

NOTA: Alterar os dados apresentados, e REMOVER TODOS os comentários deste modelo antes de enviar ao orientador(a).



UFRR

UNIVERSIDADE FEDERAL DE RORAIMA
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Título do Trabalho

Nome do aluno

Boa Vista - RR

Mês (por extenso) e ano

Nome do aluno

Título do Trabalho

Monografia de Graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Roraima como requisito parcial para a obtenção do grau de bacharel em Ciência da Computação.

Orientador(a)

Nome e titulação do(a) professor(a) orientador(a)

Universidade Federal de Roraima

Departamento de Ciência da Computação

Boa Vista - RR

Mês (por extenso) e ano

DECLARAÇÃO DE AUTORIA

NOTA: Alterar os dados apresentados abaixo.

Eu, <**Nome Completo**> (código de matrícula <**Número de Matrícula**>), autor da(o) monografia/TCC (Trabalho de Conclusão de Curso) sob o título <**Título da monografia**>, declaro que o trabalho em referência é de minha total autoria e de minha inteira responsabilidade o texto apresentado. Declaro, ainda, que as citações e paráfrases dos autores estão indicadas com as respectivas obras e anos de publicação. Declaro, para os devidos fins que estou ciente:

- dos Artigos 297 a 299 do Código Penal, Decreto-Lei n. 2.848 de 7 de dezembro de 1940;
- da Lei n. 9.610, de 19 de fevereiro de 1998, sobre os Direitos Autorais; e
- que plágio consiste na reprodução de obra alheia e submissão da mesma como trabalho próprio ou na inclusão, em trabalho próprio, de ideias, textos, tabelas ou ilustrações (quadros, figuras, gráficos, fotografias, retratos, lâminas, desenhos, organogramas, fluxogramas, plantas, mapas e outros) transcritos de obras de terceiros sem a devida e correta citação da referência.

O corpo docente responsável pela avaliação deste trabalho poderá não aceitar o referido trabalho caso os pontos mencionados acima sejam descumpridos, por conseguinte, considerar-me reprovado.

Assinatura do acadêmico(a)
Boa Vista - RR, data (por extenso).

FOLHA DE APROVAÇÃO

NOTA: Alterar os dados apresentados abaixo.

Monografia de Graduação sob o título **<Título da monografia>** apresentada por **<Nome do aluno>** e aceita pelo Departamento de Ciência da Computação da Universidade Federal de Roraima, sendo aprovada por todos os membros da banca examinadora abaixo especificada:

Titulação e nome do(a) orientador(a)

Orientador(a)

Departamento

Universidade

Titulação e nome do(a) membro da banca examinadora

Co-orientador(a), se houver

Departamento

Universidade

Titulação e nome do membro da banca examinadora

Departamento

Universidade

Titulação e nome do membro da banca examinadora

Departamento

Universidade

Boa Vista - RR, data de aprovação (por extenso).

NOTA: Alterar os dados apresentados abaixo.

Homenagem que o autor presta a uma ou mais pessoas.

Agradecimentos

NOTA: Alterar o texto abaixo. Os agradecimentos são dirigidos àqueles que contribuíram de maneira relevante à elaboração do trabalho, sejam eles pessoas ou mesmo organizações.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

NOTA: Adicionar uma citação, uma frase que lhe inspire.

Água mole e pedra dura, tanto bate até que furo.

Autor

Título do Trabalho

Autor: Nome do aluno

Orientador: Nome e titulação do(a) professor(a) orientador(a)

Resumo

LEMBRETE: O resumo deve apresentar de forma concisa os pontos relevantes de um texto, fornecendo uma visão rápida e clara do conteúdo e das conclusões do trabalho. O texto, redigido na forma impessoal do verbo, é constituído de uma sequência de frases concisas e objetivas e não de uma simples enumeração de tópicos, não ultrapassando 500 palavras, seguido, logo abaixo, das palavras representativas do conteúdo do trabalho, isto é, palavras-chave e/ou descritores. Por exemplo, deve-se evitar, na redação do resumo, o uso de fórmulas, equações, diagramas e símbolos, optando-se, quando necessário, pela transcrição na forma extensa, além de não incluir citações bibliográficas.

Palavras-chave: Palavra-chave 1, Palavra-chave 2, Palavra-chave 3.

Título do Trabalho

Autor: Nome do aluno

Orientador: Nome e titulação do(a) professor(a) orientador(a)

Abstract

NOTA: O abstract é uma versão, no geral no idioma inglês, do seu resumo. Também deve atender as mesmas restrições do resumo.

This is the english abstract.

Keywords: Keyword 1, Keyword 2, Keyword 3.

Lista de figuras

Figura 1 – Exemplo de código	19
Figura 2 – Fluxo de microserviços.	20
Figura 3 – Algoritmo para validar desalocação de endereço	21
Figura 4 – Como funciona o sistema de irrigação.	27
Figura 5 – Diagram de fluxo no modelo BPMN.	28

Lista de tabelas

Tabela 1 – Exemplo de tabela. 24

Tabela 2 – Classificação dos artigos por técnicas. 26

Tabela 3 – Cronograma de atividades 34

Lista de símbolos

λ Lambda

ω Omega

Sumário

1	INTRODUÇÃO	14
1.1	Motivação	16
1.2	Definição do Problema	16
1.3	Objetivos	17
1.3.1	Objetivo Geral	17
1.3.2	Objetivos Específicos	17
1.4	Organização do Trabalho	18
2	FUNDAMENTOS TEÓRICOS	19
2.1	Divisão a ser definida com o aluno de acordo com o tema .	19
2.1.1	Subdivisão de acordo com a necessidade de subdivisão	20
2.2	Divisão a ser definida com o aluno de acordo com o tema .	21
2.3	Destaques conceituais para pesquisa	22
3	TRABALHOS CORRELATOS	23
3.1	Trabalho 1 (Colocar o Nome do Trabalho)	23
3.2	Trabalho 2 (Colocar o Nome do Trabalho)	24
3.3	Trabalho 3 (Colocar o Nome do Trabalho)	25
3.4	Trabalho 4 (Colocar o Nome do Trabalho)	25
3.5	Correlações entre os trabalhos e a pesquisa	25
4	MÉTODO DA SOLUÇÃO PROPOSTA	27
4.1	Arquitetura	27
4.2	Ferramentas e Implementações	28
4.3	Considerações Adicionais	29
5	PLANEJAMENTO PARA AVALIAÇÃO EXPERIMENTAL	30
5.1	Projeto da Avaliação Experimental	30
6	CRONOGRAMA DE EXECUÇÃO PARA O TCC 2	32

6.1	Metas e Atividades	32
6.2	Cronograma	33
7	CONSIDERAÇÕES PARCIAIS	35
	Referências	36
	APÊNDICE A – PRIMEIRO APÊNDICE	37
	ANEXO A – PRIMEIRO ANEXO.	38

1 Introdução

NOTA: A introdução é a parte inicial do texto e que possibilita uma visão geral de todo o trabalho, devendo constar a delimitação do assunto tratado, objetivos da pesquisa, e outros elementos necessários para situar o tema do trabalho.

NOTA: O texto aqui apresentado é apenas a informação sobre algumas regras, para o escrita do seu trabalho, o texto abaixo DEVE SER REMOVIDO.

Algumas regras devem ser observadas na redação da monografia:

- ser claro, preciso, direto, objetivo e conciso, utilizando frases curtas e evitando ordens inversas desnecessárias;
- construir períodos com no máximo duas ou três linhas, bem como parágrafos com cinco linhas cheias, em média, e no máximo oito (ou seja, não construir parágrafos e períodos muito longos, pois isso cansa o(s) leitor(es) e pode fazer com que ele(s) percam a linha de raciocínio desenvolvida);
- a simplicidade deve ser condição essencial do texto; a simplicidade do texto não implica necessariamente repetição de formas e frases desgastadas, uso exagerado de voz passiva (como será iniciado, será realizado), pobreza vocabular etc. Com palavras conhecidas de todos, é possível escrever de maneira original e criativa e produzir frases elegantes, variadas, fluentes e bem alinhavadas;
- adotar como norma a ordem direta, por ser aquela que conduz mais facilmente o leitor à essência do texto, dispensando detalhes irrelevantes e indo diretamente ao que interessa, sem rodeios (verborragias);
- não começar períodos ou parágrafos seguidos com a mesma palavra, nem usar repetidamente a mesma estrutura de frase;

- desprezar as longas descrições e relatar o fato no menor número possível de palavras;
- recorrer aos termos técnicos somente quando absolutamente indispensáveis e nesse caso colocar o seu significado entre parênteses (ou seja, não se deve admitir que todos os que lerão o trabalho já dispõem de algum conhecimento desenvolvido no mesmo);
- dispensar palavras e formas empoladas ou rebuscadas, que tentem transmitir ao leitor mera ideia de erudição;
- não perder de vista o universo vocabular do leitor, adotando a seguinte regra prática: nunca escrever o que não se diria;
- usar termos coloquiais ou de gíria com extrema parcimônia (ou mesmo nem serem utilizados) e apenas em casos muito especiais, para não darem ao leitor a ideia de vulgaridade e descaracterizar o trabalho;
- ser rigoroso na escolha das palavras do texto, desconfiando dos sinônimos perfeitos ou de termos que sirvam para todas as ocasiões;
- em geral, há uma palavra para definir uma situação;
- encadear o assunto de maneira suave e harmoniosa, evitando a criação de um texto onde os parágrafos se sucedem uns aos outros como compartimentos estanques, sem nenhuma fluência entre si;
- ter um extremo cuidado durante a redação do texto, principalmente com relação às regras gramaticais e ortográficas da língua;
- geralmente todo o texto é escrito na forma impessoal do verbo, não se utilizando, portanto, de termos em primeira pessoa, seja do plural ou do singular.

Exemplo de nota para revisões, usando o comando

`\todo[inline]{comentário}`, você também pode usado o `\todo{nota}` para gera um nota em forma de balão pop-up.

1.1 Motivação

NOTA: Nesta Seção deve ser apresentado as motivações e fatos que apresentem a relevância do tema e problema abordado. Segue um **exemplo**:

Segundo Baldoni et al. (2018) diversos trabalhos na literatura tem abordado uma solução para o problema X que apresenta desafios como Y .

1.2 Definição do Problema

NOTA: Qual o problema que você está tentando resolver através do trabalho? Quais as restrições de projeto envolvidas?

NOTA: Nesta seção, você deve descrever a situação ou o contexto geral referente ao assunto em questão, devem constar informações atualizadas visando a proporcionar maior consistência ao trabalho. Segue um **exemplo**:

A verificação de gerenciamento de memória é uma tarefa importante para evitar comportamentos inesperados de programas, por exemplo, uma violação na propriedade de segurança de um ponteiro resulta em um endereço errado, que pode acabar produzindo uma saída incorreta do programa e não necessariamente um erro.

O problema considerado neste trabalho é expresso na seguinte questão: **Como complementar e aprimorar a verificação de propriedades de segurança de memória, com foco em aritmética de ponteiros e vazamentos de memória, com aplicação na linguagem de programação C?**

1.3 Objetivos

NOTA: Os objetivos constituem a finalidade de um trabalho científico, ou seja, a meta que se pretende atingir com a elaboração da pesquisa. Podemos distinguir dois tipos de objetivos em um trabalho científico, conforme apresentado nas próximas seções. **AQUI VOCÊ DEVE APRESENTAR UM TEXTO PARA INTRODUIZIR OS OBJETIVOS.**

1.3.1 Objetivo Geral

NOTA: É um objetivo mais amplos, ou seja, metas de longo alcance, as contribuições que se desejam oferecer com a execução da pesquisa. Segue um **Exemplo:**

O objetivo principal deste trabalho é aprimorar um método existente na literatura para calcular o valor de uma aplicativo para dispositivos moveis multi-plataforma.

1.3.2 Objetivos Específicos

NOTA: São a delimitação das metas mais específicas dentro do trabalho. São elas que, somadas, conduzirão ao desfecho do objetivo geral.

NOTA: Como os objetivos indicam ação, recomenda-se que eles sejam definidos por meio de verbos, tais como **analisar, avaliar, caracterizar, discutir, diagnosticar, investigar, implantar, pesquisar, realizar, determinar**, etc. Segue um **Exemplo:**

Os objetivos específicos são:

1. Propor uma técnica para instrumentação de programas escrito em C, adotando técnicas de compiladores como análise de representações intermediaria de código.

2. Analisar técnicas baseadas em execução simbólica para gerar dados de teste e identificação de localizações de erro em programas escritos em C.
3. Validar a aplicação dos métodos propostos sobre *benchmarks* públicos de programas em C, a fim de examinar a sua eficácia e aplicabilidade.

1.4 Organização do Trabalho

NOTA: Nesta seção deve ser apresentado como está organizado o trabalho, sendo descrito, portanto, do que trata cada capítulo. Segue um **exemplo**:

A introdução deste trabalho apresentou: o contexto, definição do problema, motivação, objetivos, metodologia e contribuições dessa pesquisa. Os capítulos restantes são organizados da seguinte forma:

- No Capítulo 2, **Fundamentos Teóricos**, são apresentados os conceitos abordados neste trabalho, especificamente: XXX, YYY, e ZZZ.
- No Capítulo 3, **Trabalhos Correlatos**, são analisados os trabalhos correlatos a solução proposta.
- No Capítulo 4, **Método da Solução Proposta**, é descrito as etapas de execução do método da solução proposta para XXXX.
- No Capítulo 5, **Planejamento para Avaliação Experimental**, é apresentado o planejamento e projeto para execução da avaliação da solução proposta.
- No Capítulo 6, **Cronograma de execução para o TCC 2**, é descrito o cronograma proposto para as próximas etapas do trabalho de conclusão de curso, bem como, a descrição das atividades propostas.
- E por fim no Capítulo 7, **Considerações Parciais**, apresenta-se as considerações parciais e análise das atividades já desenvolvidas.

2 Fundamentos Teóricos

NOTA: Referencial ou embasamento teórico - texto no qual se deve apresentar os aspectos teóricos, isto é, os conceitos utilizados e a definição dos mesmos. **AQUI VOCÊ DEVE APRESENTAR UM TEXTO PARA INTRODUZIR OS CONCEITOS QUE SERÃO ABORDADOS.**

2.1 Divisão a ser definida com o aluno de acordo com o tema

NOTA: A ideia de cada seção é abordar um aspecto teórico, contendo a definição, um exemplo e como este conceito se relaciona no seu trabalho. Segue **um exemplo** de apresentação de código na linguagem de programação C que pode ser utilizado na apresentação de conceitos.

Figura 1 – Exemplo de código

```

1  double custo(double entrada)
2  {
3      if (entrada < 0)
4      {
5          return -1;
6      }
7      if ((entrada * 1.5) <= 0)
8      {
9          return -1;
10     }
11     if(entrada > 50)
12     {
13         return 1.25*entrada;
14     }
15     return 1.5*entrada;
16 }
```

(a) Código C

```

1  TEST_CASE_1()
2  {
3      double result;
4      result = custo(0.49)
5      assert(result == 0.735);
6  }
7  TEST_CASE_2()
8  {
9      double result;
10     result = custo(51)
11     assert(result == 63.75);
12 }
```

(b) Código de Teste

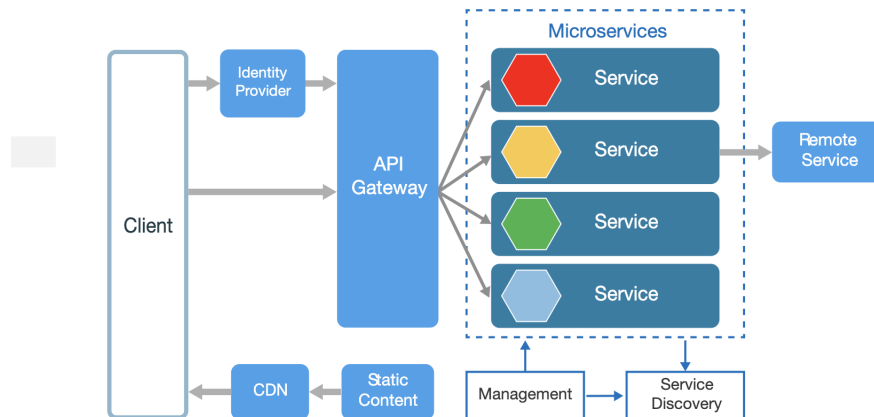
Fonte: Própria do autor.

2.1.1 Subdivisão de acordo com a necessidade de subdivisão

NOTA: Segue um exemplo de uso de figura

A Figura 2 apresenta o fluxo de microserviços.

Figura 2 – Fluxo de microserviços.



Fonte: Internet em (HYDRA1303, 2019).

NOTA: Continuação do texto. Segue um exemplo de apresentação de um algoritmo

O algoritmo na Figura 3 visa a identificação da liberação de memória:

Figura 3 – Algoritmo para validar desalocação de endereço

```

1: FUNÇÃO ISVALIDFREE(addr, mallocLog)
2:    $i \leftarrow \text{tamanhoDoLog}(\text{mallocLog}) - 1$ 
3:   ENQUANTO  $i \geq 0$  FAÇA ▷ O(m)
4:      $\text{row} \leftarrow \text{mallocLog}[i]$ 
5:      $i \leftarrow i - 1$ 
6:     SE ( $\text{row.addr} = \text{addr}$ ) ENTÃO ▷ O(1)
7:       SE  $\text{row.isFree}$  ENTÃO ▷ O(1)
8:         return FALSE
9:       SENÃO
10:        return TRUE
11:      FIM SE
12:    FIM SE
13:  FIM ENQUANTO
14:  return FALSE
15: FIM FUNÇÃO

```

Fonte: Própria do autor

2.2 Divisão a ser definida com o aluno de acordo com o tema

NOTA: Segue um exemplo de uso de referência de seção e de equações.

Referência a Seção 2.1.1. Um exemplo de exemplo de Equação 2.1:

$$f(n) = 4x^2 + 2y * 12 \quad (2.1)$$

Outro exemplo de matemática em Latex: Calcule as raízes da equação

$$x^2 + 12x - 13 = 0.$$

$$x = \frac{-12 \pm \sqrt{12^2 - (4)(1)(-13)}}{(2)(1)} = \frac{-12 \pm \sqrt{196}}{2} = \frac{-12 \pm 14}{2} = -6 \pm 7 \quad (2.2)$$

2.3 Destaques conceituais para pesquisa

NOTA: Nesta seção deve-se apresentar uma relação/contextualização para cada conceito apresentado, ou seja, como os conceitos se integram e como serão utilizados no trabalho apresentado.

3 Trabalhos Correlatos

NOTA: Nesta parte faz-se a revisão de literatura sobre o assunto, resumindo-se os resultados de estudos feitos por outros autores, cujas obras citadas e consultadas devem constar nas referências. Abaixo segue alguns exemplos do uso de citações com Latex.

Como usar citações no Latex:

- Adicionar a entrada no arquivo `referencias.bib`, que irá conter dados como: um identificador, nomes dos autores, título da publicação, local da publicação, ano e editora. Para exemplos, ver o repositório no GitHub ¹ da abnt.
- Uso de citação com nome, exemplo: Segundo Baldoni et al. (2018) bla bla ... Em latex seria: Segundo `\citeonline{SurveySymExec-CSUR:2018}` bla
- Uso de citação em final de texto, exemplo: Execução simbólica é uma técnica para explorar e examinar os caminhos de um código pela validação de predicados (BALDONI et al., 2018). Em latex a citação seria: `\cite{SurveySymExec-CSUR:2018}`.
- Exemplo de citação online (ROSEN, 2018).

3.1 Trabalho 1 (Colocar o Nome do Trabalho)

NOTA: Nesta seção deve-se apresentar uma análise do **primeiro** trabalho, que deve ser acadêmico e publicado. Sugerir-se o uso de trabalhos publicados em revistas, jornais, conferências e outros que apresentem como requisito uma revisão para publicação. Abaixo segue algumas recomendações a serem seguidas.

¹ <https://github.com/abntex/abntex2/blob/master/doc/latex/abntex2/examples/abntex2-modelo-references.bib>

Recomendações de onde buscar trabalhos a serem utilizados no seu TCC:

1. Pesquise no site google scholar em <https://scholar.google.com/>
2. Pesquise trabalhos publicados em bibliotecas digitais como: IEEE (www.ieee.org); Springer (www.springer.com); ACM (www.acm.org); periódicos CAPES (www.periodicos.capes.gov.br); e SCOPUS Elsevier (www.scopus.com), vale ressaltar que este último contém as publicações das anteriores.
3. Vale ressaltar que o acesso a algumas publicações/trabalhos nas bibliotecas enumeradas anteriormente podem requerer pagamento. Contudo, o acesso feito pela rede da UFRR possibilita um conta com acesso as bibliotecas digitais.
4. Para adicionar o trabalho as referências, recomenda-se o site dblp (<https://dblp.org/>) que já apresenta os detalhes do trabalho no formato bibtex.

3.2 Trabalho 2 (Colocar o Nome do Trabalho)

NOTA: Nesta seção deve-se apresentar uma análise do **segundo** trabalho, que deve ser acadêmico e publicado. Sugeri-se o uso de trabalhos publicados em revistas, jornais, conferências e outros que apresentem como requisito uma revisão para publicação. Abaixo também é apresentado um exemplo do uso de tabelas.

Teste de uma tabela:

Tabela 1 – Exemplo de tabela.

Título Coluna 1	Título Coluna 2
X	Y
X	W

Fonte: Própria do autor.

3.3 Trabalho 3 (Colocar o Nome do Trabalho)

NOTA: Nesta seção deve-se apresentar uma análise do **terceiro** trabalho, que deve ser acadêmico e publicado. No título acima, substituir “Trabalho 3 (Colocar o Nome do Trabalho)” pelo título do trabalho, exemplo, “SMT-Based Bounded Model Checking for Embedded ANSI-C Software”. No texto não esqueça de citar o referido trabalho, exemplo, **No trabalho de Cordeiro et al. (2012) é apresentado um software para ...**

3.4 Trabalho 4 (Colocar o Nome do Trabalho)

NOTA: Nesta seção deve-se apresentar uma análise do **quarto** trabalho, que deve ser acadêmico e publicado.

3.5 Correlações entre os trabalhos e a pesquisa

NOTA: Deve-se apresentar uma visão geral e comparativa dos trabalhos apresentados, bem como, cada trabalho apresentado pode contribuir com o seu trabalho. Sugere-se também apresentar uma tabela comparativa entre os trabalhos apresentados. Abaixo segue um exemplo:

A seguir analisamos as técnicas (ver Tabela 2) adotadas para a geração das invariantes de programa nos trabalhos apresentados anteriormente. Vale ressaltar que as técnicas identificadas foram:

- **Análise de Predicados.** Analisando os métodos baseadas na análise de predicado, podemos observar que a análise de predicado fornece um apoio significativo na análise do programa para inferir propriedades sobre o comportamento do programa;
- **Interpretação Abstratata.** É uma teoria da aproximação da semântica de linguagens de programação cuja aplicação principal é a análise estática;

- **Lógica de Mill.** A lógica de Mill é adotada para caracterizar laços por meio de uma função que define o seu espaço de estados.
- **Templates.** Analisando as publicações, notamos que a adoção de *templates* é importante para fornecer um suporte para guiar a inferência de invariantes de programas.

Tabela 2 – Classificação dos artigos por técnicas.

Artigos	Técnicas			
	Predicados	Int. Abstrata	Log. Mill	Templates
Trabalho 1	X			
Trabalho 2		X	X	
Trabalho 3	X	X		X
Trabalho 4	X	X		X

Fonte: Própria do autor.

4 Método da Solução Proposta

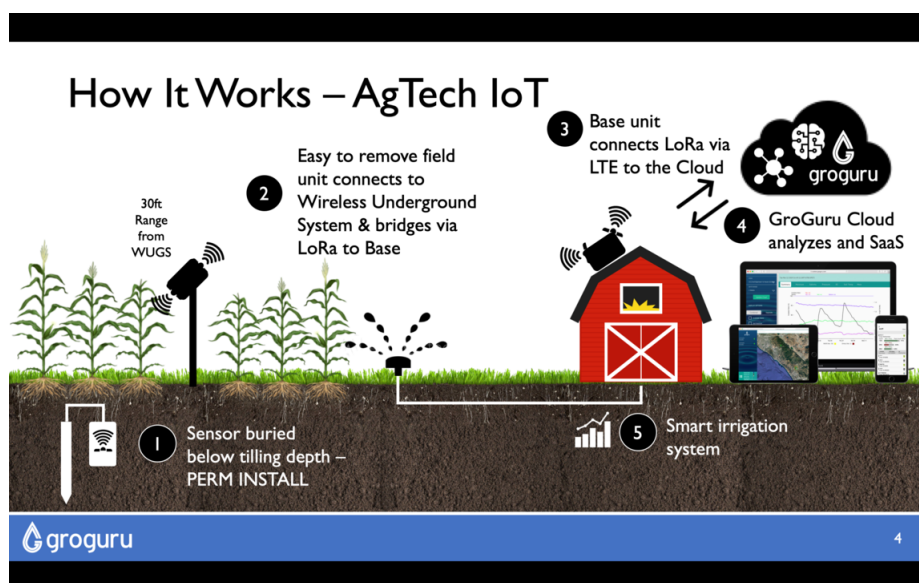
NOTA: Deve-se apresentar a visão geral da solução proposta, neste capítulo você deve apresentar metodologia do trabalho ou procedimentos metodológicos – deve constar o instrumental, os métodos e as técnicas aplicados para a elaboração do trabalho.

4.1 Arquitetura

NOTA: Deve-se apresentar uma visão geral da solução proposta, exemplo, usando uma Big Picture. Abaixo segue um **exemplo**.

A Figura 4 apresenta uma visão geral da solução proposta que visa o desenvolvimento e análise de uma sistema de irrigação usando tecnologias de IoT.

Figura 4 – Como funciona o sistema de irrigação.

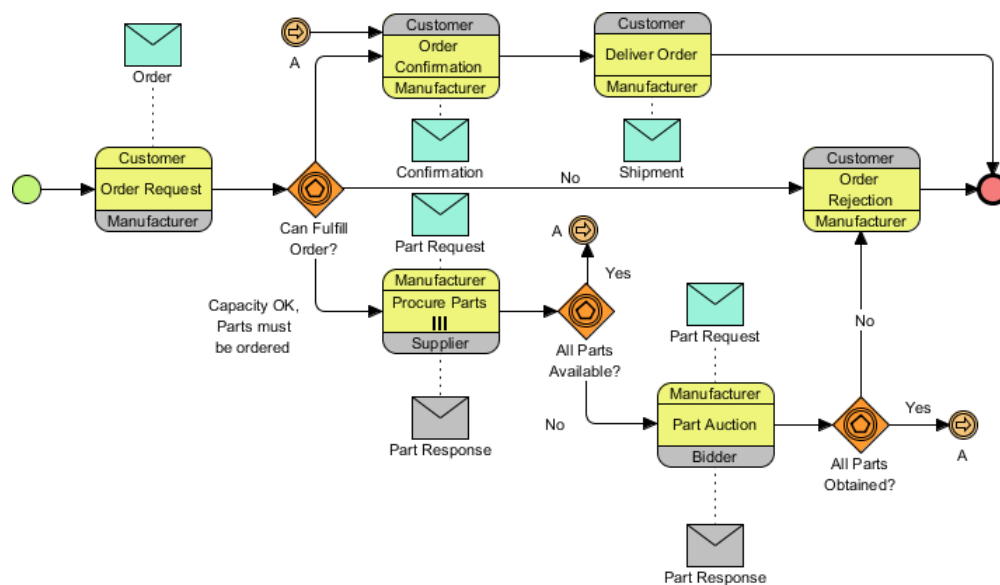


Fonte: Internet em (GROGURU, 2019)

NOTA: Nesta seção, também deve-se apresentar o fluxo da sua solução, ou seja, um diagrama contendo os artefatos de entrada e saída para cada etapa da solução proposta. Um modelo de digrama que pode ser utilizado é o BPMN (<http://www.bpmn.org>). Abaixo segue um **exemplo**.

A Figura 5 apresenta o fluxo de execução da solução proposta contendo suas respectivas entradas e artefatos gerados no modelo BPMN.

Figura 5 – Diagram de fluxo no modelo BPMN.



Fonte: Internet em (PARADIGM, 2019)

4.2 Ferramentas e Implementações

NOTA: Deve-se apresentar as ferramentas já pesquisadas que serão adotadas na solução propostas, bem como, explicar como as ferramentas serão correlacionadas. **Importante**, mencionar o nome da ferramenta, versão e onde pode ser encontrada, segue um pequeno **exemplo** de uso.

A solução proposta será uma ferramenta de verificação de código implementada como um transformador de código escrita em C/C++ usando o

framework para compiladores LLVM ¹ (v6.0) (LATTNER; ADVE, 2004). A solução utilizará como *front-end* para programas escritos em C o Clang ² (v6.0) (FANDREY, 2010) para gerar código LLVM-bitcode que será usado para a transformação de código. As ferramentas LibFuzzer (LIBFUZZER, 2019) (v6.0) e KLEE (CADAR et al., 2008) (v2.0) serão utilizadas para gerar entrada de testes para os códigos que serão analisados. Finalmente, o MetaSMT (v4.rc2) será usado como API para motores de solucionadores de satisfabilidade.

4.3 Considerações Adicionais

NOTA: Deve-se apresentar um visão geral em relação a construção da solução proposta. Analisar os pontos positivos e limitações da prototipação da prevista para a execução da solução.

¹ <https://llvm.org/>

² <https://clang.llvm.org/>

5 Planejamento para Avaliação Experimental

NOTA: Deve-se adicionar um texto para introduzir o capítulo, segue um **exemplo**

Este capítulo descreve o planejamento para a execução da avaliação experimental da solução proposta, incluindo: o planejamento e projeto para a execução de um estudo experimental para avaliar o método proposto neste trabalho.

5.1 Projeto da Avaliação Experimental

NOTA: Deve-se descrever o projeto para executar os testes para avaliar a solução proposta, incluindo o ambiente, elementos que serão utilizados, artefatos de entrada e saída, dados que serão coletados, e métricas para avaliação. Segue um **exemplo** abaixo.

Esta seção descreve o planejamento e concepção para a execução de um estudo empírico realizado com o objetivo de avaliar o método proposto para **XXXXXX**. O estudo foi conduzido aplicando o método proposto sobre *benchmarks* públicos de programas em C. Os experimentos foram conduzidos em um computador Intel Xeon CPU E5, 2.60GHz, 115GB RAM com Linux 3.13.0 - 35-generic x86_64.

NOTA: Deve-se também apresentar como será executado a avaliação e qual o seu objetivo de cada ação na avaliação. Neste momento você deve considerar as formas de testar (cenários importante) e como coletar os dados para sua avaliação. Segue um **exemplo**.

Esta avaliação empírica tem como objetivo analisar a capacidade do método proposto, sobre *benchmarks* públicos de programas em C, para contribuir

com a verificação executada pelo software X . Desta forma, nesta avaliação, investiga-se as seguintes questões de pesquisa (QP):

QP1 : As ferramentas para a geração de invariantes são capazes de suportar as diferentes estruturas da linguagem de programação C?

QP2 : As abordagens propostas para geração de dados de teste contribuem para a geração de invariantes?

QP3 : As invariantes geradas contribuem na verificação executada pelo ESBMC?

Visando responder as questões de pesquisa, a execução deste experimento foi dividida em duas partes: Na **primeira parte**, considerou-se 16 programas em ANSI-C dos seguintes *benchmarks* públicos: 6 programas do InvGen (BALDONI et al., 2018), 7 programas da categoria loops (CORDEIRO et al., 2012). A execução da **segunda parte** dos experimentos consistiu em: (1) selecionar uma das ferramentas (aquela que apresente melhor suporte aos programas em C) para gerar as invariantes dos programas do benchmark da categoria loops; e (3) executar a verificação dos programas com as invariantes usando o ESBMC (versão 1.25.2) com k indução para efetuar uma comparação dos resultados.

6 Cronograma de execução para o TCC 2

NOTA: Deve-se apresentar e descrever o cronograma proposto para a execução do TCC 2. Segue um texto de introdução.

Neste capítulo é apresentado o cronograma proposto para as próximas etapas do trabalho de conclusão de curso, bem como, a descrição das atividades.

6.1 Metas e Atividades

NOTA: Deve-se listar as atividades e metas que irão compor o cronograma para o TCC 2. Segue um **exemplo**.

As atividades propostas para a continuação deste trabalho são apresentadas a seguir.

1. **Revisão da literatura sobre invariantes de memória:** Invariantes de memória adicionam condições de pré e pós execução de operações sobre memória, sendo importante para este trabalho no que concerne a geração de *witness*;
2. **Aprimoramento do rastreamento de memória:** Comparar o rastreamento memória atual em outros modelos de memória, exemplo, o de precisão de *bit* utilizado pelo VALGRIND;
3. **Definição formal das propriedades de memória utilizado na verificação:** Criação de regras lógicas (baseada na lógica de inferência) para as propriedades verificadas.
4. **Aplicação de um *slicer* de código para o Map2Check:** No modelo atual, a instrumentação do Klee pode gerar muitos casos e facilmente gerar um

número excessivo de estados, assim será estudo como utilizar um *slicer* pra resolver esse problema.

5. **Adicionar um *bounded model checker* no método:** Com o uso de BMC será possível gerar outras propriedades de segurança de memória de forma mais otimizada, devido a aplicação de outras técnicas como *Lazy Abstraction* (CORDEIRO et al., 2012).
6. **Tradução das propriedades do *bounded model checker*:** É necessário fazer com que o Map2Check seja compatível com a propriedades geradas pelo BMC, logo serão traduzidas as propriedades.
7. **Avaliação experimental das novas propriedades suportadas pelo método:** Testes empíricos sobre as novas funcionalidades.
8. **Realização de experimentos com *benchmarks* públicos de programas escritos em C:** Com esses experimentos podemos verificar a eficácia real do método.
9. **Finalização da monografia:** Escrita sobre tudo o que foi desenvolvido e estudado no trabalho.
10. **Apresentação final:** Defesa do trabalho de conclusão de curso.

6.2 Cronograma

NOTA: Deve-se apresentar uma tabela contendo o cronograma proposto para a execução das próximas etapas do TCC. Segue um **exemplo**.

A Tabela 3 apresenta o cronograma contendo as atividades apresentadas na Seção 6.1.

Tabela 3 – Cronograma de atividades

Atividade	Agosto	Setembro	Outubro	Novembro	Dezembro
Revisão da literatura sobre invariantes de memória	×				
Aprimoramento do rastreamento de memória	×	×			
Definição formal das propriedades de memória utilizado na verificação		×	×	×	
Aplicação de um <i>slicer</i> de código para o Map2Check	×	×			
Adicionar um <i>bounded model checker</i> no método			×		
Tradução das propriedades do <i>bounded model checker</i>			×	×	
Avaliação experimental das novas propriedades suportadas pelo método			×	×	
Realização de experimentos com <i>benchmarks</i> públicos de programas escritos em C				×	
Finalização da monografia					×
Apresentação final					×

Fonte: Própria do autor.

7 Considerações Parciais

NOTA: Deve-se descrever as considerações parciais que formam a parte final (fechamento) do texto, sendo dito de forma resumida (1) o que foi desenvolvido no presente trabalho, (2) o que se espera após o desenvolvimento bem como as principais contribuições do trabalho, e (3) os desafios no desenvolvimento das próximas atividades do referido trabalho.

NOTA: A próxima seção é as referências, estas devem estar com os dados completos, visando a identificação da publicação citada ou apresentada.

Referências

- BALDONI, R.; COPPA, E.; D'ELIA, D. C.; DEMETRESCU, C.; FINOCCHI, I. A Survey of Symbolic Execution Techniques. **ACM Computing Surveys**, ACM, v. 51, n. 3, 2018.
- CADAR, C.; DUNBAR, D.; ENGLER, D. KLEE: Unassisted and Automatic Generation of High-Coverage Tests for Complex Systems Programs. In: **Symposium on Operating Systems Design and Implementation (OSDI)**. [S.l.]: USENIX, 2008. p. 209–224.
- CORDEIRO, L.; FISCHER, B.; MARQUES-SILVA, J. SMT-Based Bounded Model Checking for Embedded ANSI-C Software. In: **Transactions on Software Engineering (TSE)**. [S.l.]: IEEE, 2012. p. 957–974.
- FANDREY, D. Clang/LLVM Maturity Report. In: **Computer Science Dept., University of Applied Sciences Karlsruhe**. [S.l.: s.n.], 2010. See <<http://www.iwi.hs-karlsruhe.de>>.
- GROGURU. **AgTech IoT**. [S.l.]: <https://www.internationalbiosolutions.com/groguru>, 2019. [Online; Acessado em Março de 2019].
- HYDRA1303. **Arquitetura de Microsserviços (microservices) para Kubernetes**. [S.l.]: <https://www.hydra1303.com/language/pt/pks-pt/arquitetura-de-microsservicos-microservices-para-kubernetes-2/>, 2019. [Online; Acessado em Março de 2019].
- LATTNER, C.; ADVE, V. LLVM: A Compilation Framework for Lifelong Program Analysis & Transformation. In: **International Symposium on Code Generation and Optimization (CGO)**. [S.l.]: IEEE, 2004. p. 75–88.
- LIBFUZZER. **A library for coverage-guided fuzz testing**. 2019. <<https://llvm.org/docs/LibFuzzer.html>>. [Online; Acessado em Março de 2019].
- PARADIGM, V. **How to Draw BPMN 2.0 Business Process Diagram?** [S.l.]: <https://www.visual-paradigm.com/tutorials/businessprocessmodeling.jsp>, 2019. [Online; Acessado em Março de 2019].
- ROSEN, G. **Security Update Facebook, Inc.** [S.l.]: <https://newsroom.fb.com/news/2018/09/security-update/>, 2018. [Online; Acessado em Fevereiro de 2019].

APÊNDICE A – Primeiro apêndice

NOTA: Os apêndices são textos ou documentos elaborados pelo autor, a fim de complementar sua argumentação, sem prejuízo da unidade nuclear do trabalho. Este capítulo deve ser usado de acordo com a necessidade e em discussão com o orientador(a).

ANEXO A – Primeiro anexo.

NOTA: Os anexos são textos ou documentos não elaborados pelo autor, que servem de fundamentação, comprovação e ilustração. Este capítulo deve ser usado de acordo com a necessidade e em discussão com o orientador(a).