

Uma Introdução Interativa do \LaTeX

Parte 3: Não apenas Artigos: Apresentações & Mais

Dr John D. Lees-Miller

Tradução: Dr Luiz-Rafael Santos

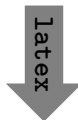
15 de agosto de 2019



Revisão de L^AT_EX

- ▶ Você escreve o documento em texto puro com **comandos** que descrevem sua estrutura ou significado
- ▶ O programa latex processa seu texto e os comandos para produzir um documento esteticamente bem formatado.

A chuva na Amazônia `\emph{cai}` na horizontal.



A chuva na Amazônia *cai* na horizontal.

Revisão de L^AT_EX: Comandos & Argumentos

- ▶ Um comando começa com uma *barra invertida* `\`.
- ▶ Alguns comandos podem receber um *argumento* entre chaves `{ }`.
- ▶ Alguns comandos também recebem *argumentos opcionais* entre colchetes `[]`.

```
\includegraphics[  
  width=0.5\textwidth]{gerbil}
```

```
\includegraphics[width=0.3\textwidth,  
  angle=270]{gerbil}
```



Revisão de \LaTeX : Ambientes

- ▶ Os comandos `\begin` e `\end` são usados para criar diferentes tipos de ambientes — ou seja contextos.
- ▶ Os ambientes `itemize` e `enumerate` fazem listas.

```
\begin{itemize} % para triângulos
```

```
\item Café
```

```
\item Chá
```

```
\end{itemize}
```

▶ Café

▶ Chá

```
\begin{enumerate} % para números
```

```
\item Café
```

```
\item Chá
```

```
\end{enumerate}
```

1. Café

2. Chá

Revisão de L^AT_EX: Textos Matemáticos

- ▶ O ambiente `equation` produz equações numeradas.

```
\begin{equation}
  \sum_{k=1}^n \frac{1}{2^k}
\end{equation}
```

$$\sum_{k=1}^n \frac{1}{2^k} \quad (1)$$

- ▶ Use cifrão `$` para escrever matemática em um texto corrido.

```
% não tão bom:
Seja a e b inteiros positivos
distintos, e seja
c = a - b + 1.

% muito melhor:
Seja  $a$  e  $b$  inteiros positivos
distintos, e seja
 $c = a - b + 1$ .
```

Seja a e b inteiros positivos distintos, e seja $c = a - b + 1$.

Seja a e b inteiros positivos distintos, e seja $c = a - b + 1$.

- ▶ Sempre utilize cifrão em pares — um para o começo do texto matemático e outro para o final.

De fato, poderíamos ter escrito $...$ como `\begin{math}...\end{math}`.

Revisão de \LaTeX : Estrutura de Documentos

- ▶ O comando `\documentclass` — define o tipo de documento.
- ▶ Metadados (`\title` e `\author`) e pacotes ficam no preâmbulo.
- ▶ O texto fica entre `\begin{document}` e `\end{document}`.
- ▶ O comando `\maketitle` cria o título; comandos `\section` criam seções numeradas.

```
\documentclass{article}
\usepackage[brazil]{babel}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
```

```
% cabeçalho do documento
```

```
\title{O Título}
```

```
\author{O Autor}
```

```
\begin{document}
```

```
% Corpo do documento
```

```
\maketitle
```

```
\section{Introdução}
```

```
Neste artigo nós \ldots
```

```
\end{document}
```

O Título

O Autor

11 de outubro de 2016

1 Introdução

Neste artigo nós ...

Revisão de \LaTeX : Exercícios

1. Aqui você encontra um texto para um pequeno artigo:¹

Clique aqui para abrir o exercícios no **Overleaf**

2. Adicione comandos \LaTeX ao texto para ficar parecido com este arquivo:

Abra para abrir o modelo do documento

Dicas

- ▶ Use os comandos `enumerate` e `itemize` para listas.
- ▶ Para digitar o símbolo $\%$ de porcentagem, *adicione* a barra invertida (`\%`).
- ▶ Para digitar a equação, use o comando `\frac` para fração e os comandos `\left(` and `\right)` para os parênteses ficarem com tamanho adequado.

¹Baseado em

Apresentações com beamer

- ▶ Beamer é um pacote para criar apresentações (tais como esta!) em \LaTeX .
- ▶ Usa-se a classe de documento **beamer**.
- ▶ Usa-se o ambiente **frame** para criar slides.

```
\documentclass{beamer}
\usepackage[brazil]{babel}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}

\title{Bem-vindo ao Beamer!}
\author{Você}
\institute{De onde você é?}
\date{Data da Apresentação}

\begin{document}

\begin{frame}
\titlepage % \maketitle do beamer
\end{frame}

\end{document}
```

Bem-vindo ao Beamer!

Você

De onde você é?

Data da Apresentação

Apresentações com `beamer`: Continuando

- ▶ A partir de agora, ao passarmos aos próximos slides, tente os exemplos digitando-os no modelo que se encontra no **Overleaf**.

Clique aqui para abrir o modelo no **Overleaf**

Apresentações com beamer: *Slides*

- ▶ Use o comando `\frametitle` para colocar título nos slides.
- ▶ Adicione então conteúdo aos slides.
- ▶ A fonte para esse slide parece-se com isso:

```
\begin{frame}
  \frametitle{Apresentações com beamer: Slides}
  \begin{itemize}
    \item Use o comando \texttt{frametitle} para colocar
      título nos slides.
    \item Adicione então conteúdo aos slides.
    \item A fonte para esse slide parece-se com isso:
  \end{itemize}
\end{frame}
```

Apresentações com **beamer**: Seções

- ▶ Use o comando `\section` para agrupar seus **slides**, e o **beamer** irá usá-los para criar um sumário automaticamente.
- ▶ Para gerar o sumário, use o comando `\tableofcontents`. Aqui está o desta apresentação. A opção `currentsection` destaca a seção atual.

```
\tableofcontents[currentsection]
```

Revisão de \LaTeX

Apresentações com **beamer**

Desenhando com **TikZ**

Notas com **todonotes**

Planilhas com **spreadtab**

Apresentações com beamer: Múltiplas Colunas

- ▶ Use os ambientes `columns` e `column` para quebrar o slide em colunas.
- ▶ O argumento para cada `column` determina sua largura.
- ▶ Veja também o pacote `multicol`, que quebra automaticamente seu conteúdo em colunas.

```
\begin{columns}
  \begin{column}{0.4\textwidth}
    \begin{itemize}
      \item Use as colunas \ldots
      \item O argumento \ldots
      \item Veja também \ldots
    \end{itemize}
  \end{column}
  \begin{column}{0.6\textwidth}
    % segunda coluna
  \end{column}
\end{columns}
```

Apresentações com beamer: Destaques

- Use `\emph` ou `\alert` para destacar:

Eu devo <code>\emph{ênfatizar}</code> que este é um ponto <code>\alert{importante}</code> .	Eu devo <i>ênfatizar</i> que este é um ponto importante .
--	---

- Ou especifique se quer negrito ou itálico:

Texto em <code>\textbf{negrito}</code> . Texto em <code>\textit{itálico}</code> .	Texto em negrito . Texto em <i>itálico</i> .
--	--

- Ou especifique a cor (em inglês Americano):

Aqui <code>\textcolor{red}{para}</code> e <code>\textcolor{green}{começa}</code> .	Aqui para e começa .
---	------------------------------------

- Veja <https://joerglenhard.wordpress.com/2011/08/01/beamer-customization-colors/> para mais cores & cores customizadas.

Apresentações com beamer: Figuras

- ▶ Usa-se `\includegraphics` do pacote `graphicx`.
- ▶ o ambiente `figure` centraliza por padrão, no beamer.

```
\begin{figure}  
\includegraphics[  
  width=0.5\textwidth]{gerbil}  
\end{figure}
```



Apresentações com beamer: Tabelas

- ▶ Tabelas em \LaTeX precisam de atenção.
- ▶ Use o ambiente `tabular` do pacote `tabularx`.
- ▶ Os argumento que especificam o alinhamento das colunas:
l para esquerda (*left*), c para centralizar e r para direita (*right*).

```
\begin{tabular}{lcr}
Item & Qtd. & Unid. \\\$ \\\
Celular & 1 & 399,99 \\\
Capa & 2 & 99,99 \\\
Cabo & 3 & 19,99 \\\
\end{tabular}
```

Item	Qtd.	Unid. \$
Celular	1	399,99
Capa	2	99,99
Cabo	3	19,99

- ▶ É necessário especificar as linhas verticais. Para as linhas horizontais, use o comando `\hline`.

```
\begin{tabular}{|l|c|r|} \hline
Item & Qtd. & Unid. \\\$ \\\hline
Celular & 1 & 399,99 \\\
Capa & 2 & 99,99 \\\
Cabo & 3 & 19,99 \\\hline
\end{tabular}
```

Item	Qtd.	Unid. \$
Celular	1	399,99
Capa	2	99,99
Cabo	3	19,99

- ▶ Use o “E comercial” `&` para separar colunas e barras invertidas duplas `\` para iniciar uma nova linha (como no ambiente `align*` que vimos na parte 1).

Apresentações com beamer: Blocos

- Um ambiente `block` produz uma caixa com título.

```
\begin{block}{Fato interessante}  
Isto é importante.  
\end{block}
```

Fato interessante
Isto é importante.

```
\begin{alertblock}{Atenção}  
Isto é realmente importante!  
\end{alertblock}
```

Atenção
Isto é realmente
importante!

- A aparência dos blocos depende do tema escolhido ...

Apresentações com beamer: Temas

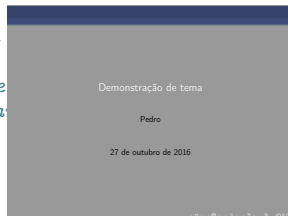
- ▶ Altere a aparência de suas apresentações usando temas.
- ▶ Veja http://deic.uab.es/~iblanes/beamer_gallery/index_by_theme.html para uma grande coleção de temas.

```
\documentclass{beamer}
\usepackage[brazil]{babel}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}

\usetheme{Darmstadt} % Tema
%experimente: Warsaw, Bergen, Madrid, ...

\usecolortheme{beetle} % Esquema de cores
% ou experimente: albatross, beaver, crane

\title{Demonstração de tema}
\author{Pedro}
\begin{document}
\begin{frame}
\titlepage
\end{frame}
\end{document}
```



Apresentações com **beamer**: Animação

- ▶ Um *frame* pode gerar múltiplos slides.
- ▶ Usa-se o comando `\pause` para mostrar somente parte do slide.

```
\begin{itemize}  
\item Você pode sentir a  
\pause \item antecipação?  
\end{itemize}
```

▶ Você pode sentir a

Apresentações com **beamer**: Animação

- ▶ Um *frame* pode gerar múltiplos slides.
- ▶ Usa-se o comando `\pause` para mostrar somente parte do slide.

```
\begin{itemize}  
\item Você pode sentir a  
\pause \item antecipação?  
\end{itemize}
```

- ▶ Você pode sentir a
- ▶ antecipação?

- ▶ Existem outras formas de fazer animações no **beamer**; veja também os comandos `\only`, `\alt`, e `\uncover`.

Apresentações com **beamer**: Exercício

Recrie a apresentação “Revolução dos Cravos” em **beamer**.

1. Abra o exercício no **Overleaf**:

[Clique para abrir esse exercício no **Overleaf**](#)

2. Baixe esta imagem no seu computador e faça o *upload* para o **Overleaf** usando o menu.

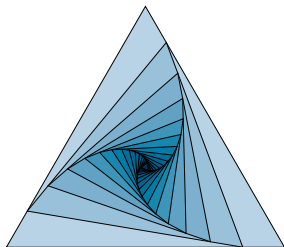
[Clique para baixar a imagem](#)

3. Adicione comandos \LaTeX para seu `.tex` e faça parecer como esta aqui:

[Clique para abrir o modelo do documento](#)

Desenhando com TikZ

- ▶ TikZ é um pacote para desenhar figuras em \LaTeX .
- ▶ Define uma poderosa linguagem de desenho dentro do \LaTeX .
Códigos curtos podem desenhar coisas surpreendentemente complicadas.



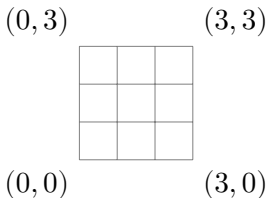
- ▶ Começaremos com coisas simples. Para desenhar um segmento de reta em TikZ:

```
\begin{tikzpicture}  
\draw (0,0) -- (1,1); % um segmento de  
\end{tikzpicture}
```



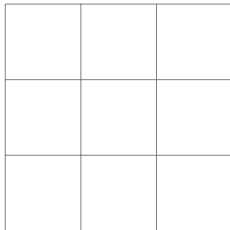
Desenhando com TikZ: Coordenadas

- O padrão para coordenadas são em centímetros, com o sentido usual:



- Quando você está trabalhando com TikZ, linhas de grade são de grande ajuda:

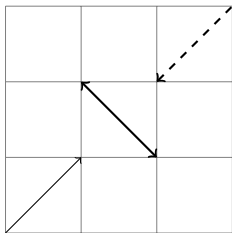
```
\begin{tikzpicture}  
  \draw[help lines] (0,0) grid (3,3);  
\end{tikzpicture}
```



Desenhando com TikZ: Lines

- ▶ Tipos de flecha e estilo de linhas são especificadas como opções ao comando `\draw`.
- ▶ Cada comando de desenho deve ser finalizado com um ponto e vírgula: `;`.

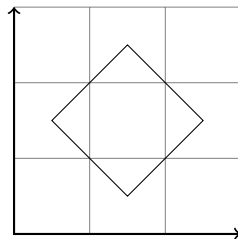
```
\begin{tikzpicture}  
  \draw[help lines] (0,0) grid (3,3);  
  \draw[->] (0,0) -- (1,1);  
  \draw[<->, thick] (2,1) -- (1,2);  
  \draw[<-, thick, dashed] (2,2)--(3,3);  
\end{tikzpicture}
```



Desenhando com TikZ: Caminhos

- ▶ Você pode especificar múltiplos pontos para formar um caminho.
- ▶ Flechas aparecerão apenas no final do caminho.

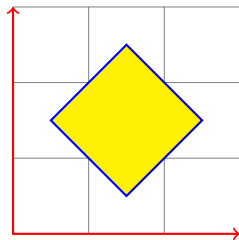
```
\begin{tikzpicture}  
  \draw[help lines] (0,0) grid (3,3);  
  % axes:  
  \draw[<->, thick] (0,3)--(0,0)--(3,0);  
  % diamond:  
  \draw (1.5,0.5) -- (2.5,1.5) --  
        (1.5,2.5) -- (0.5,1.5) --  
        cycle; % close the path  
\end{tikzpicture}
```



Desenhando com TikZ: Cores

- Cores são também especificadas como opções ao comando `\draw`.

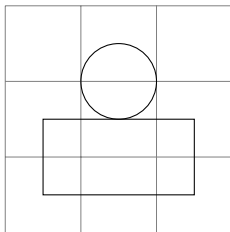
```
\begin{tikzpicture}  
  \draw[help lines] (0,0) grid (3,3);  
  % axes  
  \draw[<->, thick, red]  
    (0,3)--(0,0)--(3,0);  
  % diamond  
  \draw[thick, blue, fill=yellow]  
    (1.5,0.5) -- (2.5,1.5) --  
    (1.5,2.5) -- (0.5,1.5) --  
    cycle;  
\end{tikzpicture}
```



Desenhando com TikZ: Formas

- TikZ tem comandos internos para formas simples.

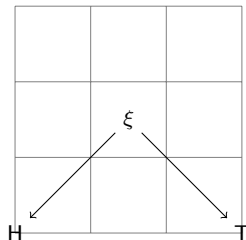
```
\begin{tikzpicture}  
  \draw[help lines] (0,0) grid (3,3);  
  \draw (1.5,2.0) circle (0.5);  
  \draw (0.5,0.5) rectangle (2.5,1.5);  
\end{tikzpicture}
```



Desenhando com TikZ: Nós & Legendas

- ▶ Utilize nós (comando `\node`) para colocar texto (e texto matemático) em desenhos do TikZ.
- ▶ Você também pode utilizar nós como coordenadas — muito útil para diagramas.

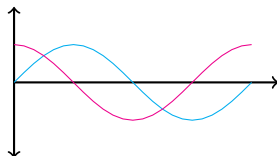
```
\begin{tikzpicture}
\draw[help lines] (0,0) grid (3,3);
\node (h) at (0,0) {H};
\node (x) at (1.5,1.5) {$\xi$};
\node (t) at (3,0) {T};
\draw[->] (x) -- (h);
\draw[->] (x) -- (t);
\end{tikzpicture}
```



Desenhando com TikZ: Funções

- Também é possível graficar algumas funções simples.

```
\begin{tikzpicture}[scale=0.5]
% y axis
\draw[<->, thick] (0,2) -- (0,-2);
% x axis
\draw[->, thick] (0,0) -- (7, 0);
% curves
\draw[cyan,domain=0:2*pi]
  plot (\x, {sin(\x r)});
\draw[magenta,domain=0:2*pi]
  plot (\x, {cos(\x r)});
\end{tikzpicture}
```



Desenhando com TikZ: Exemplos

- Dê uma olhada em TExample.net para muitos outros exemplos com TikZ:

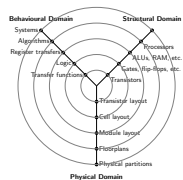
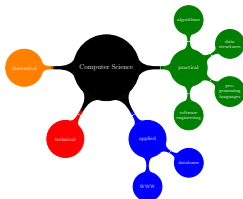
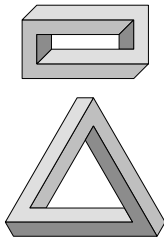
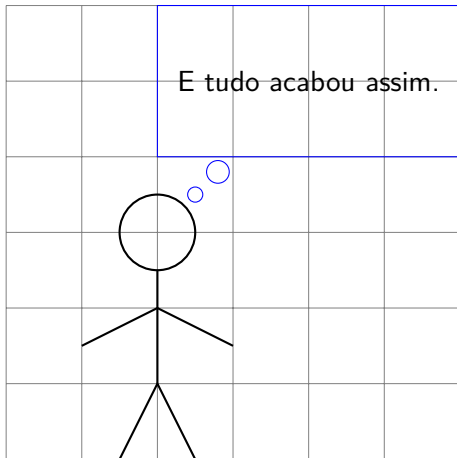


Figure 1: Gajski-Kuhn Y-chart

Desenhando com TikZ: Exercício

Desenhe a figura abaixo em TikZ:²



²Baseado em <http://xkcd.com/1022>

Notas com todonotes

- ▶ O comando `\todo` do pacote `todonotes` é ótimo para deixar notas em textos \LaTeX para você ou para seus colaboradores.

```
\todo{adicione resultados}  
\todo[color=blue!20]{arrume o método}
```

adicione resultados

arrume o método

- ▶ Dica quente: defina o seu próprio comando com `\newcommand`

```
\newcommand{\alice}[1]{\todo[color=green!40]{#1}}  
\newcommand{\bob}[1]{\todo[color=purple!40]{#1}}
```

Isso pode economizar muita digitação:

```
\alice{adicione resultados}  
\bob{arrume o método}
```

adicione resultados

arrume o método

Notas com todonotes

- ▶ Apenas notas *inline* são suportadas pelo `beamer`, mas notas na margem são suportadas por documentos normais.
- ▶ Há também um comando `\listoftodos` bem útil.

Towards the Confusing Unification of Rasterization and Local-Area Networks in State Machines

Alice Bob, Carol David, Edward Fredrick

Todo list

Are they polynomial time?	1
Realize multicast access points?	1
Instead of controlling the forward-error correction?	1
Phasellus libero ipsum, pellentesque sit amet, sem.	1

Abstract

Rasterization and Smalltalk, while important in theory, have not until recently been considered important. Given the current status of wearable methodologies, analysts clearly desire the refinement of IPv4. Purr, our new heuristic for the producer-consumer problem [1], is the solution to all of these problems.

1 Introduction

Recent advances in certifiable symmetries and Bayesian technology synchro-nize in order to realize access points. This is a direct result of the construction of multicast algorithms. This is a direct result of the analysis of active networks. The emulation of suffix trees would profoundly improve congestion control [4].

To our knowledge, our work in our research marks the first method analyzed specifically for scalable models. Existing interactive and permutable methodologies use Smalltalk to measure the construction of the partition table. The disadvantage of this type of method, however, is that hash tables can be made real-time, cooperative, and reliable. Existing “fuzzy” and concurrent algorithms use the evaluation of multicast frameworks to request access points. On the other hand, distributed archetypes might not be the

Are they polynomial time?

Realize multicast access points?

Instead of controlling the forward-error correction?

Phasellus libero ipsum, pellentesque sit amet, sem.

Planilhas com `spreadtab`

- ▶ Agora que você viu como \LaTeX pode substituir Word e PowerPoint, que tal Excel?
- ▶ Tarefa: tente o pacote `spreadtab`!

Obrigado, e seja feliz escrevendo seus próximos T_EXtos!