

UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA ESPECIAL DE TREINAMENTO
CIÊNCIAS DA COMPUTAÇÃO

INTRODUÇÃO AO L^AT_EX 2_ε

Lucas Wanner

Versão 0.666
Florianópolis, Junho de 2002

Prefácio

T_EX (Pronuncia-se “tek”) é um sistema de diagramação de textos criado no final da década de 1970 por Donald Knuth. De acordo com o próprio autor, “(*T_EX is*) a typesetting system intended for the creation of beautiful books and specially for books that contain a lot of mathematics.”

T_EX oferece um grande controle sobre a estrutura do documento a ser editado e, justamente por isso, traz uma complexidade na edição indesejável à maioria dos usuários comuns.

L^AT_EX é um conjunto de macros escritas para o T_EX que permite a diagramação de textos na mais alta qualidade tipográfica, usando um *layout* profissional pré-definido. Foi elaborado em meados da década de 1980 por Laslie Lamport, e desde então tornou-se muito popular na diagramação de livros, artigos, *journals* e documentos científicos e matemáticos em geral.

Este manual, que é uma adaptação de *The Not So Short Introduction to L^AT_EX 2_ε* [1], é destinado àqueles que desejam iniciar sua jornada pelo mundo do L^AT_EX, procurando uma solução de melhor qualidade visual, ou uma alternativa prática e elegante aos editores WYSIWYG¹ ou mais especificamente ao MS-Worst.

Por ser direcionado a iniciantes (ou será devido ao limitado conhecimento do autor?), este manual centralizará seus esforços nos conceitos básicos e em exemplos, não cobrindo tópicos avançados. Aqueles que desejarem aprofundar seus conhecimentos devem procurar os livros e manuais citados na bibliografia (a maioria deles está disponível gratuitamente em formato eletrônico na web).

Capítulo 1 oferece uma comparação entre o sistema L^AT_EX e os sistemas WYSIWYG, além de mostrar a estrutura básica de um documento L^AT_EX 2_ε², explicando o processo de “compilação” deste tipo de documento.

Capítulo 2 apresenta o restante dos conceitos necessários para editar um texto no *mundo real*.

Capítulo 3 mostra algumas características especiais, como inclusão de imagens e geração de bibliografia, além de mostrar algumas dicas de como alterar o visual *padrão* dos documentos L^AT_EX

Apêndice A trata da instalação e configuração de um sistema L^AT_EX em um PC rodando Windows.

¹What You See Is What You Get, O que você vê na tela é o que será impresso.

²L^AT_EX 2_ε é o nome da última atualização do sistema L^AT_EX, atualmente mantido pelo grupo L^AT_EX3.

Sumário

Prefácio	1
1 Introdução	4
1.1 L ^A T _E X <i>versus</i> WYSIWYG	4
1.2 Arquivos de Entrada L ^A T _E X	5
1.2.1 Espaços	5
1.2.2 Caracteres Especiais	5
1.2.3 Comandos	5
1.2.4 Comentários	6
1.3 Estrutura dos Arquivos de Entrada	6
1.4 Compilação	7
1.5 Layout	7
1.5.1 Classes de Documentos	7
1.5.2 Estilos de Página	8
1.5.3 Pacotes	9
1.6 Arquivos	9
1.7 Divisão de Projetos	10
2 Edição de Texto	11
2.1 Quebras de Linha e Página	11
2.2 Caracteres Especiais e Símbolos	11
2.2.1 Aspas	11
2.2.2 Traços e Hífens	11
2.2.3 Til (~)	12
2.2.4 Reticências	12
2.3 Edição em Português	12
2.3.1 Acentos	12
2.4 Títulos, Capítulos e Seções	13
2.5 Referências Cruzadas	14
2.6 Notas de Rodapé	14
2.7 Enfatização	14
2.8 Ambientes	15
2.8.1 Itemizar, Enumerar e Descrição	15
2.8.2 Alinhamento	15
2.8.3 Citações e Versos	16
2.8.4 Verbatim	17
2.8.5 Tabular	17
2.9 Corpos Flutuantes	18

3	Especialidades e Customização	20
3.1	Geração de Arquivos PDF	20
3.2	Inclusão de Imagens	21
3.3	Bibliografia	22
3.3.1	Fontes e Tamanhos	23
3.4	Espaçamento	23
3.5	Layout de Página	24
	Apêndice	25
A	Instalação e Configuração do Sistema MiKTeX	25
A.1	Download e Instalação	25
A.2	Configuração	26
A.3	WinEdt	26
	Referências Bibliográficas	27

Capítulo 1

Introdução

1.1 \LaTeX *versus* WYSIWYG

Em um sistema WYSIWYG, como o MS-Worst ou o Corel Word Perfect, o autor define o *layout* do documento através de comandos interativos enquanto digita o texto na tela. O resultado final da formatação é visualizado durante todo o processo. O que é visto na tela é o que será impresso. Os comandos para modificar o *layout* são bastante intuitivos de compreensão fácil. Por que alguém iria querer algo diferente?

Normalmente os documentos editados com WYSIWYG são verdadeiras obras de arte. Vários tamanhos e faces de fontes diferentes, tamanhos de página não-ortodoxos (“Se eu aumentar um pouco o tamanho da página e diminuir o tamanho da fonte, esta seção vai caber em uma página...”), e não raramente cores, muitas cores. Obviamente isto não é uma regra absoluta, mas neste tipo de editor, parece ser muito mais fácil fazer a coisa errada. Mesmo aqueles poucos que tem um senso estético apurado são castigados pela (falta de) filosofia destes editores.

Em \LaTeX , o autor não se preocupa com a formatação durante a digitação do texto. O importante é a *estrutura lógica*, e esta é talvez a grande força deste sistema, já que força o autor a criar documentos bem estruturados. Uma vez definidos os capítulos, seções, subseções, citações, fórmulas, bibliografia, etc., entra em cena o formatador, que escolhe, a partir de algumas diretivas, o melhor *layout* para o documento. O resultado final é um documento com aparência profissional e logicamente consistente.

Para efeito de comparação, seguem algumas vantagens de \LaTeX sobre WYSIWYG (e o oposto). Estas e mais algumas podem ser encontradas em [1].

Vantagens:

- A inclusão de fórmulas matemáticas é suportada de maneira conveniente.
- O usuário apenas tem de aprender alguns comandos fáceis de entender que especificam a estrutura do documento. Quase nunca é necessário pensar no *layout* final do documento.
- Estruturas complexas como notas de rodapé, referências, sumários e bibliografias são geradas facilmente.

Desvantagens:

- Aparentemente \LaTeX não funciona bem para pessoas que venderam suas almas.
- Apesar de ser possível ajustar parâmetros de um *layout* pré-definido, a construção de um novo *layout* é difícil e toma muito tempo.
- É muito difícil escrever documentos sem estrutura e desorganizados.

1.2 Arquivos de Entrada \LaTeX

Os arquivos de entrada para o \LaTeX são arquivos de texto ASCII plano, normalmente com a extensão `.tex`. Estes arquivos contém o texto do documento bem como instruções para a formatação do mesmo.

1.2.1 Espaços

Caracteres de espaçamento como espaço e tabulação são tratados igualmente como “espaço em branco” pelo \LaTeX . Vários espaços consecutivos são tratados como somente um espaço.

Uma linha em branco indica o início de um novo parágrafo. Várias linhas em branco são tratadas como uma linha em branco.

1.2.2 Caracteres Especiais

Os seguintes símbolos são caracteres reservados, que não podem ser digitados diretamente em um texto:

\$ % ^ & _ { } ~ \

É possível inserir estes caracteres em um texto adicionando uma contrabarra antes de cada caracter.¹

1.2.3 Comandos

Comando em \LaTeX iniciam com uma contrabarra e um nome constituído de letras ou um caracter especial. Por exemplo, o comando `\LaTeX` produz \LaTeX . Comandos são *case sensitive*, e podem ter parâmetros especificados entre chaves. Alguns comandos suportam parâmetros adicionais, que são especificados entre colchetes.

Espaços inseridos após comandos são ignorados. Para evitar que o \LaTeX coma os espaços, deve-se inserir duas chaves depois do comando, como em `\TeX{}`.

¹Isto não é válido para a própria contrabarra, já que `\\` é um comando usado para quebra de linha. Para produzir uma contrabarra, pode-se usar `\backslash`.

1.2.4 Comentários

Todo o texto em uma linha inserido após um % é ignorado pelo \LaTeX (incluindo a quebra de linha). Isto pode ser usado para inserir comentários em um texto que não aparecerão na versão para impressão.

1.3 Estrutura dos Arquivos de Entrada

Todo arquivo de entrada do \LaTeX inicia com um *preâmbulo*, que é constituído por uma série de comandos que especificam a formatação do documento. Os preâmbulos do $\text{\LaTeX}_{2\epsilon}$ são iniciados por um comando que especifica o tipo de documento que será criado:

```
\documentclass{...}
```

Além de diretivas para formatação, o preâmbulo também pode conter comandos para inclusão de *pacotes*, que modificam a estrutura global do documento ou adicionam novos comandos:

```
\usepackage{...}
```

O fim do preâmbulo e início do documento propriamente dito é marcado pelo comando:

```
\begin{document}
```

O corpo do documento é constituído pelo texto e comandos \LaTeX para inclusão de fórmulas matemáticas, citações, tabelas, código de programação, figuras, etc. O final do documento é marcado pelo comando:

```
\end{document}
```

Um arquivo de entrada mínimo de entrada pode ser:

```
\documentclass{article}           % preâmbulo
\begin{document}
I wanna see monkeys.              % corpo do documento
\end{document}
```

Agora um mais complicado:

```
\documentclass[a4paper,11pt]{article} % tipo de documento
\author{Bart Simpson}                  % define o autor
\title{Why I want to go to Brazil}     % define o título
\begin{document}
\maketitle                             % produz o título
\tableofcontents                       % insere o sumário
\section{Wild Life}                    % inicia uma seção
I wanna see monkeys.
\section{Carnival}                      % outra seção
I wanna see dancing monkeys.
\end{document}
```

1.4 Compilação

Chegou a hora de compilar seu primeiro documento \LaTeX . Existem *front-ends* gráficos, como o WinEdt, com os quais é possível compilar seu documento através de botões e itens de menu, mas o sistema \LaTeX em si é simplesmente um programa de linha de comando que produz um arquivo de visualização/impressão a partir de um arquivo `.tex` de texto plano. Mas Homens de Verdade não Clicam², então vamos executar o processo de compilação via linha de comando. A descrição deste processo assume que há uma distribuição do \LaTeX instalada e devidamente configurada em seu computador.

1. Crie um arquivo de entrada em um editor de texto plano (Notepad, vi, emacs, etc), e salve-o com a extensão `.tex`
2. Através de uma linha de comando, execute

```
latex teste.tex
```

3. O comando acima cria um arquivo `teste.dvi`, que contém o resultado final da formatação. É possível visualizar este arquivo com uma ferramenta que lide com arquivos `dvi`, como o Yap(Windows) ou o `xdvi(X11)`, ou convertê-lo para o formato PostScript para impressão.

```
yap teste.dvi           % para visualizar
```

```
dvips teste.dvi         % para converter para .ps
```

1.5 Layout

1.5.1 Classes de Documentos

A primeira informação incluída em um preâmbulo de documento \LaTeX é o tipo ou classe de documento a ser criado, que é especificado pelo comando `\documentclass[parametro,parametro]{classe}`. As classes (especificadas entre chaves) do documento são fornecidas juntamente com a distribuição \LaTeX . As mais comuns são:

article para artigos em *journals* científicos, relatórios curtos, ...

report para relatórios longos contendo vários capítulos, teses de doutorado,
...

book para livros.

slides para transparências.

²Esta frase pode ser substituída por “Mulheres de Verdade não Clicam” ou “Homossexuais de Verdade não Clicam”, conforme o caso. Tanto o sistema \LaTeX quanto o autor não são preconceituosos, e acreditam que cada um pode fazer o que quiser com sua vida, exceto clicar.

Os parâmetros costumizam o comportamento da classe, e os mais comuns para as classes padrão são:

10pt, 11pt, 12pt Ajusta o tamanho da fonte principal do documento. Se nada for especificado, 10pt é usado.

a4paper, letterpaper, ... Define o tamanho do papel. O tamanho *default* é letterpaper. Além destes, **a5paper, b5paper, executivepaper** e **legalpaper** podem ser especificados.

titlepage, notitlepage Especifica se uma nova página deve ser iniciada após a colocação do título. A classe **article** não inicia uma nova página por *default*, enquanto **report** e **book** iniciam.

onecolumn, twocolumn Texto em uma ou duas colunas.

twoside, oneside Especifica um *layout* para impressão em frente-única ou frente-verso. Por *default*, **report** e **article** são frente-única, e **book**, frente-verso. Obviamente, esta opção diz respeito somente à formatação, e não indica à impressora que a impressão será ou não frente-verso.

openany, openright Faz os capítulos iniciarem na próxima página ou na próxima página do lado direito. Esta opção não é válida para a classe **article**, já que esta não conhece capítulos.

Um arquivo de entrada poderia, por exemplo iniciar com a linha:

```
\documentclass[twocolumn,a4paper,titlepage]{article}
```

Esta especificação produziria um artigo de duas colunas em papel A4, com uma página separada para o título.

1.5.2 Estilos de Página

O L^AT_EX suporta três combinações de cabeçalho/rodapé, que são chamadas estilos de página. O estilo é definido pelo *parametro* do comando `\pagestyle{parametro}`

Os três parâmetros possíveis são:

plain Imprime o número da página centralizado no rodapé. Este é o estilo padrão, e é utilizado neste documento.

headings Imprime o título do capítulo e o número da página no cabeçalho.

empty Deixa o cabeçalho e rodapé vazios.

O comando `\pagestyle{parametro}` afeta todo o documento a partir da página onde foi inserido. Para afetar somente a página atual, é possível utilizar o comando `\thispagestyle{parametro}`, com os mesmos parâmetros.

1.5.3 Pacotes

Áreas específicas de editoração, como inclusão de gráficos, texto colorido, *hyperlinks* ou código fonte não tem suporte nativo no \LaTeX . Para incluir estas especialidades, é necessário o uso de pacotes, que são expansões do \LaTeX básico. Pacotes são ativados com o comando:

```
\usepackage[opcoes]{pacote}
```

Muitos pacotes são fornecidos com a distribuição básica, e outros são disponibilizados separadamente. Instruções a respeito da instalação de novos pacotes em seu sistema podem ser obtidas em [2]. Alguns dos pacotes mais comuns são:

inputenc Permite a especificação da codificação de entrada, como ASCII, ANSI-Windows, ISO Latin-1, Apple Macintosh, Next, ou uma definida pelo usuário. Este pacote é especialmente útil para documentos em português, já que permite o processamento de caracteres acentuados.

fontenc Especifica a codificação de fonte a ser usada.

ifthen Permite comandos da forma if-then-else.

babel Modifica os strings pré-definidos³ e as regras de hifenização de acordo com a localidade de entrada.

1.6 Arquivos

O \LaTeX trabalha com uma grande quantidade de arquivos com extensões diversas, que podem facilmente assustar um iniciante. Algumas das extensões mais encontradas estão listadas a seguir.

.tex Arquivo fonte \TeX ou \LaTeX , que pode ser compilado com o comando `latex`.

.ltx Alternativa à extensão anterior `.tex`, somente para \LaTeX .

.sty Pacote de macros, pode ser usado com o comando `usepackage`.

.cls Arquivos de classe, definem a aparência final do documento. Podem ser usados com o comando `documentclass`

Os seguintes arquivos são gerados no processamento de um arquivo fonte.

.dvi Device Independent File ou Arquivo Independente de Dispositivo. Principal formato de distribuição de documentos \LaTeX .

.log Log do que ocorreu na última compilação

.toc Guarda os títulos de seções. É usado na próxima compilação para gerar o sumário.

³Sumário, Lista de Figuras, Data, ...

- .lof** Semelhante ao **.toc**, armazena títulos de tabelas.
- .lot** O mesmo, para figuras.
- .aux** Usado para transportar informações como referências cruzadas de uma compilação para outra.
- .idx** Arquivo de índice.
- .ind** Arquivo de índice processado, pronto para inclusão na próxima compilação.

1.7 Divisão de Projetos

Ao trabalhar com projetos grandes, pode ser útil dividir o documento em várias partes. Há dois comandos em \LaTeX para *juntar* as partes:

`\include{file}` insere uma quebra de página e o arquivo de com nome `file.tex` no documento.

`\input{file}` simplesmente insere o arquivo `file.tex` no documento.

Capítulo 2

Edição de Texto

2.1 Quebras de Linha e Página

Para manter o alinhamento justificado nos parágrafos, o \LaTeX otimiza a distribuição das palavras entre as linhas, hifenizando-as, se necessário. A separação entre parágrafos depende o estilo do documento, mas normalmente a primeira linha o parágrafo é indentada, e não há espaço adicional entre parágrafos.

Em casos especiais pode ser necessário inserir uma quebra de linha sem terminar o parágrafo. Isto pode ser conseguido através do comando `\` ou `\newline`. O comando `\` impede ainda uma quebra de página após a quebra de linha.

Nas edições americanas, ao contrário das brasileiras, normalmente o primeiro parágrafo de uma seção não é indentado. Este é também o padrão do \LaTeX , mas é possível modifica-lo usando o pacote `indentfirst`. Isto é conseguido adicionando a linha `\usepackage{indentfirst}` no preâmbulo do documento.

Para inserir uma quebra de página usa-se o comando `\newpage`

2.2 Caracteres Especiais e Símbolos

2.2.1 Aspas

Não deve-se usar aspas simples como em uma máquina de escrever para indicar citações. Para abrir aspas usa-se dois ‘ (acentos graves), e para fechar, dois ’ (apóstrofes). Para aspas simples utiliza-se somente um de cada.

2.2.2 Traços e Hífens

Há quatro tipos de traços, três deles sendo acessados por uma combinação de hífens e o quarto o sinal matemático de menos.

atingindo-se
nas páginas 15--21
sim---ou não?
\$0\$, \$1\$ ou \$-1\$

atingindo-se
nas páginas 15–21
sim—ou não?
0, 1 ou −1

2.2.3 Til (~)

O til é muito usado em endereços da internet. Pode-se usar `\~`, mas o resultado (~) pode não ser o esperado:

`http://www.rico.edu/\~{}bush`

`http://www.rico.edu/~bush`

`http://www.inteligente.edu/\simdemocrata`

`http://www.inteligente.edu/~democrata`

2.2.4 Reticências

Em uma máquina de escrever, pontos e vírgulas ocupam o mesmo espaço que letras, o que não acontece em editoração, onde estes caracteres ocupam muito pouco espaço. Sendo assim, não é possível inserir reticências simplesmente digitando três pontos. Há um comando especial para isto (`\ldots`).

Não assim... mas assim:

João, Maria, José, `\ldots`

Não assim... mas assim:

João, Maria, José, ...

2.3 Edição em Português

Para a edição em línguas diferentes do inglês, o \LaTeX precisa ser configurado em duas áreas:

1. Todos os textos gerados automaticamente¹ precisam ser adequados à língua de edição. Isto é conseguido através do pacote Babel.
2. As regras de hifenização precisam ficar de acordo com a nova língua. Esta parte é mais difícil de configurar, já que a configuração depende da distribuição utilizada. O Apêndice A apresenta um breve guia para configurar as regras de hifenização no MiKTeX. Mais informações sobre configurações específicas de sistema podem ser obtidas em [2].

Caso o sistema já esteja devidamente configurado, é possível acionar o pacote Babel incluindo a seguinte linha no preâmbulo.

`\usepackage[lingua]{babel}`²

2.3.1 Acentos

O \TeX original suportava somente os 128 caracteres do código ASCII original e a acentuação era conseguida através da sobreposição de caracteres. Por exemplo, `\'o` produzia ó. Este tipo de construção construção ainda é válido, mas a partir do lançamento do $\text{\LaTeX} 2_{\epsilon}$, várias codificações de caracteres são suportadas através do pacote `inputenc`, tornando possível a inserção de caracteres acentuados diretamente no texto. Este pacote pode ser usado incluindo no preâmbulo a linha:

¹Nomes de Seções, Sumário, Bibliografia, etc.

²No caso do Português Brasileiro, língua deve ser substituída por `brazil`

```
\usepackage[codificacao]{inputenc}
```

Ao usar o pacote `inputenc` deve-se levar em conta que é possível que outras pessoas não consigam ler seus arquivos de entrada, por usarem uma codificação diferente. Por exemplo, `ã` é codificado como 132 em um PC, mas em alguns sistemas UNIX usando ISO-LATIN 1 é codificado como 228. A tabela a seguir apresenta algumas codificações que podem ser utilizadas no parâmetro `codificacao` do `inputenc`.

Sistema Operacional	codificação
Mac	<code>applemac</code>
Unix	<code>latin1</code>
Windows	<code>ansinew</code>
OS/2	<code>cp850</code>

Outro problema da acentuação é que internamente, caracteres acentuados ainda serão produzidos através de combinações entre caracteres, já que a codificação de fonte *default* do \LaTeX somente suporta o código ASCII de 7 bits. A consequência disso é que palavras acentuadas não podem ser hifenizadas, o que pode causar problemas estéticos no documento. Isto pode ser contornado com o uso de outra codificação de fonte, com o pacote `fontenc`.

A codificação EC tem aparência idêntica à Computer Modern (a codificação original do \TeX), mas tem suporte nativo para a maioria das línguas latinas, possibilitando a hifenização de palavras acentuadas. Esta pode ser usada usando o pacote `fontenc` com o parâmetro `T1`:

```
\usepackage[T1]{fontenc}
```

2.4 Títulos, Capítulos e Seções

Para facilitar a compreensão do leitor, um texto deve ser dividido em capítulos, seções e subseções. O \LaTeX suporta esta divisão através de comandos que tomam como parâmetro o título da seção.

As seguintes divisões estão disponíveis na classe `article`:

```
\section{...}
```

```
\subsection{...}
```

```
\subsubsection{...}
```

```
\paragraph{...}
```

```
\subparagraph{...}
```

Dois comandos adicionais são suportados pelas classe `book` e `report`:

```
\part{...}
```

```
\chapter{...}
```

O espaçamento entre seções, a numeração e o tamanho da fonte serão ajustados automaticamente, de acordo com a classe do documento.

Existe ainda o comando `\appendix`, que não tem argumentos, e muda a numeração dos capítulos (ou seções, na classe `article`) para letras.

O \LaTeX cria o sumário tomando o nome das seções e o número das páginas gerados na última compilação. O comando `\tableofcontents` insere o sumário gerado no documento. Para gerar um sumário correto, é necessário compilar o documento 2 ou 3 vezes.

Todos os comandos de particionamento tem uma forma *estrelada*, que consiste em adicionar um * após o nome do comando. O efeito disto é que a seção não será incluída no sumário, e não será numerada. O comando `\section{Prefácio}`, por exemplo, ficaria `\section*{Prefácio}`.

Normalmente os títulos de seção aparecem da mesma forma no corpo do texto e no sumário. Entretanto, isto pode não ser possível quando o título é muito grande para caber no sumário. Isto pode ser contornado com um argumento adicional no comando de particionamento:

```
\chapter[Título Para o Sumário]{Título Longo, Inútil e
                               Chato para o Corpo do Texto}
```

O título do documento é gerado através do comando `\maketitle`. Os conteúdos do título devem ser definidos pelos comandos `\title{...}` (para o título), `\author{...}` (para o autor) e opcionalmente `\date{...}` (para a data), antes de chamar o `\maketitle`. No campo `author`, vários autores podem ser separados através de comandos `\and`.

2.5 Referências Cruzadas

É comum em livros ou artigos a existência de referências cruzadas para figuras, tabelas ou partes do texto. Os seguintes comandos suportam estas referências:

```
\label{marcador}    Insere um marcador para o ponto atual do texto.
\ref{marcador}       Insere no texto o número da Seção de marcador.
\pageref{marcador}   Insere no texto o número da Página de marcador.
```

Uma referência `\label{sec:esta}` a esta seção fica assim:
“veja a Seção~`\ref{sec:esta}` na Pag.~`\pageref{sec:esta}`”.

Uma referência a esta seção fica assim: “veja a Seção 2.5 na Pag. 14”. O til entre Seção e a referência impede uma quebra de linha entre as duas palavras.

2.6 Notas de Rodapé

Notas de rodapé podem ser inseridas com o commando `\footnote{Nota}`.

2.7 Ênfatização

Há várias maneiras de enfatizar uma palavra:

<code>\underline{sublinhando}</code>	<u>sublinhando</u>
<code>\emph{enfatizando}</code>	<i>enfatizando</i>
<code>\textit{italico}</code>	<i>italico</i>

2.8 Ambientes

Ambientes são utilizados através de um par

```
\begin{ambiente}  
  % conteudo  
\end{ambiente}
```

Os principais ambientes são explicados nas subseções seguintes.

2.8.1 Itemizar, Enumerar e Descrição

O ambiente `itemize` pode ser utilizado para listas, `enumerate` para listas numeradas e `description` para descrições:

```
\begin{enumerate}  
\item Você pode misturar os ambientes  
de lista como desejar  
\begin{itemize}  
\item Mas pode começar a ficar meio bobo.  
\item[-] Com um hífen.  
\end{itemize}  
\item Portanto lembre-se  
\begin{description}  
\item[Coisas idiotas] não vão ficar interessantes  
só por estarem em uma lista.  
\item[Coisas Inteligentes] podem ser muito bem  
apresentadas em uma lista  
\end{description}  
\end{enumerate}
```

1. Você pode misturar os ambientes de lista como desejar

- Mas pode começar a ficar meio bobo.
- Com um hífen.

2. Portanto lembre-se

Coisas idiotas não vão ficar interessantes só por estarem em uma lista.

Coisas inteligentes podem ser muito bem apresentadas em uma lista.

2.8.2 Alinhamento

Os ambientes `flushleft` e `flushright` alinham o texto à esquerda e à direita, respectivamente. O `center` centraliza o texto. Nestes ambientes, o `LATEX` não tenta fazer todas as linhas do mesmo tamanho:

```
\begin{flushleft}  
Este texto está\\ alinhado à esquerda. O \LaTeX{} não está  
tentando fazer as linhas do mesmo tamanho.  
\end{flushleft}
```


Este texto está
alinhado à esquerda. O \LaTeX não está tentando fazer as linhas do mesmo
tamanho.

```
\begin{flushright}
Este texto está\\ alinhado à direita. O \LaTeX{} não está tentando
fazer as linhas do mesmo tamanho.
\end{flushright}
```

Este texto está
alinhado à direita. O \LaTeX não está tentando fazer as linhas do mesmo
tamanho.

```
\begin{center}
No centro \\ da Terra.
\end{center}
```

No centro
da Terra.

2.8.3 Citações e Versos

O ambiente `quote` é útil para citações, frases importantes ou exemplos:

```
Uma regra para o comprimento de linha em tipografia é
\begin{quote}
Na média, nenhuma linha deve ter mais de 66 caracteres
\end{quote}
É por isso que as páginas do \LaTeX{} tem margens tão grandes e
por que eles usam várias colunas em jornais.
```

Uma regra para o comprimento de linha em tipografia é

Na média, nenhuma linha deve ter mais de 66 caracteres

É por isso que as páginas do \LaTeX tem margens tão grandes e por que eles
usam várias colunas em jornais.

Para poemas, onde a quebra de linhas é importante, existe o ambiente
`verse`:

```
\begin{verse}
Rosas são verdes, \\
Violetas são azuis. \\
Feliz Dia das Bruxas, \\
Querida Professora. \\
\end{verse}
```

Rosas são verdes,
Violetas são azuis.
Feliz Dia das Bruxas,
Querida Professora.

2.8.4 Verbatim

Um texto entre `\begin{verbatim}` e `\end{verbatim}` será inserido como digitado, sem ser processado pelo L^AT_EX. Dentro de um parágrafo pode-se usar `\verb"`, sendo que o texto entre aspas será inserido diretamente. A maioria dos exemplos neste manual é inserida desta forma.

2.8.5 Tabular

O ambiente tabular pode ser usado para criar tabelas com linhas horizontais e verticais. A largura das colunas é determinada automaticamente.

`\begin{tabular}{especificacao}`

O argumento `especificacao` determina a forma da tabela. Pode-se usar `l` para uma coluna com texto alinhado à esquerda, `r` para alinhamento à direita, `c` para texto centralizado. Usa-se `p{largura}` para uma coluna com texto justificado com quebras de linha e `|` para produzir uma linha vertical.

Dentro do ambiente tabular, um `&` pula para a próxima coluna, `\\` inicia uma nova linha e `\hline` insere uma linha horizontal. Pode-se adicionar linhas horizontais parciais com a função `\cline{i-j}`, onde `i` e `j` são os números das colunas sobre as quais a linha deve passar.

```
\begin{tabular}{|r|l|}  
\hline  
7C0          & hexadecimal    \\  
3700         & octal             \\  
\cline{2-2}  
11111000000  & binary           \\  
\hline  
\hline  
1984         & decimal          \\  
\hline  
\end{tabular}
```

7C0	hexadecimal
3700	octal
11111000000	binary
1984	decimal

```
\begin{tabular}{|p{4.7cm}|}  
\hline  
Eu sou um parágrafo com quebras  
de linha dentro de uma coluna.\\  
\hline  
\end{tabular}
```

Eu sou um parágrafo com quebras de linha dentro de uma coluna.
--

O separador de colunas pode ser alterado através de um `@{}`, onde o valor entre chaves será o novo separador.

```
\begin{tabular}{@{} l @{}}
\hline
sem espaço entre colunas\\
\hline
\end{tabular}
```

sem espaço entre colunas

Através do comando `\multicolumn{n}{alinhamento}{conteudo}` pode-se escrever através de duas ou mais colunas:

```
\begin{tabular}{|c|c|c|} \hline
Ano      & \multicolumn{2}{|c|}{Índice Acadêmico}      & \\
\cline{2-3}
        & I.A.      & I.A.A      & \\
\hline
2001     & 10.0      & 9.99      & \\
\hline
2002     & 9.99      & 9.98      & \\
\hline
\end{tabular}
```

Ano	Índice Acadêmico	
	I.A.	I.A.A
2001	10.0	9.99
2002	9.99	9.98

O ambiente `tabular` não permite tabelas em várias páginas, o que pode ser obtido com os ambientes `supertabular` e `longtabular`. Para exemplos de uso destes ambientes, além de outros exemplos de tabelas, consulte [3].

2.9 Corpos Flutuantes

Corpos flutuantes são figuras ou tabelas, posicionadas de maneira a otimizar a ocupação das páginas. Estes elementos não podem conter quebras de página, e são inseridos pelo `LATEX` em uma fila de colocação. De acordo com alguns parâmetros especificados pelo usuário (veja Tab. 2.1), os corpos flutuantes são posicionados no documento. Para evitar que os corpos flutuantes tornem-se uma grande fonte de frustração (o `LATEX` nunca os posiciona onde você deseja), é importante entender um pouco do mecanismo interno de posicionamento de *floats*.

Os corpos flutuantes são ambientes, e podem ser iniciados, por exemplo, da seguinte forma:

```
\begin{table}[!htb]
```

ou

`\begin{figure}[hp]`

O primeiro exemplo inicia uma tabela, que será posicionada onde ocorreu no texto (h) mesmo que isto prejudique a estética do documento (!). Se não for possível posicionar a tabela neste ponto, ela será colocada no topo (t) ou na parte inferior (b) de uma página.

O segundo exemplo inicia uma figura que será posicionada neste ponto (h) ou em uma página especial somente com tabelas e figuras.

Se nenhum parâmetro de posicionamento é fornecido, é utilizado o padrão `tbp`.

O \LaTeX posiciona todo *float* que encontra de acordo com o parâmetro de posicionamento fornecido. Se um corpo flutuante não puder ser colocado na página atual, ele é inserido em uma fila de figuras ou de tabelas. Quando uma nova página é iniciada, o primeiro *float* é tratado como se tivesse ocorrido neste ponto do texto: o \LaTeX tenta outra vez posicioná-lo de acordo com seu parâmetro (exceto ‘h’, que não é mais possível). Qualquer novo *float* que ocorra é colocado em sua respectiva fila. A ordem de ocorrência dos corpos flutuantes é mantida estritamente, por isso uma figura que não pode ser posicionada “atrasa” a colocação de todas as figuras seguintes. Portanto:

Se o \LaTeX não está posicionando os *floats* como você deseja, é provável que exista um *float* trancando uma das filas.

Mesmo sendo possível dar ao \LaTeX uma única possibilidade de posicionamento, isso causa problemas. Se o *float* não pode ser posicionado no lugar especificado, ele fica preso, bloqueando todos os corpos flutuantes seguintes. Em particular, nunca deve-se usar a opção [h], que nas últimas versões do \LaTeX é substituída automaticamente por [ht]. Sob algumas circunstâncias, pode-se usar o comando `\clearpage`, que ordena que todas as figuras e tabelas nas filas sejam posicionadas, e inicia uma nova página.

Dentro dos ambientes `table` e `figure`, a função `\caption{texto}` especifica o rótulo do *float*. Para incluir listas de corpos flutuantes no documento, usa-se as funções `\listoftables` e `\listoffigures`. Como no sumário, é preciso compilar o documento três vezes para que estas listas sejam inseridas corretamente.

Tabela 2.1: Parâmetros de Posicionamento.

Parêmtro	Permissão para posicionar o corpo flutuante ...
h	no lugar onde ocorreu no texto. Funciona normalmente para tabelas pequenas
t	no topo de uma página
b	na parte inferior de uma página
p	em uma página especial contendo somente corpos flutuantes
!	sem considerar a maioria dos parâmetros ^a que poderiam impedir este corpo de ser posicionado.

^aComo o número máximo de corpos flutuantes por página.

Capítulo 3

Especialidades e Customização

3.1 Geração de Arquivos PDF

O formato PDF (*Portable Document Format*) da Adobe é muito popular para a distribuição de documentos via Internet. A produção destes documentos a partir de arquivos fonte \LaTeX é muito simples: basta usar o compilador `pdflatex` no lugar do tradicional `latex`. A saída será um arquivo PDF, e não um DVI.

Para incluir marcadores no formato PDF, além de alterar algumas propriedades do documento neste formato, pode-se incluir no preâmbulo (diretamente ou através de um `\input` de outro arquivo), um código semelhante ao seguinte:

```
\newif\ifpdf
\ifx\pdfoutput\undefined
\pdffalse
\else
\pdfoutput=1
\pdftrue
\fi

\ifpdf \RequirePackage[pdftex]{hyperref}

\hypersetup{colorlinks,
  debug=false,
  linkcolor=black,   %%% cor do tableofcontents,
                    %%% \ref, \footnote, etc
  citecolor=black,   %%% cor do \cite
  urlcolor=black,    %%% cor do \url e \href
  pdftitle={Relatório Anual de Atividades 2000},
  pdfauthor={PET/CCO - UFSC},
  pdfsubject={Relatório Anual de Atividades do PET/CCO},
  pdfkeywords={PET,Relatório,Atividades}}
\fi
```

3.2 Inclusão de Imagens

A inclusão de imagens no \LaTeX depende da distribuição e dos pacotes disponíveis. Neste manual será utilizado o pacote `graphicx`, que está disponível na maioria das distribuições. Estando disponível este pacote, deve-se incluir no preâmbulo do documento uma declaração de uso do mesmo:

```
\RequirePackage[pdftex]{graphicx}
```

ou

```
\RequirePackage[dvips]{graphicx}
```

Os parâmetros `pdftex` e `dvips` indicam qual compilador será utilizado. O primeiro indica o `pdflatex` e o segundo o `latex`. Quando for indicado na declaração que o compilador será `pdflatex` e for usado o `latex`, o documento provavelmente não será compilado corretamente. Existem ainda outras incompatibilidades entre os compiladores, como suporte a diferentes formatos de imagens.

A maioria dos visualizadores de arquivos `dvi` não exibirá as imagens, deixando somente o espaço reservado para as mesmas. A exibição de fato só ocorrerá se o arquivo `dvi` for convertido para um arquivo `ps`. Os arquivos compilados com `pdflatex`, entretanto exibirão as imagens corretamente.

A inclusão das imagens no documento é com o `includegraphics`:

```
\includegraphics[width=12cm]{graficos/tipoComp.jpg}
```

Este comando aceita alguns parâmetros adicionais como: `width` (largura), `height` (altura), `angle` (rotacionar no sentido anti-horário), `scale` (apresenta o gráfico na escala indicada).

Da forma apresentada acima, o `includegraphics` inclui a imagem diretamente no corpo do documento, porém é mais comum que as imagens sejam dispostas na forma de corpos flutuantes, possivelmente com legendas. Isto é conseguido com o ambiente `figure`:

```
\begin{figure}[hbt]
\begin{center}
\includegraphics[height=2cm]{figuras/platypus.jpg}
\caption{Um ornitorrinco}
\end{center}
\end{figure}
```

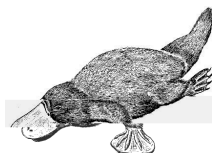


Figura 3.1: Um ornitorrinco

3.3 Bibliografia

Referências bibliográficas podem ser produzidas com ajuda do ambiente `thebibliography`. Cada entrada inicia com

```
\bibitem{marcador}
```

onde `marcador` é usado para citar o item no texto, através de um

```
\cite{marcador}
```

A numeração das entradas é gerada automaticamente. O parâmetro após seta o número máximo de referências, usado para a formatação.

Segue um exemplo de bibliografia.

```
\begin{thebibliography}{9}

\addcontentsline{toc}{chapter}{\bibname}
%% adiciona a bibliografia no sumário

\bibitem{think}
Bruce Eckel. \emph{Thinking In Java, 2nd Edition}.
Prentice-Hall, Segunda Edição, 2000.
Disponível para download em http://www.BruceEckel.com

\bibitem{tutorial}
Lisa Friendly, Mary Campione, Kathy Walrath, Alison Huml.
\emph{The Java Tutorial}.
Sun Microsystems, Segunda Edição, 2000.
Disponível para download e online em
http://java.sun.com/docs/books/tutorial/

\bibitem{api}
Sun Microsystems \emph{Java 2 Platform, Standard Edition,
v 1.3.1 API Specification}.
Sun Microsystems, 2001. Disponível online
e para download em http://java.sun.com/docs/

\bibitem{spec}
James Gosling, Bill Joy, Guy Steele, Gilad Bracha.
\emph{The Java Language Specification}.
Sun Microsystems, Segunda Edição, 2000.
Disponível online em http://java.sun.com/docs/

\end{thebibliography}
```

3.3.1 Fontes e Tamanhos

O \LaTeX escolhe o tamanho e tipo de fontes baseado na estrutura lógica do documento (seções, notas de rodapé, ...). É possível, entretanto, alterar estes tamanhos e tipos a mão. Para fazer isto, deve-se usar os comandos listados nas tabelas 3.1 e 3.2.

<code>textrm{...}</code>	roman
<code>textsf{...}</code>	sans serif
<code>texttt{...}</code>	máquina de escrever
<code>textmd{...}</code>	médio
<code>textbf{...}</code>	negrito
<code>textup{...}</code>	não-inclinado
<code>textit{...}</code>	<i>itálico</i>
<code>textsl{...}</code>	<i>inclinado</i>
<code>textsc{...}</code>	MAIÚSCULAS
<code>emph{...}</code>	<i>ênfatizado</i>
<code>textnormal{...}</code>	normal

Tabela 3.1: Fontes

<code>tiny</code>	fonte minúscula
<code>scriptsize</code>	fonte muito pequena
<code>footnotesize</code>	fonte bem pequena
<code>small</code>	fonte pequena
<code>normalsize</code>	fonte normal
<code>large</code>	fonte grande
<code>Large</code>	fonte maior
<code>LARGE</code>	fonte ainda maior
<code>huge</code>	fonte enorme
<code>Huge</code>	a maior de todas

Tabela 3.2: Tamanhos de fontes.

Lembre-se! Quanto **MAIS** fontes **VOCÊ** usar em um documento, *mais* LEGÍVEL e *bonito* ele *ficará*.

3.4 Espaçamento

Para mudar o espaçamento entre linhas em um documento usa-se o comando

`\linespread{fator}`

no preâmbulo do documento. Pode-se usar fator 1.3 para espaçamento 1,5. 1.5 para espaçamento duplo.

3.5 Layout de Página

O \LaTeX permite a especificação do tamanho de página no comando `documentclass`. Ainda assim, é possível modificar estes valores pré-definidos, através de comandos do tipo

```
\setlength{\medida}{tamanho}
```

que devem ser incluídos no preâmbulo, onde `medida` é a medida a ser alterada, e `tamanho` o novo valor desta medida. As principais medidas de páginas estão especificadas na tabela 3.3.

medida	significado
<code>voffset</code>	deslocamento na vertical
<code>hoffset</code>	deslocamento na horizontal
<code>textheight</code>	altura do texto
<code>textwidth</code>	largura do texto

Tabela 3.3: Medidas do Layout de Páginas.

Apêndice A

Instalação e Configuração do Sistema MiKTeX

O sistema \LaTeX está disponível em várias distribuições, para diversos sistemas operacionais. Esta diversidade é muito desejável, mas traz alguns problemas, sendo um dos principais a diferença entre os processos de instalação e configuração dos diferentes sistemas. As instruções para estes processos estão no *\LaTeX Local Guide* [2], mas a compreensão deste documento nem sempre é fácil, especialmente para o usuário iniciante.

Este apêndice cobre a instalação do sistema MiKTeX para Windows, versão 2.1.7, que é atualmente uma das mais populares distribuições \LaTeX . É um guia específico e superficial, e deve ser complementado (ou substituído, para usuários de outras distribuições) por [2].

A.1 Download e Instalação

O instalador do MiKTeX pode ser obtido em <http://www.miktex.org> ou em `CTAN:/systems/win32/miktex`¹. Após fazer o download, execute o programa e escolha a opção *Download Only*. Em *Package Set*, a opção *Small* deve servir à maioria dos usuários. Escolha um dos servidores de arquivo disponível e aguarde o download².

Depois de baixar os arquivos, execute novamente o instalador, selecionando a opção *Install*. Em *Package Set* escolha novamente *Small*. Selecione o local onde os arquivos baixados estão armazenados e escolha a opção *Shared environment*. Escolha um diretório para a instalação do programa e uma pasta de atalhos. Selecione as opções *Create local TEXMF folder* e *Don't incorporate existing TEXMF folder trees now*, e aguarde o término da instalação.

Após a instalação, verifique se os executáveis estão no caminho do sistema. Isto pode ser feito executando o comando `tex`. Se o aplicativo executar, digite `q` para sair. Se não for possível executar o aplicativo, adicione o caminho `<local da instalacao>/miktex/bin`³ à variável de sistema `PATH`.

¹verificar

²É só isso mesmo?

³Normalmente o local da instalação é `c:/texmf`

A.2 Configuração

Execute o aplicativo *MiKTeX Options* (através do atalho ou do comando `mo`). Na guia *Languages*, selecione as línguas desejadas e mova sua língua de uso principal (provavelmente português) para o topo da lista. Clique em *Apply* e volte à guia *General*. Clique em *Refresh Now* e depois em *Update Now*. Com um pouco de sorte, seu sistema está pronto para ser utilizado⁴.

A.3 WinEdt

Documentos L^AT_EX pode ser editados com qualquer editor de texto, mas há editores com facilidades específicas para edição deste tipo de arquivos.

Para Windows, o mais popular destes aplicativos é o WinEdt, um shareware que pode ser obtido em <http://www.winedt.com>. Este editor oferece facilidades como:

- Suporte a dicionários
- *Syntax Highlight*
- Atalhos para executáveis
- Interface gráfica para símbolos
- Gerenciamento de projetos

O uso do programa é bastante intuitivo para alguém que já tenha editado e compilado documentos L^AT_EX “no braço”, mas é recomendável ler o conteúdo da janela *Configuration Wizard* após a instalação.

⁴Por razões misteriosas, às vezes é necessário repetir estas operações. Se, ao editar seus documentos, você não conseguir hifenização correta em português (ou na língua de sua escolha), volte a este passo.

Referências Bibliográficas

- [1] Tobias Oetiker. *The Not So Short Introduction to \LaTeX 2 $_{\epsilon}$ or \LaTeX 2 $_{\epsilon}$ in 95 minutes*. Introdução bastante completa, em inglês. Disponível em <http://www.ctan.org/tex-archive/info/lshort/english/>.
- [2] *The \LaTeX Local Guide*. Guia incluído na maioria das distribuições, que explica o processo de instalação e configuração do \LaTeX .
- [3] Steding-Jessen, Klaus *\LaTeX demo: Exemplos com \LaTeX 2 $_{\epsilon}$* . Ótimo material de consulta, cobrindo vários ambientes e pacotes. Disponível (em português) em <http://biquinho.furg.br/tex-br/doc/LaTeX-demo/>
- [4] Donald E. Knuth. *The \TeX book*, , Addison-Wesley, segunda edição, 1984. Livro escrito pelo autor do \TeX , cobre virtualmente todos seus aspectos.