantless	≤	\eqslantgtr	≥	\leqq	≤
\nleqq	⋈	\geqq	≥	\ngeqq	⋈
\lesssim	≈	\lessapprox	≈	\gtrsim	≈
\gtrapprox	≈	\prec	⋈	\nprec	⋈
\succ	⋈	\nsucc	⋈	\preceq	⋈
\npreceq	⋈	\succeq	⋈	\nsucceq	⋈
\in	∈	\notin	∉	\owns	∋
\subset	⊂	\supseteq	⊃	\subsetq	⊂
\nsubseteq	⋈	\supseteq	⊃	\nsupsetq	⋈
\subseteqq	⊂	\nsupseteqq	⋈	\supseteqq	⊃
\nsupseteqq	⋈	\sqsubset	⊏	\sqsubsetq	⊏
\sqsupseteq	⊏	\sqsupseteq	⊏	\smile	⌞
\smallsmile	⌞	\frown	⌞	\smallfrown	⌞
\perp	⊥	\models	⊨	\mid	∣
\nmid	⋈	\parallel	∥	\nparallel	⋈
\shortmid	∣	\nshortmid	⋈	\shortparallel	∥
\nshortparallel	⋈	\vdash	⊢	\nvdash	⋈
\dashv	⊢	\vDash	⊨	\nvDash	⋈
\Vdash	⊨	\nVdash	⋈	\propto	∝
\asymp	⋈	\bowtie	⋈	\Join	⋈
\vartriangleleft	◁	\ntriangleleft	⋈	\vartriangleright	▷
\ntriangleright	⋈	\trianglelefteq	◁	\ntrianglelefteq	⋈
\trianglerighteq	▷	\ntrianglerighteq	⋈	\blacktriangleleft	◀
\blacktriangleright	▶	\between	⊘	\pitchfork	⋈
\therefore	∴	\because	∴		

Enquanto que \mid é um limitador, \mid é um operador que corresponde a expressão “tal que”.

Tabela 8.8: Operadores binários

Com.	Res.	Com.	Res.	Com.	Res.
+	+	-	-	\pm	±
\mp	∓	\times	×	\cdot	·
\div	÷	\And	&	\setminus	\
\smallsetminus	↖	\dagger	†	\ddagger	‡
\ast	*	\star	*	\wedge	∧
\vee	∨	\cap	∩	\cup	∪
\sqcap	⊓	\sqcup	⊔	\oplus	⊕
\ominus	⊖	\otimes	⊗	\oslash	⊘
\odot	⊙	\bigcirc	○	\circ	◦
\bullet	•	\bigtriangleup	△	\bigtriangledown	▽
\triangleleft	◁	\triangleright	▷	\diamond	◇
\wr	ℳ	\amalg	∐		

Tabela 8.9: Operadores puros.

Com.	Res.	Com.	Res.	Com.	Res.
\log	log	\ln	ln	\exp	exp
\arccos	arccos	\arcsin	arcsin	\arctan	arctan
\cos	cos	\sin	sin	\tan	tan
\csc	csc	\sec	sec	\cot	cot
\cosh	cosh	\sinh	sinh	\tanh	tanh
\lg	lg	\arg	arg	\hom	hom
\dim	dim	\ker	ker	\det	det
\gcd	gcd				

Tabela 8.10: Operadores com intervalos.

Com.	Res.	Com.	Res.	Com.	Res.
\int	∫	\iint	∬	\iiint	∭
\iiint	∭	\idotsint	∫⋯∫	\oint	∮
\prod	∏	\coprod	∐	\bigcap	⋂
\bigcup	⋃	\bigwedge	⋀	\bigvee	⋁
\bigsqcup	⊔	\biguplus	⊕	\bigotimes	⊗
\bigoplus	⊕	\bigodot	⊙	\sum	∑

Tabela 8.11: Operadores similares ao limites.

Com.	Res.	Com.	Res.	Com.	Res.
\lim	lim	\inf	inf	\sup	sup
\max	max	\injlim	injlim	\liminf	lim inf
\limsup	limsup	\min	min	\varinjlim	lim _→
\varliminf	lim _←	\varlimsup	lim _→	\Pr	Pr
\projlim	projlim	\varprojlim	lim _←		

Tabela 8.12: Outros símbolos matemáticos

Com.	Res.	Com.	Res.	Com.	Res.
<code>\Re</code>	\Re	<code>\Im</code>	\Im	<code>\nabla</code>	∇
<code>\partial</code>	∂	<code>\infty</code>	∞	<code>\emptyset</code>	\emptyset
<code>\varnothing</code>	\emptyset	<code>\forall</code>	\forall	<code>\exists</code>	\exists
<code>\nexists</code>	\nexists	<code>\angle</code>	\angle	<code>\measuredangle</code>	\measuredangle
<code>\sphericalangle</code>	\sphericalangle	<code>\top</code>	\top	<code>\bot</code>	\perp
<code>\diagup</code>	\diagup	<code>\diagdown</code>	\diagdown	<code>\triangle</code>	\triangle
<code>\triangledown</code>	\triangledown	<code>\blacktriangle</code>	\blacktriangle	<code>\blacktriangledown</code>	\blacktriangledown
<code>\Diamond</code>	\diamond	<code>\lozenge</code>	\lozenge	<code>\blacklozenge</code>	\blacklozenge
<code>\bigstar</code>	\bigstar	<code>\Box</code>	\square	<code>\square</code>	\square
<code>\blacksquare</code>	\blacksquare	<code>\clubsuit</code>	\clubsuit	<code>\diamondsuit</code>	\diamondsuit
<code>\heartsuit</code>	\heartsuit	<code>\spadesuit</code>	\spadesuit		

Tabela 8.13: Alfabeto Grego, letras minúsculas

Com.	Res.	Com.	Res.	Com.	Res.	Com.	Res.
<code>\alpha</code>	α	<code>\beta</code>	β	<code>\gamma</code>	γ	<code>\delta</code>	δ
<code>\epsilon</code>	ϵ	<code>\zeta</code>	ζ	<code>\eta</code>	η	<code>\theta</code>	θ
<code>\iota</code>	ι	<code>\kappa</code>	κ	<code>\lambda</code>	λ	<code>\mu</code>	μ
<code>\nu</code>	ν	<code>\xi</code>	ξ	<code>\pi</code>	π	<code>\rho</code>	ρ
<code>\sigma</code>	σ	<code>\tau</code>	τ	<code>\upsilon</code>	υ	<code>\phi</code>	ϕ
<code>\chi</code>	χ	<code>\psi</code>	ψ	<code>\omega</code>	ω	<code>\digamma</code>	\digamma
<code>\varepsilon</code>	ε	<code>\vartheta</code>	ϑ	<code>\varkappa</code>	\varkappa	<code>\varpi</code>	ϖ
<code>\varrho</code>	ϱ	<code>\varsigma</code>	ς	<code>\varphi</code>	φ		

Tabela 8.14: Alfabeto Grego, letras maiúsculo

Com.	Res.	Com.	Res.	Com.	Res.	Com.	Res.
<code>\Gamma</code>	Γ	<code>\Delta</code>	Δ	<code>\Theta</code>	Θ	<code>\Lambda</code>	Λ
<code>\Xi</code>	Ξ	<code>\Pi</code>	Π	<code>\Sigma</code>	Σ	<code>\Upsilon</code>	Υ
<code>\Phi</code>	Φ	<code>\Psi</code>	Ψ	<code>\Omega</code>	Ω		
<code>\varGamma</code>	\varGamma	<code>\varDelta</code>	\varDelta	<code>\varTheta</code>	\varTheta	<code>\varLambda</code>	\varLambda
<code>\varXi</code>	\varXi	<code>\varPi</code>	\varPi	<code>\varSigma</code>	\varSigma	<code>\varUpsilon</code>	\varUpsilon
<code>\varPhi</code>	\varPhi	<code>\varPsi</code>	\varPsi	<code>\varOmega</code>	\varOmega		

Capítulo 9

Desenhos utilizando o L^AT_EX

Neste capítulo abordaremos brevemente o pacote `tikz` utilizado para desenhar. Este pacote é bastante complexo de modo que abordaremos apenas uma minúscula parcela deste e para maiores informações, recomenda-se o respectivo manual.

9.1 TikZ

O pacote `tikz` permite produzir desenhos vetoriais ao informar as linhas que devem ser produzidas. Os comandos definidos por este pacote devem ser delimitados pelo ambiente `tikzpicture` que pode ser incluído no ambiente `figure` apresentado anteriormente.

9.1.1 Ambiente `tikzpicture`

Ao utilizar o TikZ para desenhar uma figura você precisa informar ao LaTeX que deseja-se iniciar uma figura. Para isso utiliza-se o ambiente `tikzpicture`. A seguir encontra-se um pequeno exemplo do ambiente `tikzpicture`. Ao utilizar TikZ para desenhar uma figura você precisa informar ao LaTeX que deseja-se iniciar uma figura. Para isso utiliza-se o ambiente `tikzpicture`. A seguir encontra-se um pequeno exemplo do ambiente `tikzpicture`.

Construindo uma reta.

```
\begin{tikzpicture}
  \draw[color=blue] (0,0) -- (1,0);
\end{tikzpicture}
```

Construindo uma reta. 

No exemplo acima podemos notar que, dentro do ambiente `tikzpicture`, os comandos devem terminar com um ponto e vírgula.

Também no exemplo acima, observamos que o ambiente `tikzpicture` não é flutuante. Uma maneira de torná-lo flutuante é envolvendo-o pelo ambiente `figure`.

Uma outra característica do ambiente `tikzpicture` é que comandos recentes são sobrepostos aos comandos antigos. No exemplo a seguir observamos essa característica.

```
\begin{tikzpicture}
  \draw[color=blue] (0,0) -- (4,0);
  \draw[color=red] (0,0) -- (3,0);
  \draw[color=black] (0,0) -- (2,0);
\end{tikzpicture}
```



9.1.2 Sistema de coordenadas

A construção de qualquer figura usando o TikZ requer que seja informado coordenadas de acordo com algum sistema. O TikZ aceita o sistema de coordenadas cartesianas, que corresponde a forma (x, y) , onde x corresponde a coordenada horizontal e y a vertical, e o sistema de coordenadas polares, que corresponde a forma $(a: r)$, onde a a direção em graus e r corresponde ao comprimento do raio.

```
\begin{tikzpicture}
  \draw[color=blue] (0,0) -- (1,0);
  \draw[color=red] (0:0) -- (45:1);
\end{tikzpicture}
```



Além de coordenadas absolutas, o TikZ também aceita coordenadas relativas. Coordenadas relativas devem ser precedidas por +, que significa “adicionar as seguintes coordenadas à coordenada absoluta previamente informada”, ou ++, que significa “adicionar as seguintes coordenadas à coordenada absoluta previamente informada e tornar esta a nova coordenada absoluta previamente informada”.

```
\begin{tikzpicture}
  \draw[color=blue] (0,0) -- +(1,0) -- +(0,1)
  ;
  \draw[color=red] (2,0) -- ++(1,0) --
  ++(0,1);
\end{tikzpicture}
```



O TikZ aceita uma vasta variedade de unidades de medida para as coordenadas, por exemplo: pt, cm, mm ...

```
\begin{tikzpicture}
  \draw (0,0) -- (4,0);
  \draw (0,-1) -- (4pt,-1);
  \draw (0,-2) -- (4cm,-2);
  \draw (0,-3) -- (4mm,-3);
\end{tikzpicture}
```




Pelo exemplo acima verifica-se que caso nenhuma unidade seja especificada é utilizada cm.

Outra característica do TikZ é que ele ajusta a figura criada para ocupar o espaço mínimo necessário. Essa característica é observada no exemplo a seguir que corresponde ao primeiro exemplo com um deslocamento de 5 unidades horizontais e o resultado produzido é idêntico ao do primeiro exemplo.

Construindo uma reta.

```
\begin{tikzpicture}
  \draw[color=blue] (5,0) -- (6,0);
\end{tikzpicture}
```

Construindo uma reta. 

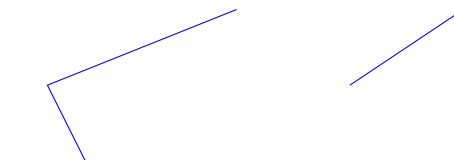
9.1.3 Linhas

Nesta seção iremos tratar da construção de linhas com o TikZ. Pelos exemplos anteriores o leitor já deve ter inferido que o comando `\draw` é responsável pela construção de linhas.

No primeiro exemplo, o comando `\draw` é seguido por um conjunto de opções envolvidas em colchetes, pelas coordenadas do ponto inicial, um operador (no caso `--`) e pelas coordenadas do ponto final.

É possível utilizar o mesmo comando `\draw` com pontos intermediários, a seguir apresentamos um exemplo deste uso.

```
\begin{tikzpicture}
  \draw[color=blue] (0,0) -- (1,0) -- (0.5,
  1) -- (3,2);
  \draw[color=blue] (4,0) -- (5,0) (4.5, 1)
  -- (6,2);
\end{tikzpicture}
```



Além da opção `color` que corresponde a cor da linha e do operador `--` que corresponde a uma linha entre dois pontos existem muitos outros. A seguir apresentamos algumas opções e depois alguns operadores.

Escala

Uma das grandes vantagens do TikZ é a capacidade de reescalar uma figura sem perder qualidade no processo.

A opção `scale` é responsável por escalar a linha a ser desenhada e deve receber o fator de escala a ser utilizado.

```
\begin{tikzpicture}
  \draw[color=blue, scale=2] (0,0) -- (2,0);
  \draw (0,0) -- (2,0);
  \draw[color=red, scale=0.5] (0,0) -- (2,0);
\end{tikzpicture}
```



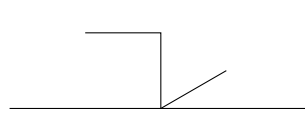
Rotação

A opção `rotate` é responsável por rotacionar a linha a ser desenhada e deve receber a medida em grau a ser utilizada.

```

\begin{tikzpicture}
  \draw (-2,0) -- (2,0);
  \draw[rotate=30] (0,0) -- (1,0);
  \draw[rotate=90] (0,0) -- (1,0) -- (1,1);
\end{tikzpicture}

```



Como podemos observar pelo exemplo acima, o ponto fixo da rotação corresponde ao primeiro ponto do comando.

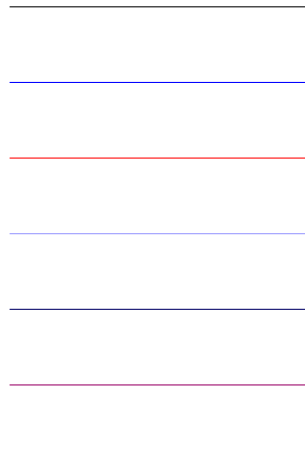
Cores

A opção `color` é responsável pela cor da linha a ser desenhada e deve receber o nome de uma cor previamente definida. No \LaTeX o nome das cores previamente definidas encontram-se disponíveis no pacote `color` e a criação de novas cores pode ser feita utilizando o pacote `xcolor` (um resumo deste pacote é encontrado em <http://en.wikibooks.org/wiki/LaTeX/Colors>).

```

\begin{tikzpicture}
  \draw (0,0) -- (4,0);
  \draw[color=blue] (0,-1) -- (4,-1);
  \draw[color=red] (0,-2) -- (4,-2);
  \draw[color=blue!40] (0,-3) -- (4,-3);
  \draw[color=blue!40!black] (0,-4) -- (4,-4)
  ;
  \draw[color=blue!40!red] (0,-5) -- (4,-5);
  \draw[color=blue!40!red!40!black] (0,-6) --
  (4,-6);
\end{tikzpicture}

```



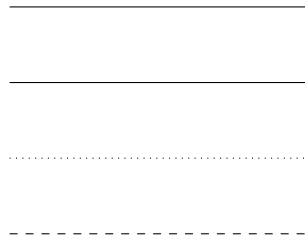
Padrão

Encontram-se predefinidos alguns padrões de linha, alguns deles são: `solid` (contínuo), `dotted` (pontilhado), `dashed` (tracejado), ...

```

\begin{tikzpicture}
  \draw (0,0) -- (4,0);
  \draw[solid] (0,-1) -- (4,-1);
  \draw[dotted] (0,-2) -- (4,-2);
  \draw[dashed] (0,-3) -- (4,-3);
\end{tikzpicture}

```



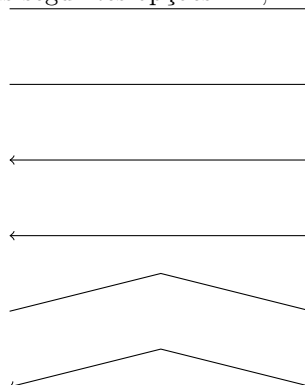
Setas

Para a construção de setas pode-se utilizar uma dentre as seguintes opções: `->`, `<-` e `<->`.

```

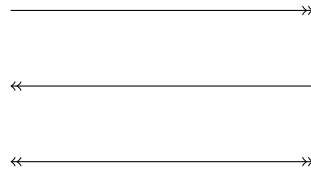
\begin{tikzpicture}
  \draw (0,0) -- (4,0);
  \draw[->] (0,-1) -- (4,-1);
  \draw[<-] (0,-2) -- (4,-2);
  \draw[<->] (0,-3) -- (4,-3);
  \draw[->] (0,-4) -- (2,-3.5) -- (4,-4);
  \draw[<->] (0,-5) -- (2,-4.5) -- (4,-5);
\end{tikzpicture}

```



Também é possível duplicar o indicador da seta utilizando uma dentre as seguintes opções: `->>`, `<<-` e `<<->>`.

```
\begin{tikzpicture}
  \draw (0,0) -- (4,0);
  \draw[->>] (0,-1) -- (4,-1);
  \draw[<<-] (0,-2) -- (4,-2);
  \draw[<<->>] (0,-3) -- (4,-3);
\end{tikzpicture}
```

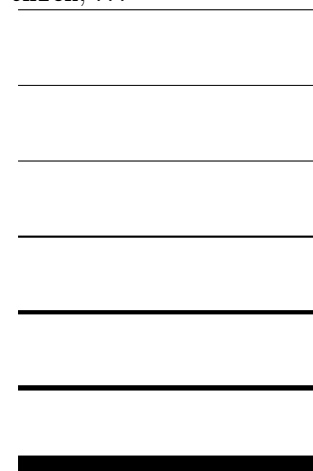


Espessura

A opção `line width` é responsável pela espessura da linha a ser desenhada e deve receber uma medida para a espessura da linha.

Encontram-se predefinidos alguns estilos que fornecem uma maneira mais “natural” de informar a espessura da linha, alguns deles são: `ultra thin`, `thin`, `thick` `ultra thick`, ...

```
\begin{tikzpicture}
  \draw (0,0) -- (4,0);
  \draw[ultra thin] (0,-1) -- (4,-1);
  \draw[thin] (0,-2) -- (4,-2);
  \draw[thick] (0,-3) -- (4,-3);
  \draw[ultra thick] (0,-4) -- (4,-4);
  \draw[line width=2pt] (0,-5) -- (4,-5);
  \draw[line width=6pt] (0,-6) -- (4,-6);
\end{tikzpicture}
```

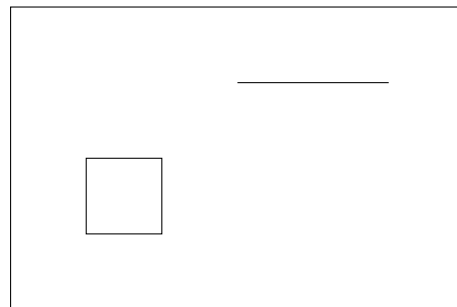


9.1.4 Operadores

Retângulos

Para a construção de retângulos pode-se utilizar o operador `rectangle` sendo que as coordenadas correspondem dois vértices não adjacentes do retângulo.

```
\begin{tikzpicture}
  \draw (0,0) rectangle (6,4);
  \draw (1,1) rectangle (2,2);
  \draw (3,3) rectangle (5,3);
\end{tikzpicture}
```

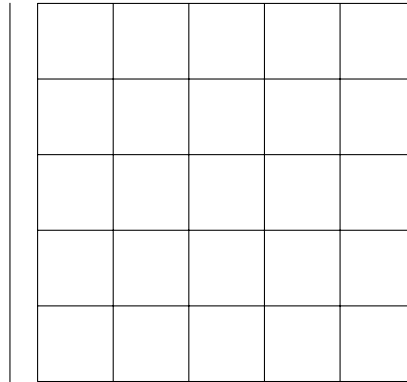


No exemplo acima observamos a ocorrência de um retângulo degenerado em uma linha.

Malha retangular

Algumas vezes deseja-se incluir na figura uma malha retangular. Para isso pode-se utilizar o operador `grid` sendo que, de maneira análoga ao operador `rectangle`, as coordenadas correspondem a dois vértices não adjacentes do retângulo maior.

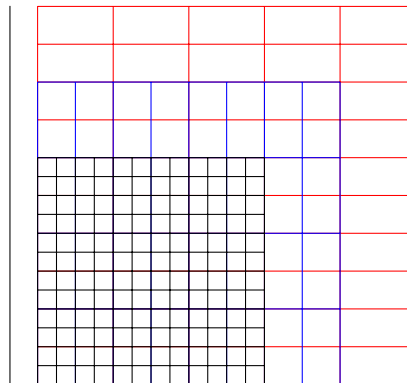
```
\begin{tikzpicture}
  \draw (0,0) grid (5,5);
\end{tikzpicture}
```



Para o operador `grid` estão disponíveis as três opções a seguir:

1. `step`: especifica a distância horizontal e vertical dos elementos da malha retângular;
2. `xstep`: especifica a distância horizontal dos elementos da malha retângular;
3. `ystep`: especifica a distância vertical dos elementos da malha retângular.

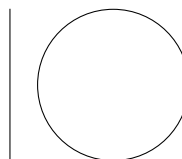
```
\begin{tikzpicture}
  \draw[color=red, ystep=0.5] (0,0) grid
    (5,5);
  \draw[color=blue, xstep=0.5] (0,0) grid
    (4,4);
  \draw[step=0.25] (0,0) grid (3,3);
\end{tikzpicture}
```



Circunferências

Para a construção de circunferências pode-se utilizar o operador `circle` sendo que o operador é seguido pela medida do raio.

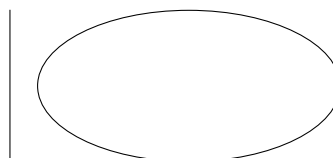
```
\begin{tikzpicture}
  \draw (0,0) circle (1);
\end{tikzpicture}
```



Elipse

Para a construção de uma elipse pode-se utilizar o operador `ellipse` sendo que o operador é seguido pela medida dos raios horizontais e verticais.

```
\begin{tikzpicture}
  \draw (0,0) ellipse (2 and 1);
\end{tikzpicture}
```



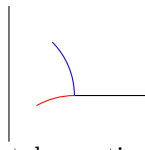
Arcos

Para a construção de parte de circunferência ou de elipse, i.e., um arco pode-se utilizar o operador `arc` que sendo que o operador é seguido por uma tripla separada por dois pontos referentes ao grau inicial, grau final e

```

o raio.
\begin{tikzpicture}
  \draw (0,0) -- (1,0);
  \draw[color=blue] (0,0) arc (0:45:1);
  \draw[color=red] (0,0) arc (90:120:1);
\end{tikzpicture}

```

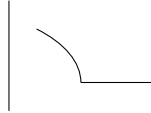


Para o caso de elipses deve-se especificar o raio horizontal e vertical.

```

\begin{tikzpicture}
  \draw (0,0) -- (1,0);
  \draw (0,0) arc (0:45:2 and 1);
\end{tikzpicture}

```



9.1.5 Nó e texto

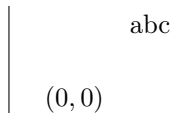
Na seção anterior apresentamos como construir linhas e algumas figuras geométricas como retângulos e circunferências. Nesta seção iremos apresentar como adicionar um pequeno texto próximo a uma linha.

No TikZ o comando `\node` é responsável por inserir um pequeno texto em uma posição específica. A seguir encontra-se um exemplo bastante simples.

```

\begin{tikzpicture}
  \node at (0,0) {$(0,0)$};
  \node at (1,1) {abc};
\end{tikzpicture}

```

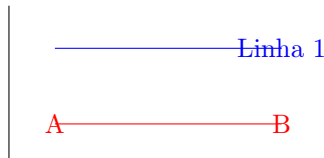


Além do uso apresentado no exemplo acima, o comando `\node` também pode ser utilizado em conjunto com o comando `\draw` como apresentado a seguir.

```

\begin{tikzpicture}
  \draw[color=blue] (0,0) -- (3,0) node {
    Linha 1};
  \draw[color=red] (0,-1) node {A} -- (3,-1)
    node {B};
\end{tikzpicture}

```



Assim como o comando `\draw`, o comando `\node` permite algumas opções que possibilitam aprimorar o exemplo acima. Tais opções serão descritas a seguir.

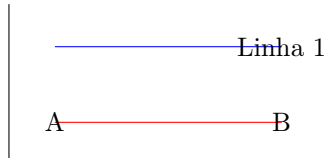
Cores

A cor do texto de um nó é definido pela opção `text` que recebe o nome de uma cor.

```

\begin{tikzpicture}
  \draw[color=blue] (0,0) -- (3,0) node [text
    =black] {Linha 1};
  \draw[color=red, text=black] (0,-1) node {A
    } -- (3,-1) node {B};
\end{tikzpicture}

```



Pelo exemplo acima verificamos que a opção `text` pode ser utilizada tanto como opção do comando `\node` como do comando `draw`.

Ancoras

Muitas vezes não deseja-se colocar o nó nas coordenadas indicada mas próximo dela. Nestes casos deve-se utilizar a opção `anchor` que recebe uma das seguintes orientações:

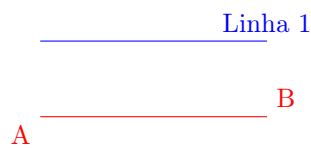
1. north,
2. south,
3. east,
4. west.

É possível combinar as orientações tomando o cuidado da primeira orientação sempre corresponder ao eixo vertical, e.g., `north east`.


```

\begin{tikzpicture}
  \draw[color=blue] (0,0) -- (3,0) node [
    anchor=south] {Linha 1};
  \draw[color=red] (0,-1) node [anchor=north
    east] {A} -- (3,-1) node [anchor=south
    west] {B};
\end{tikzpicture}

```



Como o uso de âncoras costuma ser pouco intuitivo existem algumas opções que são equivalente:

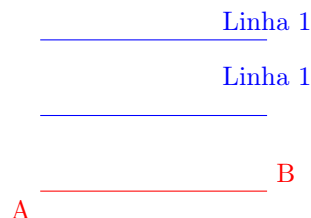
1. `below` é equivalente a `anchor=north`,
2. `above` é equivalente a `anchor=south`,
3. `right` é equivalente a `anchor=west`,
4. `left` é equivalente a `anchor=west`.

Também é possível combinar as opções enumeradas acima seguindo o mesmo cuidado do uso de âncoras, i.e., a primeira orientação sempre corresponde ao eixo vertical. Além disso, essas opções permitem atribuir uma medida para o deslocamento em cada uma das direções.

```

\begin{tikzpicture}
  \draw[color=blue] (0,0) -- (3,0) node [
    above] {Linha 1};
  \draw[color=blue] (0,-1) -- (3,-1) node [
    above=8] {Linha 1};
  \draw[color=red] (0,-2) node [below left] {
    A} -- (3,-2) node [above right] {B};
\end{tikzpicture}

```



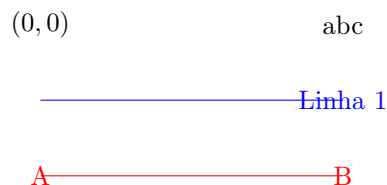
Nomeação

Os nós possuem uma característica muito útil que é a possibilidade de nomeá-los. Para atribuir um nome a um nó utiliza-se parênteses logo em seguida do comando `\node`.

```

\begin{tikzpicture}
  \node (origin) at (0,0) {$(0,0)$};
  \node (abc) at (4,0) {abc};
  \draw[color=blue] (0,-1) -- (4,-1) node (
    Linha 1) {Linha 1};
  \draw[color=red] (0,-2) node (A) {A} --
    (4,-2) node (B) {B};
\end{tikzpicture}

```



Após nomear um nó podemos utilizar sua posição a partir de seu nome.

```

\begin{tikzpicture}
  \node (A) at (0,0) {A};
  \node (B) at (4,0) {B};
  \draw (A) -- (B);
\end{tikzpicture}

```



No exemplo acima nota-se que a linha desenhada não inicia exatamente nas coordenadas correspondentes aos nós mas na fronteira do nó, i.e., a linha inicia-se no contorno do nó.

```

\begin{tikzpicture}
  \node[draw] (A) at (0,0) {A};
  \node[draw] (B) at (4,0) {B};
  \draw (A) -- (B);
\end{tikzpicture}

```



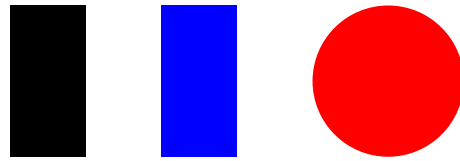
9.1.6 Preenchimento

Até o momento apenas contruímos linhas e algumas figuras geométricas. Como devemos proceder para preencher uma figura? Para preencher uma figura utiliza-se a opção `fill`.

```

\begin{tikzpicture}
  \path[fill] (0,0) -- (4,0);
  \path[fill] (0,-1) rectangle (1,-3);
  \path[fill=blue] (2,-1) rectangle (3,-3);
  \path[fill=red] (5,-2) circle (1);
\end{tikzpicture}

```

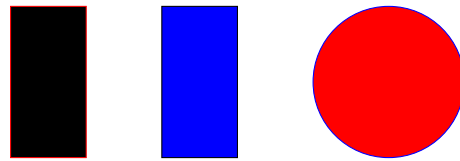


Pelo exemplo acima verifica-se que a opção `fill` apenas preenche a figura sem tratar o contorno. Isso ocorre pois o contorno é determinado pela opção `draw` vista anteriormente. No exemplo a seguir utilizamos as opções `fill` e `draw` em conjunto.

```

\begin{tikzpicture}
  \path[fill,draw] (0,0) -- (4,0);
  \path[fill,draw=red] (0,-1) rectangle
    (1,-3);
  \path[fill=blue,draw] (2,-1) rectangle
    (3,-3);
  \path[fill=red,draw=blue] (5,-2) circle (1)
    ;
\end{tikzpicture}

```



Ao invés de utilizar o comando `\path` com a opção `fill` é possível utilizar o comando `\fill` e o comando `\filldraw` no lugar do comando `\path` com as opções `fill` e `draw`.

De maneira geral, é permitido utilizar qualquer opção do comando `\path` como um comando correspondente a uma opção do comando `\path`, portanto as seguintes construções são válidas:

```
\fill[draw=red] (0,-1) rectangle (1,-3);
```

e

```
\draw[fill=blue] (2,-1) rectangle (3,-3);
```

e equivalentes a construção utilizada no exemplo anterior.

Padrão

No capítulo anterior foi apresentado alguns padrões para linhas como pontilhado e tracejado. Agora vamos apresentar alguns padrões de preenchimento que são definidos pela opção `pattern`.

Para utilizar os padrões predefinidos é necessário carregar a biblioteca `patterns`, i.e, adicionar a seguinte linha.

```
\usetikzlibrary{patterns}
```

no preâmbulo do documento.

```

\begin{tikzpicture}
  \path[pattern=dots] (0,0) rectangle (1,-2);
  \path[pattern=fivepointed stars] (2,0)
    rectangle (3,-2);
  \path[pattern=bricks] (5,-1) circle (1);
\end{tikzpicture}

```



Para atribuir um cor ao padrão a ser utilizado deve-se utilizar a opção `pattern color`.

```

\begin{tikzpicture}
  \path[pattern=dots] (0,0) rectangle (1,-2);
  \path[pattern=fivepointed stars, pattern
    color = blue] (2,0) rectangle (3,-2);
  \path[pattern=bricks, pattern color=red]
    (5,-1) circle (1);
\end{tikzpicture}

```



Capítulo 10

Referência bibliográfica

O ambiente `thebibliography` é utilizado para a inclusão da referência bibliográfica. Como ele exige um grande trabalho para ser utilizado e é difícil reutilizá-lo foi desenvolvido o BibTeX (um banco de dados plano para referências bibliográfica e um executável para construção do ambiente `thebibliography`). Posteriormente foi criado o pacote `biblatex` que estende o BibTeX. A seguir será apresentado um pouco do BibTeX e do `biblatex`.

10.1 BibTeX

O “banco de dados” corresponde a um arquivo de texto com a extensão `.bib`. Cada referência no BibTeX segue a seguinte estrutura:

```
@tipo{identificador ,
campo1 = {valor do campo 1},
campo2 = {valor do campo 2},
campo3 = {valor do campo 3},
...
}
```

Uma lista com alguns dos `tipo`'s permitido pelo BibTeX é apresentada na Tabela 10.1.

Tabela 10.1: `tipo`'s disponíveis no BibTeX padrão.

Código	Descrição
<code>article</code>	Um artigo presente em algum periódico, revista, jornal que forme uma unidade própria e possua título.
<code>book</code>	Um livro com um ou mais autores que levam crédito pela obra.
<code>inbook</code>	Uma parte de um livro que forme uma unidade própria e possua título.
<code>booklet</code>	Material com as características de um livro, mas que não foi formalmente publicado.
<code>incollection</code>	Uma parte de um livro composto dos trabalhos de vários autores, normalmente possui um editor.
<code>proceedings</code>	Uma palestra de uma conferência.
<code>inproceedings</code>	Um artigo apresentado em uma conferência.
<code>manual</code>	Um documento técnico, pode não estar disponível em versão impressa.
<code>techreport</code>	Um documento técnico produzido por uma instituição de ensino, comércio ...
<code>mastersthesis</code>	Uma tese de mestrado escrita para uma instituição de ensino.
<code>phdthesis</code>	Uma tese de doutorado escrita para uma instituição de ensino.
<code>unpublished</code>	Um trabalho que não foi formalmente publicado, como um manuscrito.
<code>misc</code>	Utilizado quando a obra não se encaixa nos <code>tipo</code> 's anteriores.

Uma lista com alguns dos `campo`'s permitido pelo BibTeX é apresentada na Tabela 10.2.

Uma das grandes vantagens de se utilizar o BibTeX é que as chances de encontrar o BibTeX de algum material na internet é extremamente alta. Tanto o Google Scholar como o Google Books disponibilizam o

Tabela 10.2: campo's disponíveis no BibTeX padrão.

Código	Descrição
author	Autor(es) da obra.
editor	Editor da obra, caso exista.
publisher	Editadora da obra.
title	Título da obra.
booktitle	Quando a obra encontra-se como parte de um livro utiliza-se este campo para o título do livro.
journal	Título do jornal ou periódico que contem a obra.
month	Mês da publicação da obra.
year	Ano da publicação da obra, deve ser um inteiro.
edition	Edição da obra. Deve ser um número inteiro.
howpublished	Tipo de publicação não usual.
school	Instituição detentora da obra.
pages	Uma página ou mais de um trabalho.
note	Alguma informação que não adequa-se aos camp's anteriores.

BibTeX para todos os materiais indexados em suas respectivas bases de dados.

10.2 biblatex

O pacote `biblatex` define o comando `\addbibresource{referencias.bib}` que é inserido no *preâmbulo* e especifica o arquivo que armazena as referências bibliográficas, nesse caso `referencias.bib` e o comando `\printbibliography` que é inserido na posição onde deseja-se incluir as referências.

O estilo a ser utilizado nas referências bibliográficas é informado como uma opção do pacote `biblatex` como indicado a seguir:

```
\usepackage[style=estilo]{biblatex}
```

Alguns dos estilos existentes são:

- `numeric`,
- `alphabetic`,
- `authoryear`, ...

Para que uma entrada do bando de dados seja incluído na referência bibliográfica ele precisa ser mencionada em algum dos arquivos `.tex` que compõe a obra. Para mencionar uma referência utiliza-se uma das variantes do comando `\cite{id}`, onde `id` corresponde ao identificador utilizado na entrada do BibTeX para a referência desejada.

O comando `\cite{id}` insere o número da referência entre colchetes, como mostrado abaixo:

Comando	Resultado
<code>\cite{Sauer:2004:Parcolumns}</code>	[9]
<code>\cite{Neves:AprendendoLaTeX}</code>	[6]
<code>\cite{Pakin:2009:Symbol}</code>	[7]
<code>\cite{Moses:2007>Listings}</code>	[5]

Para inserir o nome dos autores e o número da referência entre colchetes, utiliza-se o comando `\textcite{id}`, como mostrado abaixo:

Para inserir apenas o nome dos autores utiliza-se o comando `\citeauthor{id}`, como mostrado abaixo:

Para inserir apenas o título da referência utiliza-se o comando `\citetitle{id}`, como mostrado abaixo:

Para inserir apenas o ano de publicação da referência utiliza-se o comando `\citeyear{id}`, como mostrado abaixo:

Para citações múltiplas, utiliza-se os comandos `\cites{id1,id2,id3}` ou `\textcites{id1,id2,id3}`, como mostrado abaixo:

Comando	Resultado
<code>\textcite{Sauer:2004:Parcolumns}</code>	Sauer [9]
<code>\textcite{Neves:AprendendoLaTeX}</code>	Neves [6]
<code>\textcite{Pakin:2009:Symbol}</code>	Pakin [7]
<code>\textcite{Moses:2007:Listings}</code>	Moses [5]

Comando	Resultado
<code>\citeauthor{Sauer:2004:Parcolumns}</code>	Sauer
<code>\citeauthor{Neves:AprendendoLaTeX}</code>	Neves
<code>\citeauthor{Pakin:2009:Symbol}</code>	Pakin
<code>\citeauthor{Moses:2007:Listings}</code>	Moses

Comando	Resultado
<code>\citetitle{Sauer:2004:Parcolumns}</code>	<i>The parcolumns package</i>
<code>\citetitle{Neves:AprendendoLaTeX}</code>	<i>O que vou aprendendo em LaTeX</i>
<code>\citetitle{Pakin:2009:Symbol}</code>	<i>The Comprehensive LaTeX Symbol List</i>
<code>\citetitle{Moses:2007:Listings}</code>	<i>The Listings Package</i>

Comando	Resultado
<code>\citeyear{Sauer:2004:Parcolumns}</code>	2004
<code>\citeyear{Neves:AprendendoLaTeX}</code>	
<code>\citeyear{Pakin:2009:Symbol}</code>	2009
<code>\citeyear{Moses:2007:Listings}</code>	2007

Comando	Resultado
<code>\cites{Neves:AprendendoLaTeX,Sauer:2004:Parcolumns}</code>	[6, 9]
<code>\cites{Moses:2007:Listings,Pakin:2009:Symbol}</code>	[5, 7]
<code>\textcites{Neves:AprendendoLaTeX,Sauer:2004:Parcolumns}</code>	Neves [6], Sauer [9]
<code>\textcites{Moses:2007:Listings,Pakin:2009:Symbol}</code>	Moses [5], Pakin [7]

Por último, caso deseje incluir uma referência na referência bibliográfica mas suprimi-la ao longo do texto você deve utilizar o comando `\nocite{id}`.

Bibliografia

- [1] Johannes Braams. *Babel, a multilingual package for use with LaTeX's standard document class*. <http://www.ctan.org/pub/tex-archive/macros/latex/required/babel/babel.pdf>. Jul. de 2008.
- [2] George Grätzer. *More Math into LaTeX*. 4^a ed. Springer, 2007.
- [3] A.J. Hildebrand. *TeX Resources*. [Online; accessed 22-July-2012]. 2012. URL: [%5Curl%7Bhttp://www.math.uiuc.edu/~hildebr/tex/%7D](http://www.math.uiuc.edu/~hildebr/tex/).
- [4] L. Lamport. *LATEX: a document preparation system : user's guide and reference manual*. p. 2. Addison-Wesley Pub. Co., 1994. ISBN: 9780201529838. URL: <http://books.google.com.br/books?id=khVUAAAAMAAJ>.
- [5] Brooks Moses. *The Listings Package*. <ftp://ftp.tex.ac.uk/tex-archive/macros/latex/contrib/listings/listings.pdf>. Fev. de 2007.
- [6] Antero Neves. *O que vou aprendendo em LaTeX*. <http://aprendolatex.wordpress.com/>.
- [7] Scott Pakin. *The Comprehensive LaTeX Symbol List*. 2009.
- [8] Reginaldo J. Santos. *Introdução ao LaTeX*. <http://www.mat.ufmg.br/~regi/topicos/intlat.pdf>. Set. de 2009.
- [9] Jonathan Sauer. *The parcolumns package*. www.ctan.org/tex-archive/macros/latex/contrib/sauerj/parcolumns.pdf. Nov. de 2004.
- [10] Gilberto Souto. *Curso de LaTeX*. www.ufsm.br/petfisica/extras/arquivos/latex.pdf.
- [11] E. Swanson, A.A. O'Sean e A.T. Schleyer. *Mathematics Into Type*. American Mathematical Society, 1999. ISBN: 9780821819616. URL: <http://books.google.com.br/books?id=5YKiwrpJntoC>.
- [12] Till Tantau. *The TikZ and PGF Packages - Manual for version 2.10*. 2010.
- [13] Hideo Umeki. *The geometry package*. <ftp://ftp.tex.ac.uk/tex-archive/macros/latex/contrib/geometry/geometry.pdf>. Set. de 2010.
- [14] Wikipedia. *Comparison of TeX editors — Wikipedia, The Free Encyclopedia*. [Online; accessed 22-July-2012]. 2012. URL: [%5Curl%7Bhttp://en.wikipedia.org/wiki/Comparison_of_TeX_editors%7D](http://en.wikipedia.org/wiki/Comparison_of_TeX_editors).
- [15] Wikipedia. *ENIAC — Wikipédia: a enciclopédia livre*. [Online; accessed 22-July-2012]. 2012. URL: [%5Curl%7Bhttp://pt.wikipedia.org/wiki/ENIAC%7D](http://pt.wikipedia.org/wiki/ENIAC).
- [16] Wikipedia. *LaTeX — Wikipedia, The Free Encyclopedia*. [Online; accessed 14-July-2012]. 2012. URL: [%5Curl%7Bhttp://en.wikipedia.org/wiki/LaTeX%7D](http://en.wikipedia.org/wiki/LaTeX).
- [17] Wikipedia. *TeX — Wikipedia, The Free Encyclopedia*. [Online; accessed 14-July-2012]. 2012. URL: [%5Curl%7Bhttp://en.wikipedia.org/wiki/TeX%7D](http://en.wikipedia.org/wiki/TeX).
- [18] Wikipedia. *UNIX — Wikipedia, The Free Encyclopedia*. [Online; accessed 22-July-2012]. 2012. URL: [%5Curl%7Bhttp://en.wikipedia.org/wiki/UNIX%7D](http://en.wikipedia.org/wiki/UNIX).

Índice

- .tex, 7
- alinhamento, 20
- ambiente
 - align, 35
 - enumerate, 16
 - equation, 34
 - figure, 28, 41
 - frame, 11
 - itemize, 16
 - quotation, 15
 - quote, 15
 - table, 17
 - tabular, 16
 - tikzpicture, 41
 - verbatim, 15
- aspas, 10
- beamer
 - overlay, 12
 - tema, 13
- comando
 - %, 10
 - &, 16, 35
 - \\, 9, 16, 34, 35
 - \caption, 17, 28
 - \clearpage, 17, 29
 - \color, 27
 - \documentclass, 8
 - class, 8
 - \draw, 42
 - \footnote, 15
 - \hspace, 20
 - \hyphenation, 10
 - \includegraphics, 28
 - \input, 8
 - \item, 15
 - \label, 17, 34
 - \newline, 9
 - \newpage, 9
 - \pause, 12
 - \ref, 17
 - \tabularnewline, 16
 - \tag, 34
 - \text, 33
 - \textcolor, 27
 - \url, 27
 - \usepackage, 23
 - \verb, 15, 27
 - \vspace, 20
- comentários, 10
- Emacs, *ver* IDE
- espaços em branco, 19
- figura, 27
- fonte, 19
 - cor, 27
 - tamanho, 19
- hifenização, 10
- IDE, 7
 - informação, 7, 9
 - instalação, 7
- Kile, *ver* IDE
- lista, 15
- Mac OS X, *ver* instalação
- margens, 24
- MikTeX, *ver* instalação
- modo matemático
 - acento, 32
 - binômio, 37
 - chaves, *ver* delimitadores
 - colchetes, *ver* delimitadores
 - congruência, 37
 - delimitadores, 32
 - demonstração, 35
 - displayed, 31
 - espaçamento, 33
 - expoente, 32
 - funções definidas por partes, 36
 - índice, 32
 - inline, 31
 - matrizes, 33
 - múltiplas equações, 35
 - nomes longos para variáveis, 33
 - novos operadores, 33
 - numeração, 34
 - operações aritméticas básicas, 32
 - parênteses, *ver* delimitadores

- raiz quadrada, 37
- sistemas de equações, 36
- tag*, 34
- teorema, 34
- texto, 33
- vetores, *ver* matrizes

múltiplos arquivos, 8

nota de rodapé, 15

nova linha, 9

pacote

- amsmath*, 31
- babel*, 24
- color*, 27
- fontenc*, 24
- geometry*, 24
- graphicx*, 27, 28
- inputenc*, 23
- tikz*, 41
- url*, 27

parágrafo, 9

pontuação, 10

preâmbulo, 7, 8, 23

proTeXt, *ver* instalação

referência cruzada, 17

tabela, 16

TeX Live, *ver* instalação

Texmaker, *ver* IDE

TeXworks, *ver* IDE

TikZ

- âncora, 46
- arco, 45
- circunferência, 45
- coordenadas relativas, 42
- cor, 43
- elipse, 45
- escala, 42
- espessura, 44
- nó, 46
- preenchimento, 47
- retângulo, 44
- rotação, 42
- seta, 43
- sistema de coordenadas cartesianas, 41
- sistema de coordenadas polares, 41
- texto, *ver* nó

traço, 10