

# 1 Introdução

Este material é apresentado como parte de uma série de minicursos que serão ministrados sobre tópicos variados com o intuito de *nivelar* o conhecimento dos alunos com relação a conceitos básicos que não estão inclusos nas ementas das disciplinas regulares do curso, bem como colocá-los em contato com ferramentas úteis e boas práticas da Computação.

Os conteúdos são dispostos aqui apenas como um apanhado geral com o objetivo de prover um embasamento mínimo e recomenda-se que outras fontes de pesquisa sejam utilizadas para o aprofundamento nestes assuntos.

## 1.1 A escrita científica

A necessidade da escrita científica é constante. Na universidade, ela é tida como o principal meio de troca de informações no processo de difusão do conhecimento. Autores a utilizam em seus livros, pesquisadores divulgam suas descobertas através da publicação em revistas científicas, professores verificam o desempenho de seus alunos em avaliações escritas e os próprios acadêmicos a utilizam durante seus estudos. Na indústria, a escrita é valorizada na forma de relatórios técnicos, documentação de software e especificação de produtos.

Devido à sua importância, é natural que existam certas diretivas para se produzir um texto de forma correta e aqueles que não estão habituados a estas regras muitas vezes encontram dificuldades. Um problema frequente é o da apresentação visual dos documentos. No ambiente acadêmico, o interesse está no conteúdo de um trabalho e não em sua estética. Os artigos devem ser formatados seguindo uma norma padrão, geralmente definida pela instituição ou evento/revista no qual o trabalho está sendo divulgado.

## 1.2 Normas para escrita de trabalhos científicos

As normas são definidas não de maneira aleatória, mas levando em consideração vários fatores como legibilidade do texto, tamanho do documento, local de publicação, presença de fórmulas matemáticas ou imagens, etc. Alguns dos detalhes presentes nas normas para escrita de artigos e trabalhos acadêmicos são: requisitos com relação à

fontes, à disposição de parágrafos e seções, exibição de fórmulas e figuras, formatação de sumário e bibliografia.

Como estes detalhes são difíceis de serem capturados em palavras e regras, mostra-se abaixo um mesmo documento formatado de maneira a atender as regras de formatação do *Institute of Electrical and Electronics Engineers* (**IEEE**, acima) e da Sociedade Brasileira de Computação (**SBC**, abaixo).

Observe na página seguinte as diferenças com relação aos elementos de formatação do texto:

- Margens.
- Estilos e tamanhos de fonte.
- Centralização.
- Forma de apresentar o nome dos autores.
- Numeração das seções.
- Tabulação dos parágrafos.

Além disso, outro detalhe importante é a seção de referências. Cada congresso dispõe de maneiras diferentes para apresentação e citação das mesmas. Após a comparação entre a formatação dos textos, mostra-se, na página 4 as diferenças entre as referências das normas.

Repare como a citação do IEEE, na parte superior, ainda utiliza o formato de duas colunas, possui citação numérica, cita mais do que um autor e mostra maiores informações sobre a publicação em si. A norma da SBC, no meio, se parece mais com o padrão da UTFPR por ambas terem base na ABNT. As citações são por nome do autor e ano da publicação, e com menos informações sobre a obra. Por fim, as normas da UTFPR ressaltam o congresso publicado em negrito, autor todo em maiúsculo e mais metadados.

# Comparação básica entre as normas ABNT, IEEE e SBC para elaboração de artigos

CASELLA, G. B.<sup>\*</sup>, HAN, L. Y.<sup>†</sup>, CARVALHO, W.<sup>‡</sup>

Departamento de Ciência da Computação

UTFPR, Ponta Grossa, Paraná

{\*gbc921, †nilyuhan, ‡wevertoncarvalho34}@gmail.com

**Resumo**—Este artigo faz uma comparação entre as três diretrizes mais utilizadas para a elaboração de artigos científicos no Brasil. ABNT, IEEE e SBC. As regras para a produção de artigos científicos são brevemente apresentadas e comparadas através de uma tabela.

## I. INTRODUÇÃO

Norma pode ser definida como um documento onde se contém especificação técnica ou outros critérios precisos desenvolvidos para serem utilizados consistentemente com uma regra, diretriz ou definição. Logo ela proporciona uma maior simplicidade, aumentando a confiabilidade e a efetividade na elaboração de um artigo. As mesmas normalmente são desenvolvidas por um órgão oficial voltada para tal.

em torno de 425 mil membros em 160 países e foi fundada em 1963

As normas do IEEE visam tornar padrão as formas de publicação, criando um consenso único em que o foco está no conteúdo e não no texto. Suas políticas também são criadas com base no retorno que essa publicação terá ao autor da obra.

## IV. SBC

A SBC (Sociedade Brasileira de Computação) é a sociedade científica sem fins lucrativos, que reúne pesquisadores, professores, estudantes e profissionais da computação. Ela faz parte da SBPC (Sociedade Brasileira para o Progresso da Ciência) e da IFIP (*International Federation for Information Processing*).

A Sociedade induz o crescimento de pesquisas científicas

# Comparação básica entre as normas ABNT, IEEE e SBC para elaboração de artigos

Gabriel B. Casella, Lin Y. Han, Weverton Carvalho

<sup>1</sup>Departamento de Informática – Universidade Tecnológica Federal do Paraná - UTFPR

{gbc921, nilyuhan, wevertoncarvalho34}@gmail.com

**Resumo.** Este artigo faz uma comparação entre as três diretrizes mais utilizadas para a elaboração de artigos científicos no Brasil. ABNT, IEEE e SBC. As regras para a produção de artigos científicos são brevemente apresentadas e comparadas através de uma tabela.

## 1. Introdução

Norma pode ser definida como um documento onde se contém especificação técnica ou outros critérios precisos desenvolvidos para serem utilizados consistentemente com uma regra, diretriz ou definição. Logo ela proporciona uma maior simplicidade, aumentando a confiabilidade e a efetividade na elaboração de um artigo. As mesmas normalmente são desenvolvidas por um órgão oficial voltada para tal.

## REFERÊNCIAS

- [1] P. Motta Pires and D. Rogers, "Free/open source software: an alternative for engineering students," in *Frontiers in Education, 2002. FIE 2002. 32nd Annual*, vol. 1, 2002, pp. T3G-7-T3G-11 vol.1.
- [2] A. Gaudoul, "Do open source developers respond to competition? the latex case study," *Review of Network Economics*, vol. 6, no. 2, 2007.
- [3] L. Lamport, "Latex: User's guide & reference manual," 1986. [Online]. Available: <http://www.statistik.lmu.de/~poessnecker/Lehre/SoSe2011/Latexkurs/latex11-bibliography-print.pdf>

---

## Referências

Gaudoul, A. (2007). Do open source developers respond to competition? the latex case study. *Review of Network Economics*, 6(2).

Lamport, L. (1986). Latex: User's guide & reference manual.

Motta Pires, P. and Rogers, D. (2002). Free/open source software: an alternative for engineering students. In *Frontiers in Education, 2002. FIE 2002. 32nd Annual*, volume 1, pages T3G-7-T3G-11 vol.1.

---

## REFERÊNCIAS

GAUDEUL, A. Do open source developers respond to competition? the latex case study. **Review of Network Economics**, v. 6, n. 2, 2007.

LAMPORT, L. Latex: User's guide & reference manual. Addison-Wesley, 1986. Disponível em: <<http://www.statistik.lmu.de/~poessnecker/Lehre/SoSe2011/Latexkurs/latex11-bibliography-print.pdf>>.

PIRES, P. M.; ROGERS, D. Free/open source software: an alternative for engineering students. In: **Frontiers in Education, 2002. FIE 2002. 32nd Annual**. [S.l.: s.n.], 2002. v. 1, p. T3G-7-T3G-11 vol.1. ISSN 0190-5848.

Tendo em vista todos estes detalhes a serem levados em consideração bem como o fato de que os textos muitas vezes são escritos de maneira apressada devido a prazos curtos, como no caso de artigos ou relatórios de iniciação científica, fica claro que a utilização de ferramentas que automatizem e facilitem este processo de formatação serão sempre bem vindas. Uma destas ferramentas é o  $\text{\LaTeX}$ .

## 2 $\text{\LaTeX}$

O  $\text{\LaTeX}$  é um sistema para a editoração de documentos de alta qualidade tipográfica. Utiliza-se do  $\text{\TeX}$ , um programa que foi criado pelo matemático Donald Knuth que, ao revisar o segundo volume da sua série de livros *The Art of Computer Programming* verificou que a sua qualidade era muito baixa e já que a produção de um texto em formato digital era basicamente uma combinação de zeros e uns (com tinta e sem tinta), decidiu criar um sistema próprio.

Knuth inicialmente previu que a sua tarefa, que consistia em aprender quais eram as formas tradicionais de se exibir fórmulas matemáticas, a maneira correta de se imprimir letras em um papel, como criar suas próprias fontes e por fim integrar tudo isso em um programa de computador, levaria aproximadamente 6 meses. Donald levou quase 10 anos e o produto final de seu trabalho foi o sistema  $\text{\TeX}$  e o programa de criação de fontes *Metafont*.

$\text{\LaTeX}$  é baseado na filosofia de que autores devem se focar no conteúdo sem precisar se preocupar com a apresentação visual do que estão escrevendo. Desta forma, ao escrever um documento usando  $\text{\LaTeX}$ , o autor precisa apenas especificar a estrutura lógica de seu texto, na forma de conceitos básicos como capítulos, seções, tabelas, figuras, etc. A apresentação fica a cargo do  $\text{\LaTeX}$ .

### 2.1 Funcionamento

O funcionamento do  $\text{\LaTeX}$  é o seguinte: a partir de um arquivo de texto com extensão `.tex` contendo comandos e o texto que se quer exibir, o programa `latex` produz um arquivo independente de plataforma `.dvi`. Este arquivo então é convertido para diversos formatos dependendo de sua utilização desejada. No caso de uso mais comum, o arquivo `.dvi` é convertido para `.pdf`. Para se evitar este processo de conversão, normalmente utiliza-se o comando `pdflatex` que gera o `.pdf` diretamente a partir do `.tex`.

Um exemplo de sintaxe dos comandos  $\text{\TeX}$  é visto a seguir:

```
\documentclass{article}
```

```

\usepackage[brazil]{babel}

\title{Exemplo de um documento}
\author{Nome Autor}
\date{\today}
\begin{document}
  \maketitle
  \section{Primeiro exemplo}
  Oi. Aqui temos um exemplo de uso do \LaTeX.
\end{document}

```

Este trecho de código produz um .pdf com esta aparência:

# Exemplo de um documento

Nome Autor

29 de janeiro de 2014

## 1 Primeiro exemplo

Oi. Aqui temos um exemplo de uso do  $\text{\LaTeX}$ .

Observe como, no código apresentado, não há nenhuma informação sobre que tipo de fonte utilizar, tamanho da letra, espaçamento entre linhas, posicionamento do título, etc. Estes detalhes são tratados pelo  $\text{\LaTeX}$  e o usuário não precisa se preocupar com eles.

## 2.2 Instalação

O  $\text{\LaTeX}$  nada mais é do que um compilador, convertendo comandos e texto no formato final de apresentação escolhido. Sendo assim, existem compiladores diferentes para cada sistema operacional e várias implementações diferentes baseadas originalmente no  $\text{\LaTeX}$ / $\text{\TeX}$ . Para citar dois exemplos, temos  $\text{\XeTeX}$  (HOSNY, 2004), que possui melhor suporte a fontes e tipografias e  $\text{\LuaTeX}$  (HOEKWATER, 2007) que inclui a linguagem de script Lua e é baseada no  $\text{\pdflatex}$ .

Utilizaremos basicamente nesse minicurso o `pdflatex` que caracteriza-se basicamente por ser uma extensão do `TeX`. Para instalação no Windows normalmente utiliza-se o grupo de pacotes `MiKTeX`(SCHENK, 2009) que pode ser baixada aqui <http://miktex.org/download>. Este pacote pode automaticamente baixar as os plugins (extensões do `LaTeX`) que forem necessárias para funcionamento.

Para ambientes Linux a maioria das distribuições inclui um pacote a ser instalado pelo próprio gerenciador de pacotes da distro. Normalmente é utilizado o `TeXLive` que agrega os compiladores, plugins, macros e fontes.

Nos sistemas Debian e derivados (Ubuntu) o `LaTeX` pode ser instalado pelos pacotes `texlive`, `texlive-base` ou pelo meta-pacote `texlive-full`. Para o Ubuntu, ainda existe um *script* de instalação(KOSTYSHAK, 2013) para obter a última versão do `TeXLive` (atualmente 2013). A URL de acesso com instruções de instalação pode ser acessada em <https://github.com/scottkosty/install-tl-ubuntu>.

Finalmente, para edição dos textos utiliza-se um editor próprio para textos `LaTeX` de forma a facilitar o uso dos comandos e macros disponíveis. Dentre os vários disponíveis(WIKIPEDIA, 2014) utilizaremos aqui o `TeXStudio`(ZANDER et al., 2009) que pode ser baixado em <http://texstudio.sourceforge.net/>. O `TeXStudio` possui várias funções que facilitam na edição de textos, como compilação integrada, visualizador rápido do pdf, *auto-complete* de comandos, dicionário, localização de erros no `.tex`, entre outros.

### 3 WYSIWYG

A maioria dos softwares responsáveis por processamento de palavras utilizam uma interface que permite que o usuário veja algo muito similar ao resultado final enquanto o documento está sendo criado. Essa interface pode ser chamada de WYSIWYG (“What You See Is What You Get”, ou em uma tradução literal, “O que você vê é o que você obtém”). Um exemplo de programa que se encaixa nesta classificação é o **MS Word** ou **LibreOffice**.

Em editores WYSIWYG, pedaços de códigos são inseridos no documento para indicar onde a fonte deve mudar de tamanho, onde usar itálico, ou negrito, etc. Esses pedaços de código não são vistos pelo usuário em momento algum, portanto só é possível editar esse arquivo utilizando o próprio software que o criou. Com o `LaTeX`, o documento consiste de apenas um arquivo de texto, que pode ser modificado com qualquer editor. É possível dizer, então, que WYSIWYW (“What You See Is What You Want”, ou “O que você vê é o que você quer”).

### 3.1 $\text{\LaTeX}$ vs WYSIWYG

Embora existam diversas situações onde a utilização de um programa é preferida à de outro por inúmeras razões, existem alguns motivos que podem servir como motivação na escolha do  $\text{\LaTeX}$  como sua ferramenta para criação de textos.

- Velocidade.

Qualquer ação que requer um mouse ou cliques para acesso de menus será mais lento do que digitar algumas teclas do teclado. Isso significa que digitar caracteres especiais, como uma letra grega, ou principalmente funções serão muito mais rápidos no  $\text{\LaTeX}$ .

Na edição de arquivos muito grandes, e com muitas figuras e equações, o MS Word, por exemplo, se torna muito mais lerdo, já que ações como mover a barra de rolagem de arquivos pesados requerem muito do processador e da memória. Editar um simples arquivo de texto já não causa tantos problemas para a máquina e a modificação pode ser feita rapidamente.

- Segurança.

Os arquivos dos editores comuns são armazenados em uma forma binária. Se este arquivo for corrompido por qualquer motivo, o usuário pode perder muitas horas de trabalho. Com arquivos de texto, como no  $\text{\LaTeX}$ , se um editor falhar, basta abrir o arquivo em outro.

- Separação de conteúdo e formatação.

A separação em seções e subseções de texto no  $\text{\LaTeX}$  torna muito mais fácil para o escritor se concentrar no texto e na sua ordem de apresentação do que em detalhes superficiais como tamanhos de fonte e estilos.

- Integração com Sistema de Controle de Versões.

O  $\text{\LaTeX}$  propicia o uso de sistemas de controles de versão, já que seu formato em texto puro pode ser gerenciado de forma simples por estes sistemas. Desta forma a cooperação entre múltiplos usuários durante a escrita de um texto se torna mais fácil.

- Controle

Alguns editores WYSIWYG possuem configurações prévias que almejam facilitar a digitação de um texto e nestes casos os editores agem sem permissão do usuário, por exemplo: auto-capitalizando as primeiras palavras de uma frase ou



selecionando automaticamente sentenças inteiras. Muitas vezes essas edições são indesejadas como no caso dos sinais de subtração, hífen e travessão. Os símbolos -, – e — representam três coisas diferentes e quando detalhes deste tipo fazem diferença no contexto da frase, como é frequente em textos científicos, o editor de texto acaba se tornando um problema.

- Flexibilidade

Por possuir um formato de texto simples, o  $\text{\LaTeX}$  permite a utilização do vasto ferramental que está disponível para a manipulação de arquivos de texto, como por exemplo, substituição usando expressões regulares. Além disso, este formato permite que um arquivo `.tex` seja transportado de forma flexível e compilado em qualquer ambiente que possua o  $\text{\LaTeX}$  instalado. Modificações no texto também são bastante simples já que o sistema faz a numeração de seções automaticamente, bem como indexação de termos e geração de sumário.

Estas são apenas algumas das vantagens deste sistema sobre os editores de texto convencionais e muitos outros detalhes existem, principalmente quanto mais técnico é o conteúdo da escrita. Porém, não seria correto afirmar que o  $\text{\LaTeX}$  é melhor em todos os sentidos que editores WYSIWYG, já que em várias situações, a simplicidade de operação dos editores convencionais faz com que eles sejam melhores em certas tarefas, como quando se deseja fazer um protótipo de design. As interações estéticas são muito mais fáceis de serem percebidas no editor WYSIWYG, porém a manutenção do conteúdo de acordo com o design se torna mais difícil. A escolha, portanto de uma ferramenta certa para a ocasião e para a aplicação que se deseja é importante.

## 4 Conclusão e dicas finais

Brevemente apresentou-se a parte lógica de como o  $\text{\LaTeX}$  funciona, qual seu foco principal, onde e porque utilizar, instalação básica e suas diferenças entre editores comuns de texto WYSIWYG. Como apontado inicialmente este material não é de nenhuma forma ostensivo e deve ser utilizado apenas de maneira introdutória.

Aprofundamentos podem ser encontrados em <http://tug.org/>, enquanto que maiores detalhes da produção de texto podem ser explorados em <http://en.wikibooks.org/wiki/LaTeX>. Para dúvidas gerais, erros de compilação e questões similares o site `.tex` da StackExchange <http://tex.stackexchange.com/> possui grande base de questões.

Nós da UTFPR-PG também possuímos uma lista de email para dúvidas em gerais sobre Software Livre e essa pode ser acessada em [https://groups.google.com/forum/#!forum/psl\\_pg](https://groups.google.com/forum/#!forum/psl_pg) ou simplesmente mandando um email para [psl\\_pg@gmail.com](mailto:psl_pg@gmail.com). Todo o material do minicurso pode ser acessado em <https://bitbucket.org/fabianorosas/minicursos-utfpr/overview>

## Referências

HOEKWATER, H. H. Hartmut Henkel T. *LuaTeX*. 2007. Disponível em: <http://www.luatex.org/>.

HOSNY, J. K. K. *XeTeX*. 2004. Sourceforge. Disponível em: <http://xetex.sourceforge.net/>.

KOSTYSHAK, S. *Install script for TeX Live on Ubuntu*. 2013. Github. Disponível em: <https://github.com/scottkosty/install-tl-ubuntu>.

SCHENK, C. *MiKTeX*. 2009. Disponível em: <http://miktex.org/>.

VOGAN, B.; PLOTNICK, J. *The Transition from High School to University Writing*. 2013. Disponível em: <http://www.writing.utoronto.ca/advice/general/transition-to-university>.

WIKIPEDIA. *Comparison of TeX editors*. 2014. Wikipedia. Disponível em: [http://en.wikipedia.org/wiki/Comparison\\_of\\_TeX\\_editors](http://en.wikipedia.org/wiki/Comparison_of_TeX_editors).

ZANDER, B. van der et al. *TexStudio*. 2009. Sourceforge. Disponível em: <http://texstudio.sourceforge.net/>.