Introdução ao LaTeX

Pedro F. Silva

Departamento de Matemática
Faculdade de Ciências da Universidade do Porto

O que é o LaTeX?

- Do ponto de vista funcional podemos definir, abusivamente, o LaTeX como um processador de texto.
- De um ponto de vista formal ele pode ser visto como uma linguagem de programação de alto nível.
- A maneira mais correta é defini-lo como um conjunto de macros com uma estrutura associada;
- Foi inventado na decada de 80 do século passado por Leslie Lamport e tem origem no TeX cujo pai foi Donald E. Knuth;
- Os principais utilizadores do LaTeX são: matemáticos, engenheiros, filosófos, advogados, linguistas, economistas, investigadores e académicos em geral.

Porquê utilizar LaTeX?

Algumas vantagens:

- disponível gratuitamente e para quase todos os sistemas operativos;
- cross-plattform;
- qualidade tipográfica profissional;
- facilidade no manuseamento de entes matemáticos;
- a estruturação dos documentos segue um critério funcional ao invés de um critério físico;
- estabilidade (a versão publicada em 1994 permanece quase inalterada até aos dias de hoje);

Porquê utilizar LaTeX?

Algumas desvantagens:

- o tempo necessário à aprendizagem é severamente superior ao que se necessita para aprender a utilizar um processador de texto tipo *Word*;
- não existe um suporte técnico (se bem que isto não constitui um problema real);
- qualidade tipográfica profissional;
- necessidade de um conhecimento específico acerca da natureza do LaTeX para criar estruturas radicalmente diferentes das usuais (problema que raramente se coloca).

Filosofia do processamento de texto

Existem várias filosofias sobre o processamento e a edição de texto. Pretende-se apenas abordar os dois principais paradigmas antagónicos, uma vez que grande parte dos utilizadores de processadores de texto se concentra entre estes dois. Temos então os seguintes pontos de vista para a edição de texto:

- **"What you see is what you get"**;
- **"What you see is what you mean"**.

O que é necessário para utilizar LaTeX?

Para utilizar LaTeX necessitamos de um compilador que podemos instalar de maneira diferente consoante o sistema operativo que estejamos a utilizar. Vejamos os casos dos três grandes OS da atualidade:

- **Windows:** Pode-se utilizar uma distribuição como o MikTeX;
- **Linux:** Normalmente os sistemas Linux trazem o LaTeX por default, caso contrário a instalação é muito fácil através de algum gestor de pacotes como o *Synaptics* (sistemas GNOME) ou análogos noutros sistemas;
- **Mac:** Existem distribuições análogas ao caso Windows, sendo um exemplo, o MacTeX.

Alguns editores

- Produzir um ficheiro TEX é uma tarefa pouco exigente em termos de aplicações necessárias que devemos ter instaladas no computador;
- Tudo o que necessitamos é um editor de texto simples, por exemplo, *GEdit* em Linux, *Notepad* em Windows e *TextEdit* em Mac;
- Contudo, sendo o LATEX um conjunto muito variado de macros, é extremamente útil poder dispor de um programa que nos ajude na escrita do código-fonte (impede que sejamos obrigados a saber muita coisa de memória).

Alguns editores

Existem diversos editores de LaTeX que se ajustam, consoante a sua especificidade, mais ou menos bem a todos os tipos de utilizadores. Vejamos alguns exemplos:

- Texmaker
- TexnicCenter
- LyX

Considerações

Devemos ter em atenção o comportamento do compilador quando editamos texto. Para isso convém ter em mente, entre outros, o seu funcionamento nos seguintes aspetos:

- Carateres brancos;
- Carateres especiais: % $ _ # & ;
- Comandos LaTeX.

Comandos em LaTeX

- Em LaTeX os comandos começam sempre com uma contrabarra (*backslash*) e normalmente são seguidos de dois tipos de parêntesis, a saber, parêntesis retos e chavetas;
- O compilador é sensível a maiúsculas e minúsculas;
- Entre chavetas indicamos alguma instância da qual o comando depende, por exemplo, o comando $\backslash begin\{center\}$ permite-nos indicar que o segmento seguinte no texto deverá ser centrado. Qualquer comando *begin* tem de ser finalizado por um comando *end*;
- Entre parêntesis retos indicamos algum atributo do comando/pacote que estamos a utilizar por forma a modificar o comportamento global da função, por exemplo, o comando $\backslash usepackage\,[portuges]\,\{babel\}$ permite indicar que a língua em que escrevemos o documento é Português.

Estrutura fundamental de um ficheiro TEX

O compilador espera que o utilizador introduza o seu documento seguindo uma lógica predefinida que ele conhece bem. Assim, os documentos devem começar com a instrução:

```
\documentclass[]{}
```

- Entre chavetas devemos indicar o parâmetro obrigatório, ou seja, qual a classe que queremos utilizar para este documento;
- Entre parêntesis retos podemos colocar atributos que aproximem o comportamento do documento da melhor maneira ao que pretendemos fazer;
- Este comando abre a parte do documento a que usualmente chamamos *preâmbulo*.

Algumas classes de documentos

Como acabámos de ver a instrução *documentclass* é a primeira instrução que temos de definir nos nossos documentos. É assim fundamental conhecer os tipos de classes de documento que temos ao dispor.
Veja-se a tabela seguinte:

article	report
book	slides
proc	minimal

Nota: Mencionar os comandos: \tableofcontents, \listoffigures e \listoftables.

Opções das classes de documentos

Listam-se em seguida algumas opções das classes de documentos:

- **Tamanho:** *10pt* (default) ,*11pt*, *12pt*, etc;
- **Tipo de papel:** *a4paper* , *letterpaper* (default), *a5paper*, etc;
- **Impressão frente e verso:** *twoside* ou *oneside*;
- **Orientação:** *landscape*;
- **Documento em coluna dupla:** *twocolumns*;

Generalidades sobre o preâmbulo

O preâmbulo começa então com a instrução

`\documentclass[]{}`

e acaba com a instrução

`\begin{document}`

Esta instrução dá início à parte do documento a que chamamos *corpo*. O corpo do documento é terminado pelo comando

`\end{document}`

que deve ser a última instrução de todo o documento. Qualquer linha depois deste comando será terminantemente ignorada pelo compilador.

Generalidades sobre o preâmbulo

- O preâmbulo serve para declarar todos os pacotes que vamos utilizar no nosso documento. Devemos utilizar a instrução *usepackage* [opções] {*pacote*};
- Se utilizarmos um editor conveniente, a tarefa de declarar todos os pacotes de que necessitamos é simplificada através de *Wizards* que nos permitem inicializar rapidamente os nossos ficheiros segundo *templates* padrão ou personalizados que podemos criar;
- Ainda assim, importa conhecer alguns pacotes fundamentais que estão ao nosso dispor para produzir os nossos documentos.

Preâmbulo - Pacotes fundamentais

Para utilizar um pacote devemos utilizar a instrução *begin*{*pacote*} no corpo do documento após devida declaração do pacote no preâmbulo.
Vejamos alguns pacotes úteis:

- **inputenc** : permite indicar a codificação dos carateres. Para escrever em Português podemos utilizar como opções *utf8* e *latin1* (sendo a primeira a recomendada);
- **babel** : permite definir a língua na qual o documento está escrito (torna a hifenização automática);
- **geometry** : permite fazer alterações às configurações da página;
- **graphicx** : permite incluir objetos gráficos;
- **color** : permite modificar a cor do texto;
- **multicols** : permite escrever em várias colunas;

Preâmbulo - Pacotes matemáticos

Vejamos agora alguns pacotes especialmente adequados à edição de texto matemático:

- **amsmath** : fornece diversas macros úteis para a introdução de entes matemáticos;
- **amsfonts** : fornece suporte ao nível dos tipos de letra para texto matemático;
- **amssymb** : fornece suporte ao nível dos símbolos matemáticos;
- **listings** : permite a edição automática de algoritmos em diversas linguagens de programação.

Preâmbulo - Autor, Título e Data

O LaTeX tem uma função muito interessante que nos permite fazer automaticamente a folha de rosto de qualquer documento. A instrução para chamar a função é *maketitle* e para a sua correta utilização devemos fornecer no preâmbulo:

- O título: utilizando a instrução *title*{ };
- O nome do autor: utilizando a instrução *author*{ };
- A data: utilizando a instrução *date*{ }.

Podemos omitir um qualquer destes dados não escrevendo um argumento, por exemplo, *date*{} inibirá a data na folha de rosto.

Exemplo minimalista de um documento

```
\documentclass[a4paper,11pt,portuges]{article}
\usepackage[portuges]{babel}
\usepackage[utf8x]{inputenc}
\begin{document}
%Espero que o curso nao esteja a ser uma seca!
\end{document}
```

Ambientes em LaTeX

Abordaremos alguns ambientes úteis para a edição de texto, nomeadamente:

- **itemize**: útil em situações em que pretendemos elencar sem que haja uma ordem numérica associada;
- **enumerate**: útil em situações em que pretendemos elencar reforçando uma ordem numérica;
- **description**: útil para apresentar, por exemplo, um conjunto de definições;
- **verbatim**: permite transcrever (i.e. inibe a compilação);
- **tabular**: permite criar uma tabela
- **figure**: permite introduzir figuras e outros *corpo*

Ambientes em LaTeX

- **Posicionamento de parágrafos**:
  - **flushleft** : permite alinhar à esquerda
  - **flushright**: permite alinhar à direita
  - **center**: permite centrar
- **abstract**: permite fazer um resumo da publicação (especialmente útil em teses).

Perguntas:

1. Como se indica que pretendemos utilizar um destes ambientes?
2. Qual o ambiente utilizado para criar estas perguntas?

Exemplos: o ambiente *itemize*

```
\begin{itemize}
\item Item 1;
\item Item 2;
\item Item 3.
\end{itemize}
```

Exemplos: o ambiente *enumerate*

```
\begin{enumerate}
\item Item 1;
\item Item 2;
\item Item 3.
\end{enumerate}
```

Exemplos: o ambiente *figure*

```
\begin{figure}
\caption{Figura 1}
\includegraphics[scale=1]{fotografia.png}
\end{figure}
```

Exemplos: o ambiente *tabular*

Para fazer uma tabela em LaTeX temos de declarar o número de colunas e o alinhamento.
Vejamos:

```
\begin{tabular}{|c|c|c|}
\hline i & $a_i$ & $b_i$ \\
\hline 0 & $a_0$ & $b_0$ \\
\hline 1 & $a_1$ & $b_1$ \\
\hline 2 & $a_2$ & $b_2$ \\
\hline 3 & $a_3$ & $b_3$ \\
\hline 4 & $a_4$ & $b_4$ \\
\end{tabular}
```

Escrita de fórmulas matemáticas

A escrita de fórmulas matemáticas é, sem dúvida, um dos pontos fortes do LaTeX e é também uma das características que nos faz estar tão interessados nele. Vejamos como se introduz uma fórmula F:

- **dentro de um parágrafo**:
  - \$ F \$
  - \(F \ )
  - \begin{math} F \end{math}
- **fazendo uma quebra**:
  - \$ \$ F \$ \$
  - \[F \]
  - \begin{displaymath} F \end{displaymath}

Podemos utilizar ainda o ambiente *equation* para produzir equações sequencialmente ordenadas

Exemplos de fórmulas matemáticas

```
$ f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty}
{a_n \cdot \cos{\left( \frac{n \pi t}{L} \right)}
+b_n \cdot \sin{\left( \frac{n \pi t }{L} \right)}
} $

\begin{displaymath}
f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty}
{a_n \cdot \cos{\left( \frac{n \pi t}{L} \right)}
+ b_n \cdot \sin{\left( \frac{n \pi t }{L} \right)}
\end{displaymath}

\begin{equation}
x^n+y^n+z^n=2 \, \forall n \in \mathbb{N}
\end{equation}
```

Exemplos de fórmulas matemáticas

Vejamos agora alguns exemplos de fórmulas matemáticas:

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cdot \cos\left(\frac{n\pi t}{L}\right) + b_n \cdot \sin\left(\frac{n\pi t}{L}\right)$$

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cdot \cos\left(\frac{n\pi t}{L}\right) + b_n \cdot \sin\left(\frac{n\pi t}{L}\right)$$

$$x^n + y^n + z^n = 2 \forall n \in \mathbb{N} \tag{1}$$

Considerações sobre o modo matemático

- No modo matemático o espaçamento é feito automaticamente de acordo com a expressão, obedecendo assim apenas a critérios lógicos. Todo o espaçamento extra que pretendamos tem de ser declarado com expressões como $\backslash quad$, $\backslash qquad$ ou ainda $\backslash,$.
- Cada letra é considerada uma variável, assim se pretendermos introduzir texto a meio de uma fórmula temos de recorrer ao comando $\backslash textrm\{\ \}$.
- A generalidade dos comandos no modo matemático afeta apenas o carater imediatamente a seguir à instrução, utilizam-se *chavetas* para corrigir esta situação.

Alguns comandos clássicos

- **Letras gregas:** são introduzidas geralmente como $\backslash$nome da letra;
- **Expoentes:** utiliza-se o acento de circunflexo;
- **Índices:** utiliza-se o *underscore*;
- **Raízes:** introduz-se como $\backslash\mathrm{sqrt}\,[\mathrm{indice}]\,\{n\}$
- **Barras superiores e inferiores:** utilizam-se os comandos $\backslash\mathrm{overline}$ e $\backslash\mathrm{underline}$;
- **Chavetas horizontais:** utilizam-se os comandos $\backslash\mathrm{overbrace}$ e $\backslash\mathrm{underbrace}$ para produzir chavetas superiores e inferiores respetivamente.
- **Sinais de multiplicação:** utilizam-se os comandos $\backslash\mathrm{cdot}$ para produzir o ponto centrado e $\backslash\mathrm{times}$ para o sinal da escola primária.

Exemplos

- $\alpha, \beta, \gamma, \theta$
- $x^y, 2^3, \alpha^\theta, 2^{2^{2^{2^{2^2}}}}$
- $x_t, x_{t_j}, \alpha_\lambda$
- $\underbrace{x + y + z + 2}_{A}, \quad \overbrace{14 + x^2 + t_j}^{R}$
- $2 \cdot 3 = 6, \quad 5 \times 10^{-t}$

Algumas comandos comummente utilizados

- \arccos
- \arcsin
- \arctan
- \arg
- \sinh
- \sec
- \cos
- \cosh
- \cot
- \coth
- \sup
- \sin
- \csc
- \deg
- \det
- \dim
- \tan
- \exp
- \gcd
- \hom
- \inf
- \tanh
- \ker
- \lg
- \lim
- \liminf
- \min
- \limsup
- \ln
- \log
- \max
- \Pr
- \bmod
- \int
- \sum
- \frac

Ambientes matriciais

- **matrix:** cria uma matriz sem parêntesis;
- **pmatrix:** cria uma matriz com parêntesis curvos;
- **bmatrix:** cria uma matriz com parêntesis ;
- **vmatrix:** cria uma matriz com linhas verticais simples;
- **Vmatrix:** cria uma matriz com linhas verticais duplas.

Nota: Para separadores e quebras usam-se os mesmos símbolos que reservamos para o ambiente *tabular* já introduzido.

Exemplo de uma matriz

```
$ H_3 =
\begin{pmatrix}
1 &  \frac{1}{2} & \frac{1}{3} \\
\frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\
\frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\
\end{pmatrix} $
```

Exemplo de uma matriz

$$H_3 = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{pmatrix}$$

A classe *Beamer*

- A classe *Beamer* permite-nos fazer apresentações de diapositivos;
- Para a utilizar devemos inicializar o documento com $\backslash\mathrm{documentclass}\{\mathrm{beamer}\}$;
- Para criar um slide utilizamos o ambiente *frame*;
- A função *maketitle* vista anteriormente continua disponível para esta classe;
- Existem diversos temas que podemos utilizar;
- Muitas das caraterísticas que abordámos no contexto da classe *Article* continuam válidas nesta classe.

Exemplo de um documento *Beamer*

```
\documentclass{beamer}
\usepackage[portuges]{babel}
\usepackage[utf8]{inputenc}
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{amssymb}
\usepackage{color}
\begin{document}
%\begin{frame} Forma extensa
\frametitle{Diapositivo 1}
Diapositivo 1
%\end{frame}
\frame{ %Forma curta
\frametitle{Diapositivo 2}
Conteudo
}
\end{document}
```

Estilos do Beamer

- Existem inúmeros estilos criados do Beamer dos quais podemos tirar imenso partido;
- Veja-se o seguinte link: Beamer Theme Gallery ;
- Depois de escolhido um estilo, para o utilizar tudo o que temos de fazer é incluir a instrução $\backslash\text{usetheme}\{\text{nome do tema}\}$ no preâmbulo;
- A menos que o utilizador procure algo verdadeiramente específico, os *default* do LaTeXneste campo são bastante bons.

O pacote *Listings*

Para finalizar o curso falo de um pacote que considero muito útil. Este pacote permite a escrita de algoritmos em diversas linguagens de uma forma muito sugestiva (indentação automática, realce das palavras-chave da linguagem, etc.)

- Para utilizarmos o pacote temos de o declarar no preâmbulo com a instrução $\backslash\text{usepackage}\{\text{listings}\}$;
- No corpo do documento para começarmos um ambiente do *listings* devemos utilizar a instrução $\backslash\text{begin}\{\text{lstlisting}\}$;
- Se utilizarmos apenas uma linguagem ao longo de todo o texto podemos declará-la no início do documento logo após a instrução $\backslash\text{begin}\{\text{document}\}$ através do comando $\backslash\text{lstset}\{\text{language}\}$;
- Entre as linguagens suportadas estão, por exemplo, TEX, *Scilab*, *Python*, *Mathematica*, *R*, *C++*, *C*, *SQL*, *Pascal*, *Fortran* e *HTML*.

Webgrafia

Algumas páginas de internet utilizadas para a produção deste tutorial:

- **Folha de Exercícios:** http://web.mit.edu/rsi/www/pdfs/latex-assignment.pdf
- **Manual de LaTeX:** http://www.ctan.org/tex-archive/info/lshort/portuguese/ptlshort.pdf

Referências

Indico alguns links que considero úteis para aqueles que pretendam aprender LaTeX:

- **Página do projeto:** http://www.latex-project.org/ ;
- **Manual LaTeX:** tobi.oetiker.ch/lshort/lshort.pdf (versão estendida em Inglês) ;
- **Manual LaTeX:** http://www.ctan.org/tex-archive/info/lshort/portuguese/ptlshort.pdf (versão estendida em Português).