

Considerações sobre o $\text{\LaTeX}2_\epsilon$

António Batel Anjo e Delfim F. Marado Torres

Departamento de Matemática
Universidade de Aveiro
{batel, delfim}@mat.ua.pt

1 Antes de mais, o que é o \TeX ?

O \TeX é um programa de computador, criado por Donald Knuth em finais dos anos 70, projectado para trabalho tipográfico, nomeadamente para a escrita de documentos com fórmulas matemáticas. Trata-se pois de um programa dirigido aos autores e desde sempre foi muito popular entre os matemáticos, físicos, astrónomos, informáticos e, de um modo geral, entre os cientistas.

O \TeX , tal como o usamos hoje, foi tornado público em 1982 e é considerado como uma das ferramentas de software mais estáveis do mundo. Está neste momento na versão 3.14159 e será perfeito quando atingir o valor numérico de π .

O \TeX tem-se desenvolvido no mundo académico, graças aos esforços colectivos de centenas de programadores de todo o mundo, e encontra-se disponível, gratuitamente, em quase todas as plataformas e sistemas operativos.

2 Um passo em frente: o \LaTeX

O \LaTeX é um acrescento ao \TeX , que proporciona um sistema de preparação de documentos com uma excelente qualidade tipográfica, usando minutas profissionais pré-definidas.

Foi criado, originalmente, por Les-

lie Lamport e usa, tal como dissemos, o \TeX como base. A versão mais recente do \LaTeX é o $\text{\LaTeX}2_\epsilon$ e será sobre esta versão que nos iremos aqui debruçar.

3 Um pouco de filosofia

Autores pouco experientes, cometem erros de formatação quando assumem que o design, por exemplo de um livro, é principalmente uma questão de estética: “Se um documento tem ‘bom aspecto’, então está bem concebido”. Mas como os documentos devem ser feitos para serem lidos e não para estarem em galerias de arte, a facilidade de leitura e a facilidade de compreensão são factores muito mais importantes do que o facto de serem bonitos.

Nos populares ambientes WY-SIWYG — “*What You See Is What You Get*” — normalmente são os autores que tratam de todo o design. O resultado são trabalhos ‘muito bonitos’ mas com uma estrutura pobre.

O \LaTeX previne este tipo de erros, forçando o utilizador a declarar as estruturas lógicas do texto. O \LaTeX encarregar-se-á, depois, do resto.

Falamos então de uma filosofia WY-SIWYM — “*What You See Is What You Mean*” — em oposição ao WYSIWYG.

Para escrever em \LaTeX usamos simples ficheiros ASCII, criados por um qualquer editor, contendo o texto do documento propriamente dito e certos

comandos que irão servir para o \LaTeX produzir um ficheiro de extensão `.dvi`. As distribuições de \LaTeX disponibilizam programas para visualizar e imprimir estes ficheiros, assim como uma série de conversores para outros formatos, como por exemplo o PostScript.

4 Porque devo aprender \LaTeX ?

As vantagens do \LaTeX tornam-se evidentes quando os nossos documentos se tornam grandes. Uma tese, de mestrado ou doutoramento, em \LaTeX cabe perfeitamente numa disquete. Uma colega nossa, precisava de 40 disquetes para guardar a sua dissertação de mestrado escrita em Word.

O \LaTeX é também muito vantajoso quando mudamos, constantemente, os nossos ficheiros. No \LaTeX , índices, citações, colocação de figuras, são tratados automaticamente.

Outra desvantagem do Word é produzir documentos – de extensão `.doc` – não portáteis. Quem ainda não teve problemas em ler um ficheiro de Macintosh em PC ou vice versa? Quem já tentou abrir um documento de Word entre diferentes versões de Word? Vejam o último exemplo: o Word 97 não é capaz de ler os ficheiros do Word 2000. Por outro lado, um documento criado em \LaTeX , há dez anos atrás, ainda pode ser publicado, sem qualquer tipo de edição, com os sistemas actuais de \LaTeX .

Por último, a qualidade tipográfica dos documentos produzidos pelo \LaTeX é maior do que a que se obtém, por exemplo, com o Word. Claro que pudemos sempre mudar os “templates” no Word, mas então teremos trabalho para mais uns dias ...

Não vos queremos esconder nada:

embora o \LaTeX seja, em geral, fácil de usar, tem também as suas dificuldades; e exige um tempo de aprendizagem maior que o Word. No entanto, o esforço será compensado.

Se ainda não o conseguimos convencer das vantagens do \LaTeX , não continue. Dê primeiro uma vista de olhos em:

<http://www.cudenver.edu/~hgreenbe/courses/texinfo/wordvslatex.html>

5 Generalidades sobre os ficheiros \LaTeX

5.1 Espaços

Caracteres de espaço, como sendo o “espaço branco” ou o `tab`, são tratados todos como “espaço” pelo \LaTeX . Vários espaços consecutivos são tratados como um “espaço”. Um carácter de espaço no início duma frase é ignorado e a mudança de uma linha é tratada como um “espaço”. Uma linha vazia entre duas linhas de texto define fim de parágrafo. *Várias* linhas vazias entre duas linhas de texto definem, na mesma, um parágrafo.

5.2 Caracteres Especiais

Os símbolos seguintes são caracteres reservados, que ou têm um significado especial para o \LaTeX ou não estão disponíveis em todas as fontes. Se os introduzir directamente no texto, os caracteres não irão aparecer na impressão. São eles:

`$ & % # _ { } \`

Como poderá constatar, a maioria destes símbolos poderão ser produzidos se antes do símbolo introduzir uma barra (`\`). No caso de querer introduzir o

carácter `\` no seu texto, não poderá fazer `\\`, pois este comando é usado para mudança de linha. Para produzir este carácter terá de recorrer ao comando

```
$$\backslashslash$
```

5.3 Comandos

Os comandos de \LaTeX são “*case sensitive*”, isto é, diferem se estiverem com letras maiúsculas ou minúsculas. Podem ter dois formatos possíveis: ou começam com uma barra `\` e têm de seguida um nome constituído apenas por letras — aí os comandos terminam com um espaço, um número ou outro carácter ‘não-letra’ — ou consiste numa barra `\` e exactamente um carácter especial.

O \LaTeX ignora os espaços depois dos comandos. Se quisermos obter um espaço após um comando, teremos de colocar `\` seguido de espaço após o comando, ou então `{ }` seguido também de espaço após o referido comando.

5.4 Comentários

A partir do carácter `%`, o \LaTeX ignora o resto da presente linha, a quebra de linha, e todos os tipos de espaço no início da linha seguinte. Sendo assim, este carácter serve para introduzir no ficheiro comentários que depois na impressão não aparecem.

6 A estrutura dos ficheiros \LaTeX

6.1 O preâmbulo

Quando o \LaTeX_ϵ processa um ficheiro, ele espera que esse ficheiro siga uma certa estrutura. Todos os ficheiros devem começar com um comando do tipo

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

Com efeito, a primeira informação que o \LaTeX precisa de saber quando está a processar um ficheiro, é o tipo de documento que o autor quer criar. Alguns tipos de documentos são:

article para artigos em jornais científicos, pequenos relatórios, ...

report para relatórios longos contendo vários capítulos, pequenos livros, teses de Doutoramento e dissertações de Mestrado, ...

book para livros completos

slides para slides e acetatos. Esta classe usa letras grandes *sans serif*.

O \LaTeX_ϵ inclui outras classes, por exemplo para escrita de cartas.

Além disso, podemos especificar, entre parêntesis rectos, uma série de opções que nos permitem personalizar o comportamento do documento. Essas opções são separadas por vírgulas. As mais comuns são:

10pt, 11pt, 12pt Indicam o tamanho da fonte principal no documento. Por defeito, os documentos são 10pt.

a4paper, letterpaper, a5paper, b5paper Definem o tamanho do papel onde o documento será impresso. O tamanho por defeito é o letterpaper.

fleqn Coloca as fórmulas matemáticas alinhadas à esquerda em vez de centradas.

leqno Coloca a numeração das fórmulas matemáticas à esquerda destas, em vez de estarem à direita.

titlepage, notitlepage Especifica se uma nova página deve ser usada, ou não, após o título do documento. A classe `article` não começa, por defeito, uma nova página enquanto as classes `report` e `book` o fazem.

twocolumn faz o \LaTeX produzir o texto em duas colunas.

twoside, oneside Controla as margens da página adequadamente, consoante quisermos imprimir o texto apenas num ou nos dois lados da folha. O `article` e o `report` usam o `oneside` por defeito enquanto o `book` usa o `twoside`. Atenção: o comando `twoside` não indica à impressora que deve imprimir nos dois lados da folha.

openany, openright Faz com que os capítulos comecem em qualquer página ou apenas em página ímpar, do lado direito. Estas opções não funcionam no modo `article`, pois este modo não reconhece capítulos. O modo `report` usa por defeito o `openany` enquanto o modo `book` usa por defeito o `openright`.

Exemplo: um ficheiro em \LaTeX pode começar com a linha

```
\documentclass[11pt,twoside,
                a4paper]{article}
```

indicando que pretendemos escrever um artigo com o tamanho da fonte base de *onze pontos* e que pretendemos imprimi-lo dos dois *lados* em páginas *A4*.

De seguida, podemos:

- Incluir comandos que *influenciam o estilo de todo o documento*. Por exemplo:

alterar o espaçamento normal entre linhas com o comando

```
\renewcommand{%
  \baselinestretch}{...}
```

onde `...` deverá ser substituído por um valor numérico. Alguns valores constantes da tabela seguinte.

Tamanho letra	Espaçamento duplo	Espaçamento 1.5
10pt	1.67	1.25
11pt	1.62	1.21
12pt	1.66	1.24

- *Definir novos comandos*. Por exemplo podemos definir os comandos \mathbb{R} e \mathbb{N} por

```
\def\mathbb{R}{\rm I\kern-.17em R}}
\def\mathbb{N}{\rm I\kern-.20em N}}
```

para produzir, respectivamente, os símbolos \mathbb{R} e \mathbb{N} . (Um efeito semelhante pode ser conseguido pela inclusão do

“*package*” `amssymb`. Ver comentário à frente.)

- *Carregar “packages”* que acrescentam novas opções ao sistema \LaTeX . Para carregar um “package” usa-se o comando

```
\usepackage[...]{...}
```

Um exemplo pode ser

```
\usepackage[portuges]{babel}
```

quando pretendemos escrever documentos em língua portuguesa. Desta maneira iremos ter *Capítulo* em vez de *Chapter*, etc. Podemos também usar o *package* `babel` para escrever documentos multi-língua. Por exemplo

```
\usepackage[portuges,
english]{babel}
```

Fazemos a comutação da língua, conforme necessário, com o comando

```
\selectlanguage{...}
```

Deste modo podemos ter diferentes regras de hifenização. Um outro *package* útil é o

```
\usepackage{isolatin1}
```

que nos permite introduzir palavras acentuadas, de imediato, sem o uso de quaisquer comandos para o efeito. Para escrita de documentos matemáticos, poderá dar jeito incluir:

```
\usepackage{amssymb}
```

Poderemos então, a título de exemplo, produzir os símbolos \mathbb{R} e \mathbb{N} , respectivamente com os comandos `$$\mathbb{R}$$` e `$$\mathbb{N}$$`.

Existem milhares de *packages*: potencialmente para tudo o que precisamos e viremos a precisar. A este respeito, a referência certa é sem dúvida o livro [1]. Se a sua distribuição de L^AT_EX não disponibilizar um determinado *package* que pretende usar, poderá sempre encontrá-lo na internet nos servidores CTAN (ver secção 8 mais à frente).

Aqui vamos apenas indicar mais dois *packages*: um para inclusão de imagens e outro para produzir um índice.

Para inclusão de gráficos no formato `.eps` pode usar o `graphicx`

```
\usepackage{graphicx}
```

que lhe permite depois incluir as imagens com o comando

```
\includegraphics[Largura,Altura,
Angulo,Escala]{Imagem.eps}
```

Não se esqueça que tudo o que vem entre parêntesis rectos é sempre opcional.

Para geração de índices deve incluir os comandos

```
\usepackage{makeidx}
\makeindex
```

e depois, no corpo do texto e na altura que desejar, escrever

```
\tableofcontents
```

6.2 O corpo

Estamos nesta altura prontos para começar a trabalhar no texto propriamente dito. Um exemplo:

```
\begin{document}
Um texto muito curto!
\end{document}
```

Enquanto a área entre o `\documentclass` e o `\begin{document}` é chamada de *preâmbulo*; chamamos *corpo do texto* ao compreendido entre os comandos `\begin{document}` e `\end{document}`.

Tudo o que vier a seguir ao comando `\end{document}` é pura e simplesmente ignorado pelo L^AT_EX.

6.2.1 Alguns Exemplos de Escrita

O texto que se segue, é um documento independente que se inicia da seguinte forma:

```
\documentclass[11pt]{article}
\usepackage{isolatin1}
\begin{document}
\title{A arte de bem
      excrever \LaTeX}
\author{Delfim e António}
\date{\today}
\maketitle
```

A primeira instrução define o tipo e o tamanho da letra que vamos usar.

```
\documentclass[11pt]{article}
```

A instrução

```
\usepackage{isolatin1}
```

é o exemplo de como chamar um arquivo utilitário: a codificação de sinais diacríticos varia de computador para computador; o arquivo

```
isolatin1.sty
```

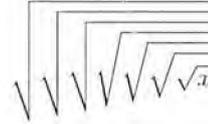
efectua a descodificação. Segue-se o título do trabalho, o autor e a data. Aqui acaba o preâmbulo do documento e começa o texto, sempre depois de um

```
\begin{document}
```

Uma expressão matemática é escrita, e destacada, no texto da seguinte forma

```
$$
\sqrt{
  \sqrt{
    \sqrt{
      \sqrt{
        \sqrt{
          \sqrt{x}}
        }
      }
    }
  }
}
$$
```

que produz o seguinte resultado



Pode no entanto ser conveniente reter essa expressão matemática para referência posterior – ou anterior ...

```
\begin{equation}
\sqrt{\sqrt{\sqrt{\sqrt{x}}}}
\label{raiz}
\end{equation}
```

Desta forma podemos referir-nos à expressão (6.1) em qualquer parte do texto,

$$\sqrt{\sqrt{\sqrt{\sqrt{x}}}} \quad (6.1)$$

usando o comando do L^AT_EX

```
\ref{raiz}
```

A escrita de expressões mais elaboradas como por exemplo o teorema 6.1, é conseguida à custa de alguma novas definições.

Teorema 6.1 *Se n é um inteiro positivo, então $D_x(x^{-n}) = -nx^{-n-1}$.*

Demonstração: Utilizando a definição de x^{-n} , a regra do cociente e a regra da potência,

$$\begin{aligned} D_x(x^{-n}) &= D_x\left(\frac{1}{x^n}\right) \\ &= \frac{x^n(0) - 1(nx^{n-1})}{(x^n)^2} \\ &= \frac{-nx^{n-1}}{x^{2n}} = (-n)x^{(n-1)-2n} \\ &= (-n)x^{-n-1} \end{aligned} \quad (6.2)$$

◊

Vejamos como se produz este texto

```
\begin{teor}
Se $n$ é um inteiro positivo,
então

$$D_x(x^{-n}) = -n x^{-n-1}.$$

\label{teorema}
\end{teor}
```

```
\begin{demo}
Utilizando a definição de
 $x^{-n}$ , a regra do cociente
e a regra da potência,
```

```
\begin{eqnarray}
D_x(x^{-n}) & \&= &
D_x(\frac{1}{x^n}) \\
\&= & \frac{x^n(0) - 1(nx^{n-1})}{(x^n)^2} \\
\&= & \frac{-nx^{n-1}}{(x^n)^2} = \\
& (-n) x^{(n-1)-2n} \\
\&= & (-n)x^{-n-1}
\end{eqnarray}
```

```
\end{demo}
```

Os ambientes *teorema* e *demonstração*, colocados no preâmbulo do documento, são definidos da seguinte forma:

```
\newtheorem{teor}{%
  {Teorema}[section]}

\newenvironment{demo}{%
  \endsloppypar\noindent
  \bgroup\small
  {\bf{Demonstração:}} }
  {\samepage\null\hfill
  $\diamond$\endsloppypar\egroup}
```

No próximo número, continuaremos a tratar de escrita de documentos em \LaTeX .

7 Breves pensamentos

Será que nos deveríamos ver obrigados a alinhar na infundável corrida de

atualizações de hardware e software — que tão bem caracteriza a indústria informática dos dias de hoje — se o que pretendemos é escrever textos de qualidade profissional?

A manipulação de ficheiros ASCII (e o uso do próprio \LaTeX) pode bem ser feito com aquele 'velhinho' 486 que tem para aí num canto ...

Não é verdade que as técnicas de escrita, por exemplo de um curriculum vitae, pouco ou nada se alteraram nos últimos 10 anos? Então porque será que a produção de documentos tem sido o pretexto de compra de programas e computadores cada vez mais recentes?

Os ficheiros ASCII, por seu lado, garantem também a portabilidade dos documentos entre sistemas operativos e máquinas diferentes!

8 Onde encontrar uma distribuição livre de \LaTeX ?

O que se segue é a nossa opinião, obviamente limitada aos nossos conhecimentos, de quais as 'melhores' distribuições de \LaTeX disponíveis em regime livre, na internet, para a plataforma que usa.

Qualquer uma delas pode ser encontrada num dos servidores CTAN: por exemplo <ftp.tex.ac.uk>, <ftp.dante.de> ou <ctan.tug.org>, a partir da directoria </tex-archive/systems/>.

Linux: *teTeX*, disponível em <unix/teTeX>. Já vem, por exemplo, com o RedHat Linux.

MS-DOS: *emTeX*, disponível em <msdos/emtex>.

Win32: (Win95, Win98 e NT) *MikTeX*, disponível em win32/miktex.

Power PC: *MikTeX*, disponível em win32/miktexppc.

Mac: *OzTeX*, disponível em mac/oztex. Deve correr em qualquer Macintosh Plus, SE, II, ou modelos mais novos. Não trabalha em Macs com apenas 128K ou 512K.

Existem também versões não comerciais para outras plataformas, tais como *Ataris*, *Amigas*, etc. Não há mesmo desculpa para não produzir documentos de qualidade profissional!

9 Para saber mais

O local ideal de partida, diríamos mesmo um sítio obrigatório para todos os interessados em *L^AT_EX*, é a página do 'T_EXUsers Group':

<http://www.tug.org>

Referências

- [1] M. Goossens, F. Mittelbach and A. Samarin, *The L^AT_EX Companion*, Addison-Wesley, 1994.
- [2] H. J. Greenberg, *A Simplified Introduction to L^AT_EX*, July 21, 1999, <http://www.cudenver.edu/~hgreenbe/>.
- [3] T. Oetiker, H. Partl, I. Hyna and E. Schlegl, *The Not So Short introduction to L^AT_EX₂_ε*, Version 3.2, September 21, 1998, CTAN:/tex-archive/info/lshort.

Aí poderá encontrar: respostas às perguntas mais frequentes; informação sobre como instalar o *L^AT_EX* no seu computador; livros a ler; tutoriais disponíveis na rede; etc, etc.

Na remota possibilidade de não encontrar a informação que precisa, peça ajuda aos outros utilizadores. A comunidade *L^AT_EX*iana é muito prestável. Recomendamos o fórum de notícias e discussão `comp.text.tex`.

Agradecimentos

A escrita deste documento despoletou o interesse de Marcos Daniel, aluno do ensino secundário, para o *L^AT_EX*. Queremos agradecer-lhe o facto de nos ter mostrado que mesmo para um jovem de 16 anos, 'educado nas filosofias visuais', a aprendizagem de um sistema como o *L^AT_EX*, orientado para o conteúdo e não para o aspecto do texto na página, não é difícil e torna-se mesmo produtiva num abrir e fechar de olhos. Continua!