INSTRUÇÕES PARA O ARTIGO DO TRABALHO DE CONCLUSÃO DE CURSO – BCC

João Pedro da Silva, José Carlos Pereira – Orientador

Curso de Bacharel em Ciência da Computação

Departamento de Sistemas e Computação

Universidade Regional de Blumenau (FURB) – Blumenau, SC – Brasil

maildoaluno@furb.br, maildoorientador@furb.br

**Resumo:** O resumo é uma apresentação concisa dos pontos relevantes de um texto. Informa suficientemente ao leitor, para que este possa decidir sobre a conveniência da leitura do texto inteiro. Deve conter OBRIGATORIAMENTE o OBJETIVO, METODOLOGIA, RESULTADOS e CONCLUSÕES. O resumo não deve ultrapassar 10 linhas e deve ser composto de uma sequência corrente de frases concisas e não de uma enumeração de tópicos. O resumo deve ser escrito em um único texto corrido (sem parágrafos). Deve-se usar a terceira pessoa do singular. As palavras-chave, a seguir, são separadas por ponto, com a primeira letra maiúscula. Caso uma palavra-chave seja composta por mais de uma palavra, somente a primeira deve ser escrita com letra maiúscula, sendo que as demais iniciam com letra minúscula, desde que não sejam nomes próprios.]

**Palavras-chave**: Ciência da computação. Sistemas de informação. Monografia. Resumo. Formato.

# Introdução

O presente *template* deve ser seguido para a confecção dos trabalhos finais de conclusão dos cursos de Ciência da Computação e Sistemas de Informação da FURB. Ele é fortemente (mas não totalmente) baseado no modelo da Sociedade Brasileira de Computação de modo a facilitar posteriores adaptações para publicações das produções dos TCCs em eventos científicos. O *template* proposto segue as normas da ABNT para citações e referências bibliográficas, bem como para referências à figuras, quadros e tabelas.

Para o TCC dos cursos, **o presente artigo está limitado a 20 páginas**, incluindo as referências bibliográficas e excluindo os apêndices e anexos. Os principais elementos de formatação estão explicados no anexo no fim deste *template*. O formato pode ser diferente caso o artigo seja submetido a um evento científico da área. Neste caso o estudante pode manter o artigo no formato do evento, mas deve anexar o comprovante de submissão. Dúvidas ou problemas devem ser sanados com a coordenação do TCC do curso.

A introdução deve despertar no leitor o interesse pelo texto, apresentando os assuntos que serão tratados e o enfoque que será dado ao tema central. Deve iniciar com uma **contextualização** do estudo a ser realizado, explicando claramente sua origem/motivação. A visão geral do tema deve então ser afunilada até se chegar ao problema a ser pesquisado. Após o problema ter sido identificado, deve-se delimitar que aspectos ou elementos serão tratados. Deve-se deixar bem claro o problema que se quer resolver, com o desenvolvimento do trabalho e **justificar** porque o assunto merece ser estudado.

Deve esclarecer a **formulação do problema** a ser investigado e deve ser finalizado com os **objetivos do trabalho** que podem ser subdivididos **em geral (obrigatório)** e **específicos (opcional)**. O objetivo principal deve ser descrito em uma frase única, usando o verbo no infinitivo. Os objetivos específicos detalham o objetivo principal ou definem subprodutos do trabalho. Também se relacionam a formas de validação ou avaliação do trabalho realizado. Os objetivos devem ser mensuráveis quanto a se e como foram ou não atingidos. Os objetivos específicos devem ser enumerados, usando verbos no infinitivo.

Também deve(m) ser apresentado(s) **o(s) método(s) de testes** do trabalho desenvolvido.

# FUNDAMENTAÇÃO TEÓRICA

## Este capítulo contém os aspectos que fundamentam o desenvolvimento deste estudo, o capítulo está dividido em cinco seções. A seção 2.1 identifica o que é a Criatividade Computacional. Na seção 2.2 é apresentada uma visão geral sobre o que é arte e a evolução destes conceitos nas últimas décadas. A seção 2.3 está focada em descrever o que é arte generativa e arte generativa computacional, respectivamente. A seção 2.4 apresenta o software e linguagem utilizadas para o desenvolvimento. Na seção 2.5 são apresentados os trabalhos correlatos a este estudo.

## CRIATIVIDADE COMPUTACIONAL

Para entender o que é Criatividade Computacional deve-se primeiro entender o que é criatividade e o que pode ser considerado criatividade, neste sentido, para os seres humanos a definição do que é criatividade é algo ainda muito abrangente. Entretanto, a criatividade pode ser entendida como a forma ou método de resolução de um problema de uma maneira não convencional, ou seja, a habilidade de encontrar diferentes soluções para um problema (PEREIRA, 2004).

A Criatividade Computacional é uma subárea da Inteligência Artificial que trabalha com sistemas computacionais capazes de gerar artefatos e ideias. Sendo que estes sistemas são geralmente aplicados em domínios criativos, tais como: matemática e ciência, poesia, artes visuais e design gráfico (COLTON; WIGGINS, 2012). A Criatividade Computacional é um campo da Inteligência Artificial interdisciplinar, combinado principalmente com matérias como filosofia, psicologia, matemática e engenharia, e ainda podendo ser aplicada a diversos domínios (ACKERMAN et al., 2017).

Existe mais de uma abordagem para o estudo da Criatividade Computacional. Estas abordagens apesar de serem diversificadas convergem para o entendimento do que pode ser considerado um produto criativo. Neste contexto o produto seria o software que é capaz de produzir algo. Este ideal de produzir algo está de acordo com a maioria das contribuições científicas para o tema que tem a tendência de trabalhar com sistemas generativos. Apesar da maioria dos trabalhos estarem focados nos sistemas generativos, existem duas características básicas para se classificar um sistema como criativo, são elas geração e evolução (ACKERMAN et al., 2017). Portanto podemos considerar a computação criativa de acordo com Prosecco (2017) como “Um campo emergente que estuda e explora o potencial dos computadores para serem mais do que ferramentas ricas em recursos e para atuar como criadores autônomos e cocriadores.”, tendo em vista então que em um sistema criativo, o ímpeto criativo vem da máquina, não do usuário, embora em um sistema híbrido um ímpeto possa vir de ambos (PROSECCO, 2017).

## O QUE É ARTE?

“O que é arte?” esta é uma pergunta que obteve várias respostas, mas mesmo atualmente, existe uma certa dificuldade em defini-la e em entender qual a sua importância para a sociedade (COREIA, 2019). No século XIX Tolstoy (1996) definiu a arte como “A forma que um ser humano encontra de compartilhar o sentimento ou a sensação que obteve ao realizar uma ação, permitindo assim que outro ser humano tenha o mesmo sentimento que o autor”. Apesar desta ser uma definição do século XIX ela se repete para outros autores ao longo dos últimos séculos. Como para Langer de acordo com o estudo de Coreia (2019), em que é apresentada a definição de que “arte é a criação de formas simbólicas de sentimento humano”.

Enquanto alguns autores se esforçam para definir o que é arte outros dizem que a arte por si só não pode ser definida. O ato de tentar definir a arte e as características que devam ser encontradas em obras de arte pode ser considerado o motivo de falha de teóricos. Ao tentarem definir algo do qual não tem conhecimento falham com definições vazias por não entenderem sua essência (MOROKAWA, 2018). Como definir o contexto generalizado de uma palavra que se quer existe em todos idiomas é um desafio para filósofos (DISSANAYAKE, 2002), o problema então pode ser dito como não “o que é arte”, mas qual tipo de conceito pode ser considerado arte e para que ela é utilizada (DANIELE; SONG, 2019). Para entendimento geral deste trabalho a arte será considerada como “uma forma de expressão humana com o propósito de transmitir algum sentimento”, levando em consideração noções do final do século XX, suportado por autores como Dissanayake (2002) e Angelini (2017).

## ARTE GENERATIVA

Arte generativa é um tipo de arte onde em algum momento de sua concepção há alguma automatização do processo, ou, o processo como todo é fruto de um sistema independente, neste contexto não necessariamente um sistema computacional. O elemento principal deste cenário é a passagem do controle de criação da obra para um sistema ou método. É importante lembrar que o termo “arte generativa” diz respeito apenas a forma como a obra foi produzida e não o porquê ou o que ela contém (GALANTER, 2003).

Na criação da arte generativa o papel do artista está em manipular este sistema gerador e ser capaz de filtrar ou produzir algo significante através de uma produção que foge do seu controle de maneira parcial ou total (MCORMACK; DORIN, 2001). Existe mais de uma classificação para produção de conteúdo generativo indo de sistemas randômicos até sistemas mais complexos. O sistema ainda pode ter sua criação fixa, mas se modifica dada a interação com outros sistemas ou pessoas (GALANTER, 2003). Na Anexo 1 é possível visualizar uma obra do artista Hans Haacke, pioneiro na crítica institucional criou trabalhos conceituais que expõe as conexões entre dinheiro, arte e política. A obra foi produzida entre os anos de 1964 e 1965, e é considerada uma instalação artística. Esta obra é considerada como generativa por ter um sistema independente (o ventilador) capaz de modificar a obra durante sua exibição e esse sistema não está sob o controle do artista. A obra se modifica conforme a direção que o vento toma (SFMOMA, 2005).

### ARTE GENERATIVA COMPUTACIONAL

Tendo em vista que a arte generativa consiste na automatização do processo de criação de alguma obra através de um sistema, ao trazer esse conceito para a computação o sistema passa a ser considerado como um algoritmo que tipicamente faz uso de aleatorização para determinar as características da criação. Estas características podem ser fixadas por algum artista generativo e outras podem ser parametrizáveis pela própria máquina gerando assim um trabalho generativo diferente a cada interação do algoritmo (PARIKH, 2020).

Um exemplo de trabalho generativo computacional é o do artista Anders Hoff com o projeto *Inconvergent*. Conforme pode ser observado no Quadro 3, estão dispostos dois diferentes algoritmos generativos, no primeiro algoritmo é feita uma geração generativa que não trabalha com mutações de resultado, neste contexto, mutações seriam erros ou padrões não esperados (HOFF, 2017). A execução do primeiro algoritmo com as regras demonstradas no Quadro 3 gera a primeira imagem do Quadro 4. O segundo algoritmo disposto no Quadro 3 é semelhante ao primeiro, porém contém a adição de um comando de mutação, ou a inclusão de erros ao resultado gerado pelos processos generativos. O comando pode ser observado destacado na imagem. O uso da mutação pode ser observado na segunda imagem do Quadro 4, onde o resultado gerado pelo segundo algoritmo é uma produção mais complexa e com mais detalhes.

Quadro 3 – Relação entre códigos generativos com suas regras

|  |  |
| --- | --- |
| Algoritmo inicial | Algoritmo com mudança de uma regra |
|  |  |

Fonte: Hoff (2017).

Quadro 4 – Resultados da execução dos algoritmos generativos

|  |  |
| --- | --- |
| Resultado da execução do primeiro algoritmo | Resultado da execução após a mutação |
|  |  |

Fonte: Hoff (2017).

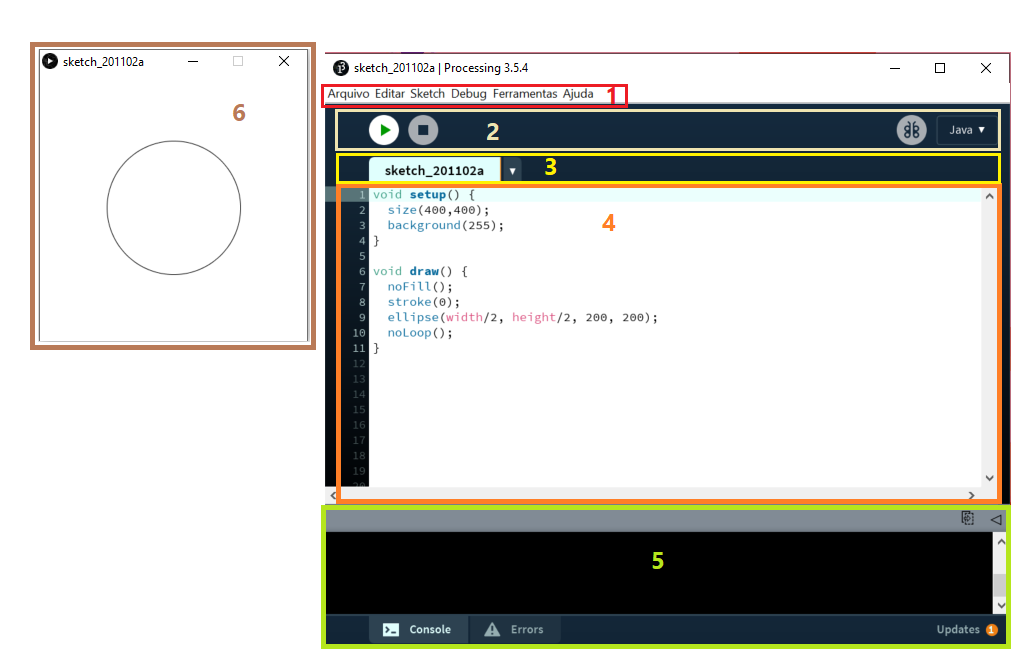
## PROCESSING

Processing é um software de *sketchbook* e uma linguagem criada para trabalhos relacionados a artes visuais. Foi desenvolvido como um projeto de código aberto, iniciado por Casey Reas e Benjamin Fry em 2001 com o objetivo de ensinar habilidades de programação por meio de um feedback instantâneo de recursos visuais. Foi construído inicialmente com base no Java, mas atualmente conta com extensões para Python, Javascript, Android e Arduino. Com a popularização da ferramenta recebeu adição de bibliotecas de terceiros que adicionaram recursos sofisticados como o desenho em 3D e leitura de XML (PEARSON, 2011).

### PROCESSING DEVELOPMENT ENVIRONMENT

O Processing Development Enviroment (PDE) tem o objetivo de facilitar o desenvolvimento de programas em Processing. Os programas são escritos em um editor de texto e iniciados ao clicar no botão de *run*. Conforme pode ser observado na Figura 1, o PDE pode ser dividido em seis partes sendo elas: 1- Menu de opções, 2 – Barra de ferramentas, 3 – Abas, 4 – Editor, 5 – Mensagens e erros e 6 – Janela de resultado. Na barra de ferramentas estão contidos os botões de execução, pause e depuração e na lateral direita tem o selecionador de modo de desenvolvimento, neste exemplo Java (PROCESSING, 2020?).

Figura 1 – Interface do PDE



Fonte: Adaptado de Processing (2020).

### SKETCHES AND SKETCHBOOK

Todos os projetos desenvolvidos com Processing são chamados de *sketches*, cada um destes *sketches* está contido em sua própria pasta. Um programa desenvolvido em Processing pode conter diferentes arquivos, porém a pasta deve conter um arquivo Processing principal que recebe o mesmo nome que o da pasta. Os arquivos desenvolvidos com o Processing possuem a extensão .pde que é um acrônimo para Processing Development Enviroment. Os sketches podem ser salvos em qualquer lugar do computador, mas de forma padrão eles são salvos em uma *sketchbook*, que ficará localizado em diferentes lugares dependendo do sistema operacional utilizado, as preferências para armazenamento dos arquivos .pde podem ser modificadas através do próprio ambiente (PROCESSING, 2020?).

### RENDERIZAÇÃO

O Processing contém quatro renderizadores de tela integrados para renderização bidimensional e tridimensional. O PD2 é o redenrizador mais rápido, mas o menos preciso para desenhar formas bidimensionais. O PD3 é para geometria tridimensional com ele também é possível ter acesso e controle a câmera, efeitos de iluminação e materiais e texturas. Os renderizadores são acelarados caso o computador contenha uma placa gráfica compatível com OpenGL. O renderizador para cada *sketch* é especificado na função **size()**, caso não seja passado o valor de um renderizador como parâmetro para a função a renderização será feita com o renderizador padrão (PROCESSING, 2020?).

## TRABALHOS CORRELATOS

Na segunda parte da fundamentação devem ser descritos os trabalhos correlatos. Devem ser incluídos preferencialmente trabalhos acadêmicos com características e funcionalidades semelhantes ao que está sendo produzido. A descrição deve ser feita em quadros conforme apresentado no Quadro 1. Os itens apresentados no quadro são obrigatórios. Quando não for possível identificar, justificar no quadro.

Quadro 1 – Trabalho Correlato 1

|  |  |
| --- | --- |
| Referência |  |
| Objetivos |  |
| Principais funcionalidades |  |
| Ferramentas de desenvolvimento |  |
| Resultados e conclusões |  |

Fonte: elaborado pelo autor.

# DESCRIÇÃO

Nesta seção devem ser descritos os **aspectos mais relevantes de especificação e implementação** para a compreensão sobre o trabalho desenvolvido. O título “DESCRIÇÃO” pode ser complementado com “DO SOFTWARE”, “DA FERRAMENTA” ou “DO PROTÓTIPO” ou aquilo que melhor representar o que foi desenvolvido. Esta seção deve estar organizada em pelo menos duas subseções: especificação e implementação

Reitera-se que, em função da limitação do número de páginas, a descrição deve contemplar o que é mais significativo para a compreensão do que foi desenvolvido.

## Especificação

**Os diagramas inseridos devem considerar o modelo de estrutura e de comportamento do que foi desenvolvido**. Se é um sistema com usuário final, inclua a lista de requisitos funcionais e não funcionais. Destaca-se que os diagramas desenvolvidos bem como outros aspectos de especificação deverão obrigatoriamente constar nos apêndices **quando não couberem nesta seção**.

## implementação

Descreve os aspectos fundamentas para a compreensão da implementação do TCC. Considere a inserção dos códigos mais relevantes da implementação, bem como as telas do trabalho desenvolvido.

# RESULTADOS

De modo a ampliar o seu caráter científico, todos os TCCs devem apresentar e discutir resultados não limitados à comparação com os trabalhos correlatos. Devem ser apresentados os casos de testes do software, destacando objetivo do teste, como foi realizada a coleta de dados e a apresentação dos resultados obtidos, preferencialmente em forma de gráficos ou tabelas, fazendo comentários sobre os mesmos. Também é sugerida a comparação com os trabalhos correlatos apresentados na fundamentação teórica.

# CONCLUSÕES

As conclusões devem refletir os principais resultados alcançados, realizando uma avaliação em relação aos objetivos previamente formulados. Deve-se deixar claro se os objetivos foram atendidos, se as ferramentas utilizadas foram adequadas e quais as principais contribuições do trabalho sociais ou práticas para o seu grupo de usuários bem como para o desenvolvimento científico e ou tecnológico da área.

Deve-se incluir também as limitações e as possíveis extensões do TCC.

Referências

As referências devem ser apresentadas em ordem alfabética. Só podem ser inseridas nas referências os documentos citados ao longo do artigo. Todos os documentos citados obrigatoriamente têm que estar inseridos nas referências. A seguir são apresentados alguns exemplos de referências bibliográficas. Destaca-se que deve ser seguida a norma da ABNT.

[parte de um documento:]

AMADO, Gilles. Coesão organizacional e ilusão coletiva. In: MOTTA, Fernando C. P.; FREITAS, Maria E. (Org.). **Vida psíquica e organização**. Rio de Janeiro: FGV, 2000. p. 103-115.

[trabalho acadêmico ou monografia (TCC/Estágio, especialização, dissertação, tese):]

AMBONI, Narcisa F. **Estratégias organizacionais**: um estudo de multicasos em sistemas universitários federais das capitais da região sul do país. 1995. 143 f. Dissertação (Mestrado em Administração) - Curso de Pós-Graduação em Administração, Universidade Federal de Santa Catarina, Florianópolis.

[norma técnica:]

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6023**: informação e documentação: referências - elaboração. Rio de Janeiro, 2002a. 24 p.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 10520**: informação e documentação: citações em documentos: apresentação. Rio de Janeiro, 2002b. 7 p.

[livro:]

BASTOS, Lília R.; PAIXÃO, Lyra; FERNANDES, Lúcia M. **Manual para a elaboração de projetos e relatórios de pesquisa, teses e dissertações**. Rio de Janeiro: Zahar, 1979.

[trabalho acadêmico ou monografia (TCC/Estágio, especialização, dissertação, tese):]

BRUXEL, Jorge L. **Definição de um interpretador para a linguagem Portugol, utilizando gramática de atributos**. 1996. 77 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

[verbete de enciclopédia em meio eletrônico:]

EDITORES gráficos. In: WIKIPEDIA, a enciclopédia livre. [S.l.]: Wikimedia Foundation, 2006. Disponível em: http://pt.wikipedia.org/wiki/Editores\_graficos. Acesso em: 13 maio 2006.

[artigo em evento:]

FRALEIGH, Arnold. The Algerian of independence. In: ANNUAL MEETING OF THE AMERICAN SOCIETY OF INTERNATIONAL LAW, 61, 1967, Washington. **Proceedings…** Washington: Society of International Law, 1967. p. 6-12.

[norma técnica:]

IBGE. **Normas para apresentação tabular**. 3. ed. Rio de Janeiro, 1993. 61 p. Disponível em: http://biblioteca.ibge.gov.br/visualizacao/monografias/GEBIS%20-%20RJ/normastabular.pdf. Acesso em: 27 ago. 2013.

[artigo em periódico:]

KNUTH, Donald E. Semantic of context-free languages. **Mathematical Systems Theory**, New York, v. 2, n. 2, p. 33-50, jan./mar. 1968.

[trabalho acadêmico ou monografia (TCC/Estágio, especialização, dissertação, tese):]

SCHUBERT, Lucas A. **Aplicativo para controle de ferrovia utilizando processamento em tempo real e redes de Petri**. 2003. 76 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

[página da internet com autor]

SCHULER, João P. S. **Tutorial de Delphi**. Porto Alegre, [2002]. Disponível em: http://www.schulers.com/jpss/pascal/dtut/. Acesso em: 27 ago. 2013.

[página da internet sem autor]

SCHRATCH. **Program, imagine, share**. [S.l.], [2013?]. Disponível em: <https://scratch.mit.edu/>. Acesso em: 27 maio 2013.

[relatório de pesquisa:]

VARGAS, Douglas N. **Editor dirigido por sintaxe**. 1992. Relatório de pesquisa n. 240 arquivado na Pró-Reitoria de Pesquisa, Universidade Regional de Blumenau, Blumenau.

APÊNDICE A – DIAGRAMAS DE ESPECIFICAÇÃO

É fundamental que todo projeto apresente alguma forma de especificação do que foi desenvolvido. A descrição é opcional. Assim, **este apêndice deve conter os diagramas de especificação que não couberam ao longo do texto**. Os diagramas devem conter legendas numeradas na sequência do artigo.

Cada apêndice deve iniciar em uma nova página.

APÊNDICE B – XXX

Podem ser inseridos outros apêndices no artigo tais como códigos de implementação, telas de interface, instrumentos de coleta de dados, entre outros. **Apêndices são** **textos elaborados pelo autor** a fim de complementar sua argumentação. Os apêndices são identificados por letras maiúsculas consecutivas, seguidas de um travessão e pelos respectivos títulos. Deve haver no mínimo uma referência no texto anterior para cada apêndice. Colocar sempre um preâmbulo no apêndice. Caso existam tabelas ou ilustrações, identifique-as através da legenda, seguindo a numeração normal das legendas do artigo.

ANEXO A – DESCRIÇÃO

Elemento opcional, **anexos são documentos não elaborados pelo autor**, que servem de fundamentação, comprovação ou ilustração, como mapas, leis, estatutos, entre outros. Os anexos são identificados por letras maiúsculas consecutivas, seguidas de um travessão e pelos respectivos títulos. Deve haver no mínimo uma referência no texto anterior para cada anexo. Colocar sempre um preâmbulo no anexo. Caso existam tabelas ou ilustrações, identifique-as através da legenda, seguindo a numeração normal das legendas do artigo.

# DESCRIÇÃO DA FORMATAÇÃO

A seguir são apresentadas observações gerais sobre o texto do artigo do Trabalho de Conclusão de Curso (TCC). **Observa-se que esta descrição deve ser retirada do texto final.**

Na confecção do texto deve-se:

1. usar frases curtas. Segundo Teodorowitsch (2003, p. 3), “Frases com mais de duas linhas aumentam o risco de o leitor não compreender a ideia ou de entendê-la de forma equivocada.”;
2. usar linguagem impessoal (usar a terceira pessoa do singular) e verbo na voz ativa (a ação é praticada pelo sujeito), com conexão entre os parágrafos;
3. não usar palavras coloquiais;
4. não usar palavras repetidas em demasia;
5. usar verbos no presente quando for referir-se a partes do trabalho que já se encontram disponíveis no texto;
6. destacar palavras em língua estrangeira em itálico, conforme descrito abaixo:
   1. nome de software, ferramenta, aplicativo, linguagem de programação, plataforma, empresa: não deve ser escrito em itálico (exemplos: Delphi 7, Pascal, Object Pascal, Java, JavaScript, Java 2 Micro Edition, Basic, Microsoft Visual C++, C, Windows, Linux, MySQL, Oracle, Eclipse 3.0, Enterprise Architect, Rational Rose, Microsoft, Sun Microsystems),
   2. citações: o sobrenome do autor ou o nome da instituição responsável pela autoria do documento citado não deve ser escrito em itálico (exemplo: Segundo Sun Microsystems (2004), ...),
   3. palavras em língua estrangeira encontradas nos dicionários nacionais: não devem ser grafadas em itálico (exemplos: software, hardware, web, Internet),
   4. demais palavras em língua estrangeira: devem ser escritas em itálico (exemplos: *palmtop*, *classpath*, *play*, etc.). No entanto, Teodorowitsch (2003, p. 7), sugere que alguns termos em língua inglesa devem ser substituídos por termos em português (exemplos: núcleo em vez de *kernel*, aprendizagem de máquina em vez de *machine learning*, etc.);
7. observar as seguintes regras quanto ao uso de siglas:
   1. colocar as siglas entre parênteses precedidas pela forma completa do nome, quando aparecem pela primeira vez no texto (exemplos: Associação Brasileira de Normas Técnicas (ABNT), Trabalho de Conclusão de Curso (TCC)). Caso exista uma lista de siglas na parte pré-textual do volume final, pode-se usar somente a sigla, quando aparecer pela primeira vez no texto,
   2. usar apenas a sigla nas demais ocorrências no texto,
   3. escrever as siglas em letras maiúsculas e não usar itálico,
   4. escrever o plural das siglas sem apóstrofo (exemplos: PCs, APIs, PDAs) e determinar o gênero da sigla conforme o gênero do primeiro substantivo do seu nome (exemplo: o TCC – o Trabalho de Conclusão de Curso).

## formatação

A formatação geral para apresentação do documento, descrita na NBR 14724 (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2011), é a seguinte:

1. o texto divide-se em capítulos, seções e subseções (até cinco divisões);
2. a apresentação de citações em documentos deve seguir a NBR 10520 (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2002b);
3. a descrição das referências bibliográficas deve estar de acordo com a NBR 6023 (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2002a).

Observa-se ainda que todo capítulo, seção ou subseção deve ter no mínimo um texto relacionado.

O artigo deve ser digitado usando as fontes e formatação de parágrafos deste modelo, indicadas no Quadro 2.

Quadro 2– Estilos do modelo

|  |  |
| --- | --- |
| **USO** | **FORMATO** |
| título de capítulo ou seção primária (1) | TF-TÍTULO 1 (Times New Roman, 10pt, negrito, maiúsculas) |
| título de seção secundária (1.1) | TF-TÍTULO 2 (Times New Roman, 10pt, maiúsculas) |
| título de seção terciária (1.1.1) | TF-Título 3 (Times New Roman, 10pt, minúsculas, exceto a 1a letra da 1a palavra do título e de nomes próprios) |
| título de seção quaternária (1.1.1.1) | TF-Título 4 (mesma formatação seção ternária) |
| título de seção quinária (1.1.1.1.1) | TF-Título 5 (mesma formatação seção ternária) |
| texto | TF-TEXTO (Times New Roman, 10pt) |
| citação direta com mais de três linhas | TF-CITAÇÃO (Times New Roman, 9pt, com recuo de 4 cm) |
| itens (alíneas) | ver descrição abaixo (Times New Roman, 10pt) |
| referência bibliográfica | TF-REFERÊNCIA ITEM (Times New Roman, 10pt, alinhada à margem esquerda) |
| fonte, legenda, texto de quadro/tabela e figura | TF-FONTE (Times New Roman, 9pt, centralizada)  TF-LEGENDA, (Times New Roman, 10pt, centralizada)  TF-TEXTO- QUADRO (Times New Roman, 10pt)  TF-FIGURA (Times New Roman, 10pt, centralizada) |

Fonte: elaborado pelo autor.

O espaçamento, também definido no modelo, deve ser conforme indicado no Quadro 3.

Quadro 3 - Espaçamento

|  |  |
| --- | --- |
| **USO** | **ESPAÇAMENTO** |
| título de capítulo ou seção primária (1)  título de seção secundária (1.1)  título de seção terciária (1.1.1)  título de seção quaternária (1.1.1.1)  título da seção quinária (1.1.1.1.1) | espaço simples, com 12pt antes do parágrafo |
| texto | espaço simples, com 6 pt antes do parágrafo |
| citação direta com mais de três linhas | espaço simples com 6pt antes e depois do parágrafo |
| itens (alíneas) | espaço simples, com 6 pt antes do parágrafo |
| referência bibliográfica | espaço simples, com 6 pt antes do parágrafo |
| legenda e texto de ilustração/tabela | espaço simples, com 6 pt antes do parágrafo |
| fonte | espaço simples, com 0pt antes do parágrafo |

Fonte: elaborado pelo autor.

Na disposição gráfica de itens (alíneas) devem ser observados os seguintes quesitos:

1. o texto que antecede os itens termina com dois pontos;
2. cada item deve iniciar com uma letra minúscula seguida de fecha parênteses e terminar com um ponto e vírgula, sendo que o último item termina com ponto (FORMATO: TF-ALÍNEA);
3. o texto de cada item inicia com letra minúscula, exceto nomes próprios;
4. quando contiver subitens, os mesmos devem iniciar com hífen colocado sob a primeira letra do texto do item correspondente (FORMATO: TF-SUBALÍNEA nível 1 ou TF-SUBALÍNEA nível 2, conforme o caso). Nesse caso, cada subitem deve terminar com uma vírgula, exceto o último que termina com ponto ou com ponto e vírgula.

Segue um exemplo:

1. cada item inicia com letra minúscula, cada item inicia com letra minúscula, cada item inicia com letra minúscula, cada item inicia com letra minúscula (FORMATO: TF-ALÍNEA);
2. cada item inicia com letra minúscula, cada item inicia com letra minúscula, cada item inicia com letra minúscula, cada item inicia com letra minúscula (FORMATO: TF-ALÍNEA):
   1. cada subitem (nível 1) inicia com letra minúscula, cada subitem (nível 1) inicia com letra minúscula (FORMATO: TF-SUBALÍNEA nível 1):
      1. cada subitem (nível 2) inicia com letra minúscula, cada subitem (nível 2) inicia com letra minúscula (FORMATO: TF-SUBALÍNEA nível 2),
      2. cada subitem (nível 2) inicia com letra minúscula, cada subitem (nível 2) inicia com letra minúscula (FORMATO: TF-SUBALÍNEA nível 2),
   2. cada subitem (nível 1) inicia com letra minúscula, cada subitem (nível 1) inicia com letra minúscula (FORMATO: TF-SUBALÍNEA nível 1);
3. cada item inicia com letra minúscula, cada item inicia com letra minúscula, cada item inicia com letra minúscula, cada item inicia com letra minúscula (FORMATO: TF-ALÍNEA).

#### Exemplo de título de seção quaternária [FORMATO: TF-TÍTULO 4]

Formato: TF-TEXTO.

##### Exemplo de título de seção quinária [FORMATO: TF-TÍTULO 5]

Formato: TF-TEXTO.

### Formatação de quadros, figuras e tabelas

Um quadro contém apenas informações textuais, que podem ser agrupadas em colunas. Uma figura contém, além das informações textuais, pelo menos um elemento gráfico. Uma tabela é uma apresentação tabular de informações **numéricas** relacionadas.

Os quadros, figuras e tabelas são identificados na parte superior por uma legenda (a qual deve estar centralizada) composta pela palavra designativa (Figura, Quadro ou Tabela, conforme o caso), seguida de seu número em algarismo arábico (usar numeração progressiva, uma sequência para os quadros, outra para as figuras e outra para as tabelas), de hífen e do título. As ilustrações devem:

1. aparecer centralizadas no texto;
2. estar delimitadas por uma moldura simples (com exceção das tabelas não quais não devem ser usadas bordas (linhas) verticais em suas extremidades);
3. aparecer numa única página (quando o tamanho não exceder o da página), inclusive a legenda;
4. serem inseridas o mais próximo possível do trecho a que se referem pela primeira vez.

Toda ilustração deve ter fonte, centralizada. Quando foi o próprio autor que fez a ilustração, deve inserir o texto: “Fonte: elaborado pelo autor”.

Observa-se que quando um código fonte for descrito dentro de um quadro, deve-se utilizar letra do tipo courier new 9pt. (TF-CÓDIGO-FONTE)

Exemplos de como se deve referenciar uma figura, um quadro e uma tabela bem como descrevê-los são mostrados a seguir.

Um exemplo de uma rede de Petri pode ser visto na Figura 1.

Figura 2– Exemplo de uma rede de Petri



Fonte: Schubert (2003, p. 18).

Um exemplo de código fonte gerado a partir de uma especificação pode ser visto no Quadro 4.

Quadro 4 – Funções que verificam se as transições estão sensibilizadas

|  |
| --- |
| Function TEstruturaMalha.T1Sensibilizada: boolean;  begin  result := (Fp2 and Fp4);  end;  function TEstruturaMalha.T2Sensibilizada: boolean;  begin  result := (Fp1 and Fp3);  end;  function TEstruturaMalha.T3Sensibilizada: boolean;  begin  result := (Fp2 and Fp4);  end; |

Fonte: Schubert (2003, p. 63).

A quantidade de trabalhos finais realizados no Curso de Ciência da Computação (de 2010 até 2014) é apresentada na Tabela 1.

Tabela 1 – Trabalhos finais realizados no Curso de Ciência da Computação

|  |  |  |  |
| --- | --- | --- | --- |
| Ano | Estágios | TCC´s | Totais |
| 2010/1 | 0 | 16 | 16 |
| 2010/2 | 0 | 21 | 21 |
| 2011/1 | 0 | 25 | 25 |
| 2011/2 | 0 | 23 | 23 |
| 2012/1 | 0 | 23 | 23 |
| 2012/2 | 0 | 22 | 22 |
| 2013/1 | 0 | 25 | 25 |
| 2013/2 | 0 | 16 | 16 |
| 2014/1 | 0 | 18 | 18 |
| 2014/2 | 0 | 13 | 13 |
|  | **0** | **202** | **202** |

Fonte: elaborado pelo autor.

### Exemplos de citações retiradas de documentos ou de nomes constituintes de uma entidade

A apresentação de citações em documentos deve seguir a NBR 10520 (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2002b). O sistema a ser usado é o alfabético. Exemplos de citações são: “Numa publicação recente (SEBESTA, 2000) é exposto ...” e “Segundo Silva et al. (1987), execução controlada de programas é ...”.

Quando a citação referir-se a uma parte específica do documento consultado, especificar no texto do artigo a(s) página(s). Esta(s) deverá(ão) seguir a data, separada(s) por vírgula(s) e precedida(s) pelo designativo que a(s) caracteriza(m). Como exemplo, mostra-se: “(SCHIMT, 1999, p. 50)” ou “... visto que Schimt (1999, p. 50) implementou ...”.

As citações diretas (transcrição textual de parte da obra do autor consultado), no texto, com mais de três linhas, devem ser destacadas com recuo de 4 cm da margem esquerda, com letra menor que a do texto utilizado e sem as aspas (FORMATO: TF-CITAÇÃO), conforme o exemplo a seguir.

A Associação Brasileira de Normas Técnicas (ABNT) é o Fórum Nacional de Normalização. As Normas Brasileiras, cujo conteúdo é de responsabilidade dos Comitês Brasileiros (ABNT/CB) e dos Organismos de Normalização Setorial (ABNT/ONS), são elaboradas por Comissões de Estudo (CE), formadas por representantes dos setores envolvidos, delas fazendo parte: produtores, consumidores e neutros (universidades, laboratórios e outros). (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2002b, p. 1).

Quando da citação de um nome (identificador) constituinte de uma entidade ou de um elemento de interface em um texto, deve-se utilizar o tipo de letra *courier new*, com tamanho nove (9). Para facilitar a formatação, existe o estilo de palavra denominado TF-COURIER9. Como exemplo cita-se nome de classe, atributo ou método. A seguir são apresentados exemplos.

As classes TTabelaTransicao e TExpressaoRegular são classes de interface, porém estão sendo consideradas como classes de domínio da aplicação.

Ao clicar no botão Confirmar, o software abre uma nova tela.