

# Produção de Documentos e Apresentações com $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$

**Prof. Alcemir Rodrigues Santos**

Laboratório de Engenharia de Software

Universidade Estadual do Piauí



6 de maio de 2023



## 1 Introdução

## 2 Documentos

- Comandos Básicos
- Elaboração de documentos

## 3 Apresentações com Beamer

- Estrutura
- Aparência

## 4 Conclusão

# Breve História do $\text{\LaTeX}$



- Processador de textos x Editor de textos



- $\text{\TeX}$ (1977) – Donald E. Knuth



- $\text{\LaTeX}$ (1985) – Leslie Lamport
- $\text{\LaTeX}2_{\epsilon}$  (1994) – LaTeX3 Team

# Por que usar $\text{\LaTeX}$ ?



- Conteúdo x Formatação
- Portabilidade
- Acabamento gráfico superior
- Estabilidade
- Escalabilidade
- Disponibilidade e custo
- Utilização de arquivos texto
- Suporte referências bibliográficas
- Fácil manejo de documentos grandes

# Limitações $\text{\LaTeX}$ ?



- Personalização exige mais estudo
- São necessárias várias ferramentas
- Legibilidade reduzida
- Aprendizagem mais lenta



- Distribuições para instalação
  - Unix/Linux (TeXLive): <http://www.tug.org/texlive/>
  - Windows (MikTeX): <http://www.miktex.org/>
  - MAC OS (MacTeX): <http://www.tug.org/mactex/>
- Pessoas dispostas a ajudar
  - T<sub>E</sub>X StackChange: <http://tex.stackexchange.com/>
- Pacotes: arquivos e documentação
  - CTAN: <http://www.ctan.org/>



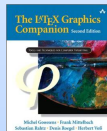
**LaTeX: A Document Preparation System** Leslie Lamport e Duane Bibby



**The LaTeX Companion** Michel Goossens, Frank Mittelbach e Alexander Samarin



**A Guide to LaTeX: Document Preparation for Beginners and Advanced Users** Helmut Kopka e Patrick W. Daly



**The LaTeX Graphics Companion** Michel Goossens, Sebastian Rahtz e Frank Mittelbach

# Estrutura lógica dos arquivos $\text{\LaTeX}$



## Preâmbulo

Tipos de documento, pacotes adicionais e comando gerais.

```
\documentclass[12pt,a4paper]{article}
\usepackage{graphicx}
\newcommand{\titulo}[1]{\large\bf #1}
...
```

## Corpo

O texto do documento.

```
\begin{document}
...
\end{document}
```





- Comandos: `\command`, `\command{}`, `\command[]{}{}`
- Ambientes: `\begin{ambiente}...\end{ambiente}`
- Caracteres especiais: `$& \%#_{}~^\\` devem ser precedidos por `\` ou o comando `\verb`
- Espaçamento automático
- Comentários: usa-se o caractere `%` ou `\begin{comment}...\end{comment}`
- Delimitador de contexto: `{ ... }`
- Referência a arquivos: `/igual/ao/linux`

# Exemplo funcional mínimo!



$\text{\LaTeX}$  hello world!

```
\documentclass[12pt,a4paper]{article}  
\begin{document}  
Hello world !  
\end{document}
```



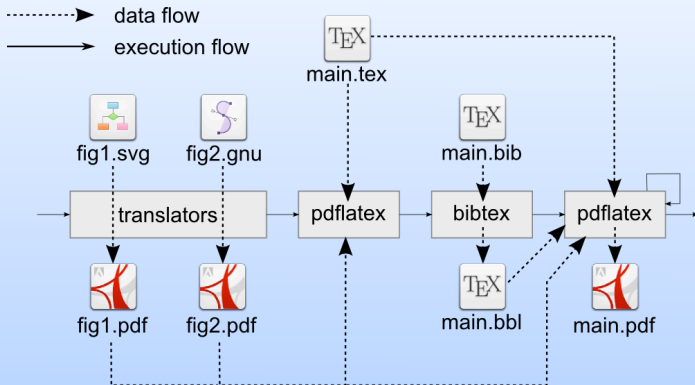
## Opções

`10pt, 12pt, oneside, twoside, a4paper, letterpaper, titlepage, twocolumn`

## Documentos comuns

`article, book, report, slides, letter`

# O Processo de Compilação





- O programa `latex` gera o arquivo `.dvi`: `latex arquivo.tex`
- A inclusão de referências bibliográficas feita através do programa `bibtex`: `bibtex arquivo`
- O *PostScript* final pode ser gerado pelo `dvips`: `dvips arquivo.dvi -o arquivo.ps`
- O *PostScript* pode ser visualizado e impressão pelo `gsview32.exe` (Windows) ou `gv` (Linux/Unix).
- Uma outra alternativa é utilizar o comando `pdflatex`



- `.tex`: Arquivos fontes
- `.log`: Relatório da compilação
- `.dvi`: Resultado da compilação dos arquivos fonte via *latex*
- `.aux`: Arquivos auxiliar utilizado na geração documento final (*.dvi* ou *.pdf*)
- `.cls`: Arquivos de classe
- `.sty`: Pacotes



- `.toc`: Itens para o sumário
- `.lof`: Itens para a lista de figuras
- `.lot`: Itens para a lista de tabelas
- `.bbl`: Itens para a lista de bibliografias
- `.blg`: Arquivos auxiliar utilizado na geração de bibliografias



- Tipos de divisões: `\section{}`, `\subsection{}`,  
`\subsubsection{}` `\paragraph{}`,  
`\subparagraph{}`
- Classe book: `\part{}`, `\chapter{}`
- Apêndices: `\appendix`



- Utilizar o pacote babel e fontes especiais:

```
\documentclass[12pt,a4paper]{article}  
\usepackage[latin1]{inputenc}  
\usepackage[T1]{fontenc}  
\usepackage[brazil,english]{babel}  
\begin{document}  
\selectlanguage{brazil}  
  
...  
\end{document}
```

# Aplicando Formatações ao Texto



- Novo parágrafo: é suficiente deixar uma linha em branco
- Negrito: `\textbf{text}` → **text**
- Itálico: `\textit{text}` → *text*
- Texto centralizado, esquerda e direita: Usar ambientes *center*, *flushleft* e *flushright*.

```
\begin{center}  
... texto ...  
\end{center}
```



- Listas numeradas:

```
\begin{enumerate}  
\item Banana  
\item Batata  
\end{enumerate}
```

- 1 Banana
- 2 Batata



- Listas de itens:

```
\begin{itemize}  
\item Banana  
\item Batata  
\end{itemize}
```

- Banana
- Batata



- Listas de descrição:

```
\begin{description}  
\item[Fruta:] Banana  
\item[Ferramenta:] Martelo  
\end{description}
```

Fruta: Banana  
Ferramenta: Martelo



- Modo texto V.S modo matemático
- Separadores \$ ... \$ e \$\$ ... \$\$:

Tem-se que  $x=0$ .

Tem-se que  $x = 0$ .

Tem-se que:  $xx=0xx$ .

Tem-se que:

$$x = 0$$

.

- Sobrescrito e Subscrito:

$X^{\sup}=Y_{\inf}=Z^{\sup}_{\inf}$

$$X^{\sup} = Y_{\inf} = Z_{\inf}^{\sup}$$

- Espaços em modo matemático:

$a\ b, a\ ;\ b, a\ ;\ ;\ ;\ b$

$$ab, a\ b, a\ b$$

- Negrito:

$\mathbf{x} = [x_1\ ;\ ;\ x_2]^T$

$$\mathbf{x} = [x_1\ x_2]^T$$

- Vetores:

$$\vec{a}, \hat{a}, \bar{a}, \tilde{a}, \dot{a}, \ddot{a}$$

$$\vec{a}, \hat{a}, \bar{a}, \tilde{a}, \dot{a}, \ddot{a}$$

- Somatórios e Integrais:

$$\sum_{i=1}^n f(x_i) \Delta x \approx \int_a^b f(x) dx$$

$$\sum_{i=1}^n f(x_i) \Delta x \approx \int_a^b f(x) dx$$



- Frações:

$$(y+2) \frac{x+1}{x-1}$$

$$(y+2) \frac{x+1}{x-1}$$

- Limites e derivadas parciais:

$$\frac{\partial f(x,y)}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{f(x+\Delta x, y) - f(x, y)}{\Delta x}$$

$$\frac{\partial f(x,y)}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{f(x+\Delta x, y) - f(x, y)}{\Delta x}$$



- Parênteses, chaves e colchetes:

```
$$ \left[  
  \left\{ \right.  
    \left( \right.  
      {1 \over x}  
    \right)^2 - 3  
  \right\} + x^2  
\right]^3  
$$
```

$$\left[ \left\{ \left( \frac{1}{x} \right)^2 - 3 \right\} + x^2 \right]^3$$

- Matrizes: *//Definição*

```
$$\mathbf{I} =  
\left[  
  \begin{array}{cccc}  
    1      & 0      & \ldots & 0      \\  
    0      & 1      & \ldots & 0      \\  
    \vdots & \vdots & \ddots & \vdots \\  
    0      & 0      & \ldots & 1      \\  
  \end{array}  
\end{array}  
\right]$$
```

- Matrizes: *//Resultado*

$$I = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}$$

- Ambiente tabular: *//Definição*

```
\begin{tabular}{|l|c|c|r|}  
\hline  
Item & Preço & Quantidade & Total \\  
\hline \hline  
Banana & 0,55 & 5 & 2,75 \\  
\hline  
Batata & 0,35 & 3 & 1,05 \\  
\hline \hline  
      &      & Total & 3,80 \\  
\hline  
\end{tabular}
```

- Ambiente tabular: *//Resultado*

Item	Preço	Quantidade	Total
Banana	0,55	5	2,75
Batata	0,35	3	1,05
		Total	3,80



- Declarar o pacote `graphicx`: `\usepackage{graphicx}`
- Inserir o comando  
`\includegraphics[options]{path}`:
- Exemplo:  
`\includegraphics[scale=.3]{figs/leslie.ps}`
- Outras opções disponíveis: *scale*, *width*, *height* e *angle*.

# Mudando o tipo de fonte



Comando	Família de fonte
<code>\textit{Itálico}</code>	<i>Itálico</i>
<code>\textsc{Small Caps}</code>	SMALL CAPS
<code>\textbf{Negrito}</code>	<b>Negrito</b>
<code>\texttt{Typewriter}</code>	Typewriter
<code>\textsf{Sans Serif}</code>	Sans Serif
<code>\textrm{Romano}</code>	Romano
<code>\textsl{Inclinado}</code>	<i>Inclinado</i>



# Mudando o tamanho da fonte



Comando	Tamanho resultante
<code>{\tiny LES}</code>	LES
<code>{\scriptsize LES}</code>	LES
<code>{\footnotesize LES}</code>	LES
<code>{\small LES}</code>	LES
<code>{\normal LES}</code>	LES
<code>{\large LES}</code>	LES
<code>{\Large LES}</code>	LES
<code>{\LARGE LES}</code>	LES
<code>{\huge LES}</code>	LES
<code>{\Huge LES}</code>	LES



- O comando `\pagestyle{ }` define a aparência das páginas:
  - `\pagestyle{plain}`: Numeração no rodapé e sem cabeçalho.
  - `\pagestyle{headings}`: Numeração no rodapé e cabeçalho.
  - `\pagestyle{empty}`: Sem numeração ou cabeçalho.
  - `\pagestyle{myheadings}`: Permite que o usuário especifique através dos comandos `\markboth{cab_esq}{cab_dir}` e `\markright{cab_dir}`.
  - Use `\thispagestyle{estilo}` para mudar somente uma determinada página.

# Uma capa mínima e sumário



- Incluir `titlepage` nas opções de classe
- Definir o *título* do trabalho, *autor* e *data*:  
`\title{Curso de \LaTeX}`  
`\author{Alcemir Santos} \date{}`,  
`\date{\today}` ou `\date{Outubro/2008}`
- Colocar o comando `\maketitle` depois do início do documento.
- Acrescentar a seguir o comando `\tableofcontents`

- Horizontais:

Efeito do comando `\hspace{.83cm}` na linha

Efeito do comando `\hfill` na linha

Efeito do comando `\hrulefill` na linha

Efeito do comando `\dotfill` na linha

- Verticais:

Espaçamento fixo: `\vspace{0.3cm}`

Preenchimento vertical: `\vfill`

- `\hspace*{ }` e `\vspace*{ }` → evitam problemas com linha nova e página nova



- Se a hifenação falhar, colocar no preâmbulo:  
`\hyphenation{hi-fen ma-nu-al}`
- O comando `\pagebreak` inicia um nova página
- Notas de rodapé<sup>1</sup> podem ser feitas com `\footnote{texto}`

---

<sup>1</sup>como esta aqui em baixo.



## Tabelas

```
\begin{table}[h|t|b|p]
\begin{tabular}
...
\end{tabular}
\end{table}
```

## Figuras

```
\begin{figure}[h|t|b|p]
...
\includegraphics{}
...
\end{figure}
```

- `\clearpage`

Finaliza a página e força o aparecimento dos objetos flutuantes restantes

Permite que várias figuras sejam agrupadas em uma só área.

- `\usepackage{subfigure}`

```
\begin{figure}
\mbox{
  \subfigure[Caption (a)]{
    \includegraphics[scale=.3]{fig-a.ps}
  }
  \subfigure[caption (b)]{
    \includegraphics[scale=.3]{fig-b.ps}
  }
}
\caption{Caption geral}
\end{figure}
```

Permite a inclusão de arquivos com código-fonte no documento, com formatação dependente da linguagem.

```
\usepackage{listings}, \lstloadlanguages{C},  
\lstset{language=C}, \lstinputlisting{filename}
```

```
#include <stdio.h>  
/* Comment block */  
int main(){  
    // Line comment.  
    printf("LaTeX is great for programmers!");  
    return 0;  
}
```





- `\label{ELEM-ID}`: Relaciona o elemento corrente do documento com a chave `ELEM-ID`.
- Pode ser tabelas, figuras, seções, subseções, item de lista, *etc.*
- `\ref{ELEM-ID}`: Referencia o elemento relacionado com a chave `ELEM-ID`
- `\pageref{ELEM-ID}`: Referencia a página onde está o elemento relacionado com a chave `ELEM-ID`
- As chaves devem ser únicas e são sensíveis à caixa
- Deve-se compilar duas vezes

# Referências Cruzadas: tabelas



```
\begin{table}
\centering
\begin{tabular}{|c|c|}\hline
Quant & R$ \\ \hline
10     & 2.3 \\ \hline
\end{tabular}
\caption{Valores}
\label{tab:valores}
\end{table}
```

Quant	R\$
10	2.3

Tabela 1: Valores

A Tabela 1 mostra ...

```
A Tabela~\ref{tab:valores}
mostra \ldots
```

# Referências Cruzadas: figuras



```
\begin{figure}  
  \centering  
  \includegraphics[scale=.3]  
  {./img/les}  
  \caption{Logo do LES.}  
  \label{fig:les}  
\end{figure}
```

```
A Figura~\ref{fig:les}  
(Pág. \pageref{fig:les})  
mostra \ldots
```

```
A Figura~\ref{fig:les}  
mostra \ldots
```



Figura 1: Logo do LES.

A Figura 1 (Pág. 43) mostra ...

# Referências Cruzadas: equações



A Equação~\ref{eq:logn} mostra a definição da função logaritmo , válida para  $x > 0$ .

```
\begin{equation}
\ln(x)=\int_1^x \frac{1}{t} dt
\label{eq:logn}
\end{equation}
```

A Equação 1 mostra a definição da função logaritmo, válida para  $x > 0$ .

$$\ln(x) = \int_1^x \frac{1}{t} dt \quad (1)$$

# Referências Cruzadas: equações



Na início da seção adicionei o comando `\label{}` após a definição da seção com `\section{}` assim:

```
\section{Minha seção} \label{sec:minha}
```

A referência a esta seção deve ser feita assim:

A Seção `\ref{sec:minha}`  
apresenta `\ldots`

A Seção 2 apresenta ...



- 1 Criar um arquivo de bibliografias (*.bib*)
- 2 Utilizar o comando `\cite{chave}` para indicar a referência bibliográfica desejada
- 3 Definir o estilo de referência utilizada com `\bibliographystyle{estilo}`
- 4 Especificar o arquivo de bibliografias e o ponto de inserção com `\bibliography{arquivo}`
- 5 Utilizar o `bibtex`, compilador de referências

- Formato:

```
@tipo_de_citação{chave,  
  campo_1 = {Valor 1},  
  campo_2 = {Valor 2},  
  ...,  
  campo_n = {Valor n},  
}
```

- Tipos mais comuns: *book*, *article*, *inproceedings*, *inbook*, *masterthesis*, *phdthesis*, *techreport*.



- `\cite{chave}`: coloca a chamada da referência e inclui na lista final
- `\nocite{chave}`: não coloca a chamada mas inclui na lista
- `\nocite{*}`: lista todas as referências bibliográficas sem chamada no texto
- Leitura adicional: pacote `natbib`.





- 1 Elaborar um documento com as estruturas vistas até aqui.
- 2 Criar artigo com template<sup>2</sup> da Sociedade Brasileira de Computação.

---

<sup>2</sup>Disponível aqui: <http://bit.ly/1BQBTq9>



- Os comandos padrões e  $\text{\LaTeX}$  2 $\epsilon$  também funcionam no Beamer
- Súcarios podem ser gerados automaticamente
- Você pode facilmente criar efeitos dinâmicos
- A aparência pode ser mudada com uso de temas à seu gosto
- Os temas disponíveis por padrão são bem estruturados e fáceis de ler. O que torna a apresentação mais profissional e fácil da audiência seguir.



- A aparência, cores e fontes utilizada na apresentação podem ser facilmente alterada de forma *global*, mas alterações podem ser feitas de forma *local*
- Você pode criar apresentações usando o mesmo código utilizado no seu artigo  $\text{\LaTeX}$
- A saída produzida é tipicamente um *.pdf* file, o que facilita a apresentação em qualquer plataforma
- Sua apresentação irá ter a mesma estrutura, independente de qual computador ou visualizador está sendo utilizado

# Onde achar o Beamer?



Beamer está disponível para download *gratuitamente* em:

<https://bitbucket.org/rivanvx/beamer/wiki/Home>

Existe bastante coisa sobre Beamer na Internet e existe também uma *documentação* Beamer disponível no repositório acima e no endereço abaixo:

<http://www.ctan.org/tex-archive/macros/latex/contrib/beamer/doc/>



- A maneira mais rápida de iniciar a desenvolver apresentações com Beamer é utilizar-se de templates prontos.
- Vários templates prontos estão disponíveis no repositório do Beamer
- Um exemplo pode ser encontrado seguindo este caminho:  
`beamer/solutions/conference-talks/  
conference-ornate-20min.en.tex`
- Copie o arquivo e modifique os conteúdos.

# Para testar suas apresentações



- Para ver como é uma apresentação, compile o código  $\text{\LaTeX}$  duas vezes
- Abra o arquivo *.pdf* com o visualizador disponível e utilize em modo “Tela Cheia”
- O sumário gerado tem *hyperlinks* nas seções e subseções, além de uma linha auxiliar com botões de navegação



- Cada projeto Beamer é feito de uma série de *frames*. Cada frame produz um ou mais slides, dependendo da existência ou não de “*overlays*”, as quais serão discutidas mais tarde.
- A opção `[plain]` causa a supressão de “cabeçalho”, “rodapé”, e “barra lateral”. Útil pra exibir figuras grandes.

## Um frame básico

```
\begin{frame}[<alignment>]
\frametitle{Frame Title Goes Here}
  Texto do frame e/ou o código LaTeX.
\end{frame}
```



- Para compor frames basta escrever seu texto ou código  $\text{\LaTeX}$  entre os comandos `\begin{}` e `\end{}` frame.
- Os frames são centralizados [c] por padrão. Os valores [t] (alinhamento superior) e [b] (alinhamento inferior) também são aceitos.

## Um frame básico

```
\begin{frame}[t]  
\frametitle{Frame Title Goes Here}  
  Texto do frame e/ou o código LaTeX.  
\end{frame}
```



# “Capa” para a apresentação



O frame de capa mostra somente as informações inserida no início do documento:

## Um frame básico

```
\begin{frame}  
  \titlepage  
\end{frame}
```

# “Capa” para a apresentação



Por padrão, o comando `\titlepage` cria uma página que inclui:

- Título
- Autor
- Afiliação
- Data
- Imagem (logo)

Caso algum desses valores não seja declarados no preâmbulo, eles não serão incluídos do slide de capa.



O comando `\tableofcontents` cria dinamicamente o sumário baseado na estrutura que você definiu

## Slide de Sumário

```
\begin{frame}  
  \frametitle{Sumário}  
  \tableofcontents[ pausesections]  
\end{frame}
```

Perceba que o argumento `pausesections` permite que os itens apareçam seção à seção.



## Exemplo

```
\begin{frame}
  \titlepage
\end{frame}
\begin{frame}
  \frametitle{Sumário}
  \tableofcontents[ pausesections]
\end{frame}

\begin{frame}
  \frametitle{Introdução}
  Corpo do texto ou código LaTeX.
\end{frame}
```



- *Overlays* permitem que seus slides apareçam incrementalmente.
- Mais especificamente, em Beamer, *overlays* controlam a ordem na qual as partes do frame aparecem.
- Uma maneira fácil de implementar overlays é usar o comando `\pause` entre as partes que devem aparecer separadamente

Por exemplo:

`\textbf{Step1:}` Compute the maximal suffix of  $w$  with respect to  $\preceq_l$  (say  $v$ ) and the maximal suffix of  $w$  with respect to  $\preceq_r$  (say  $v'$ ).

`\pause`

`\textbf{Step 2:}` Find words  $u$ ,  $u'$  such that  $w = uv = u'v'$ .

`\pause`

`\textbf{Step 3:}` If  $|v| \leq |v'|$ , then output  $(u, v)$ . Otherwise, output  $(u', v')$ .



**Step1:** Compute the maximal suffix of  $w$  with respect to  $\preceq_l$  (say  $v$ ) and the maximal suffix of  $w$  with respect to  $\preceq_r$  (say  $v'$ ).

**Step 2:** Find words  $u, u'$  such that  $w = uv = u'v'$ .

**Step 3:** If  $|v| \leq |v'|$ , then output  $(u, v)$ . Otherwise, output  $(u', v')$ .



**Step1:** Compute the maximal suffix of  $w$  with respect to  $\preceq_l$  (say  $v$ ) and the maximal suffix of  $w$  with respect to  $\preceq_r$  (say  $v'$ ).

**Step 2:** Find words  $u, u'$  such that  $w = uv = u'v'$ .

**Step 3:** If  $|v| \leq |v'|$ , then output  $(u, v)$ . Otherwise, output  $(u', v')$ .





**Step1:** Compute the maximal suffix of  $w$  with respect to  $\preceq_l$  (say  $v$ ) and the maximal suffix of  $w$  with respect to  $\preceq_r$  (say  $v'$ ).

**Step 2:** Find words  $u, u'$  such that  $w = uv = u'v'$ .

**Step 3:** If  $|v| \leq |v'|$ , then output  $(u, v)$ . Otherwise, output  $(u', v')$ .

# Especificação de Overlays



São feitas com os símbolos ( $<$ ,  $>$ ) e indicam quais partes devem aparecer

A especificação  $<1->$  diz “mostre do slide 1 em diante.”  $<1-3>$  diz “mostre do slide 1 ao 3.”  $<-3, 5-6, 8->$  diz “mostre todos os slides, exceto os slides 4 e 7.”

Um exemplo:

```
\begin{itemize}
```

```
\item<1>      $abcadcabca$
```

```
\item<1-2>    $abcabcabca$
```

```
\item<1-2>    $accaccacca$
```

```
\item<1>      $bacabacaba$
```

```
\item<1,3>    $cacdaccacc$
```

```
\item<1-2>    $caccaccacc$
```

```
\end{itemize}
```

- *abcadcabca*

- *abcabcabca*

- *accaccacca*

- *bacabacaba*

- *cacdaccacc*

- *caccaccacc*

# Especificação de Overlays



São feitas com os símbolos ( $<$ ,  $>$ ) e indicam quais partes devem aparecer

A especificação  $<1->$  diz “mostre do slide 1 em diante.”  $<1-3>$  diz “mostre do slide 1 ao 3.”  $<-3, 5-6, 8->$  diz “mostre todos os slides, exceto os slides 4 e 7.”

Um exemplo:

```
\begin{itemize}
  \item<1>      $abccadccabca$
  \item<1-2>    $abcabccabca$
  \item<1-2>    $accaccacca$
  \item<1>      $bacabacaba$
  \item<1,3>    $caccdaccacc$
  \item<1-2>    $caccaccacc$
\end{itemize}
```

- *abccadccabca*
- *abcabccabca*
- *accaccacca*
- *bacabacaba*
- *caccdaccacc*
- *caccaccacc*

# Especificação de Overlays



São feitas com os símbolos ( $<$ ,  $>$ ) e indicam quais partes devem aparecer

A especificação  $<1->$  diz “mostre do slide 1 em diante.”  $<1-3>$  diz “mostre do slide 1 ao 3.”  $<-3, 5-6, 8->$  diz “mostre todos os slides, exceto os slides 4 e 7.”

Um exemplo:

```
\begin{itemize}
  \item<1>      $abccadcabca$
  \item<1-2>    $abcabcbabca$
  \item<1-2>    $accaccacca$
  \item<1>      $bacabacaba$
  \item<1,3>    $cacdaccacc$
  \item<1-2>    $caccaccacc$
\end{itemize}
```

- *abccadcabca*
- *abcabcbabca*
- *accaccacca*
- *bacabacaba*
- *cacdaccacc*
- *caccaccacc*

Podem também ser utilizadas para dar efeito em partes do texto. Por exemplo, o código abaixo aplica o comando `\alert{}` somente nos slides especificados:

```
\alert{Todos slides}  
\alert<2>{Slide 2}  
\alert<3>{Slide 3}  
\alert<1,3>{Slides 1 e 3}  
\alert<-2,4>{Slides 1, 2 e 4}
```

**Todos slides**

**Slide 2**

**Slide 3**

**Slides 1 e 3**

**Slides 1, 2 e 4**

**Nota:** Se quiser que cada item de uma lista apareça em ordem, basta usar a opção `[<+>]`. Exemplo: `\begin{itemize}[<+>]`

Podem também ser utilizadas para dar efeito em partes do texto. Por exemplo, o código abaixo aplica o comando `\alert{}` somente nos slides especificados:

```
\alert{Todos slides}  
\alert<2>{Slide 2}  
\alert<3>{Slide 3}  
\alert<1,3>{Slides 1 e 3}  
\alert<-2,4>{Slides 1, 2 e 4}
```

**Todos slides**

**Slide 2**

Slide 3

Slides 1 e 3

**Slides 1, 2 e 4**

**Nota:** Se quiser que cada item de uma lista apareça em ordem, basta usar a opção [`<+>`]. Exemplo: `\begin{itemize} [<+>]`

Podem também ser utilizadas para dar efeito em partes do texto. Por exemplo, o código abaixo aplica o comando `\alert{}` somente nos slides especificados:

```
\alert{Todos slides}  
\alert<2>{Slide 2}  
\alert<3>{Slide 3}  
\alert<1,3>{Slides 1 e 3}  
\alert<-2,4>{Slides 1, 2 e 4}
```

**Todos slides**

**Slide 2**

**Slide 3**

**Slides 1 e 3**

**Slides 1, 2 e 4**

**Nota:** Se quiser que cada item de uma lista apareça em ordem, basta usar a opção [`<+>`]. Exemplo: `\begin{itemize} [<+>]`

Podem também ser utilizadas para dar efeito em partes do texto. Por exemplo, o código abaixo aplica o comando `\alert{}` somente nos slides especificados:

```
\alert{Todos slides}  
\alert<2>{Slide 2}  
\alert<3>{Slide 3}  
\alert<1,3>{Slides 1 e 3}  
\alert<-2,4>{Slides 1, 2 e 4}
```

**Todos slides**

Slide 2

Slide 3

Slides 1 e 3

**Slides 1, 2 e 4**

**Nota:** Se quiser que cada item de uma lista apareça em ordem, basta usar a opção [`<+>`]. Exemplo: `\begin{itemize} [<+>]`





Overlays também podem ser utilizados em ambientes

```
\begin{theorem}<1->  
  Um teorema.  
\end{theorem}
```

```
\begin{proof}<2->  
  Uma prova.  
\end{proof}
```

Theorem

*Um teorema.*

Demonstração.

Uma prova.

Overlays também podem ser utilizados em ambientes

```
\begin{theorem}<1->  
  Um teorema.  
\end{theorem}
```

Theorem

*Um teorema.*

```
\begin{proof}<2->  
  Uma prova.  
\end{proof}
```

Demonstração.

Uma prova.



Beamer provê muitas formas de estruturar seus slides de forma que ele fiquem bem organizados e fácil de sua audiência seguir. Como exemplos, temos:

- Columns
- Blocks
- Boxes (Borders)



O ambiente pode ser chamado como segue:

```
\begin{columns}
  \column{.xx\textwidth}
  Texto ou código da segunda coluna
  \column{.xx\textwidth}
  Texto ou código da segunda coluna
\end{columns}
```

Onde **.xx** é porcentagem do slide.



Blocos podem ser utilizados para separar uma porção específica do texto do restante do slide:

```
\begin{block}{Introdução à {\LaTeX}}  
``Beamer é uma classe {\LaTeX} para criar  
apresentações\ldots''  
\end{block}
```

Introdução à  $\text{\LaTeX}$

“Beamer é uma classe  $\text{\LaTeX}$  para criar apresentações. . .”

# Estruturas dos Slides: Blocos



Outros ambientes podem ser utilizados como blocos:

## Introduction to $\text{\LaTeX}$

Conteúdo	Ambiente correspondente
Genérico	<code>block</code>
Teoremas	<code>theorem</code>
Lemas	<code>lemma</code>
Provas	<code>proof</code>
Corolários	<code>corollary</code>
Exemplos	<code>example</code>
Título em destaque	<code>alertblock</code>

# Estruturas dos Frames: Colunas e Blocos



Podemos combinar “colunas” e “blocos” para fazer uma apresentação mais limpa.

```
\begin{columns}[t]
  \column{.5\textwidth}
    \begin{block}{Cabeçalho da Coluna 1}
      Corpo do texto da Coluna 1
    \end{block}
  \column{.5\textwidth}
    \begin{block}{Cabeçalho da Coluna 2}
      Corpo do texto da Coluna 2
    \end{block}
\end{columns}
```

E temos como resultado...

# Estruturas dos Frames: Colunas e Blocos



Cabeçalho da Coluna 1

Corpo do texto da Coluna 1

Cabeçalho da Coluna 2

Corpo do texto da Coluna 2

Perceba que a opção `[t]` adicionado ao ambiente de colunas alinha os blocos por cima para que eles fiquem na mesma linha vertical, diferentemente de centralizado no slide.



# Estruturas dos Frames: Colunas e Blocos



Bordas também podem ser utilizadas para adicionar uma organização à sua apresentação. Com o uso do pacote `fancybox` (lembre-se de declarar `\usepackage{fancybox}` no preâmbulo).

## Borda de Textos

Comando	Resultado
<code>\shadowbox{Texto}</code>	
<code>\fbox{Texto}</code>	
<code>\doublebox{Texto}</code>	
<code>\ovalbox{Texto}</code>	
<code>\Ovalbox{Texto}</code>	

Temas podem mudar completamente a aparência de sua apresentação. Você escolhe o tema a ser utilizados usando o comando `\usetheme{}` com um dos seguintes argumentos:

Antibes	Boadilla	Frankfurt	Juanlespins
Montpellier	Singapore	Bergen	Copenhagen
Goettingen	Madrid	Paloalto	Warsaw
Berkeley	Darmstadt	Hannover	Malmoe
Pittsburgh	Berlin	Dresden	Ilmenau
Marburg	Rochester		



Se você gosta do “layout” de um tema, mas não gosta da cor, você pode facilmente invocar uma nova cor para o tema substituindo `default` no comando `\usetheme{default}` inserido no preâmbulo por um dos seguintes argumentos:

albatross	crane	beetle	dove
fly	seagull	wolverine	beaver



Existe também a possibilidade de especificar cores para a parte *interna* ou *externa* da mesma forma da cor geral do tema: substituindo `default` no comando `\usetheme{default}`.

## Opções parte interna

`lily`   `orchid`   `rose`

## Opções parte externa

`whale`   `seahorse`   `dolphin`



- 1 Elaborar uma apresentação com as estruturas vistas.



- Este material foi criado com base em duas referencias principais:
  - Curso de extensão em  $\text{\LaTeX}$  ministrado por *Messias Alves* em 2008.  
/\*Parte sobre a criação de documentos\*/
  - Tutorial de Beamer em Beamer, do *Prof. Charles T. Batts* de 2007.  
/\*Parte sobre a criação de apresentações\*/



- <http://latex.simon04.net/>
- [http://deic.uab.es/~iblanes/beamer\\_gallery/index\\_by\\_theme.html](http://deic.uab.es/~iblanes/beamer_gallery/index_by_theme.html)
- <http://texdoc.net/texmf-dist/doc/latex/beamer/doc/beameruserguide.pdf>
- <http://www.stdout.org/~winston/latex/latexsheet.pdf>
- <http://en.wikibooks.org/wiki/LaTeX>
- <http://tex.stackexchange.com/>

Happy  $\text{\LaTeX}$  coding!

Obrigado por ter tirado um tempo para estar aqui e acompanhar este tutorial de  $\text{\LaTeX}$ . Agora você deve ter um conhecimento básico para começar a criar seus documentos e apresentações com alta qualidade.

