

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
FLUMINENSE *CAMPUS* ITAPERUNA**

BACHARELADO EM SISTEMAS DE INFORMAÇÃO

NOME DO DISCENTE

**TEMPLATE PARA TRABALHOS DE CONCLUSÃO DE
CURSO DO BACHARELADO EM SISTEMAS DE
INFORMAÇÃO DO IFF**

Itaperuna/RJ

2024

NOME DO DISCENTE

TEMPLATE PARA TRABALHOS DE CONCLUSÃO DE CURSO DO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO DO IFF

Trabalho de Conclusão de Curso apresentado
no Instituto Federal Fluminense *campus* Ita-
peruna como requisito parcial para conclusão
do Curso de Bacharelado em Sistemas de In-
formação.

Orientador: Nome do orientador
Coorientador: Nome do coorientador

Itaperuna/RJ
2024

NOME DO DISCENTE

TEMPLATE PARA TRABALHOS DE CONCLUSÃO DE CURSO DO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO DO IFF

Trabalho de Conclusão de Curso apresentado
no Instituto Federal Fluminense *campus* Ita-
peruna como requisito parcial para conclusão
do Curso de Bacharelado em Sistemas de In-
formação.

Aprovado em 12 de abril de 2025.

Banca Avaliadora:

Nome do orientador (orientador)
IFF

Membro da banca A
IFF

Membro da banca B
UFF

Itaperuna/RJ
2024

*Aos que acreditam que o \LaTeX é o melhor
sistema tipográfico existente.*

AGRADECIMENTOS

Agradeço a Deus pelo Dom da Vida...

*E não vos conformeis com este século, mas transformai-vos
pela renovação da vossa mente, para que experimenteis qual
seja a boa, agradável e perfeita vontade de Deus.*
(Bíblia Sagrada, Romanos 12:2)

RESUMO

Resumo em português.

Palavras-chave: Latex. Template. Editoração de Texto.

ABSTRACT

English abstract.

Keywords: Latex. Template. Text Editing.

LISTA DE FIGURAS

Figura 1 – Logo do T _E X	15
Figura 2 – Logo do LaT _E X	15

LISTA DE QUADROS

Quadro 1 – Complexidades de tempo em algumas estruturas de dados	17
--	----

LISTA DE TABELAS

Tabela 1 – As 10 Linguagens de Programação mais populares em Maio 2021 . . .	16
Tabela 2 – Tabela Padrão IBGE	17

LISTA DE ABREVIATURAS E SIGLAS

WYSIWYG: What You See Is What You Get

WYSIWYM: What You See Is What You Mean

LISTA DE SÍMBOLOS

Γ	Letra grega maiúscula Gama
Λ	Letra grega maiúscula Lambda
ζ	Letra grega minúscula zeta
\in	Pertence

SUMÁRIO

LISTA DE FIGURAS	8
LISTA DE QUADROS	9
LISTA DE TABELAS	10
1 SEÇÃO PRIMÁRIA	14
1.1 SEÇÃO SECUNDÁRIA	14
1.1.1 Seção terciária	14
1.1.1.1 Seção quaternária	14
1.1.1.1.1 Seção quinária	14
2 IMAGENS	15
2.1 TEX	15
2.2 LATEX	15
3 TABELAS E QUADROS	16
4 ALGORITMOS	18
4.1 BUSCA BINÁRIA	18
5 MATERIAL DE APOIO	19
 REFERÊNCIAS BIBLIOGRÁFICAS	 20
 Apêndices	 21
APÊNDICE A PRIMEIRO APÊNDICE	22
APÊNDICE B SEGUNDO APÊNDICE	23
 Anexos	 24
ANEXO A PRIMEIRO ANEXO	25
ANEXO B SEGUNDO ANEXO	26

1 SEÇÃO PRIMÁRIA

Conteúdo da seção primária

1.1 SEÇÃO SECUNDÁRIA

Conteúdo da seção secundária

1.1.1 Seção terciária

Conteúdo da seção terciária

1.1.1.1 Seção quaternária

Conteúdo da seção quaternária

1.1.1.1.1 Seção quinária

Conteúdo da seção quinária

2 IMAGENS

2.1 TEX

TeX (= tau epsilon chi, pronunciado como “tech”) é um sistema de tipografia criado por Donal Knuth em 1978. É popular no meio acadêmico, principalmente entre os físicos, matemáticos e cientistas da computação, devido sua sua capacidade de produzir fórmulas e símbolos matemáticos de uma maneira elegante.

O principal motivo da criação do TeX foi a insatisfação de Knuth com a qualidade de impressão de seu segundo volume do magnum opus multivolume *The Art of Computer Programming*¹. A logo do T_EX pode ser vista na Figura 1.

Figura 1 – Logo do T_EX



Fonte: <https://en.wikipedia.org/wiki/TeX>

2.2 LATEX

LaT_EX (pronuncia-se <<Lah-Tech>> ou <<Lay-Tech>>) é um **conjunto de macros** criadas por Leslie Lamport para o processador de textos T_EX. O LaT_EX fornece ao usuário comandos de alto nível, facilitando assim sua utilização.

Figura 2 – Logo do LaT_EX



Fonte: <https://en.wikipedia.org/wiki/LaTeX>

¹ <https://www.tug.org/whatis.html>

3 TABELAS E QUADROS

Existem diferenças entre tabelas e quadros, as principais estão na formatação e natureza do conteúdo. A tabela deve ser utilizada de forma majoritária quando os dados forem numéricos, em contrapartida, o quadro é geralmente utilizado quando temos dados não numéricos.

Um quadro é composto por linhas verticais e horizontais, sendo, portanto “fechado”. Já as tabelas possuem apenas linhas horizontais, sendo, desta forma “abertas”. Abaixo encontram-se exemplos de uso de tabelas e quadros.

Tabela 1 – As 10 Linguagens de Programação mais populares em Maio 2021

Maio 2021	Maio 2020	Modificação	Linguagem
1	1		C
2	3	∧	python
3	2	∨	Java
4	4		C++
5	5		C#
6	6		Visual Basic
7	7		JavaScript
8	14	∧	Assembly Language
9	8	∨	PHP
10	9	∨	SQL

Fonte: Tiobe Index - Tiobe (2021)

Na Tabela 1 podem ser vistas as 10 linguagens de programação mais populares em maio de 2021.

O abnT_{EX}¹ disponibiliza o comando **IBGEtab** que permite criar tabelas com formatação padronizadas de acordo com o IBGE (ABNT NBR 14724:2011²). Na Tabela 2 podemos ver a Tabela 1 no formato IBGE.

¹ <https://www.abntex.net.br/>

² <https://bit.ly/2T9jqat>

Tabela 2 – As 10 Linguagens de Programação mais populares em Maio 2021

Maio 2021	Maio 2020	Modificação	Linguagem
1	1		C
2	3	^	python
3	2	v	Java
4	4		C++
5	5		C#
6	6		Visual Basic
7	7		JavaScript
8	14	^	Assembly Language
9	8	v	PHP
10	9	v	SQL

Fonte: Tiobe Index - Tiobe (2021)

No Quadro 1 é apresentada a análise de pior caso das complexidades de tempo das principais operações em algumas estruturas de dados.

Quadro 1 – Complexidades de tempo em algumas estruturas de dados

Estrutura de dados	Complexidade de tempo (Pior caso)			
	Acesso	Busca	Inserção	Remoção
Array	$O(1)$	$O(n)$	$O(n)$	$O(n)$
Stack	$O(n)$	$O(n)$	$O(1)$	$O(1)$
Queue	$O(n)$	$O(n)$	$O(1)$	$O(1)$
Singly-Linked List	$O(n)$	$O(n)$	$O(1)$	$O(1)$
Doubly-Linked List	$O(n)$	$O(n)$	$O(1)$	$O(1)$
Hash Table	N/A	$O(n)$	$O(n)$	$O(n)$
Binary Search Tree	$O(n)$	$O(n)$	$O(n)$	$O(n)$
B-Tree	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$
Red-Black Tree	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$
AVL Tree	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$

Fonte: BigO Cheatsheet - <https://www.bigocheatsheet.com/>

4 ALGORITMOS

4.1 BUSCA BINÁRIA

O algoritmo de busca binária possui complexidade de tempo $O(\log n)$ e pode ser utilizado, por exemplo, para verificar de modo eficiente se um determinado *array* ordenado possui um dado elemento fornecido.

Um modo tradicional de implementarmos a busca binária é realizar a checagem do elemento central do *subarray* e verificarmos se ele corresponde ao que estamos procurando, se sim, então o elemento procurado foi encontrado, caso contrário verificaremos se o elemento central é maior ou menor que o elemento procurado, caso seja maior, então a busca continua no *subarray* esquerdo, senão no *subarray* direito. No algoritmo 1 podemos ver uma possível implementação deste raciocínio de forma iterativa.

Algoritmo 1: buscaBinaria(vet, tam, x)

Entrada: Vetor (vet), tamanho do vetor (tam) e elemento a ser procurado (x)

Saída: Posição onde o elemento se encontra ou -1 caso contrário

```

1   $a \leftarrow 0$ ;
2   $b \leftarrow tam - 1$ ;
3   $meio \leftarrow -1$ ;
4  enquanto ( $a \leq b$ ) faça
5       $meio = (a + b) / 2$ ;
6      se ( $vet[meio] == x$ ) então
7          retorna meio ; // elemento encontrado
8      se ( $vet[meio] < x$ ) então
9           $a \leftarrow meio + 1$ ;
10     senão
11          $b \leftarrow meio - 1$ ;
12 retorna meio ; // elemento não encontrado
```

Em Laaksonen (2017), o autor traz uma excelente abordagem do algoritmo da busca binária e propõe um outro método de implementação. Também em (PROGRAMIZ, 2021) é realizada uma boa abordagem do algoritmo em suas versões iterativa e recursiva.

5 MATERIAL DE APOIO

Abaixo seguem algumas referências para se trabalhar com a classe `abnTEX2` (Da qual este template foi estendido) e da classe `memoir`, na qual o `abnTEX2` foi baseado. Também são mostradas referências para o pacote `biblatex-abnt`, no qual foi utilizado para gerenciar as referências deste template e que possui suporte para as regras de citação exigidas pela ABNT.

- a) A classe `abnTEX2`
 - Site oficial: <https://www.abntex.net.br/>
 - Repositório GitHub: <https://github.com/abntex/abntex2>
 - Página CTAN: <https://www.ctan.org/pkg/abntex2>
- b) A classe `memoir`
 - Página CTAN: <https://www.ctan.org/pkg/memoir>
- c) O pacote `biblatex-abnt`
 - Repositório GitHub: <https://github.com/abntex/biblatex-abnt>
 - Página CTAN: <https://www.ctan.org/pkg/biblatex-abnt>

REFERÊNCIAS BIBLIOGRÁFICAS

LAAKSONEN, A. **Guide to Competitive Programming: Learning and Improving Algorithms**. 1. ed. Suíça: Springer, 2017.

PROGRAMIZ, E. **Learn to code for free: Binary Search**. [S.l.], 2021. Disponível em: <<https://www.programiz.com/dsa/binary-search>>.

TIOBE. **Tiobe - The Software Quality Company**. [S.l.], 2021. Disponível em: <<https://www.tiobe.com/tiobe-index/>>.

Apêndices

APÊNDICE A – PRIMEIRO APÊNDICE

APÊNDICE B – SEGUNDO APÊNDICE

Anexos

ANEXO A – PRIMEIRO ANEXO

ANEXO B – SEGUNDO ANEXO