

CENTRO UNIVERSITÁRIO DA FEI
LEONARDO ANJOLETTA FERREIRA & DOUGLAS DE RIZZO MENEGHETTI

**CLASSE \LaTeX DA FEI PARA CRIAÇÃO DE DISSERTAÇÕES E TESES DE
PÓS-GRADUAÇÃO STRICTO SENSU: de acordo com o manual de 2007 da biblioteca**

São Bernardo do Campo
2014

**Classe L^AT_EX da FEI para criação de dissertações e teses de pós-graduação stricto sensu:
de acordo com o manual de 2007 da biblioteca**

Dissertação/tese de Mestrado/Doutorado apresentada ao Centro Universitário da FEI para obtenção do título de Mestre/Doutor em [Curso], orientado pelo Prof. Dr. [Orientador].

A quem eu quero dedicar o texto.

AGRADECIMENTOS

A quem eu quero agradecer.

“A good scientist is a person with original ideas. A good engineer is a person who makes a design that works with as few original ideas as possible. There are no prima donnas in engineering.”

Freeman Dyson

RESUMO

Resumo vai aqui.

Palavras-chave: Palavras. Chave. Vão. Aqui.

ABSTRACT

Abstract goes here.

Keywords: Keywords. Go. Here.

LISTA DE TABELAS

2.1	Exemplo de tabela com legenda acima e fonte abaixo	16
2.2	Comandos originais e suas substituições	19

LISTA DE ALGORITMOS

2.1	Exemplo de algoritmo usando algorithm2e em português	17
-----	--	----

LISTA DE ABREVIATURAS

ABNT Associação Brasileira de Normas Técnicas.

abnT_EX *Absurd Norms for T_EX*.

FEI Fundação Educacional Inaciana.

SUMÁRIO

1	INTRODUÇÃO	12
1.1	A classe da FEI (fei.cls)	12
1.2	Organização do Texto	13
2	CLASSE FEI.CLS	14
2.1	Pacotes necessários	14
2.2	Pacotes que faltam importar	15
2.3	Comandos e ambientes modificados	15
2.3.1	\maketitle	15
2.3.2	Ambientes itemize e enumerate	15
2.3.3	\listofalgorithms	15
2.3.4	Ambiente table	16
2.3.5	Variáveis do pacote algorithm2e	16
2.3.6	Outros comandos/ambientes internos	17
2.4	Novos ambientes	17
2.4.1	\folhaderosto	17
2.4.2	\resumo	18
2.4.3	\agradecimentos	18
2.5	Novos comandos	18
2.5.1	\subtitulo{ }	18
2.5.2	\sumario,\figuras,\tabelas	18
2.5.3	\folhadeaprovacao{ }	19
2.5.4	\fichacatalografica{ }	19
2.5.5	\dedicatoria{ }	19
2.5.6	\epigrafe{ }{ }	20
2.5.7	\aspas{ }	20
2.5.8	\marca	20
2.5.9	\palavraschave{ } e \keyword{ }	21
2.5.10	\apendice{ }, e \anexo{ }	21
2.5.11	\bibliografia{ }	22
3	REFERÊNCIA USANDO O abnT_EX	23
3.1	O que é o abnT _E X e como foi utilizado	23
3.2	Citação no final de linha	23
3.3	Citação durante o texto	23
3.4	Citação indireta	24

3.5	Citação no rodapé	24
3.6	Citações múltiplas	24
3.7	Citações de campos específicos	24
3.8	Outros exemplos	25
3.9	Citações com mais de três linhas	25
4	ÍNDICES USANDO MAKEINDEX	26
4.1	Indexando palavras	26
4.2	Imprimindo o índice	26
5	LISTAS DE SÍMBOLOS E ABREVIATURAS	27
5.1	Indexando abreviaturas	27
5.2	Indexando símbolos	27
5.3	Utilizando abreviaturas e símbolos indexados	28
5.4	Imprimindo as listas	28
6	COMPILANDO O PROJETO	29
6.1	Compilação utilizando arara	29
7	INSTALAÇÃO DOS PACOTES E PROGRAMAS	31
7.1	Windows	31
7.2	Ubuntu	31
	REFERÊNCIAS	32
	APÊNDICE A - Arquivos criados pelo processo de compilação	34
	ÍNDICE	34

1 INTRODUÇÃO

Para o desenvolvimento de trabalhos acadêmicos, a biblioteca do Centro Universitário da Fundação Educacional Inaciana (FEI) utiliza um modelo baseado na norma da Associação Brasileira de Normas Técnicas (ABNT). O modelo da FEI é baseado na ABNT pois este não é em si um padrão fixo, mas uma série de opções cuja escolha fica a cargo da instituição.

Existem diversas instituições que utilizam modelos baseados na ABNT e até mesmo a classe *Absurd Norms for T_EX* (abnT_EX), porém certas configurações são necessárias para que o texto se torne o mais próximo possível do modelo requisitado pela biblioteca da FEI.

O único arquivo necessário para escrever o documento seguindo o modelo proposto é a classe `fei.cls` que realiza a formatação de todo o texto, iniciando pela capa, passando pelos elementos de pré-texto, textuais, referências e, finalmente, pela formatação de anexos e apêndices.

É necessário enfatizar a necessidade de instalação de determinados pacotes, listados e descritos mais abaixo, dos quais a classe `fei.cls` depende para seu funcionamento correto. Estes pacotes estão disponíveis nas distribuições do MikT_EX (para Windows), MacT_EX (para Mac OS) e T_EX Live (para Linux e Mac OS).

1.1 A classe da FEI (`fei.cls`)

A escrita da classe que formata o texto foi realizada seguindo apenas o manual disponível pela biblioteca (nesta versão, o manual utilizado desde 2007).

Toda a formatação foi realizada a partir da importação e configuração de pacotes já existentes e disponíveis nas diversas distribuições de L^AT_EX existentes. Durante o desenvolvimento desta classe, buscou-se utilizar o menor número de pacotes possíveis e sempre os mais comuns de serem encontrados.

Para facilitar a escrita do texto final, alguns comandos/ambientes já existentes foram modificados e novos comandos e ambientes foram adicionados. Desta forma, espera-se que o autor tenha menos trabalho com a formatação do texto do que com a escrita do mesmo.

1.2 Organização do Texto

Este capítulo descreveu a ideia geral utilizada para criar um pacote que forneça os recursos necessários para desenvolver um trabalho escrito segundo o modelo da ABNT seguido pela biblioteca do Centro Universitário da FEI.

O capítulo 2 trata da classe `fei.cls`, explicando os comandos e ambientes criados e modificados. O capítulo 3 apresenta uma breve explicação de como o pacote `abnTeX` foi utilizado e seus comandos de citação que seguem os padrões utilizados no modelo da FEI. O capítulo 4 ensina os comandos básicos para a criação de um índice remissivo de palavras. O capítulo 4 ensina como utilizar o pacote `glossaries` na indexação e utilização de abreviaturas e símbolos, assim como a impressão das suas respectivas listas. O capítulo 6 ensina a ordem em que o projeto deve ser compilado, assim como introduz o leitor à ferramenta `arara` de compilação automatizada de projetos `LATEX`. Por fim, o capítulo 7 guia o leitor na instalação dos diversos pacotes e programas necessários para utilização de todas as funcionalidades da classe `fei.cls`.

2 CLASSE FEI.CLS

2.1 Pacotes necessários

- a) `geometry`: utilizado para formatar as margens da folha;
- b) `fancyhdr`: utilizado na formatação do cabeçalho;
- c) `babel`: escolha de línguas (importado pacote para português e inglês);
- d) `fontenc`: codificação da fonte;
- e) `algorithm2e`: provê comandos para a escrita de algoritmos;
- f) `mathtools`: extensões para facilitar a escrita de fórmulas matemáticas (inclui o pacote `amsmath`);
- g) `times`: carrega fonte Times New Roman;
- h) `graphicx`: importação e utilização de imagens;
- i) `paralist`: para gerar listas sem quebra de linha;
- j) `multirow`: permite que uma coluna ocupe várias linhas em uma tabela;
- k) `xcolor`: utilizado para alterar cores em células de tabela;
- l) `hyperref`: gera os links entre referências no PDF;
- m) `setspace`: espaçamento entre linhas;
- n) `caption`: altera a formatação de certas legendas;
- o) `tocloft`: permite melhor personalização de itens do sumário, lista de figuras e tabelas;
- p) `pdfpages`: faz a inclusão de páginas em PDF no documento final;
- q) `ifthen`: permite a utilização de condições na geração do texto;
- r) `imakeidx`: permite a criação de um índice remissivo ao fim do texto;
- s) `glossaries`: permite a criação de listas de símbolos e abreviaturas;
- t) `abntex2cite`: formata citações e referências de acordo com o padrão ABNT 6023.

2.2 Pacotes que faltam importar

- a) `inputenc`: codificação dos arquivos `tex` de entrada. Depende do editor que está sendo utilizado, normalmente `latin1` ou `utf8`.

2.3 Comandos e ambientes modificados

2.3.1 `\maketitle`

O comando `\maketitle` foi modificado para criar uma página no formato da biblioteca. O comando utiliza o nome fornecido em `\author{ }`, o título em `\title{ }`, o subtítulo de `\subtitulo{ }` juntamente com o ano corrente para gerar a capa. O local (São Bernardo do Campo) é fixo (podendo ser alterado no arquivo da classe).

2.3.2 Ambientes `itemize` e `enumerate`

Segundo o padrão da biblioteca, toda lista deve utilizar a sequência de letras. Para que não houvesse problemas de formatação, o ambiente `itemize` foi redirecionado para utilizar o `enumerate` e este passa a utilizar a letras para a sequência de itens (como utilizado na seção 2.1).

2.3.3 `\listofalgorithms`

O pacote `algorithm2e` já importado permite que algumas configurações sejam feitas, como a formatação da lista de algoritmos. O comando `\listofalgorithms` foi modificado para deixar o título centralizado e em português.

2.3.4 Ambiente `table`

O manual da biblioteca é bem específico quanto às legendas de tabelas: legenda principal em cima da tabela e fonte abaixo. Para satisfazer esta regra, o pacote `caption` redefine o comando `\caption{}` e disponibiliza o comando novo `\caption*{}`. O primeiro, situado acima da tabela, representará a legenda principal e aparecerá na lista de tabelas. Já o segundo é formatado da mesma forma que uma legenda comum, porém não aparece na lista de tabelas e não recebe numeração. Um exemplo pode ser visto na tabela 2.1.

```
\begin{table}[ht!]
  \caption{Legenda da tabela}
  \begin{tabular}{|c|c|c|c|}
    [...]
  \end{tabular}
  \caption*{Fonte: Autor}
\end{table}
```

Tabela 2.1 – Exemplo de tabela com legenda acima e fonte abaixo

•	x_1	x_2	x_3
y_1	1	0	0
y_2	0	1	0
y_3	0	0	1

Fonte: Autor

2.3.5 Variáveis do pacote `algorithm2e`

O pacote `algorithm2e` fornece diversos comandos para a escrita de pseudo-código em diversos idiomas. O idioma importado pela `fei.cls` foi o português.

Exemplo:

```
\begin{algorithm}
  \Entrada{Vetor \ (X\)}
  \Saida{Vetor \ (Y\)}
  \ParaCada{variável \ (x_i \in X\)}{
    \ (y_i = x_i^2\ )
  }
  \Retorna \ (Y\)
```

```

\caption{Exemplo de algoritmo usando algorithm2e em português}
\label{lst:alg}
\end{algorithm}

```

<p>Entrada: Vetor X</p> <p>Saída: Vetor Y</p> <p>para cada <i>variável</i> $x_i \in X$ faça</p> <p style="padding-left: 20px;">$y_i = x_i^2$</p> <p>fim</p> <p>retorna Y</p>

Algoritmo 2.1: Exemplo de algoritmo usando algorithm2e em português

2.3.6 Outros comandos/ambientes internos

Alguns comandos e ambientes, como `chapter`, `abstract`, `fontsize` `quote` e `quotation`, que são comandos já definidos dentro do \LaTeX , foram modificados para seguir as descrições do manual da biblioteca.

Apesar destes comandos terem sido modificados, as mudanças foram feitas de forma que a utilização dos mesmos continuasse igual, assim um texto já escrito para outro modelo poderia ser apenas recompilado utilizando esta classe.

2.4 Novos ambientes

2.4.1 `\folhaderosto`

A folha de rosto recebe um texto já definido dependendo do tipo de texto escrito (monografia, dissertação ou tese). Este texto pode ser encontrado no manual da biblioteca e deve ser colocado entre o início e o fim do ambiente. Por exemplo,

```

\begin{folhaderosto}
Dissertação de Mestrado apresentada ao Centro Universitário
da FEI para obtenção do título de Mestre em Engenharia Elétrica,
orientado pelo Prof. Dr. Nome do Orientador.
\end{folhaderosto}

```

2.4.2 `\resumo`

O ambiente `resumo` funciona da mesma forma que o ambiente `abstract`, sendo a única diferença que o `abstract` possui o comando `\selectlanguage{english}` no início e o `resumo` utiliza `\selectlanguage{brazil}`.

2.4.3 `\agradecimentos`

O ambiente de agradecimentos não possui nenhuma propriedade especial, somente centraliza o título e deixa o texto que se encontra entre seu `begin` e `end` na formatação esperada.

2.5 Novos comandos

2.5.1 `\subtitulo{ }`

Uma vez que as normas da biblioteca demandam formatações específicas para o título e subtítulo do documento (título em letras maiúsculas na capa, seguido do subtítulo em letras normais, separados por “:”), foi criado o comando `\subtitulo{ }`, o qual recebe o texto referente ao subtítulo do texto. Este comando pode ser usado, preferencialmente, após o comando `\title{ }` no preâmbulo do documento. Título e subtítulo também aparecem na folha de rosto.

2.5.2 `\sumario,\figuras,\tabelas`

O \LaTeX já possui comandos que criam sumário, lista de figuras e lista de tabelas, porém, para seguir o modelo necessário e facilitar a manutenção do mesmo foram criados novos comandos que geram estas listas.

As substituições estão visíveis na tabela 2.2.

Tabela 2.2 – Comandos originais e suas substituições

Original	Novo
<code>\tableofcontents</code>	<code>\sumario</code>
<code>\listoffigures</code>	<code>\figuras</code>
<code>\listoftables</code>	<code>\tabelas</code>

Fonte: Autor

2.5.3 `\folhadeaprovacao{}`

O comando para a folha de aprovação pode receber o argumento `ata.pdf`. Se este argumento foi passado, o comando tenta importar uma página em PDF com o mesmo nome para utilizar como folha de aprovação. Se o argumento não for passado ou for diferente de `ata.pdf`, será inserido um texto no lugar da página para marcar a posição da folha de aprovação.

As possibilidades de utilização do comando são:

- a) `\folhadeaprovacao{}`: insere uma página com o texto, como descrito;
- b) `\folhadeaprovacao{ata.pdf}`: insere o PDF com a ata da banca.

2.5.4 `\fichacatalografica{}`

Este comando segue o mesmo princípio do `\folhadeaprovacao{}`, mas espera que o argumento passado seja `ficha.pdf`. As possibilidades de utilização do comando são:

- a) `\fichacatalografica{}`: insere uma página com o texto, como descrito;
- b) `\fichacatalografica{ficha.pdf}`: insere o PDF com a ficha catalográfica fornecida pela biblioteca.

Estes dois comandos são os únicos que dependem do pacote `pdfpages` (para importar a página em PDF).

2.5.5 `\dedicatoria{}`

O comando `\dedicatoria{}` recebe um argumento com a dedicatória desejada e o insere na posição especificada pelo manual da biblioteca. Por exemplo:

`\dedicatoria{A quem eu quero dedicar o texto}.`

2.5.6 `\epigrafe{ }{ }`

A epígrafe possui um formato especial, da mesma forma que a dedicatória. Este comando recebe dois parâmetros, sendo o primeiro a epígrafe e o segundo o autor da mesma.

Nota: O manual da FEI requer que a referência da epígrafe esteja presente no final do trabalho. O comando `\nocite{obra}` pode ser usado para que a referência apareça ao final do texto, sem aparecer na epígrafe.

Exemplo: `\epigrafe{Haw-Haw!}{Nelson Muntz \nocite{muntz_book}}`

2.5.7 `\aspas{ }`

As aspas no \LaTeX são geradas de forma diferente dos outros editores de texto e pode ser encontrada em qualquer manual sobre \LaTeX . Apenas para facilitar a inserção de aspas no formato do \LaTeX , foi criado o comando `\aspas{ }` que recebe o texto desejado e o coloca entre aspas.

Exemplo: `\aspas{Texto entre aspas} → “Texto entre aspas”`

2.5.8 `\marca`

É comum precisar que certas células de uma tabela precisem ser destacadas das demais, como em cronogramas, por exemplo. O comando `\marca` foi feito para que a célula de uma tabela ficasse com o fundo cinza. Este é o único comando que utiliza o pacote `xcolor`.

Exemplo:

```
\begin{table}[ht]
  \begin{center}
    \begin{tabular}{|c|c|c|}
      \hline
      1 & 2 & 3 \\
      \hline
      \marca & & \marca \\
      \hline
      a & b & c \\
      \hline
    \end{tabular}
  \end{center}
\end{table}
```

```
\end{center}
\end{table}
```

Resultado:

1	2	3
a	b	c

2.5.9 \palavraschave{} e \keyword{}

Segundo o modelo da biblioteca da FEI, o resumo e o abstract devem receber no máximo 3 palavras chave. Estes comandos devem ser utilizados dentro dos respectivos ambientes e as palavras devem ser passadas como argumentos.

Exemplo:

```
\begin{resumo}
Aqui deve ser escrito o resumo do trabalho.

\palavraschave{Resumo. Modelo da FEI. Latex}
\end{resumo}
```

2.5.10 \apendice{}, e \anexo{}

Para os apêndices e anexos foram criados dois comandos separados. Eles devem ser utilizados durante a abertura do apêndice e o argumento entre chaves leva o nome do apêndice ou anexo, assim como em capítulos, seções etc.

Exemplos:

```
\apendice{Primeiro apêndice}
[...]
\apendice{Segundo apêndice}
[...]
\anexo{Único anexo do trabalho}
```

2.5.11 `\bibliografia{}`

A utilização de referências bibliográficas a partir do `bibtex` depende do comando `\bibliography{}` que recebe o caminho até o arquivo `bib` utilizado. Porém, a adição da página de referências ao sumário e a formatação do título da mesma dependem de outras variáveis que precisam ser definidas durante a produção do texto (o pacote `babel` substitui o nome da página de referências e este só pode ser mudado após o início do texto).

Para facilitar, foi criado o comando `\bibliografia{}` que recebe como parâmetro o caminho para o arquivo `bib`. Este comando realiza a formatação necessária e repassa o caminho para o comando `\bibliography{}` padrão do \LaTeX .

Exemplo: `\bibliografia{minha_bibliografia.bib}`

3 REFERÊNCIA USANDO O ABNT_{TEX}

3.1 O que é o abn_{TEX} e como foi utilizado

O abn_{TEX} (<https://code.google.com/p/ABNTeX2/>) é um conjunto de macros (comandos e ambientes) que busca seguir as normas da ABNT para formatos acadêmicos. O pacote completo do abn_{TEX} fornece tanto uma classe para a formatação do texto quanto um pacote para a formatação das referências bibliográficas.

Entretanto, a ABNT fornece certas opções para que o texto seja produzido, sendo que a biblioteca do Centro Universitário da FEI ficou a cargo de escolher estas formatações para seus trabalhos.

Para a formatação correta das citações e referências de acordo com o padrão da biblioteca da FEI, foi importado o pacote `abntex2cite-alf`, que utiliza o modelo autor-data.

As seções a seguir disponibilizam exemplos dos comandos mais comuns. Para uma lista detalhada, o leitor é referenciado ao manual do `abntex2cite-alf`.

3.2 Citação no final de linha

A citação no final de linha deve deixar os nomes dos autores, seguido do ano, entre parênteses e em letras maiúsculas. Este resultado pode ser obtido utilizando o comando `\cite{obra}`.

Exemplo: Este texto deveria ser uma referência `\cite{j:turing50}`. → Este texto deveria ser uma referência (TURING, 1950).

3.3 Citação durante o texto

Para que a citação seja feita durante o texto, o nome do autor é formatado somente com as iniciais maiúsculas e o ano entre parênteses. O pacote da abn_{TEX} fornece o comando `\citeonline{obra}` para este caso.

Exemplo: Segundo `\citeonline{haykin99a}`, este texto deveria ser uma referência. → Segundo Haykin (1999), este texto deveria ser uma referência.

3.4 Citação indireta

Quando se deseja citar uma obra a qual o autor não possui acesso direto a ela, pode-se citar uma outra obra que, por sua vez, cita a primeira. O abnT_EX disponibiliza esse tipo de citação através do comando `\apud{obra_inacessivel}{obra_acessivel}`.

Exemplo: `\apud{Mcc43}{RusselNo10}` formata a citação de forma semelhante a (MCCULLOCH; PITTS, 1943 apud RUSSELL; NORVIG, 2010).

3.5 Citação no rodapé

Citações no rodapé¹ são feitas usando o comando `\footciteref{obra}`.

3.6 Citações múltiplas

Os comandos `\cite{obra_1, ..., obra_n}` e `\citeonline{obra_1, ..., obra_m}` também possibilitam a utilização de citações múltiplas.

Exemplos:

`\cite{Mcc43, RusselNo10, haykin99a}` → (MCCULLOCH; PITTS, 1943; RUSSELL; NORVIG, 2010; HAYKIN, 1999).

`\citeonline{Mcc43, RusselNo10, haykin99a}` → McCulloch e Pitts (1943), Russell e Norvig (2010), Haykin (1999).

3.7 Citações de campos específicos

Para citar o nome do autor em linha, utilize o comando `\citeauthoronline{obra}`.

`\citeauthoronline{galilei_dialogue_1953}` → Galilei

Para citar o nome do autor em letras maiúsculas, utilize `\citeauthor{obra}`.

`\citeauthor{galilei_dialogue_1953}` → GALILEI

Para citar o ano de uma obra, utilize `\citeyear{obra}`.

`\citeyear{galilei_dialogue_1953}` → 1953

¹TURING, A. M. Computing machinery and intelligence. **Mind**, v. 59, p. 433–460, 1950.

3.8 Outros exemplos

```

\Idem[p.~2]{j:turing50} → Id., 1950, p. 2
\Ibidem[p.~2]{j:turing50} → Ibid., p. 2
\opcit[p.~2]{j:turing50} → TURING, op. cit., p. 2
\passim{j:turing50} → TURING, 1950, passim
\loccit{j:turing50} → TURING, 1950, loc. cit.
\cfcite[p.~2]{j:turing50} → Cf. TURING, 1950, p. 2
\etseq[p.~2]{j:turing50} → TURING, 1950, p. 2, et seq.

```

3.9 Citações com mais de três linhas

O único tipo de citação que independe do pacote `abntex2cite` é a citação com mais de três linhas. De acordo com o manual da biblioteca, ela deve ter recuo de 4 cm da margem esquerda, letra de tamanho 10 pt, espaçamento simples e não deve conter aspas nem recuo ao início do parágrafo. No \LaTeX , os ambientes responsáveis por tais citações são `quote` (para citações de um parágrafo) e `quotation` (para citações com mais de um parágrafo).

Exemplo:

I propose to consider the question, ‘Can machines think?’ This should begin with definitions of the meaning of the terms ‘machine’ and ‘think’. The definitions might be framed so as to reflect so far as possible the normal use of the words, but this attitude is dangerous. If the meaning of the words ‘machine’ and ‘think’ are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, ‘Can machines think?’ is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words. (TURING, 1950)

4 ÍNDICES USANDO MAKEINDEX

Para a criação de índices remissivos, foi importado o pacote `imakeidx`. Este pacote utiliza o programa `makeindex` na criação de arquivos auxiliares que indexam as palavras a serem colocadas no índice. Normalmente, o `makeindex` vem junto com a instalação do \LaTeX .

Nota: pelo fato do `makeindex` não processar palavras que não sejam do idioma inglês muito bem, foi criado um sucessor espiritual para ele, o `xindy`. No entanto, o `xindy` não possui versões para Windows e Mac OS, então foi optado utilizar o `makeindex` neste projeto. Para aprender dicas avançadas na criação de índices mais complexos, recomenda-se a leitura da documentação do `makeindex` (<http://www.ctan.org/pkg/makeindex>).

4.1 Indexando palavras

Para que uma palavra apareça posteriormente no índice, é necessário indexá-la. Para isso, usa-se o comando `\index{palavra}`, o qual inclui “palavras” no arquivo auxiliar de indexação.

Exemplo: [...] a biblioteca do Centro Universitário da `\index{FEI}`FEI utiliza um modelo baseado na norma da `\index{ABNT}`ABNT [...]

É possível indexar uma palavra mais de uma vez, para que todas as páginas nas quais esta palavra apareceu apareçam no índice.

4.2 Imprimindo o índice

A impressão do índice é feita utilizando o comando `\indice`, o qual, além de imprimir o índice, inclui uma entrada para o mesmo no sumário.

5 LISTAS DE SÍMBOLOS E ABREVIATURAS

Para a criação das listas de símbolos e abreviaturas, foi utilizado o pacote `glossaries`, responsável por indexar termos de diferentes categorias e gerar listas destes termos. Ao contrário do índice, que indexa as palavras no decorrer do texto, o pacote `glossaries` exige que os termos sejam declarados antes de serem referenciados durante o texto. Uma boa prática para organizar tais termos consiste em declará-los ao início do documento, ou em um documento separado, o qual pode ser chamado utilizando os comandos `\input` ou `\include`. Estas opções ficam a cargo do leitor. As próximas duas seções ensinarão os comandos básicos para indexação de símbolos e abreviaturas.

Nota: assim como descrito no capítulo 4, o pacote `glossaries` depende das ferramentas `makeindex`.

5.1 Indexando abreviaturas

A indexação de abreviaturas é feita utilizando o comando

`\newacronym[longplural=1]{2}{3}{4}`, onde:

- a) 1: o significado a abreviatura no plural, escrito por extenso (*opcional*);
- b) 2: código que será utilizado para referenciar a abreviatura no decorrer do texto;
- c) 3: a abreviatura em si;
- d) 4: o significado a abreviatura, escrito por extenso.

Exemplo: `\newacronym[longplural=Associações
Brasileiras de Normas Técnicas]
{abnt}{ABNT}{Associação Brasileira de Normas Técnicas}`

5.2 Indexando símbolos

A indexação de símbolos é feita utilizando o comando

`\newglossaryentry{1}{type=simbolos,
name={2}, sort=3, description={4}}`, onde:

- a) 1: código que será utilizado para referenciar a abreviatura no decorrer do texto);

- b) 2: o símbolo; caso a notação matemática seja necessária, use `\ensuremath{2}` (Cf. exemplo abaixo);
- c) 3: uma sequência de caracteres para indicar a ordenação alfabética do símbolo na lista;
- d) 4: a descrição do símbolo, que aparecerá na lista.

Exemplo: `\newglossaryentry{pi}{type=simbolos,
name={\ensuremath{\pi}}, sort=p,
description={número irracional que representa [...]}}`

5.3 Utilizando abreviaturas e símbolos indexados

O pacote `glossaries` disponibiliza os seguintes comandos para chamar os itens indexados durante o texto:

- a) `\gls{codigo}`: imprime a entrada em letras minúsculas;
- b) `\Gls{codigo}`: imprime a entrada em letras maiúsculas;
- c) `\glspl{codigo}`: imprime a entrada no plural;
- d) `\Glspl{codigo}`: imprime a entrada no plural e em letras maiúsculas.

Repare que, no caso das siglas, quando estas são usadas pela primeira vez, são impressas a definição seguida da sigla entre parênteses. Nas demais vezes, a sigla aparecerá sozinha. É importante ressaltar que o pacote `glossaries` adiciona às listas somente os termos que foram utilizados durante o texto. Para que todos os termos declarados apareçam, basta usar o comando `\glsaddall` no corpo do texto.

5.4 Imprimindo as listas

O comando `\printglossaries` imprime ambas as listas em sequência.

6 COMPILANDO O PROJETO

Para a utilização de todos os recursos que a classe `fei.cls` disponibiliza, é necessário compilar o projeto utilizando diversas ferramentas diferentes em ordem específica:

$$\underset{a}{\text{pdfL}\text{\texttt{A}}\text{T}\text{\texttt{E}}\text{X}} \rightarrow \underset{b}{\text{BibT}\text{\texttt{E}}\text{X}} \rightarrow \underset{c}{\text{MakeIndex}} \rightarrow \underset{d}{\text{MakeGlossaries}} \rightarrow \underset{e}{\text{pdfL}\text{\texttt{A}}\text{T}\text{\texttt{E}}\text{X}} \rightarrow \underset{f}{\text{pdfL}\text{\texttt{A}}\text{T}\text{\texttt{E}}\text{X}}$$

onde:

- a) gera o PDF e arquivos auxiliares básicos;
- b) lê os arquivos auxiliares criados em *a*, gerando a bibliografia apenas com as referências utilizadas do arquivo `bib` utilizado. Necessário apenas se citações e referências forem usadas no texto;
- c) lê os arquivos auxiliares, criando um ou mais arquivos de índice. Necessário apenas se houve indexação de palavras para serem adicionadas ao índice;
- d) lê os arquivos auxiliares, criando um ou mais arquivos de listas. Necessário apenas se houve indexação e utilização de símbolos e abreviaturas no texto;
- e) atualiza todas as referências através do texto, utilizando os arquivos gerados em *b*, *c* e *d* (*desnecessário se os passos b–d não foram realizados*);
- f) gera o PDF final (*desnecessário se os passos b–e não foram realizados*).

Um exemplo de comando que executa todas essas funções é o seguinte:

```
pdflatex manual.tex
bibtex manual.aux
makeindex manual.idx
makeglossaries manual
pdflatex manual.tex
pdflatex manual.tex
```

6.1 Compilação utilizando arara

O `arara` (cereda.github.io/arara/) é uma ferramenta que automatiza a compilação de projetos em $\text{\texttt{L}}\text{\texttt{A}}\text{\texttt{T}}\text{\texttt{E}}\text{X}$. Seu trabalho consiste em compilar um arquivo `tex` utilizando

comentários colocados nas primeiras linhas do arquivo como instruções. Esta abordagem permite a abstração por parte do usuário de chamadas complexas a programas como `makeindex` e `glossaries`. Abaixo, um exemplo de como os comandos para o processo de compilação exemplificado acima ficariam em `arara`:

```
% arara: pdflatex
% arara: makeindex
% arara: makeglossaries
% arara: bibtex
% arara: pdflatex
% arara: pdflatex
```

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

E a chamada ao `arara` seria feita da seguinte forma:

```
arara manual.tex
```

A escolha de como compilar o projeto fica a cargo do leitor.

7 INSTALAÇÃO DOS PACOTES E PROGRAMAS

Este capítulo guia o leitor na instalação dos diferentes pacotes e programas necessários para utilizar todas as funcionalidades da classe `fei.cls`.

7.1 Windows

A opção mais simples para instalação do \LaTeX no Windows é o aplicativo Mik \TeX (<http://miktex.org>). Tenha certeza de escolher a opção que permite ao software baixar pacotes em falta do repositório online e, na primeira vez que compilar seu projeto, todos os pacotes serão baixados.

Alternativamente, é possível utilizar o gerenciador de pacotes do Mik \TeX para selecionar os pacotes a serem baixados. A lista destes pacotes está disponível na seção 2.1.

7.2 Ubuntu

No Ubuntu, é necessária a instalação dos seguintes pacotes através do `apt-get`:

- a) `texlive`: pacotes essenciais do \TeX Live;
- b) `texlive-science`: instala pacotes científicos, como `algorithm2e` e `mathtools`;
- c) `texlive-lang-portuguese`: idioma português do `babel`;
- d) `texlive-publishers`: pacote `abntex2cite` da `abn\text{\TeX}`.
- e) `texlive-extra-utils`: `arara`;

REFERÊNCIAS

DYSON, F. **Disturbing the Universe**. Basic Books, 1979. (Basic Books 50 years). ISBN 9780465016778. Disponível em: <<http://books.google.com.br/books?id=RHzoMeU2bxsC>>.

GALILEI, G. **Dialogue concerning the two chief world systems: Ptolemaic and Copernican**. [S.l.]: University of California Pr, 1953.

HAYKIN, S. **Neural Networks: A Comprehensive Foundation**. [S.l.]: Prentice Hall, 1999.

MCCULLOCH, W.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **Bulletin of Mathematical Biophysics**, n. 5, p. 115–133, 1943.

RUSSELL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach**. 3. ed. [S.l.]: Prentice Hall, 2010.

TURING, A. M. Computing machinery and intelligence. **Mind**, v. 59, p. 433–460, 1950.

APÊNDICE A - ARQUIVOS CRIADOS PELO PROCESSO DE COMPILAÇÃO

A seguir, as descrições dos arquivos auxiliares gerados durante o processo de compilação de um documento utilizando a classe `fei.cls` e todos os seus recursos.

a) `pdflatex manual.tex`

- i) `alg`: *log* do `makeindex`;
- ii) `aux`: arquivo com as referências a serem processadas pelo `BibTeX`;
- iii) `glo`, `acn`, `sym`: listas de abreviaturas e símbolos.
- iv) `idx`: arquivo com os termos a serem adicionados no índice pelo `MakeIndex`;
- v) `loa`: lista de algoritmos;
- vi) `out`: atalhos (*bookmarks*) utilizados pelo leitor de PDF.
- vii) `toc`: sumário;

b) `bibtex manual.aux`

- i) `bbl`: arquivo contendo as citações utilizadas no texto, prontas a serem incluídas na próxima execução do `pdfLATEX`;
- ii) `blg`: *log* do `BibTeX`.

c) `makeindex manual.idx`

- i) `ilg`: *log* do `MakeIndex`.
- ii) `ind`: contém, em linguagem *tex*, a formação do índice a ser inserida na chamada a `\printindex`;

d) `makeglossaries manual`

- i) `acr`, `sbl`, `gls`: contém, em linguagem *tex*, a formação das listas a serem inseridas na chamada a `\printglossaries`;
- ii) `glg`: *log* do `MakeGlossaries`.

ÍNDICE

A

ABNT, 12–14, 23, 26

F

FEI, 12, 13, 21, 23, 26

L

L^AT_EX, 12, 13, 17, 18, 20, 22, 25, 29