

Aula Exercício de Introdução ao L^AT_EX

Thiago de Gouveia Nunes

7 de outubro de 2011

1 Introdução ao L^AT_EX

LaTeX é um programa de formatação de texto e uma expansão do programa TeX, criado por Donald Knuth. Mas o que é um programa de formatação de texto?

A maioria dos processadores de texto cuidam de 4 estágios para preparar um texto: 1. O texto entra no computador. 2. O texto é formatado em linhas, parágrafos e páginas. 3. O texto é impresso na tela. 4. O texto é impresso.

O LaTeX e o TeX se preocupam somente com o segundo estágio do processamento. Para formatar um texto usando o LaTeX, escrevemos o texto em um editor de texto e compilamos. A partir daí, o LaTeX cuidará da formatação do texto.

1.1 Tipos de Documentos

Os tipos mais comuns de documentos que o LaTeX aceita são

1. Book
2. Article
3. Letter
4. Report

1.2 Fontes

Podemos mudar o tamanho e o estilo das fontes usando os seguintes comandos:

- `\textrm{...}` roman
- `\textsf{...}` sans serif
- `\texttt{...}` typewriter
- `\textmd{...}` medium
- `\textbf{...}` **boldface**
- `\textup{...}` upright
- `\textit{...}` *italic*
- `\textsl{...}` *slanted*
- `\textsc{...}` SMALL CAP

2 Primeiro arquivo

Agora, vamos criar um arquivo simples no \LaTeX . Copie as linhas abaixo no seu editor de texto favorito e rode o compilador de \LaTeX para gerar o pdf.

```
\documentclass{article}
\begin{document}
Hello, world!
\end{document}
```

Os comandos nesse arquivo são:

- `\documentclass{.}` é o comando que define qual o tipo de documento estamos escrevendo.
- `\begin{...}` e `\end{...}` são os comandos que definem quando algo começa e termina. No nosso caso, eles definem quando o documento começa e termina.

Agora, vamos acrescentar novos elementos nesse documento, e transformá-lo em um documento real.

3 Estrutura de um Documento

3.1 Modificações para o Documento

Vamos estudar mais detalhadamente a estrutura de um documento no \LaTeX . Já vimos acima o `\documentclass[...]{...}`. Ele aceita várias opções para modificar o documento. Essas opções, que colocadas entre os colchetes, são:

- `\documentclass[10pt]{article}` Assim, o tamanho das letras do documento é setado para 10pt.
- `documentclass[letterpaper]{article}` Esse comando faz com que o \LaTeX molde o seu texto para ser impresso em papel de carta.
- `documentclass[twocolumn]{article}` Esse aqui faz com que o texto seja dividido em duas colunas (como nos dicionários).
- `documentclass[oneside]{article}` Quando o \LaTeX montar o texto para impressão, ele vai configurá-lo para imprimir somente nas páginas ímpares.

3.2 Modificações para Páginas

Podemos modificar todas as páginas de um documento, ou alguma página específica, usando os comandos `\pagestyle{...}` e o `\thispagestyle{...}`, respectivamente.

Os argumentos que eles aceitam são:

- `plain` - O cabeçalho fica vazio e o rodapé contém o número da página. Esse é o padrão para o `article`,
- `empty` - Tanto o cabeçalho como o rodapé ficam vazios,
- `headings` - O rodapé fica vazio e o cabeçalho contém o número da página e o nome do capítulo ou seção ou subseção,
- `myheadings` - Igual ao `headings`, mas o que é mostrando no lugar do nome do capítulo pode ser configurado.

Para modificar o modo com que as páginas são numeradas, usamos o comando `\pagenumbering{...}`. Seus argumentos são:

- arabic - Numerais arábicos
- roman - Numerais romanos em minúsculo,
- Roman - Numerais romanos em maiúsculo,
- alph - Numerais por extenso em inglês em minúsculo,
- Alph - Numerais por extenso em inglês em maiúsculo.

3.3 Atualizar Exemplo

Esse são só alguns exemplos do que podemos mudar nos nossos documentos. Agora, vamos ver como fica o nosso documento de teste se colocarmos as opções *twocolumn* e *a4paper* para o documento, opções *headings* e *Roman* para as páginas. E vamos incrementá-lo com dois parágrafos de texto.

Para criar novos parágrafos no L^AT_EX só precisamos deixar uma linha totalmente em branco entre os dois parágrafos.

3.4 Título

O L^AT_EX também cuida da criação do título do documento. Para isso usamos os seguintes comandos

```
\title{título}
\author{autor}
\date{data}
\maketitle
```

Note que é preciso usar o comando `\maketitle` ou o título não será gerado.

3.5 Resumo

Nos documentos do tipo *article* e *report* podemos gerar um resumo usando os comandos `\begin{abstract}` e `\end{abstract}`. Tudo o que estiver escrito dentro dessas tags vai virar o resumo do documento, e será posicionado entre o título e o resto do documento.

3.6 Divisão do Documento

Quando escrevemos textos grandes (como esse), é bom criar seções e subções para deixar tudo organizado e para facilitar a busca de informações. O \LaTeX tem uma hierarquia de comandos para criar divisões:

1. `\chapter`
2. `\section`
3. `\subsection`
4. `\subsubsection`
5. `\paragraph`
6. `\subparagraph`

O `\chapter` só pode ser usado no tipo book.

3.7 Nova Atualização

Agora vamos colocar mais informação no nosso documento. Vamos criar duas seções com duas subseções cada. Vamos criar um resumo também.

4 Fórmulas Matemáticas

O \TeX foi criado pelo Donal Kunth principalmente para processar fórmulas matemáticas. Para colocar expressões matemáticas em linha usamos o símbolo \$.

Para produzir

A raiz de $ax + b = 0$ é $-b/a$.

Devemos escrever no arquivo-fonte

A raiz de $ax+b=0$ é $-b/a$.

Esse método é antigo, e é usado pelo \TeX . O \LaTeX usa dois padrões para reconhecer expressões

`\(...\)`

ou

`\begin{math} ... \end{math}`

Então, para gerar a mesma linha acima, podemos usarmos

A raiz de $(ax+b=0)$ é $(-b/a)$.

4.1 Expoentes e Índices

Vamos usar exemplos para entender como criar expoentes e índices. Para produzir expoentes, usamos o símbolo \wedge .

Para produzir esse trecho

$$x^n + y^n = z^n$$

devemos escrever $x^n + y^n = z^n$.

O trecho,

$$(x^m)^n = x^{mn}$$

é gerado pelo código $(x^m)^n = x^{mn}$.

Podemos fazer potência de potências, por exemplo,

$$2^{2^n} + 1$$

é produzido pela linha $2^{2^n}+1$.

Agora vamos falar de índices. Para gerar um índice, usamos o símbolo $_$.

O trecho abaixo

A sequência (x_n) definida por

$$x_1 = 1, \quad x_2 = 1, \quad x_n = x_{n-1} + x_{n-2} \quad (n > 2)$$

é chamada de Fibonacci.

é feita digitando a linha

A sequência (x_n) definida por

$\$$

$x_1=1,\quad x_2=1,\quad x_n=x_{n-1}+x_{n-2}\backslash;\backslash;(n>2)$

$\$$

é chamada de Fibonacci.

.

Podemos juntar potências e índices usando a seguinte linha de código (x_n^2) , com resultado, (x_n^2) .

4.2 Raizes

Raizes são geradas usando o comando $\sqrt[n]{x}$. $\sqrt{2}$ gera $\sqrt{2}$. Para fazer raízes com raízes não quadradas usamos, por exemplo, $\sqrt[4]{2}$, que gera $\sqrt[4]{2}$. Podemos colocar qualquer fórmula matemática dentro das chaves que o \LaTeX vai arrumar a raiz para englobar a fórmula.

5 Mais Matemática

Essa seção vai explorar o pacote *amsmath*. Para incluí-lo no L^AT_EX, é só colocar a linha `\usepackage{amsmath}` abaixo da linha `\documentclass{...}` no início do arquivo fonte.

5.1 Matriz

Podemos fazer um sistema de equações usando o comando `\begin{align*}`. Um exemplo:

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

É gerado usando

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

Para escrever esse sistema como matriz, usamos esses comandos:

```
\begin{equation*}
  \begin{pmatrix}
    1 & 1 & -1 \\
    1 & -1 & 1 \\
    1 & 1 & 1
  \end{pmatrix}
  \begin{pmatrix}
    x \\
    y \\
    z
  \end{pmatrix}
\end{equation*}
```

```

=
\begin{pmatrix}
1\\
1\\
1
\end{pmatrix}.
\end{equation*}

```

Com esse resultado:

$$\begin{pmatrix} 1 & 1 & -1 \\ 1 & -1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.$$

Podemos mudar as matrizes de parênteses para chaves usando o `\begin{bmatrix}` e para barras usando o `\begin{vmatrix}`.

Exemplos:

$$\begin{bmatrix} 1 & 1 & -1 \\ 1 & -1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{vmatrix} 1 & 1 & -1 \\ 1 & -1 & 1 \\ 1 & 1 & 1 \end{vmatrix}$$

Para gerar uma matriz genérica $n \times m$, usamos:

```

\begin{equation*}
\begin{bmatrix}
a_{11} & a_{12} & \dots & a_{1n} \\
a_{21} & a_{22} & \dots & a_{2n} \\
\hdotsfor{4} \\
a_{m1} & a_{m2} & \dots & a_{mn}
\end{bmatrix}
\end{equation*}

```

Gerando,

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$$

5.2 Frações

O L^AT_EX usa o comando `\frac{X}{Y}` para inserir uma fração no texto. O exemplo seria

$$\frac{x}{y}$$

X e Y podem ser qualquer expressão matemática. Por exemplo:

$$\frac{x^2 + x + 1}{\frac{y^2}{2}}$$

Se usarmos o comando `\tfrac{X}{Y}`, geramos uma fração menos que a normal.

$$\frac{1}{2} > \frac{1}{2}$$

6 Expressões de Exemplo

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

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

$$\sum_{i=1}^3 i$$

```
\begin{equation*}
\sum_{i = 1}^3 i
\end{equation*}
```

$$\binom{n}{1}$$

```
\begin{equation*}
\binom{n}{1}
\end{equation*}
```

7 Comandos Customizados

O \LaTeX permite a programação de novos comandos. Isso é muito útil, quando, por exemplo, estamos escrevendo uma aula de álgebra linear e precisamos sempre escrever `(x_1,x_2,\dots,x_n)` para gerar (x_1, x_2, \dots, x_n) , podemos usar o comando

```
\newcommand{\nome_do_comando}{comando_a_ser_executado}
```

No exemplo anterior, podemos fazer o comando

```
\newcommand{\vect}{(x_1,x_2,\dots,x_n)}
```

e agora, toda vez que digitarmos `\vect` obteremos (x_1, x_2, \dots, x_n)

Podemos, ainda, modificar o novo comando para aceitar parâmetros, assim não estamos limitados a vetores com coordenadas X . O comando ainda é o mesmo, mas precisamos modificá-lo um pouco.

```
\newcommand{\vect}[N]{(#1_1,#1_2,\dots,#1_#i)}
```

Sendo N o número de argumentos que esse comando vai receber e $\#i$ recebe o i -ésimo argumento.

Logo, o comando

```
\newcommand{\vect}[3]{(#1_#2,#1_2,\dots,#1_#3)}
```

faz um vetor de variável $\#1$, da posição $\#2$ até $\#3$. Exemplo, (x_1, x_2, \dots, x_n)