

UNIVERSIDADE ESTADUAL PAULISTA "JÚLIO DE MESQUITA FILHO"

FACULDADE DE CIÊNCIAS - CAMPUS BAURU

DEPARTAMENTO DE COMPUTAÇÃO

BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

NOME DO AUTOR

TÍTULO DO TEXTO

BAURU

Outubro/2018

NOME DO AUTOR

TÍTULO DO TEXTO

Trabalho de Conclusão de Curso do Curso
de Ciência da Computação da Universidade
Estadual Paulista “Júlio de Mesquita Filho”,
Faculdade de Ciências, Campus Bauru.
Orientador: Prof. Dr. Seu Orientador

BAURU
Outubro/2018

Nome do Autor Título do Texto/ Nome do Autor. – Bauru, Outubro/2018-
25 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Seu Orientador

Trabalho de Conclusão de Curso – Universidade Estadual Paulista “Júlio de
Mesquita Filho”

Faculdade de Ciências

Ciência da Computação, Outubro/2018.

1. Tags 2. Para 3. A 4. Ficha 5. Catalográfica

Nome do Autor

Título do Texto

Trabalho de Conclusão de Curso do Curso de Ciência da Computação da Universidade Estadual Paulista "Júlio de Mesquita Filho", Faculdade de Ciências, Campus Bauru.

Banca Examinadora

Prof. Dr. Seu Orientador

Orientador

Universidade Estadual Paulista "Júlio de Mesquita Filho"

Faculdade de Ciências

Departamento de Ciência da Computação

Professor Convidado 1

Universidade Estadual Paulista "Júlio de Mesquita Filho"

Faculdade de Ciências

Departamento de Ciência da Computação

Professor Convidado 2

Universidade Estadual Paulista "Júlio de Mesquita Filho"

Faculdade de Ciências

Departamento de Ciência da Computação

Bauru, _____ de _____ de _____.

Espaço destinado à dedicatória do texto.

Agradecimentos

Espaço destinado aos agradecimentos.

Espaço destinado à epígrafe.

Não esquecer autor

Resumo

Espaço destinado à escrita do resumo.

Palavras-chave: Palavras-chave de seu resumo.

Abstract

Abstract area.

Keywords: Abstract keywords.

Lista de figuras

Figura 1 – Exemplo do ambiente TeXworks.	21
Figura 2 – Imagem 1 da minipage	21

Lista de quadros

Quadro 1 – Exemplo de quadro 18

Lista de tabelas

Tabela 1 – Exemplo de transações de mercado.	17
--	----

Lista de códigos

1	Exemplo de código em Python	19
---	---------------------------------------	----

Lista de abreviaturas e siglas

ABNT	Associação Brasileira de Normas Técnicas
abnTeX	ABsurdas Normas para TeX
ONU	Organização Nacional da União
SIRGAS2000	Sistema de referência geocêntrico para as américas época 2000.4
SUPRE-MISS	Suicide Prevention Multisite Intervention Study on Suicidal Behaviors
UNESP	Universidade Estadual Paulista "Júlio de Mesquita Filho"

Lista de símbolos

Γ	Letra grega Gama
Λ	Lambda
\in	Pertence
π	Número pi
ζ	Letra grega minúscula zeta

Sumário

	Lista de códigos	12
1	INTRODUÇÃO	16
1.1	Modificadores de Texto	16
1.2	Seções	17
1.2.1	Subseções	17
1.2.1.1	Sub-subseções	17
1.3	Alíneas	17
1.4	Tabelas	17
1.5	Quadros	18
1.6	Algoritmos	18
1.7	Códigos	19
1.8	Figuras	20
1.9	Equações	22
1.10	Como citar as referências	22
1.11	Entrada de Siglas e Símbolos	23
1.12	Notas	23
2	CONCLUSÃO	24
	REFERÊNCIAS	25

1 Introdução

Para iniciar a produção em .tex é necessário instalar os pacotes básicos da linguagem e seus compiladores. O MiKTeX é um pacote básico para o Windows (miktex.org/download) e o MacTeX um pacote básico para o Mac (tug.org/mactex) que contém o mínimo necessário de TeX/LaTeX para rodar. Ele já vem com os compiladores nativos da linguagem e uma IDE (TeXworks, para edição do texto) que possui o compilador integrado.

Normalmente é utilizado o modo pdfLaTeX + MakeIndex + BibTeX para compilar um arquivo .tex. Existem outros formatos de compiladores, mas essa opção é capaz de gerar um .pdf automático após a compilação e ainda por cima adicionar as funcionalidades do BibTeX (recursos para criação e montagem automática de fontes bibliográficas).

Além disso, também é necessária a instalação do pacote abnTeX2. Esse tutorial <https://github.com/abntex/abntex2/wiki/Instalacao> provém o passo a passo de como instalar cada componente do TeX, em qualquer sistema operacional (Linux, Mac OS e Windows). Caso esteja utilizando o MiKTeX, ele é capaz de efetuar o download do pacote automaticamente, apenas instale-o, abra o projeto.tex e compile-o, ele irá requisitar a autorização para baixar automaticamente os pacotes que faltam para efetuar a compilação.

Com tudo em mãos e o compilador funcionando, é hora de abrir o modelo (projeto.tex) e começar a escrever o texto. É possível perceber no código a estrutura do arquivo e os campos possíveis de edição. Ao escrever o texto, ele é escrito normalmente, sendo que existem diversos comandos para estilizá-lo, criar tabelas, figuras, dentre outros. A seguir abordaremos os principais comandos e funções que podem ser utilizadas em um projeto básico de TCC. Para outras funções e pacotes, procure no Google, a comunidade é ativa e provavelmente já deve ter feito o que é de sua necessidade.

O arquivo projeto.tex contém os pacotes e comandos básicos que definem a estrutura desse texto já no formato requisitado pela ABNT. Dentro dele é possível ver que estamos importando outros dois arquivos .tex (introducao e conclusao), ou seja, esses arquivos estão sendo basicamente concatenados com o comando “input”. A divisão não é necessária, mas pode ser que auxilie na escrita do texto ao deixar as coisas mais separadas e organizadas, não sendo um único arquivo cheio de linhas e linhas de código.

1.1 Modificadores de Texto

Os modificadores de texto mais simples utilizados são o negrito (“textbf”) **texto em negrito** e o itálico (“emph”) *texto em itálico*.

1.2 Seções

Seções podem ser criadas a partir do comando “section” e hierarquizadas abaixo do capítulo principal. É possível referenciá-las, por exemplo, Seção 1.2 corresponde a seção atual em que estamos. Já se quisermos referenciar alguma outra coisa, é só utilizarmos o comando “ref” presente no código desse texto, por exemplo, Capítulo 1.

1.2.1 Subseções

Subseções também podem ser criadas com o comando “subsection” e referenciadas 1.2.1.

1.2.1.1 Sub-subseções

Também há mais um nível que pode ser criado com o comando “subsubsection”.

1.3 Alíneas

- a) As alíneas devem ser criadas desse modo, com o comando `begin{alíneas}`. Isso é necessário para que estejam no formato definido pelo pacote `abnTeX2` e, consequentemente, no formato definido pela ABNT.
- b) Cada item da alínea pode ser invocado com um comando `item`.
- c) O fim de cada alínea é determinado por `end{alíneas}`.

1.4 Tabelas

As tabelas também podem ser referenciadas como se fossem seções ou figuras, por exemplo, esta é a Tabela 1.

Tabela 1 – Exemplo de transações de mercado.

TID	Conjunto de Itens
1	{Pão, Leite}
2	{Pão, Fralda, Cerveja, Ovos}
3	{Leite, Fralda, Cerveja, Coca-Cola}
4	{Pão, Leite, Fralda, Cerveja}
5	{Pão, Leite, Fralda, Coca-Cola}

Quando uma tabela é criada com `begin{table}`, ela é automaticamente adicionada à Lista de Tabelas.

1.5 Quadros

Este modelo vem com o ambiente quadro e impressão de Lista de quadros configurados por padrão. Verifique um exemplo de utilização:

Quadro 1 – Exemplo de quadro

Pessoa	Idade	Peso	Altura
Marcos	26	68	178
Ivone	22	57	162
...
Sueli	40	65	153

Fonte: Autor.

Este parágrafo apresenta como referenciar o quadro no texto, requisito obrigatório da ABNT. Primeira opção, utilizando autoref: Ver o [Quadro 1](#). Segunda opção, utilizando ref: Ver o Quadro [1](#).

1.6 Algoritmos

O pacote nicealgo incluído nos arquivos desse projeto é responsável por disponibilizar comandos extras, não inerentes ao básico TeX, para a criação de algoritmos. Um exemplo do Algoritmo [1](#) é escrito a seguir. Eles também pode ser referenciados como se fossem tabelas ou figuras.

Algoritmo 1 – ALGORITMO AIS

ENTRADA: Conjunto Frequente $L = 0$ e Grupo de Fronteira $F = 0$.

```

1.  Enquanto  $F \neq 0$ , faça
2.      Seja conjunto candidato  $C = 0$ ;
3.      Para cada tuplas  $t$  da base de dados, faça
4.          Para cada conjuntos de itens  $f$  em  $F$ , faça
5.              Se  $t$  contém  $f$ , então
6.                  Seja  $C_f =$  conjuntos de itens candidatos extensões de  $f$  e contidos em  $t$ ;
7.                  Para cada conjunto de itens  $c_f$  em  $C_f$ , faça
8.                      Se  $c_f \in C$ , então
9.                           $c_f.\text{contagem} = c_f.\text{contagem} + 1$ ;
10.                     Se não
11.                          $c_f.\text{contagem} = 0$ ;
12.                          $C = C + c_f$ ;
13.
14.
15.
16.      Seja  $F = 0$ ;
17.      Para cada conjunto de itens  $c$  em  $C$ , faça
18.          Se  $\text{contagem}(c)/\text{tamanho\_db} > \text{minsupport}$ , então
19.               $L = L + c$ ;
20.          Se  $c$  deve ser usado como a próxima fronteira, então
21.               $F = F + c$ ;
22.
23.

```

1.7 Códigos

Códigos podem ser criados a partir do comando `begin{lstlisting}` e `end{lstlisting}`. É possível passar parâmetros para essa função, como por exemplo, a linguagem do código e a legenda dele. Por exemplo: `\begin{lstlisting}[language=Python, caption=Exemplo de código em Python]`

Código 1 – Exemplo de código em Python

```

import numpy as np

def incmatrix(genl1, genl2):
    m = len(genl1)
    n = len(genl2)

```

```

M = None #to become the incidence matrix
VT = np.zeros((n*m,1), int) #dummy variable

#compute the bitwise xor matrix
M1 = bitxormatrix(genl1)
M2 = np.triu(bitxormatrix(genl2),1)

for i in range(m-1):
    for j in range(i+1, m):
        [r,c] = np.where(M2 == M1[i,j])
        for k in range(len(r)):
            VT[(i)*n + r[k]] = 1;
            VT[(i)*n + c[k]] = 1;
            VT[(j)*n + r[k]] = 1;
            VT[(j)*n + c[k]] = 1;

            if M is None:
                M = np.copy(VT)
            else :
                M = np.concatenate((M, VT), 1)

        VT = np.zeros((n*m,1), int)

return M

```

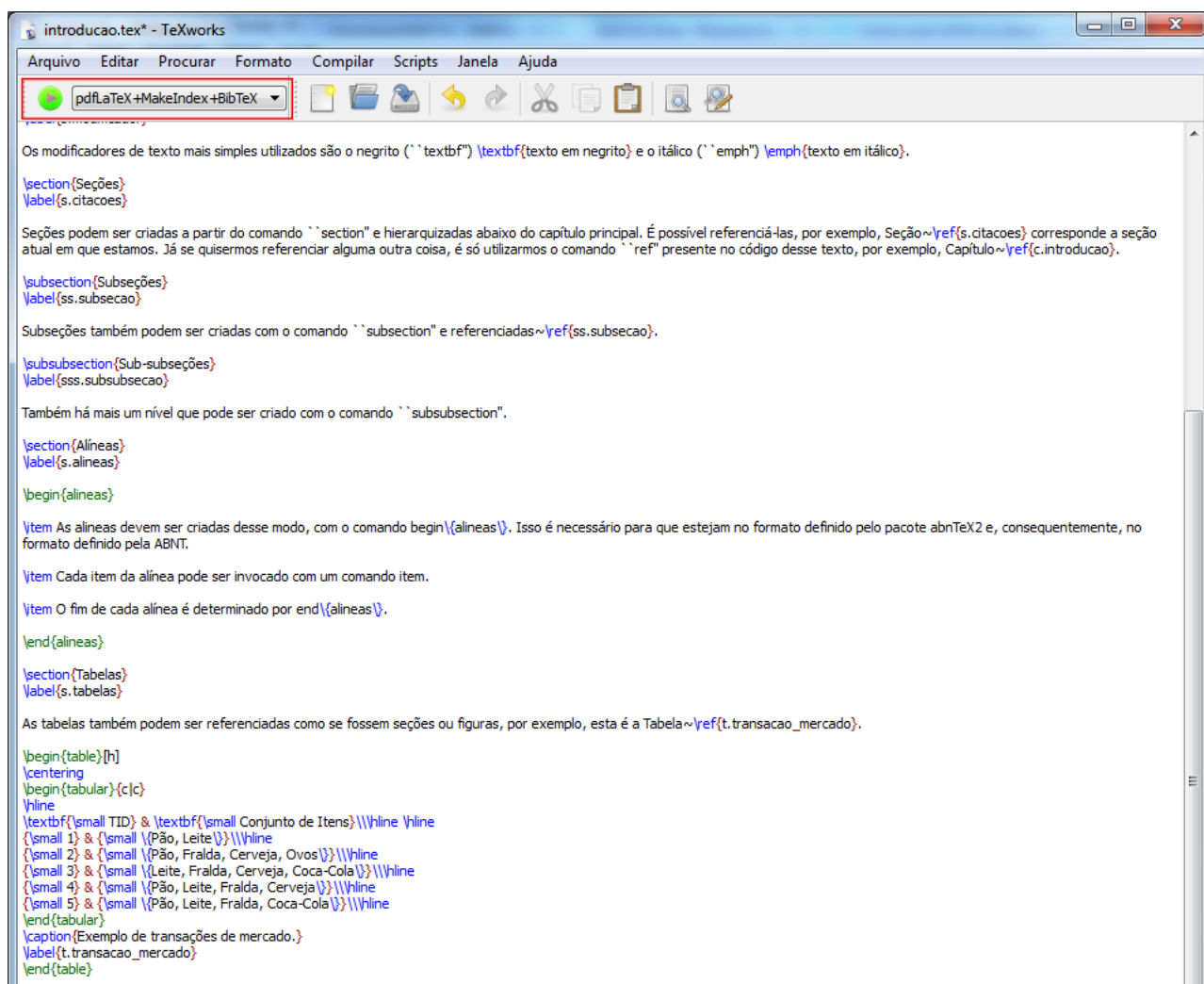
1.8 Figuras

Este parágrafo apresenta como referenciar figura no texto, requisito obrigatório da ABNT. Primeira opção, utilizando autoref: Ver a [Figura 1](#). Segunda opção, utilizando ref: Ver a Figura [1](#).

Atente-se ao código para perceber um possível redimensionamento com a função `scale` e o caminho de onde a figura deve ser retirada.

Quando uma figura é criada com `begin{figure}`, ela é automaticamente adicionada à Lista de Ilustrações.

Figura 1 – Exemplo do ambiente TeXworks.



Fonte: Elaborada pelo autor.

Figura 2 – Imagem 1 da minipage



Fonte: Produzido pelos autores

1.9 Equações

O TeX também é muito famoso pela forma em que consegue tratar funções e símbolos matemáticos. A partir da utilização de dois cifrões (`\$codigo matemático\$`) é possível identificar ao compilador que a escrita a seguir são símbolos e códigos originários do pacote matemático do TeX. Aqui estamos demonstrado um exemplo $\phi = 1 + x$ dessa utilização.

Também podemos definir equações utilizando os comandos `\begin{equation}` e `\end{equation}`. Por exemplo:

$$E(\mathbf{v}, \mathbf{h}) = - \sum_{i=1}^m a_i v_i - \sum_{j=1}^n b_j h_j - \sum_{i=1}^m \sum_{j=1}^n v_i h_j w_{ij}, \quad (1.1)$$

$$P(\mathbf{v}, \mathbf{h}) = \frac{e^{-E(\mathbf{v}, \mathbf{h})}}{\sum_{\mathbf{v}, \mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}}, \quad (1.2)$$

$$\hat{\phi}^j = \begin{cases} \hat{\phi}^j \pm \varphi_j \varrho & \text{com probabilidade PAR} \\ \hat{\phi}^j & \text{com probabilidade (1-PAR).} \end{cases} \quad (1.3)$$

Existem diversos sites no Google que contém códigos de símbolos e funções matemáticas de todos os tipos. Exemplo:

estudijas.lu.lv/pluginfile.php/14809/mod_page/content/16/instrukcijas/matematika_moodle/LaTeX_Symbols.pdf.

1.10 Como citar as referências

Aqui está um exemplo de como podemos referenciar as bibliografias utilizadas no trabalho. Elas são guardadas na forma de metadados (tags) no arquivo `.bib` a qual é importada no projeto principal (`projeto.tex`).

E podemos citá-las de acordo com os identificadores atribuídos para cada referência, por exemplo, (STONEBRAKER et al., 1993) , (ROCHA; CAPPABIANCO; FALCÃO, 2009) e (KERAS. . . , 2018).

Após citar um item de referência bibliográfica com o comando “cite”, ela será automaticamente padronizada e incluída na página de Referências de seu arquivo. Atualmente os maiores sites portadores de artigos, periódicos, dentre outros (IEEE, Springer, etc) já conseguem exportar a publicação desejado no formato BibTeX, sendo facilmente adicionado ao arquivo `.bib` de seu trabalho.

Muitas vezes não é possível exportar publicações diretamente para o formato BibTeX, como, por exemplo, na citação de sites. Para mais informações sobre como criar manualmente arquivos BibTeX: <<https://github.com/abntex/limarka/wiki/Adicionando-refer%C3%AAs>>

1.11 Entrada de Siglas e Símbolos

Quando uma sigla ou símbolo são criados com os comandos abaixo, elas são automaticamente adicionadas à Lista de abreviaturas/siglas ou Lista de símbolos.

Associação Brasileira de Normas Técnicas (ABNT),

De acordo com ABNT...

Suicide Prevention Multisite Intervention Study on Suicidal Behaviors (SUPRE-MISS)

Sistema de referência geocêntrico para as américas época 2000.4 (SIRGAS2000)

Universidade Estadual Paulista "Júlio de Mesquita Filho" (UNESP)

Organização Nacional da União (ONU)

ABsurdas Normas para TeX (abnTeX)

Γ , Λ

ζ

\in

π

A constante π ...

1.12 Notas

- O título de quadros e tabelas são sempre em cima.
- Sempre que possível, utilize o comando autoref.
- todas as figuras da monografia precisam estar referenciadas no texto pelo menos uma vez.

2 Conclusão

Os arquivos estão sendo concatenados. Podemos continuar a nossa escrita em outro arquivo .tex desde que ele seja importado no projeto principal, que é sempre o utilizado para efetuar a compilação.

Referências

KERAS Documentation. 2018. Disponível em: <<<https://keras.io/>>>. Acesso em 22 de Outubro de 2018.

ROCHA, L. M.; CAPPABIANCO, F. A. M.; FALCÃO, A. X. Data clustering as an optimum-path forest problem with applications in image analysis. *International Journal of Imaging Systems and Technology*, Wiley Periodicals, v. 19, n. 2, p. 50–68, 2009.

STONEBRAKER, M.; AGRAWAL, R.; DAYAL, U.; NEUHOLD, E. J.; REUTER, A. Dbms research at a crossroads: The vienna update. In: *VLDB '93 Proceedings of the 19th International Conference on Very Large Data Bases*. [S.l.]: Morgan Kaufmann Publishers Inc., 1993. p. 688–692.