

Cursos CAMECC
Introdução ao L^AT_EX para o Curso 29
Licenciatura em Matemática

Raniere Silva

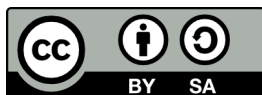
23 de julho de 2012

Licença

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 um mini-curso voltados aos aulas do curso de Licenciatura em Matemática da Universidade Estadual de Campinas (UNICAMP).

O objetivo do mini-curso é apresentar o LaTeX aos futuros licenciados em matemática e ajudá-los a dominar essa poderosa ferramenta de trabalho.

Quando me perguntam por que utilizar o LaTeX eu respondo:

1. É uma ferramenta livrer.
2. É bastante estável (lançado em 1985 por Laslie Lamport e baseado no TeX que foi lançado por Donald Knuth em 1978).
3. E possui uma ótima qualidade tipográfica, i.e., muito bonito.

Além dos três motivos mencionados acima ainda posso dizer que

- é, usualmente, utilizado na produção dos mais variados documentos técnico e científicos,
- e encoraja o autor a preocupar-se apenas com o conteúdo.

O mini-curso foi preparado para ser ministrado em quatro horas sendo que cada hora deve cobrir um dos capítulos. Além dos capítulos encontra-se no apêndice uma preve história de fatos importantes na área de computação que ajudam a entender o surgimento do LaTeX, uma explicação técnica do LaTeX e dicas de locais para procurar ajuda, e alguns exercícios.

Sumário

Prefácio	iii
1 Olá \LaTeX	1
1.1 Instalação	1
1.2 Arquivo <code>.tex</code>	1
1.3 <i>Preâmbulo</i>	2
1.4 Hello world	4
1.4.1 Teclado e Idioma	4
1.4.2 Espaços, linhas, parágrafos e páginas	5
1.4.3 Hifenização	6
1.4.4 Acentos	6
1.5 Caracteres especiais	6
1.5.1 Aspas	6
1.5.2 Traço	7
1.5.3 Pontos sucessivos	7
1.5.4 Pontuação e demais símbolos	7
1.5.5 Comentários	7
1.6 Margens	8
1.6.1 <code>geometry</code>	8
1.6.2 Estilo de página	9
1.7 Fonte	9
1.7.1 Tamanho	10
1.7.2 Cor	10
1.7.3 Edição direta	10
1.8 Espaçamento	11
1.8.1 Espaçamento horizontal	11
1.8.2 Linha horizontal	11
1.8.3 Espaçamento vertical	11
1.8.4 Linha verticais	11
1.9 Alinhamento	12
2 Aproveitando ao máximo o \LaTeX	13
2.1 Endereços da internet	13
2.2 Nota de rodapé	13
2.3 Referência cruzada	13
2.4 Listas	14

2.4.1	<code>itemize</code>	14
2.4.2	<code>enumerate</code>	14
2.5	Figuras	14
2.5.1	Arquivos de imagem	14
2.5.2	<code>figure</code>	15
2.6	Tabelas	16
2.6.1	<code>tabular</code>	16
2.6.2	<code>table</code>	17
2.6.3	Extensão Calc2LaTeX	18
2.7	Citações	18
3	Matemática no \LaTeX, amsmath	19
3.1	Modo matemático	19
3.1.1	<i>Inline</i>	19
3.1.2	<i>Displayed</i>	19
3.1.3	Uso de <i>inline</i> e <i>displayed</i>	20
3.2	Primeiros comandos no modo matemático	20
3.2.1	Operações aritméticas básicas	20
3.2.2	Índices e expoentes	20
3.2.3	Acentos	20
3.2.4	Delimitadores	21
3.2.5	Textos e espaçamentos	21
3.2.6	Matrizes	22
3.3	Comandos avançados no modo matemático	22
3.3.1	Equações, numeração e referenciação	22
3.3.2	<i>Tags</i>	23
3.3.3	Teorema	23
3.3.4	Demonstração	23
3.3.5	Alinhamento	24
3.3.6	Fórmulas longas	24
3.3.7	Ocultando termos	24
3.3.8	Funções definidas por partes	25
3.3.9	Fonte e Símbolos	25
3.4	Símbolos e operadores	26
3.4.1	Raiz quadrada	26
3.4.2	Binomial	29
3.4.3	Congruências	29
4	Apresentações e desenhos utilizando o \LaTeX	31
4.1	TikZ	31
4.1.1	Ambiente <code>tikzpicture</code>	31
4.1.2	Sistema de coordenadas	32
4.1.3	Linhas	33
4.1.4	Operadores	35
4.1.5	Nó e texto	37
4.1.6	Preenchimento	39
4.2	Classe Beamer	40

4.2.1	Primeiro <i>slide</i>	41
4.2.2	Título do <i>slide</i>	41
4.2.3	Comandos e ambientes do LaTeX	41
4.2.4	<i>Overlays</i>	42
4.2.5	Temas	43
A	História	45
B	Obtendo ajuda	47
C	Exercícios	49
	Index	56

Lista de Tabelas

1.1	Parâmetros disponíveis para <code>options</code> .	3
1.2	Parâmetros disponíveis para <code>class</code> .	4
1.3	Acentuação (utilizando a vogal “o” para exemplo).	6
1.4	Para pontuação e símbolos especiais.	7
1.5	Opções disponíveis para <code>parameter</code> , referente ao pacote <code>geometry</code> .	8
1.6	Opções disponíveis para <code>style</code> .	9
1.7	Opções disponíveis para <code>XX</code> da fonte.	9
1.8	Opções disponíveis para o tamanho da fonte, em ordem crescente.	10
2.1	Opções disponíveis para <code>parameter</code> .	15
2.2	Opções disponíveis para <code>place</code> .	16
2.3	Opções disponíveis para <code>colunas</code> .	17
2.4	Relação entre corrente e tensão para determinado circuito.	18
2.5	Opções disponíveis para <code>place</code> .	18
3.1	Acentos disponíveis no modo matemático.	21
3.2	Delimitadores disponíveis no LaTeX.	21
3.3	Espaçamento no modo matemático.	22
3.4	Opções disponíveis para <code>XX</code> da fonte para o alfabeto matemático.	25
3.5	Opções disponíveis para <code>XX</code> da fonte para o alfabeto matemático interpretado como símbolo.	26
3.6	Setas	26
3.7	Relações binárias	27
3.8	Operadores binários	28
3.9	Operadores puros.	28
3.10	Operadores com intervalos.	28
3.11	Operadores similares aos limites.	28
3.12	Outros símbolos matemáticos	29
3.13	Alfabeto Grego, letras minúsculas	29
3.14	Alfabeto Grego, letras maiúsculo	29

Capítulo 1

Olá L^AT_EX

Neste primeiro capítulo apresentamos os conhecimentos mínimos de todo usuário do LaTeX. Os capítulos A e B são uma complementação a este capítulo podendo ser lidos de maneira independente.

1.1 Instalação

Para utilizar o LaTeX você precisa das macros que compõem o LaTeX, disponíveis para

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

e de um editor de texto. É recomendado que ao invés de um editor de texto utilize-se uma IDE (*Integrated Development Environment*) própria para o LaTeX, como

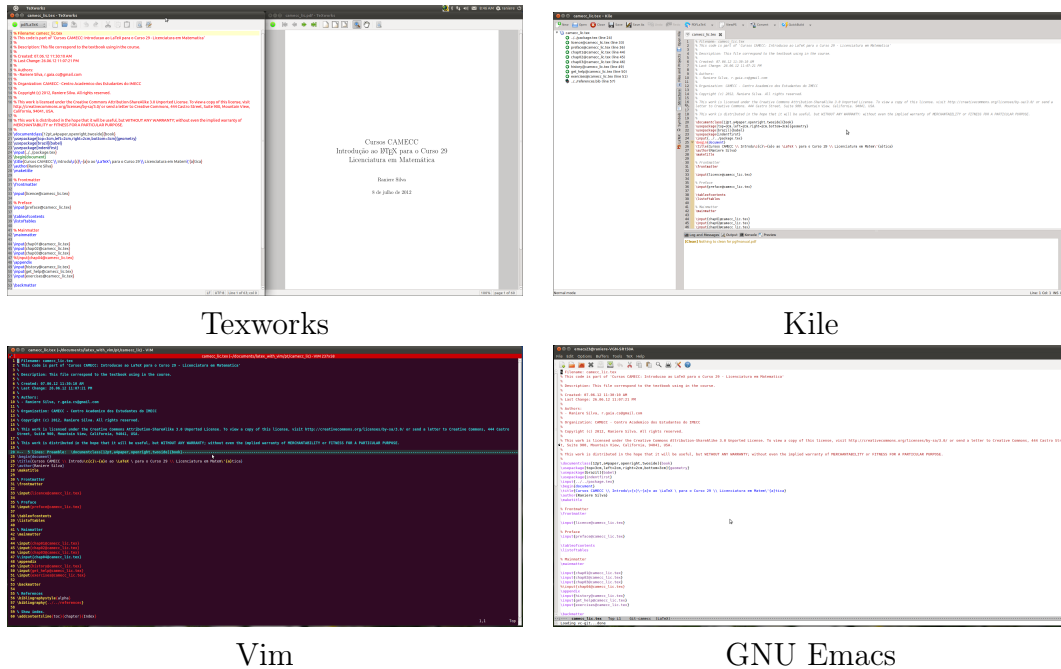
- TeXworks (<http://www.leliseron.org/texworks/>),
- Kile (<http://kile.sourceforge.net/>),
- Texmaker (<http://www.xmlmath.net/texmaker/>).

O TeXworks costuma acompanhar a maioria das distribuições do LaTeX e por isso será utilizado neste curso. Uma lista com várias IDE's encontra-se disponível em http://en.wikipedia.org/wiki/Comparison_of_TeX_editors e imagens de algumas delas são apresentadas na Figura 1.1.

1.2 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*

Figura 1.1: *Screenshots* de alguns IDE's

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.

1.3 *Preâmbulo*

O *preâmbulo* deve ser iniciado por

```
\documentclass[options]{class}
```

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

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

onde `class` indica o tipo de documento a ser criado e `options` é uma lista de palavras chaves separadas por vírgula que personaliza o compartimento de `class` (na Tabela 1.1 encontra-se algumas das palavras chaves disponíveis).

Tabela 1.1: Parâmetros disponíveis para `options`.

Função	Código	Descrição
Tamanho	10pt	Utiliza, por padrão, o tamanho 10.
	11pt	Tamanho 11.
	12pt	Tamanho 12.
Papel		Utiliza, por padrão, o tamanho da folha correspondente carta.
	letterpaper	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 1.2 e outros são indicados em <http://aprendolatex.wordpress.com/2007/07/15/mais-classes-de-documentos/>. Existe ainda alguns arquivos `.cls` personalizados disponíveis na internet, destacando-se o `abnt.cls`, disponível em <http://abntex.codigolivre.org.br/>, indicado para documentos que devem seguir as normas da ABNT e o usuário também pode escrever sua própria `class`.

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:

Tabela 1.2: Parâmetros disponíveis para `class`.

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.

```
\usepackage[options]{package}
```

onde `package` é o nome do pacote e `options` é uma lista de palavras chaves correspondente a opções do pacote.

Por último, é no *preâmbulo* que o usuário também pode definir seus próprios comandos e ambientes³.

1.4 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*.

1.4.1 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 1.4.2 abordaremos como o \LaTeX interpreta o espaço e enter (mudança de linha).

³Não será abordado neste curso, uma ótima fonte é http://en.wikibooks.org/wiki/LaTeX/Customizing_LaTeX

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 matemático⁴.

Existem, também, 13 símbolos especiais ($\#$ $\$$ $\%$ $\&$ \sim $_$ $\^$ \backslash $\{$ $\}$ $\@$ $"$ $/$) 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`. 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}
```

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`.

Além disso, deve-se utilizar o 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` que possui as seguintes opções para o idioma português: `portuges`, `portuguese`, `brazil`, `brazilian`. Maiores detalhes podem ser encontrados na documentação do pacote[Bra08].

1.4.2 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)</pre>		<pre>Hello world.(2 spaces) Hello</pre>
<pre>Hello world.(3 spaces)</pre>		<pre>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)</pre>		<pre>Hello world.(2 spaces) Hello</pre>
<pre>Hello \ world.(3 spaces)</pre>		<pre>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 mudança de linha no LaTeX é representada

⁴O modo matemático é apresentado no capítulo 3.

por `\` ou pelo comando `\newline`, como ilustrada a seguir.

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

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.

Por último é importante avisar que, por padrão, o primeiro parágrafo de capítulo, seções, ..., não é identado. Quando desejar-se indentar o primeiro parágrafo uma solução é utilizar o pacote `indentfirst`.

1.4.3 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.

1.4.4 Acentos

Embora seja possível utilizar algumas codificações de arquivo que suportam acentuação utilizando o pacote `inputenc` é importante saber como inserir os acentos utilizando apenas a tabela ASCII que é apresentado na Tabela 1.3.

Tabela 1.3: 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>	ö	<code>\c{c}</code>	ç	<code>\t{oo}</code>	ôô
<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

1.5 Caracteres especiais

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

1.5.1 Aspas

Para as aspas não deve-se usar o caracter 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)

1.5.2 Traço

LaTeX admite três tipos de traço.

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

1.5.3 Pontos sucessivos

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

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

1.5.4 Pontuação e demais símbolos

Para pontuação e demais símbolos especiais deve-se proceder como na Tabela 1.4.

Tabela 1.4: 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>	€

Destaca-se que para que o símbolo € seja impresso é necessário que o *preâmbulo* contenha a seguinte linha de código

```
\usepackage[official]{eurosym}
```

1.5.5 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 não é processado.

1.6 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.

1.6.1 `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{parameter = length, ...}
```

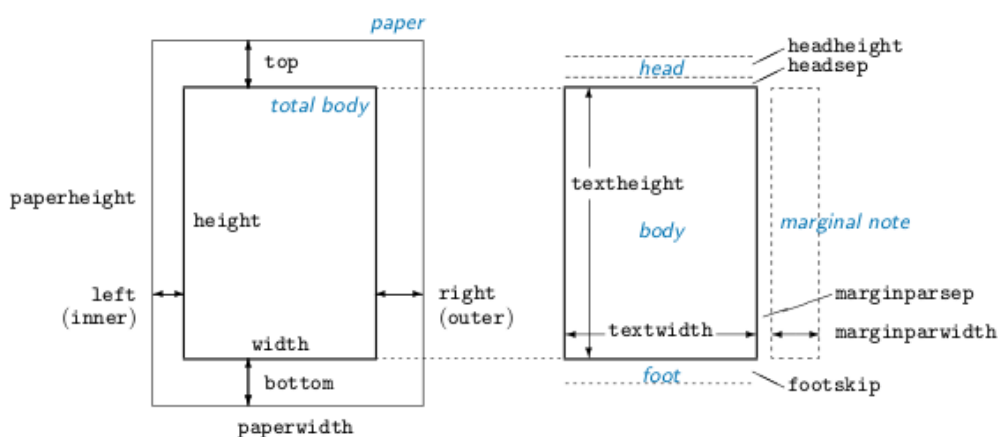
ou

```
\usepackage[parameter = length, ...]{geometry}
```

Podemos utilizar `length` em qualquer unidade disponível no \LaTeX , mm, cm e outras. Já as opções para `parameter` mais utilizadas são apresentadas na Tabela 1.5 e ilustradas na Figura 1.2.

Tabela 1.5: Opções disponíveis para `parameter`, 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: [Ume10]

Figura 1.2: Ilustração das opções disponíveis para `parameter` apresentadas na Tabela 1.5.

1.6.2 Estilo de página

Existe um estilo de página definido como padrão⁵, 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 1.6.

Tabela 1.6: 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`.

1.7 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 1.7 apresenta alguns das opções disponíveis.

Tabela 1.7: 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 1.7.

Itálico: <code>\textit{novo texto}</code> . \\	Itálico: <i>novo texto</i> .
Negrito: <code>\textbf{novo texto}</code> . \\	Negrito: novo texto .
Romano: <code>\textrm{novo texto}</code> . \\	Romano: novo texto.
Sans serif: <code>\textsf{novo texto}</code> . \\	Sans serif: novo texto.
Maquina de escrever: <code>\texttt{novo texto}</code> . \\	Maquina de escrever: novo texto.
Caixa alta: <code>\textsc{novo texto}</code> .	Caixa alta: NOVO TEXTO.

⁵Corresponde ao estilo `plain` apresentado na Tabela 1.6.

1.7.1 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 1.8 apresenta algumas opções disponíveis).

Tabela 1.8: 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.

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

<code>{\tiny muito pequeno} \\\</code>	<code>muito pequeno</code>
<code>{\small pequeno} \\\</code>	<code>pequeno</code>
<code>fonte padrao \\\</code>	<code>fonte padrao</code>
<code>{\Large grande} \\\</code>	<code>grande</code>
<code>{\Huge enorme}</code>	<code>enorme</code>

1.7.2 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.

<code>\textcolor{blue}{azul} \\\</code>	<code>azul</code>
<code>{\color{blue}azul}</code>	<code>azul</code>

1.7.3 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 no mesmo parágrafo.

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

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

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

1.8 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.

1.8.1 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*`.

1.8.2 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.

1.8.3 Espaçamento vertical

No capítulo anterior informamos como mudar de linha, nesta seção vamos trabalhar com o espaço entre as linhas.

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.

1.8.4 Linha verticais

O comando `\vrule` produz uma linha vertical.

1.9 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).

<code>\begin{flushleft}</code>	<code>esquerda</code>	
<code>\end{flushleft}</code>		
<code>\begin{center}</code>	<code>centralizado</code>	
<code>\end{center}</code>		
<code>\begin{flushright}</code>	<code>direita</code>	
<code>\end{flushright}</code>		

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

Capítulo 2

Aproveitando ao máximo o L^AT_EX

Neste capítulo apresentado ferramentas mais avançadas do L^AT_EX como listas, referências cruzadas, tabelas, figuras, bibliografia e outras.

2.1 Endereços da internet

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

2.2 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.

2.3 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 não será tratado neste curso¹.

Para alguns comandos e ambientes o L^AT_EX 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 sequência de caracteres², *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.

¹Os interessados podem dar uma olhada em http://en.wikibooks.org/wiki/LaTeX/Bibliography_Management

²Recomenda-se escolher uma sequência “amigável”.

2.4 Listas

Para a construção de listas podemos utilizar um dos quatro ambientes: `itemize`, `enumerate`, `description`³ ou `list`⁴. 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.

2.4.1 `itemize`

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

<code>\begin{itemize}</code>	
<code>\item Primeiro;</code>	• Primeiro;
<code>\begin{itemize}</code>	
<code>\item Subitem;</code>	– Subitem;
<code>\end{itemize}</code>	
<code>\item Segundo.</code>	• Segundo.
<code>\end{itemize}</code>	

2.4.2 `enumerate`

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

<code>\begin{enumerate}</code>	
<code>\item Primeiro;</code>	1. Primeiro;
<code>\begin{enumerate}</code>	
<code>\item Subitem;</code>	(a) Subitem;
<code>\end{enumerate}</code>	
<code>\item Segundo.</code>	2. Segundo.
<code>\end{enumerate}</code>	

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

2.5 Figuras

No \LaTeX é possível inserir figuras contidas em um arquivo de imagem ou desenhar uma⁵. Também podemos adicionar uma legenda para a figura.

2.5.1 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`⁶.

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 [parameter=length] {file}
```

³Não será tratado neste curso

⁴Não será tratado neste curso

⁵Ver a Seção 4.1

⁶Este formato requer instalada o \TeX Live 2011 ou superior.

em que `parameter` é um comando disponíveis (algumas opções disponíveis são apresentadas na Tabela 2.1), `length` é uma medida para `parameter` e `file` é o nome do arquivo que contém a imagem.

Tabela 2.1: Opções disponíveis para `parameter`.

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 `length` podemos utilizar medidas correspondente a folha escolhida como por exemplo `\textwidth` ou `\textheight`.

```
\includegraphics[height=2cm]{figures/
  anemonenfisch.png} \\
Imagem de Andreas Preuss / marauder,
disponível 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 Dedication.

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

2.5.2 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{legend}
  \label{P:image}
\end{figure}
```

onde `place` é o parâmetro que indica onde a figura deve ser preferencialmente inserida (as opções disponíveis são apresentadas na Tabela 2.2 e a opção padrão é `tbp`), `imagem` corresponde ao código da figura a ser inserida, `\caption` é o comando correspondente a legenda e `legend` é o texto a ser apresentado como legenda, `\label` é o comando para referência cruzada como já apresentado.

```

\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}

```

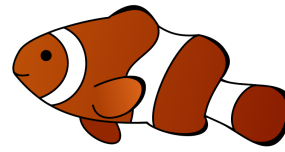


Imagem de Andreas Preuss /
marauder, disponível em <http://openclipart.org/detail/171242/anemonenfisch-by-marauder-171242> e licenciada
sobre CC0 PD Dedication.
O parâmetro H neste ambiente é
necessário para evitar error
relacionado ao ambiente
flutuante.

Figura 2.1: Um peixe.

Tabela 2.2: Opções disponíveis para `place`.

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` que força as figuras pendentes a serem inseridas.

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

2.6 Tabelas

Assim com as figuras, o \LaTeX permite construir tabelas e adicionar legendas à estas.

2.6.1 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 2.3, e `informacao` é o conteúdo de cada célula da tabela.

Tabela 2.3: Opções disponíveis para colunas.

Código	Descrição
<code>l</code>	Alinha com margem esquerda.
<code>r</code>	Alinha com a margem direita.
<code>c</code>	Centralizado.
<code>p</code>	Requer como parâmetro a largura da coluna.
<code> </code>	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|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>.

2.6.2 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, muito semelhante com a do ambiente `figure`, é

```
\begin{table}[place]
tabela
\caption{legend}
\label{P:tebela}
\end{table}
```

onde `place` é o parâmetro que indica onde a tabela deve ser preferencialmente inserida (as opções disponíveis são apresentadas na Tabela 2.5 e a opção padrão é `tbp`), `tabela` corresponde ao código da tabela a ser inserida, `\caption` é o comando correspondente a legenda e `legend` 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 2.4: 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 2.5: Opções disponíveis para `place`.

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.

2.6.3 Extensão Calc2LaTeX

Muitas vezes temos uma tabela no Calc⁷ 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.

2.7 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.

⁷O Calc é um dos aplicativos do pacote Openoffice e corresponde ao popular Excel do pacote Microsoft Office.

Capítulo 3

Matemática no L^AT_EX, amsmath

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

3.1 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*.

3.1.1 *Inline*

Expressões matemáticas *inline* 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>		$1 + 1 = 2$

3.1.2 *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*.

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

3.1.3 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, ...;
- a equação será numerada;
- a expressão que você deseja destacar/enfatizar.

3.2 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 3.2.5).

3.2.1 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.

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

3.2.2 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

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

3.2.3 Acentos

Os acentos disponíveis no modo matemático são apresentados na Tabela 3.1.

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 3.1: 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>	ă	<code>\dot{a}</code>	\dot{a}	<code>\ddot{a}</code>	\ddot{a}
<code>\dddot{a}</code>	ä	<code>\ddddot{a}</code>	ä	<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}

3.2.4 Delimitadores

Parênteses, colchetes e chaves são exemplos de delimitadores. Uma lista completa dos delimitadores disponíveis no LaTeX encontra-se na Tabela 3.2.

Tabela 3.2: Delimitadores disponíveis no LaTeX.

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

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

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

$$\begin{array}{c}
 \text{\texttt{\$}\texttt{\backslash left}\texttt{(}\texttt{\backslash frac}\texttt{\{a\}\{b\}}\texttt{\backslash right)}\texttt{= a}\texttt{\backslash left}} \\
 \text{\texttt{(}\texttt{\backslash frac}\texttt{\{1\}\{b\}}\texttt{\backslash right)}\texttt{\$}}
 \end{array}
 \quad \left| \quad
 \left(\frac{a}{b} \right) = a \left(\frac{1}{b} \right)$$

3.2.5 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.

```
$lanche = salgado + suco, \text{
  errado.}$ \\
$\mathrm{lanche} = \mathrm{salgado} +
  \mathrm{suco}, \text{ correto.}$ \\
```

$lanche = salgado + suco$, errado.
 $lanche = salgado + suco$, correto.

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

```
$a = b, \text{ por hipotese.}$
```

$a = b$, 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 3.3.

Tabela 3.3: 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\quad</code>	$\Rightarrow \Leftarrow$

3.2.6 Matrizes

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

```
$\begin{matrix}
  2 & a+b \\
  a/b & a^2
\end{matrix}$
```

$\begin{matrix} 2 & a+b \\ a/b & a^2 \end{matrix}$

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 $|||$.

3.3 Comandos avançados no modo matemático

3.3.1 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}
```

$$a^2 = b^2 + c^2 \quad (3.1)$$

No exemplo acima, `E:TeoPit` correspondente ao parâmetro do comando `\label`, como apresentado na Seção 2.3. 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
 $.$ \\
 A equacao \eqref{E:TeoPit} e conhecida
 como Teorema de Pitagoras.

Na equacao (3.1) a corresponde a
 hipotenusa de um triangulo e os
 catetos sao b e c .

A equacao (3.1) e conhecida como
 Teorema de Pitagoras.

3.3.2 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:

$$a^2 + b^2 = c^2 \quad (3.2)$$

Com tag:

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

(Teorema de Pitagoras)

(3.2) e (Teorema de Pitagoras)
 sao equivalentes.

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

3.3.3 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.

3.3.4 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 será inserido o símbolo *qed*.

```
\begin{proof}
  $$a^2 + b^2 = c^2$$
\end{proof}
\begin{proof}
  $$a^2 + b^2 = c^2 \text{\texttt{\textbackslash qedhere}}$$
\end{proof}
```

Demonstração.

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

\square

Demonstração.

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

\square

3.3.5 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 multiplas 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}
```

$$a^2 = b^2 + c^2 \quad (3.3)$$

$$a = \sqrt{b^2 + c^2} \quad (3.4)$$

Quando o alinhamento ocorrer adjacente a um sinal de $=$, $+$, \dots 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{\texttt{\textbackslash text}\{Teorema} \\
  & \text{\texttt{\textbackslash de Pit}\{a\}goras} \\
  a &= \sqrt{b^2 + c^2}
\end{align*}
```

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

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

3.3.6 Fórmulas longas

Para fórmulas muito longas que extrapolam a largura da caixa de texto deve-se utilizar o ambiente `multline`, para uma única equação, ou `split`, este último deve ser utilizado dentro de um outro ambiente matemático.

3.3.7 Ocultando termos

Ao trabalhar com fórmulas muito longas tenta-se diminuir o tamanho utilizando seqüências 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`

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

3.3.8 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}`

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

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

3.3.9 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 3.4 apresenta alguns das opções disponíveis.

Tabela 3.4: 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 3.4.

Normal: `a.`

Itálico: `a.`

Negrito: `\mathbf{a}.`

Romano: `a.`

Sans serif: `a.`

Máquina de escrever: `\mathtt{a}.`

Normal: *a*.

Itálico: *a*.

Negrito: **a**.

Romano: a.

Sans serif: a.

Máquina de escrever: a.

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

Normal: `α.`

Negrito: `$\boldsymbol{\alpha}$.`

Normal: α .

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 3.5 apresenta as opções disponíveis.

Tabela 3.5: 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 3.5.

Normal: \mathbb{R} .	Normal: R .
Caligrafico: \mathcal{R} .	Caligrafico: \mathcal{R} .
Euler Fraktur: \mathfrak{R} .	Euler Fraktur: \mathfrak{R} .
Blackboard bold: \mathbb{R} .	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}).

3.4 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 3.6: 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>\nrightarrow</code>	\nrightarrow	<code>\nLeftarrow</code>	\nLeftarrow	<code>\nRightarrow</code>	\nRightarrow
<code>\leftrightharpoonup</code>	\leftrightharpoonup	<code>\longleftrightharpoonup</code>	\longleftrightharpoonup	<code>\Leftrightarrow</code>	\Leftrightarrow
<code>\Longleftrightharpoonup</code>	\Longleftrightharpoonup	<code>\nleftrightharpoonup</code>	\nleftrightharpoonup	<code>\nLeftrightarrow</code>	\nLeftrightarrow
<code>\dashleftarrow</code>	\dashleftarrow	<code>\dashrightarrow</code>	\dashrightarrow	<code>\leftrightharpoons</code>	\leftrightharpoons
<code>\rightleftharpoons</code>	\rightleftharpoons	<code>\leftrightharpoons</code>	\leftrightharpoons	<code>\rightleftharpoons</code>	\rightleftharpoons
<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

3.4.1 Raiz quadrada

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

$\sqrt{4} = 2$	$\sqrt{4} = 2$
$\sqrt[3]{8} = 2$	$\sqrt[3]{8} = 2$

Tabela 3.7: Relações binárias

Com.	Res.	Com.	Res.	Com.	Res.
$<$	$<$	<code>\nless</code>	\nless	$>$	$>$
<code>\ngtr</code>	\ngtr	<code>\ll</code>	\ll	<code>\lll</code>	\lll
<code>\gg</code>	\gg	<code>\ggg</code>	\ggg	$=$	$=$
<code>\neq</code>	\neq	$:$	$:$	<code>\doteq</code>	\doteq
<code>\sim</code>	\sim	<code>\nsim</code>	\nsim	<code>\cong</code>	\cong
<code>\ncong</code>	\ncong	<code>\simeq</code>	\simeq	<code>\approx</code>	\approx
<code>\equiv</code>	\equiv	<code>\leq</code> ou <code>\le</code>	\leq	<code>\nleq</code>	\nleq
<code>\geq</code> ou <code>\ge</code>	\geq	<code>\ngeq</code>	\ngeq	<code>\leqslant</code>	\leqslant
<code>\nleqslant</code>	\nleqslant	<code>\geqslant</code>	\geqslant	<code>\ngeqslant</code>	\ngeqslant
<code>\eqslantless</code>	\eqslantless	<code>\eqslantgtr</code>	\eqslantgtr	<code>\leqq</code>	\leqq
<code>\nleqq</code>	\nleqq	<code>\geqq</code>	\geqq	<code>\ngeqq</code>	\ngeqq
<code>\lesssim</code>	\lesssim	<code>\lessapprox</code>	\lessapprox	<code>\gtrsim</code>	\gtrsim
<code>\gtrapprox</code>	\gtrapprox	<code>\prec</code>	\prec	<code>\nprec</code>	\nprec
<code>\succ</code>	\succ	<code>\nsucc</code>	\nsucc	<code>\preceq</code>	\preceq
<code>\npreceq</code>	\npreceq	<code>\succeq</code>	\succeq	<code>\nsucceq</code>	\nsucceq
<code>\in</code>	\in	<code>\notin</code>	\notin	<code>\owns</code>	\owns
<code>\subset</code>	\subset	<code>\supset</code>	\supset	<code>\subseteq</code>	\subseteq
<code>\nsubseteq</code>	\nsubseteq	<code>\supseteq</code>	\supseteq	<code>\nsupseteq</code>	\nsupseteq
<code>\subseteqq</code>	\subseteqq	<code>\nsubseteqq</code>	\nsubseteqq	<code>\supseteqq</code>	\supseteqq
<code>\nsupseteqq</code>	\nsupseteqq	<code>\sqsubset</code>	\sqsubset	<code>\sqsubseteq</code>	\sqsubseteq
<code>\sqsupset</code>	\sqsupset	<code>\sqsupseteq</code>	\sqsupseteq	<code>\smile</code>	\smile
<code>\smallsmile</code>	\smallsmile	<code>\frown</code>	\frown	<code>\smallfrown</code>	\smallfrown
<code>\perp</code>	\perp	<code>\models</code>	\models	<code>\mid</code>	\mid
<code>\nmid</code>	\nmid	<code>\parallel</code>	\parallel	<code>\nparallel</code>	\nparallel
<code>\shortmid</code>	\shortmid	<code>\nshortmid</code>	\nshortmid	<code>\shortparallel</code>	\shortparallel
<code>\nshortparallel</code>	\nshortparallel	<code>\vdash</code>	\vdash	<code>\nvDash</code>	\nvDash
<code>\dashv</code>	\dashv	<code>\vDash</code>	\vDash	<code>\propto</code>	\propto
<code>\Vdash</code>	\Vdash	<code>\bowtie</code>	\bowtie	<code>\Join</code>	\Join
<code>\asymp</code>	\asymp	<code>\vartriangleleft</code>	\vartriangleleft	<code>\vartriangleright</code>	\vartriangleright
<code>\vartriangleleft</code>	\vartriangleleft	<code>\trianglelefteq</code>	\trianglelefteq	<code>\ntrianglelefteq</code>	\ntrianglelefteq
<code>\ntrianglerighteq</code>	\ntrianglerighteq	<code>\ntrianglerighteq</code>	\ntrianglerighteq	<code>\blacktriangleleft</code>	\blacktriangleleft
<code>\blacktriangleright</code>	\blacktriangleright	<code>\between</code>	\between	<code>\pitchfork</code>	\pitchfork
<code>\therefore</code>	\therefore	<code>\because</code>	\because		

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

Tabela 3.8: Operadores binários

Com.	Res.	Com.	Res.	Com.	Res.
$+$	$+$	$-$	$-$	\pm	\pm
\mp	\mp	\times	\times	\cdot	\cdot
\div	\div	$\&$	$\&$	\setminus	\setminus
\smallsetminus	\smallsetminus	\dagger	\dagger	\ddagger	\ddagger
\ast	\ast	\star	\star	\wedge	\wedge
\vee	\vee	\cap	\cap	\cup	\cup
\sqcap	\sqcap	\sqcup	\sqcup	\oplus	\oplus
\ominus	\ominus	\otimes	\otimes	\oslash	\oslash
\odot	\odot	\bigcirc	\bigcirc	\circ	\circ
\bullet	\bullet	\bigtriangleup	\bigtriangleup	\bigtriangledown	\bigtriangledown
\triangleleft	\triangleleft	\triangleright	\triangleright	\diamond	\diamond
\wr	\wr	\amalg	\amalg		

Tabela 3.9: 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 3.10: Operadores com intervalos.

Com.	Res.	Com.	Res.	Com.	Res.
\int	\int	\iint	\iint	\iiint	\iiint
\iiint	\iiint	$\int \cdots \int$	$\int \cdots \int$	\oint	\oint
\prod	\prod	\coprod	\coprod	\bigcap	\bigcap
\bigcup	\bigcup	\bigwedge	\bigwedge	\bigvee	\bigvee
\bigsqcup	\bigsqcup	\biguplus	\biguplus	\bigotimes	\bigotimes
\bigoplus	\bigoplus	\bigodot	\bigodot	\sum	\sum

Tabela 3.11: Operadores similares ao limites.

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

Tabela 3.12: 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>	\varnothing	<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>	\bot
<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>	\Box	<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 3.13: 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>	F
<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 3.14: 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		

3.4.2 Binomial

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

$$\$a \equiv b \pmod{v} \quad \left| \quad a \equiv b \pmod{v}\right.$$

3.4.3 Congruências

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

$$\$a \equiv b \pmod{v} \quad \left| \quad a \equiv b \pmod{v}\right.$$

Capítulo 4

Apresentações e desenhos utilizando o L^AT_EX

Neste capítulo abordaremos brevemente o pacote `tikz` utilizado para desenhar e a classe `beamer` utilizada para produzir apresentações. Este pacote e essa classe são bastante complexas de modo que abordaremos apenas uma minúscula parcela destes e para maiores informações, recomenda-se os respectivos manuais.

4.1 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.

4.1.1 Ambiente `tikzpicture`

Ao utilizar o TikZ para desenhar uma figura você precisa informar ao L^AT_EX 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 L^AT_EX 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}
```



4.1.2 Sistema de coordenadas

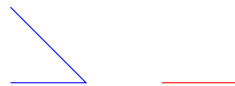
A construção de qualquer figura usando o TikZ requer que seja informado coordenadas de acordo com algum sistema. O TikZ aceita o sistema de coordenadas cartesianas, que corresponde a forma (x, y) , onde x corresponde a coordenada horizontal e y a vertical, e o sistema de coordenadas polares, que corresponde 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}
```



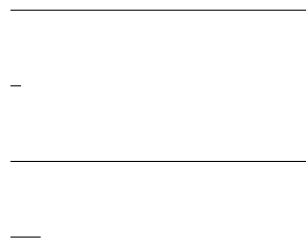
Além de coordenadas absolutas, o TikZ também aceita coordenadas relativas. Coordenadas relativas devem ser precedidas por $+$, que significa “adicionar as seguintes coordenadas à coordenada absoluta previamente informada”, ou $++$, que significa “adicionar as seguintes coordenadas à coordenada absoluta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

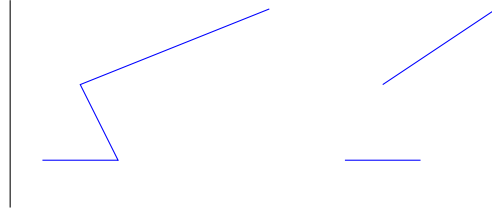
4.1.3 Linhas

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 desse 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}
```



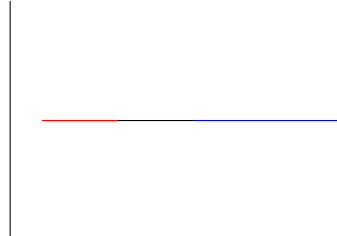
Além da opção `color` que corresponde a cor da linha e do operador `--` que corresponde a uma linha entre dois pontos existem muitos outros. A seguir apresentamos algumas opções e depois alguns operadores.

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.

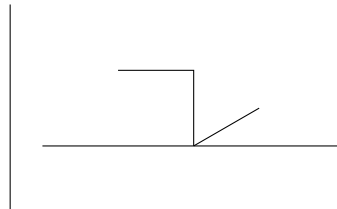
```
\begin{tikzpicture}
  \draw[color=blue, scale=2] (0,0)
    -- (2,0);
  \draw (0,0) -- (2,0);
  \draw[color=red, scale=0.5] (0,0)
    -- (2,0);
\end{tikzpicture}
```



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}
```



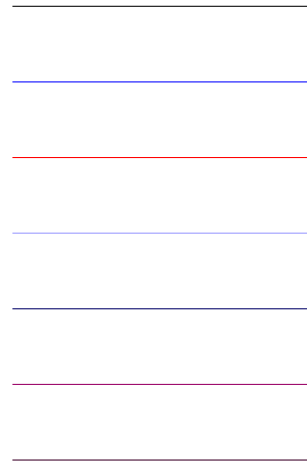
Como podemos observar pelo exemplo acima, o ponto fixo da rotação corresponde ao primeiro ponto do comando.

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>).

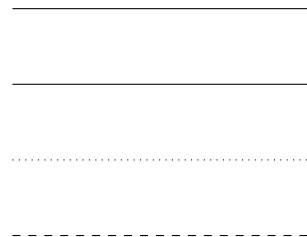
```
\begin{tikzpicture}
  \draw (0,0) -- (4,0);
  \draw[color=blue] (0,-1) -- (4,-1)
  ;
  \draw[color=red] (0,-2) -- (4,-2);
  \draw[color=blue!40] (0,-3) --
    (4,-3);
  \draw[color=blue!40!black] (0,-4)
    -- (4,-4);
  \draw[color=blue!40!red] (0,-5) --
    (4,-5);
  \draw[color=blue!40!red!40!black]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



Padrão

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

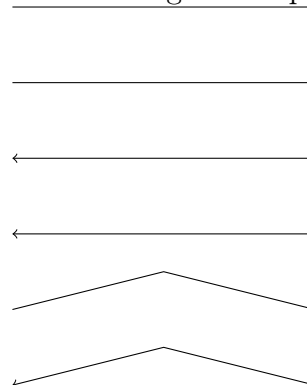
```
\begin{tikzpicture}
  \draw (0,0) -- (4,0);
  \draw[solid] (0,-1) -- (4,-1);
  \draw[dotted] (0,-2) -- (4,-2);
  \draw[dashed] (0,-3) -- (4,-3);
\end{tikzpicture}
```



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}
```

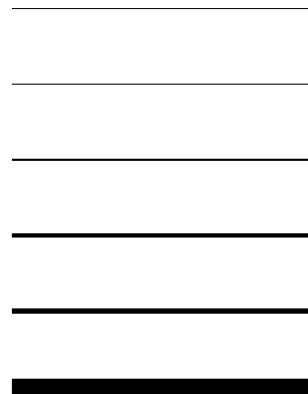


Espessura

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}
```

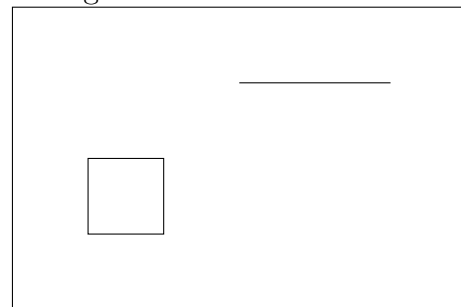


4.1.4 Operadores

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}
```

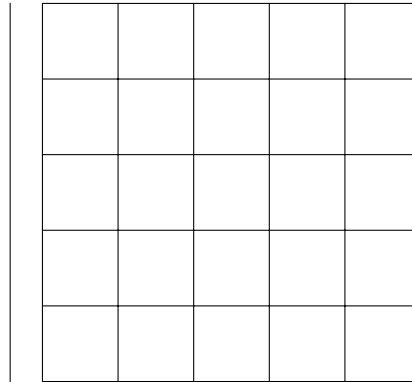


No exemplo acima observamos a ocorrência de um retângulo degenerado em uma linha.

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.

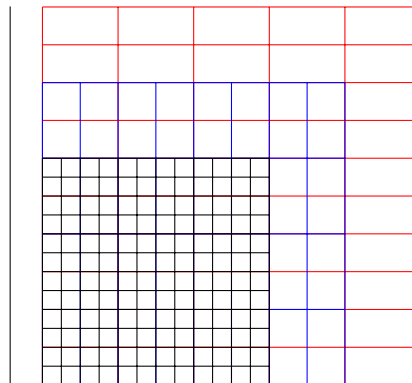
```
\begin{tikzpicture}
  \draw (0,0) grid (5,5);
\end{tikzpicture}
```



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

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

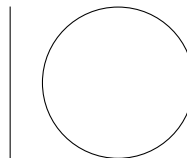
```
\begin{tikzpicture}
  \draw[color=red, ystep=0.5] (0,0)
    grid (5,5);
  \draw[color=blue, xstep=0.5] (0,0)
    grid (4,4);
  \draw[step=0.25] (0,0) grid (3,3);
\end{tikzpicture}
```



Circunferências

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

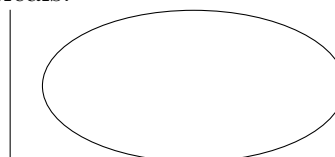
```
\begin{tikzpicture}
  \draw (0,0) circle (1);
\end{tikzpicture}
```



Ellipse

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

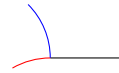
```
\begin{tikzpicture}
  \draw (0,0) ellipse (2 and 1);
\end{tikzpicture}
```



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}
```



4.1.5 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}
```

abc
(0,0)

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}
```

— Linha 1
A — B

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}
```

— Linha 1
A — B

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

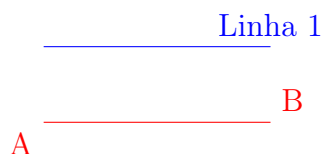
Ancoras

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}
```

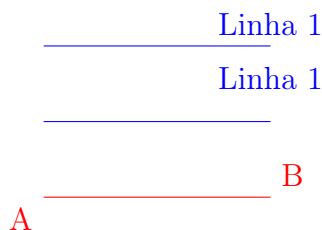


Como o uso de âncoras costuma ser pouco intuitivo existem algumas opções que são equivalentes:

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=east`.

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.

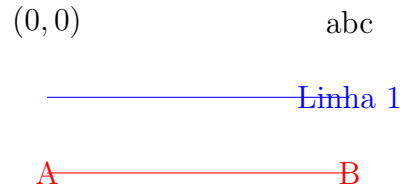
```
\begin{tikzpicture}
  \draw[color=blue] (0,0) -- (3,0)
    node [above] {Linha 1};
  \draw[color=blue] (0,-1) -- (3,-1)
    node [above=8] {Linha 1};
  \draw[color=red] (0,-2) node [
    below left] {A} -- (3,-2) node
    [above right] {B};
\end{tikzpicture}
```



Nomeação

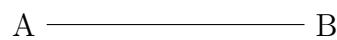
Os nós possuem uma característica muito útil que é a possibilidade de nomeá-los. Para atribuir um nome a um nó utiliza-se parênteses logo em seguida do comando `\node`.

```
\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.

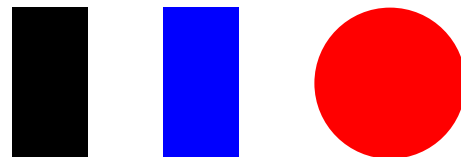
```
\begin{tikzpicture}
  \node[draw] (A) at (0,0) {A};
  \node[draw] (B) at (4,0) {B};
  \draw (A) -- (B);
\end{tikzpicture}
```



4.1.6 Preenchimento

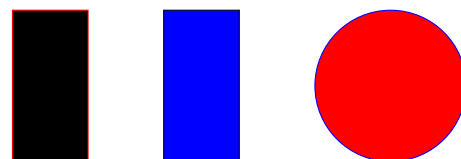
Até o momento apenas contruimos linhas e algumas figuras geométricas. Como devemos proceder para preencher uma figura? Para preencher uma figura utiliza-se a opção `fill`.

```
\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 `\draw`, 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}
```



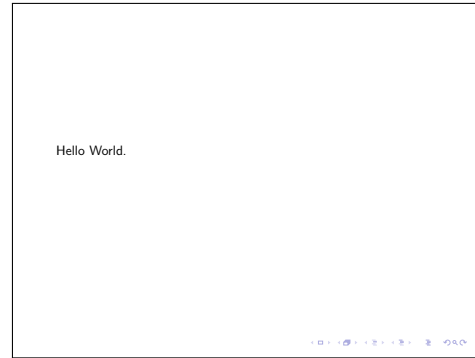
4.2 Classe Beamer

As apresentações criadas com a classe `beamerbeamer` são 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}

```



4.2.1 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}

```



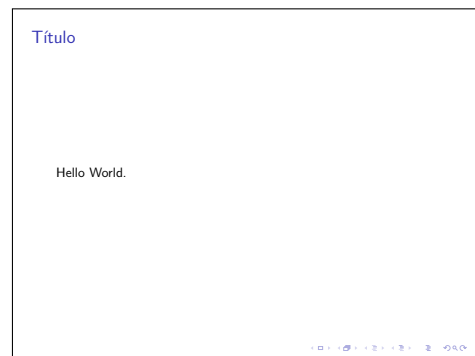
4.2.2 Título do *slide*

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}

```



4.2.3 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 apresentados anteriormente.

```

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

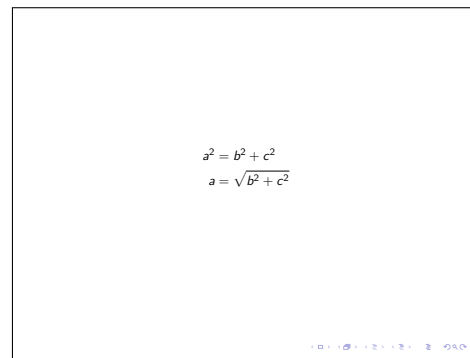
```



```

\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}

```



4.2.4 Overlays

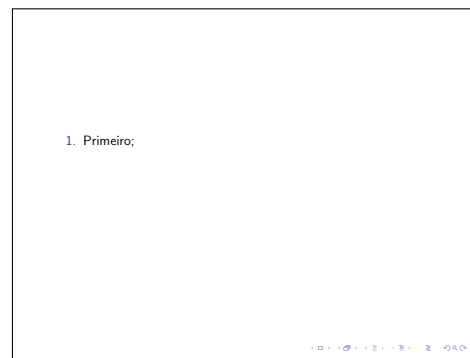
Até o momento todos os *slides* que construimos tinha sua informação apresentada em um único momento. Infelizmente não é isso que deseja-se na grande maioria das apresentações, i.e., deseja-se que fragmentos dos *slides* sejam apresentados em momentos distintos para que seja possível construir a informação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

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

```

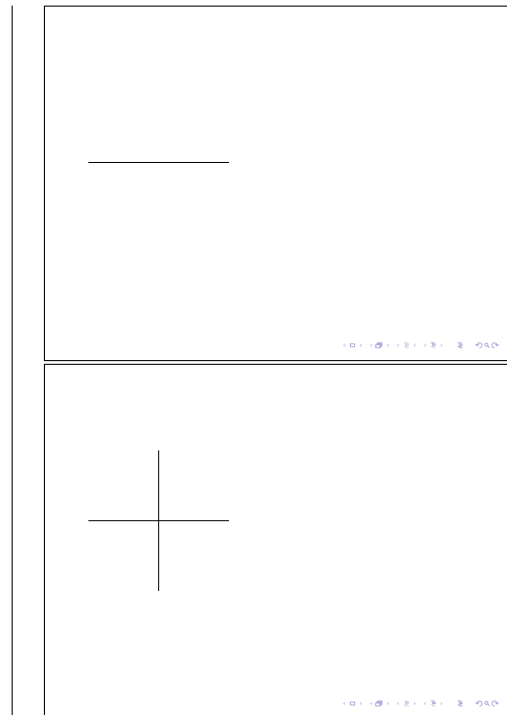
\documentclass{beamer}
\begin{document}
\begin{frame}
  \begin{enumerate}
    \item Primeiro;
    \pause
    \begin{enumerate}
      \item Subitem;
    \end{enumerate}
    \item Segundo.
  \end{enumerate}
\end{frame}
\end{document}

```



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`.

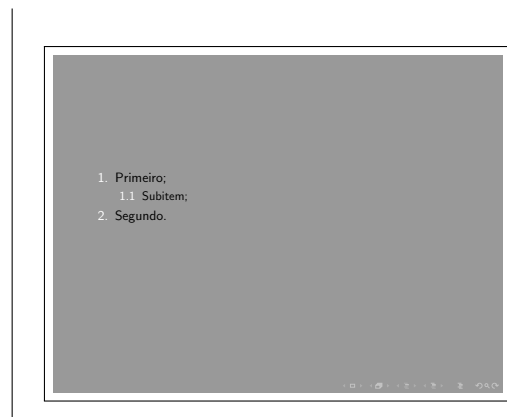
```
\documentclass{beamer}
\usepackage{tikz}
\begin{document}
\begin{frame}
  \begin{tikzpicture}
    \draw (0,0) -- (4,0);
    \pause
    \draw (2,2) -- (2,-2);
  \end{tikzpicture}
\end{frame}
\end{document}
```



4.2.5 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.

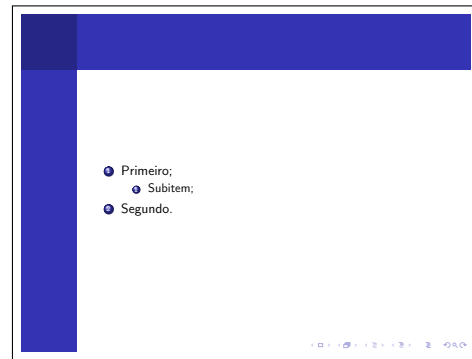
```
\documentclass{beamer}
\usecolortheme{beetle}
\begin{document}
\begin{frame}
  \begin{enumerate}
    \item Primeiro;
    \begin{enumerate}
      \item Subitem;
    \end{enumerate}
    \item Segundo.
  \end{enumerate}
\end{frame}
\end{document}
```



```

\documentclass{beamer}
\usetheme{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 `\usetheme` sugere-se <http://www.hartwork.org/beamer-theme-matrix/>. Outros temas estão disponíveis na internet e alguns deles reunidos em <http://latex.simon04.net/>.

Apêndice A

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 TeX, 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 WYSIWYG¹. Por ser WYSIWYG, utilizar o Word ou algum de seus concorrentes não exigia nenhum conhecimento prévio e isso acabou ofuscando o LaTeX.²

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.³ Desde que o Windows passou a ser o sistema operacional

¹Acrônimo 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”.

²É importante destacar que, tipicamente, os usuários do LaTeX (ou TeX) e do Word (ou concorrentes) possuem necessidades bastante diferentes.

³Nessa época a Apple ainda era uma *start-up* quando comparada a seus concorrentes como, por exemplo, a

dominante⁴ 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.

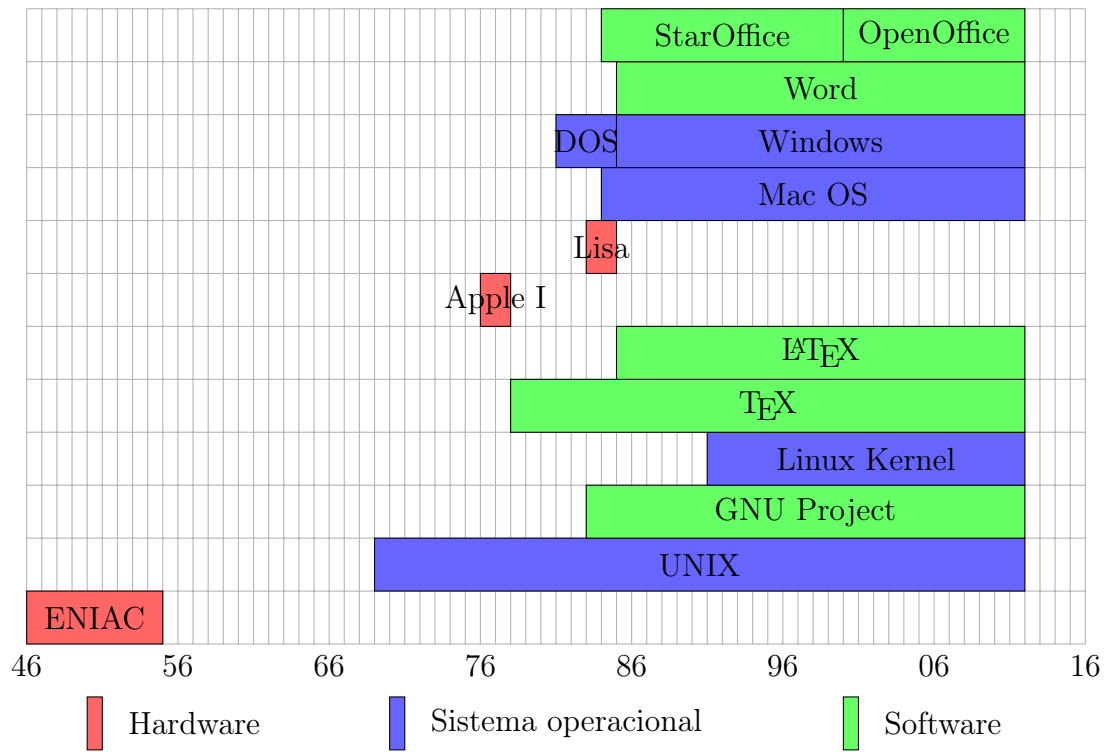


Figura A.1: Linha do tempo de alguns softwares.

IBM e ocorria a *UNIX wars* (ver detalhes em http://en.wikipedia.org/wiki/Unix_wars).

⁴Ao menos no ramo de computadores pessoais.

Apêndice B

Obtendo ajuda

Antes de mais nada é importante saber com que parte do LaTeX você precisa de ajuda pois as palavras com “TeX” são utilizada, muitas vezes, de maneira inadequada. A seguir segue uma explicação das partes do TeX apresentadas em “LaTeX vs. MiKTeX: The levels of TeX”:

Distribuições São grandes coleções de softwares relacionados ao TeX para serem baixados e instalados, e.g., MiKTeX, TeX Live, ...

Front ends São editores utilizados para criar de um documento/arquivo `.tex`, e.g., Emacs, TeXworks, TeXShop, TeXnicCenter, WinEdt, ... Os documentos/arquivos `.tex` são totalmente independentes de qualquer editor.

Engines São executáveis binários que implementam diferentes dialetos TeX, e.g., TeX, pdfTeX, XeTeX, LuaTeX, ...

Formatos São os dialetos TeX utilizados quando cria-se um documento/arquivo `.tex`, e.g., LaTeX, plain TeX, ...

Pacotes São *add-ons* para o sistema TeX básico, desenvolvidos independentemente, que fornecem funcionalidades adicionais, e.g., geometry, lm, ... O site CTAN é um repositório com a vasta maioria dos pacotes existentes.

Para dúvidas gerais recomenda-se o FAQ disponível em <http://www.tex.ac.uk/cgi-bin/texfaq2html> que é mantido pelos usuários TeX do Reino Unido.

Para dúvidas rotineiras ou iniciais uma ótima fonte é o Wikibook em inglês sobre LaTeX disponível em <http://en.wikibooks.org/wiki/LaTeX>. Também existem vários outros manuais disponíveis gratuitamente na internet (ver <http://www.latex-project.org/guides/> e alguns livros publicados sobre o assunto (ver <http://www.tug.org/interest.html>). Destaca-se também a existência de uma enciclopédia dedicada ao TeX (<http://tex.loria.fr/>).

Mesmo o melhor manual sobre LaTeX ainda pode deixar o usuário com algum “problema” a ser resolvido. Nestes casos dois ótimos lugares para procurar uma solução é o “TeX Stack Exchange” (<http://tex.stackexchange.com/>) e o “LaTeX Community” (<http://www.latex-community.org/>). Também é possível perguntar em alguma lista de emails sobre o tema (ver algumas em <http://www.tug.org/mailman/listinfo>).

Por último, quando tratar-se de algum pacote recomenda-se dar uma olhada no manual. Os manuais dos pacotes presentes na sua distribuição são facilmente acessados utilizando o Texdoc (<http://tug.org/texdoc/>), para isso execute no terminal o comando abaixo

```
texdoc <nome_pacote>
```

onde <nome_pacote> é o nome completo ou parcial do pacote desejado.

Apêndice C

Exercícios

Nas páginas a seguir encontram-se alguns exemplos a serem reproduzidos para você testar os comandos e ambientes que foram apresentados neste curso. Como ponto de partida recomendamos utilizar o código abaixo.

```
\documentclass[12pt, a4paper]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[top=3cm, left=2cm, right=2cm, bottom=3cm]{geometry}
\usepackage[brazil]{babel}
\begin{document}

\end{document}
```

A seguir algumas dicas referentes aos exemplos presentes nas próximas páginas que não foram cobertos neste curso.

1. Utilize os comandos `\title`, `\author`, `\date` e `\maketitle` para o título.
2. Utilize o comando `\section` para fazer a divisão do texto e para as referências leia um pouco sobre o BibTeX em http://en.wikibooks.org/wiki/LaTeX/Bibliography_Management.
3. Ver o item 2 e para a definição das funções seno, cosseno e tangente utilize o ambiente `description`.

Exercício 1

Bradesco e Itaú vão reduzir taxas de juros

REDAÇÃO ÉPOCA COM AGÊNCIA BRASIL

18/04/2012 às 14h42

Os bancos Bradesco e Itaú anunciaram nesta quarta-feira (18) que vão reduzir as taxas de juros a seus clientes. Nas últimas semanas, Banco do Brasil, Caixa Econômica Federal, HSBC e Santander também anunciaram reduções nas taxas de juros. As decisões foram feitas depois da presidente Dilma Rousseff defender a redução do “spread”, que é a diferença entre o que os bancos brasileiros pagam para captar o dinheiro do investidor e o que eles cobram dos tomadores de empréstimos.

Na quinta-feira (12), o ministro da Fazenda, Guido Mantega também fez críticas aos bancos privados por não reduzir as taxas e cobrar altos spreads. Hoje, a taxa básica de juro está em 9,75% ao ano, um dos níveis mais baixos da história. Mas esse valor não tem efeito na vida do cidadão comum. Os juros cobrados no cheque especial chegam a 300% ao ano, as taxas de empréstimos para negócios de menor porte estão na faixa de 60% ao ano e o spread brasileiro não tem precedente em nenhum outro lugar do mundo.

No Bradesco, a taxa mínima do crédito pessoal cairá de 2,66% para a partir de 1,97% ao mês. Na linha CDC Bens, a taxa será reduzida de 3,54% para a partir de 2,97% ao mês. No caso do financiamento de veículos, a taxa, que era 1,35%, passará a ser a partir de 0,97% ao mês. Nas operações de crédito consignado para os aposentados, o Bradesco reduziu a taxa de 1,32% para a partir de 0,9% ao mês. Os cartões de crédito emitidos em parceria com redes varejistas terão taxas para parcelamento com juros a partir de 2,49% ao mês, com prazo de até 24 meses. Além da redução das taxas, o Bradesco informou que ampliou o limite de crédito em mais R\$ 15 bilhões, sendo R\$ 9 bilhões para pessoas físicas e R\$ 5 bilhões para pessoas jurídicas.

No Itaú, no caso de financiamento de veículos, a taxa mínima sofrerá redução de 8% e será de 0,99% ao mês. A taxa será válida para clientes correntistas há mais de um ano, em operações com 50% de entrada e parcelamento em até 24 meses. Nos empréstimos consignados para beneficiários do INSS, a taxa mínima foi reduzida para 0,89%, e a máxima, para 2,2% ao mês.

As novas taxas de juros e limites, sujeitos a aprovação de crédito, entram em vigor na segunda-feira (23).

Retirado de <http://revistaepoca.globo.com/Negocios-e-carreira/noticia/2012/04/bradesco-e-ita-u-va-o-reduzir-taxas-de-juros.html>

Exercício 2

O FarmPoint fez uma análise dos dados apresentados pelo Instituto Brasileiro de Geografia e Estatística (IBGE) na última quarta-feira, dia 24, sobre a Pesquisa de Produção da Pecuária Municipal de 2009 (PPM 2009). O efetivo de ovinos em 2009 foi de 16,8 milhões de cabeças, crescimento de 1,1% frente as 16,6 milhões de cabeças de 2008 e o efetivo de caprinos foi de 9,16 milhões de cabeças, queda de 2,04% comparado as 9,35 milhões de cabeças de 2008.

1 Ovinos

O efetivo de ovinos em 2009 foi de 16,8 milhões de cabeças, crescimento de 1,1% frente as 16,6 milhões de cabeças de 2008. Em 2009, a região Nordeste deteve o maior número de cabeças ovinas, totalizando 9,56 milhões de cabeças, crescimento de 2,08% frente a 2008. A região Sul apresentou o segundo maior rebanho, 4,8 milhões de cabeças, queda de 0,81% comparado a 2008. A região Centro-Oeste apresentou o terceiro maior rebanho, 1,12 milhões de cabeças, crescimento de 1,56% frente a 2008, seguido da região Sudeste com 761.952 cabeças (queda de 0,39% frente a 2008) e da região região Norte, 547.903 cabeças, aumento de 2,51%.

2 Regiões

A região Nordeste possui 56,9% do rebanho nacional, seguida da região Sul (28,6%), região Centro-Oeste (6,71%), região Sudeste (4,53%) e região Norte (3,26%).

3 Estados

Em 2009, o Rio Grande do Sul se manteve na liderança e totalizou 3,94 milhões de cabeças, queda de 1,59% frente a 2008. A Bahia manteve o segundo lugar no ranking, com um efetivo de 3,02 milhões de cabeças e crescimento de 0,25% frente a 2008. A terceira posição foi

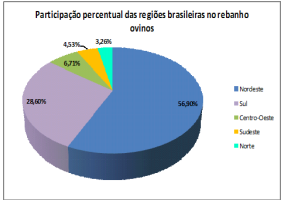


Figura 1: Ovinos

ocupada pelo Ceará, com 2,07 milhões de cabeças, crescimento de 1,98% comparado ao ano anterior. Pernambuco apresentou um crescimento de 10%, totalizando 1,48 milhões de cabeças e ocupando o quarto lugar.

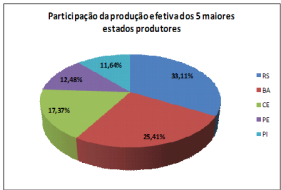


Figura 2: Estados

Tabela 1: Os 5 municípios com maior rebanho de ovinos em 2009

Municípios	# cabeças
Sant’Ana do Livramento - RS	401779
Alegrete - RS	239778
Casa Nova - BA	225832
Quaraí - RS	190744
Uruguaiana - RS	180407

Referências

[1] Raquel Maria Cury Rodrigues. Análise do desenvolvimento do rebanho ovino e caprino no brasil em 2009. <http://www.farmpoint.com.br/cadeia-produtiva/especiais/analise-do-desenvolvimento-do-rebanho-ovino-1.aspx>.

Exercício 3, folha 01 de 02

1 A função polinomial

Um polinômio (função polinomial) com coeficientes reais na variável x é uma função matemática $f : \mathbb{R} \rightarrow \mathbb{R}$ definida por:

$$p(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots + a_nx^n,$$

onde $a_0, a_1, a_2, \dots, a_n$ são números reais, denominados coeficientes do polinômio. O coeficiente a_0 é o termo constante.

Se os coeficientes são números inteiros, o polinômio é denominado polinômio inteiro em x .

O valor numérico de um polinômio $p = p(x)$ em $x = a$ é obtido pela substituição de x pelo número a , para obter $p(a)$.

Exemplo: O valor numérico de $p(x) = 2x^2 + 7x - 12$ para $x = 3$ é dado por:

$$\begin{aligned} p(3) &= 2 \cdot (3)^2 + 7 \cdot 3 - 12 \\ &= 2 \cdot 9 + 21 - 12 \\ &= 18 + 9 \\ &= 27. \end{aligned}$$

2 Sistemas de equações lineares

Um sistema de equações lineares ou sistema linear é um conjunto formado por duas ou mais equações lineares. Um sistema linear pode ser representado na forma:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_n \end{cases} \quad (1)$$

onde x_1, x_2, \dots, x_n são as incógnitas, $a_{11}, a_{12}, \dots, a_{mn}$ são os coeficientes e b_1, b_2, \dots, b_m são os termos independentes.

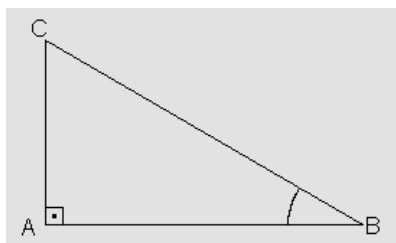
Uma sequência de números (r_1, r_2, \dots, r_n) é solução de (1) se satisfaz identicamente a todas as equações desse sistema linear.

3 Trigonometria no triângulo retângulo

Um triângulo é uma figura geométrica plana, constituída por três lados e três ângulos internos. Esses ângulos, tradicionalmente, são medidos numa unidade de medida, denominada grau e, cada um deles tem medida entre 0° e 180° , de modo que, em qualquer triângulo, a soma dessas medidas é 180° .

Num triângulo retângulo definimos as chamadas razões trigonométricas que são relações entre os lados do triângulo e que têm a propriedade de determinar a medida dos ângulos do triângulo, uma vez que seus lados sejam conhecidos.

Exercício 3, folha 02 de 02



No triângulo retângulo ABC , consideremos, por exemplo, o ângulo que tem vértice em B , cuja medida α , em graus, é um número real que está no intervalo $0, \pi/2$. Entre os lados do triângulo podemos estabelecer as seguintes razões:

seno é a razão entre o comprimento do cateto oposto ao ângulo \hat{B} e o comprimento da hipotenusa do triângulo. Indicando o seno de α por $\sin(\alpha)$, temos $\sin(\alpha) = \overline{AC}/\overline{BC}$.

cosseno é a razão entre o comprimento do cateto adjacente ao ângulo e o comprimento da hipotenusa do triângulo. Indicando o cosseno de α por $\cos(\alpha)$, temos $\cos(\alpha) = \overline{AB}/\overline{BC}$.

tangente é a razão entre os comprimentos do cateto oposto e do cateto adjacente ao ângulo \hat{B} . Indicando a tangente de α por $\tan(\alpha)$, temos $\tan(\alpha) = \overline{AC}/\overline{AB}$.

Referências

- [1] Emanuelle L.Vicente and Ulysses Sodré. Ensino médio: Sistemas lineares. <http://pessoal.sercomtel.com.br/matematica/medio/matrizes/sistemas.htm>.
- [2] Ulysses Sodré. Ensino médio: Polinômios e equações algébricas. <http://pessoal.sercomtel.com.br/matematica/medio/polinom/polinom.htm>.
- [3] USP. Trigonometria no triângulo retângulo. http://ecalculo.if.usp.br/funcoes/trigonometricas/rz_trigo_triret.htm/rz_trigo_triret.htm.

Referências Bibliográficas

- [Bra08] Johannes Braams. *Babel, a multilingual package for use with LaTeX's standard document class*, jul 2008. <http://www.ctan.org/pub/tex-archive/macros/latex/required/babel/babel.pdf>.
- [Grä07] George Grätzer. *More Math into LaTeX*. Springer, 4 edition, 2007.
- [Hil12] A.J. Hildebrand. Tex resources, 2012. [Online; accessed 22-July-2012].
- [Lam94] L. Lamport. *LATEX: a document preparation system : user's guide and reference manual*. Number p. 2. Addison-Wesley Pub. Co., 1994.
- [LaT12] LaTeX Project. Latex – a document preparation system, 2012. [Online; accessed 22-July-2012].
- [Pak09] Scott Pakin. *The Comprehensive LaTeX Symbol List*, 2009.
- [San09] Reginaldo J. Santos. Introdução ao latex, sep 2009. <http://www.mat.ufmg.br/~regi/topicos/intlat.pdf>.
- [SOS99] E. Swanson, A.A. O'Sean, and A.T. Schleyer. *Mathematics Into Type*. American Mathematical Society, 1999.
- [Sou] Gilberto Souto. Curso de latex. www.ufsm.br/petfisica/extras/arquivos/latex.pdf.
- [Tan10] Till Tantau. *The TikZ and PGF Packages - Manual for version 2.10*, 2010.
- [TeX12] TeX Users Group. Latex vs. miktex: The levels of tex, 2012. [Online; accessed 22-July-2012].
- [TWM10] Till Tantau, Joseph Wright, and Vedran Miletie. *The Beamer class - User Guide for version 3.12*, 2010.
- [Ume10] Hideo Umeke. *The geometry package*, sep 2010. <ftp://ftp.tex.ac.uk/tex-archive/macros/latex/contrib/geometry/geometry.pdf>.
- [Wik12a] Wikipedia. Comparison of tex editors — Wikipedia, the free encyclopedia, 2012. [Online; accessed 22-July-2012].
- [Wik12b] Wikipedia. Eniac — Wikipedia, the free encyclopedia, 2012. [Online; accessed 22-July-2012].

- [Wik12c] Wikipedia. Latex — Wikipedia, the free encyclopedia, 2012. [Online; accessed 14-July-2012].
- [Wik12d] Wikipedia. Tex — Wikipedia, the free encyclopedia, 2012. [Online; accessed 14-July-2012].
- [Wik12e] Wikipedia. Unix — Wikipedia, the free encyclopedia, 2012. [Online; accessed 22-July-2012].

Índice Remissivo

- .tex, 1
- alinhamento, 12
- ambiente
 - align, 24
 - enumerate, 14
 - equation, 22
 - figure, 15, 31
 - frame, 40
 - itemize, 14
 - quotation, 18
 - quote, 18
 - table, 17
 - tabular, 16
 - tikzpicture, 31
 - verbatim, 10
- aspas, 6
- beamer
 - overlay, 42
 - tema, 43
- comando
 - %, 7
 - &, 17, 24
 - \\, 6, 17, 22, 24
 - \caption, 15, 17
 - \clearpage, 16, 18
 - \color, 10
 - \documentclass, 3
 - class, 3
 - \draw, 33
 - \footnote, 13
 - \hspace, 11
 - \hyphenation, 6
 - \includegraphics, 14
 - \input, 2
 - \item, 14
 - \label, 13, 22
 - \newline, 6
 - \newpage, 6
 - \pause, 42
 - \ref, 13
 - \tabularnewline, 17
 - \tag, 23
 - \text, 22
 - \textcolor, 10
 - \url, 13
 - \usepackage, 3
 - \verb, 10, 13
 - \vspace, 11
- comentários, 7
- espaços em branco, 11
- figura, 14
- fonte, 9
 - cor, 10
 - tamanho, 10
- hifenização, 6
- IDE, 1
- informação, 2, 4
- instalação, 1
- Kile, *veja* IDE
- lista, 14
- Mac OS X, *veja* instalação
- margens, 8
- MikTeX, *veja* instalação
- modo matemático
 - acento, 20
 - binômio, 29
 - chaves, *veja* delimitadores
 - colchetes, *veja* delimitadores
 - congruência, 29
 - delimitadores, 21
 - demonstração, 23

- displayed*, 19
- espaçamento, 22
- expoente, 20
- fórmulas longas, 24
- funções definidas por partes, 25
- índice, 20
- inline*, 19
- matrizes, 22
- múltiplas equações, 24
- nomes longos para variáveis, 21
- novos operadores, 21
- numeração, 22
- operações aritméticas básicas, 20
- parênteses, *veja* delimitadores
- raiz quadrada, 26
- sistemas de equações, 25
- tag*, 23
- teorema, 23
- texto, 21
- vetores, *veja* matrizes
- múltiplos arquivos, 2
- nota de rodapé, 13
- nova linha, 5
- pacote
 - amsmath*, 19
 - babel*, 5
 - color*, 10
 - fontenc*, 5
 - geometry*, 8
 - graphicx*, 10, 14
 - inputenc*, 5
 - tikz*, 31
 - url*, 13
- parágrafo, 6
- pontuação, 7
- preâmbulo*, 1–3
- proTeXt, *veja* instalação
- referência cruzada, 13
- tabela, 16
- TeX Live, *veja* instalação
- Texmaker, *veja* IDE
- TeXworks, *veja* IDE
- TikZ
 - âncora, 38
 - arco, 37
 - circunferência, 36
 - coordenadas relativas, 32
 - cor, 33
 - elipse, 36
 - escala, 33
 - espessura, 35
 - nó, 37
 - preenchimento, 39
 - retângulo, 35
 - rotação, 33
 - seta, 34
 - sistema de coordenadas cartesianas, 32
 - sistema de coordenadas polares, 32
 - texto, *veja* nó
- traço, 7