



UNIVERSIDADE SÃO JUDAS TADEU

CONCEITOS DE TÉCNICAS DE REVISÃO DE SOFTWARE

GIULIA GABRIELA DE LIMA SANTOS | 823122979
KAUE BRITO VIEIRA | 824211851
KAUÊ DIB DE SOUZA DIAS 823149871
MURILO BONUCCELLI DE OLIVEIRA | 823148988
VICTOR IGNACIO | 823125249
VINÍCIUS SANTANA TEIXERA | 82319112

1. Introdução

- **Objetivo do Plano de Teste:** Este plano descreve as estratégias, métodos e recursos para a realização de testes de software no algoritmo de busca binária, com o objetivo de verificar sua funcionalidade, robustez e desempenho em diferentes cenários de entrada. A aplicação da técnica de teste de caixa branca visa garantir a cobertura dos fluxos de decisão do código.

2. Objetivos do Projeto

- Validar a funcionalidade do algoritmo de busca binária para garantir a correta localização de elementos em um vetor ordenado.
- Verificar a cobertura de decisões no código do algoritmo, incluindo estruturas condicionais e laços.
- Garantir a confiabilidade do algoritmo em diferentes cenários de entradas (vetores vazios, vetores com um único elemento, etc.).

3. Justificativa

A aplicação do teste de caixa branca permite validar a execução de todas as condições lógicas internas do algoritmo, garantindo que ele funcione corretamente para todos os casos previstos. Esse tipo de teste é crucial, pois a busca binária é uma técnica amplamente utilizada para otimizar a busca em vetores ordenados, sendo importante validar sua eficácia e segurança.

4. Definição do Escopo do Projeto

- **Incluído:**
 - Testes do algoritmo de busca binária.

- Testes manuais para verificação da cobertura das estruturas condicionais do código.
- Teste de fluxos de controle como `if`, `else`, `while`, e verificação de decisões de divisão do vetor.
- **Excluído:**
 - Testes de desempenho e otimização do algoritmo.
 - Testes em sistemas que não utilizem vetor ordenado.
 - Integração do algoritmo com outros sistemas ou módulos.

5. Plano de Gestão de Testes

5.1. Planejamento

Este plano de teste será executado com base na metodologia de caixa branca, com foco em garantir que o algoritmo percorra todas as estruturas de controle do código. A técnica de cobertura de decisões será aplicada, abrangendo todos os ramos de execução, como `if`, `else if`, `else`, e o laço `while`.

5.2. Metodologia

O teste será realizado manualmente com a execução de casos de teste que cubram:

- **Casos positivos:** onde o valor está presente no vetor.
- **Casos negativos:** onde o valor não está presente.
- **Casos de limites:** como vetores vazios ou com um único elemento.

5.3. Recursos

- **Equipe de Testes:** [Nome da equipe ou responsável]
- **Ferramentas:**
 - Editor de código (VSCode, Eclipse, etc.)
 - Ambiente de desenvolvimento (Python, Java, etc.)
- **Infraestrutura:** Ambiente local de desenvolvimento, sem necessidade de recursos adicionais.

6. Identificação e Descrição dos Casos de Teste

Caso de Teste	Entrada	Valor a Buscar	Resultado Esperado	Objetivo do Teste
CT01	{1, 3, 5, 7, 9}	5	2	Testar caso em que o valor está no meio do vetor.
CT02	{1, 3, 5, 7, 9}	3	1	Testar valor menor que o elemento do meio, que entra no <code>if</code> .
CT03	{1, 3, 5, 7, 9}	9	4	Testar valor maior que o meio, que entra no <code>else if</code> .
CT04	{1, 3, 5, 7, 9}	4	-1	Testar valor inexistente no vetor, percorrendo todo o laço <code>while</code> .
CT05	{10}	10	0	Testar vetor com um único elemento.
CT06	{}	10	-1	Testar vetor vazio.
CT07	{1, 2, 3, 4, 5, 6, 7}	6	5	Testar valor à direita, mais próximo do final do vetor.
CT08	{1, 2, 3, 4, 5, 6, 7}	1	0	Testar valor à esquerda, mais próximo do início do vetor.

7. Recursos Humanos e Treinamento

- **Equipe de Testes:** A equipe será composta por [nomes da equipe] que têm experiência com testes de caixa branca e validação de algoritmos.
- **Treinamento:** A equipe passará por uma breve sessão de treinamento sobre o algoritmo de busca binária e sobre como validar a cobertura das estruturas de decisão no código.

8. Cronograma

O cronograma de execução dos testes está organizado da seguinte forma:

Etapa	Data de Início	Data de Término	Responsável
Planejamento de Teste	01/05/2025	03/05/2025	Carlos
Execução dos Testes	04/05/2025	08/05/2025	Lucas
Análise e Relatório Final	09/05/2025	12/05/2025	CAIO

9. Plano de Comunicação

- **Relatório de Progresso:** Relatórios semanais serão enviados à liderança do projeto.
- **Reuniões de Status:** Reuniões de acompanhamento a cada duas semanas para revisar progresso e resultados dos testes.
- **Canal de Comunicação:** E-mail, Slack ou Microsoft Teams.

10. Riscos e Mitigações

- **Risco 1:** Inconsistência nos resultados devido a erros lógicos no algoritmo.
 - **Mitigação:** Realização de testes em diferentes cenários para garantir a robustez do algoritmo.
- **Risco 2:** Tempo de execução dos testes ser superior ao esperado.
 - **Mitigação:** Planejamento e execução eficiente dos casos de teste com a equipe treinada.
- **Risco 3:** Falta de cobertura total de decisões no código.
 - **Mitigação:** Garantir que todos os caminhos lógicos sejam cobertos, verificando o relatório de cobertura.

11. Critérios de Aceitação

- **Passagem:** Todos os casos de teste executados devem produzir resultados conforme esperado. A cobertura das estruturas de decisão deve ser 100%.
- **Rejeição:** Se algum teste falhar ou a cobertura não for atingida, o algoritmo precisa ser ajustado ou o caso de teste será reavaliado.

12. Encerramento do Projeto

- **Relatório Final:** O relatório final será gerado após a execução dos testes e avaliação dos resultados. Nele serão descritas as conclusões sobre a correção do algoritmo de busca binária e os passos para aprimoramento, caso necessário.
- **Reunião de Encerramento:** Após a conclusão, uma reunião será realizada com as partes interessadas para revisar os resultados dos testes e encerrar o projeto.