

**Universidade São Judas Tadeu - Butantã**  
**Noturno**

**Nomes: Gabriel Carvalho dos Santos - 821159957**  
**Gustavo Mendes de Lima - 821139542**  
**Matheus dos Santos de Souza - 821133576**  
**Lucas Rodrigues Santos - 823124699**

**Turma: GQS-CCP1AN-BUE1**

**Professor: Robson Calvetti**

## Exercício prático 1:

Plano de Testes;

Realizar um plano de testes para o exercício prático 1

```
public static int busca_binaria(int iVet[], int iK)
{
    int iBaixo, iAlto, iMeio;

    iBaixo=0;
    iAlto=iVet.length-1;
    while(iBaixo <= iAlto)
    {
        iMeio=(iBaixo+iAlto)/2;
        if(iK < iVet[iMeio]) iAlto=iMeio-1;
        else if(iK > iVet[iMeio]) iBaixo=iMeio+1;
        else return iMeio;
    }
    return -1;
}
```

## Plano de Teste para a Função

### Introdução

- **Identificação do Projeto:** Implementação da função de busca binária em um array de inteiros.
- **Objetivos do Documento:** Este plano de teste tem como objetivo definir os testes a serem realizados para validar a implementação da função busca binária.
- **Escopo:** Testar a funcionalidade da busca binária em arrays ordenados.

### Requisitos do Teste

- **Requisitos Funcionais:**
  1. A função deve retornar o índice do elemento procurado se ele estiver presente no array.
  2. A função deve retornar -1 se o elemento não estiver presente.
- **Requisitos Não Funcionais:**
  1. A função deve operar em tempo  $O(\log n)$  para arrays ordenados.

### Estratégias e Ferramentas

- **Tipos de Testes:**
  1. Testes de unidade: Verificação da funcionalidade da função com diferentes entradas.
  2. Testes de limite: Testes com arrays vazios e arrays com um único elemento.

- **Técnicas Empregadas:**

Análise de valores de retorno esperados e comparação com os retornos reais.

- **Critério de Finalização de Teste:**

Todos os casos de teste devem ser executados e seus resultados devem corresponder às expectativas.

- **Infraestrutura:**

Ambiente de desenvolvimento: Visual Studio (VScode).

## Casos de Teste

- **Teste com elemento presente no meio do array:**

1. Entrada: iVet = [1, 2, 3, 4, 5], iK = 3
2. Saída Esperada: 2 (índice de 3)

- **Teste com elemento presente no início do array:**

3. Entrada: iVet = [1, 2, 3, 4, 5], iK = 1
4. Saída Esperada: 0

- **Teste com elemento presente no final do array:**

1. Entrada: iVet = [1, 2, 3, 4, 5], iK = 5
2. Saída Esperada: 4

- **Teste com elemento não presente no array:**

1. Entrada: iVet = [1, 2, 3, 4, 5], iK = 6
2. Saída Esperada: -1

- **Teste com array vazio:**

1. Entrada: iVet = [], iK = 1
2. Saída Esperada: -1

- **Teste com array de um elemento presente:**

1. Entrada: iVet = [5], iK = 5
2. Saída Esperada: 0

- **Teste com array de um elemento ausente:**

1. Entrada: iVet = [5], iK = 3
2. Saída Esperada: -1

## Exercício prático 2:

Roteiro de Testes;

Realizar um roteiro de testes para o exercício prático 1

```
public static int busca_binaria(int iVet[], int iK)
{
    int iBaixo, iAlto, iMeio;

    iBaixo=0;
    iAlto=iVet.length-1;
    while(iBaixo <= iAlto)
    {
        iMeio=(iBaixo+iAlto)/2;
        if(iK < iVet[iMeio]) iAlto=iMeio-1;
        else if(iK > iVet[iMeio]) iBaixo=iMeio+1;
        else return iMeio;
    }
    return -1;
}
```

### Roteiro de Teste de Software para a Função

#### Objetivo:

Validar a implementação da função busca\_binaria, que realiza uma busca binária em um vetor ordenado.

- **Documentos de Referência:**
  1. Especificação Funcional
  2. Guia de Interface

#### Estrutura do Roteiro de Teste:

- **Identificação do Objeto de Teste**
  1. **Função:** busca\_binaria
  2. **Tipo:** Algoritmo de busca
- **Pré-requisitos**
  1. O vetor iVet[] deve estar ordenado em ordem crescente.
  2. A função deve ser chamada com um vetor não nulo e um valor de busca iK.

## Casos de Teste

Caso de Teste	Descrição	Entrada	Resultado Esperado
CT01	Busca um elemento presente no vetor	iVet = [1, 2, 3, 4, 5], iK = 3	2 (índice do elemento 3)
CT02	Busca um elemento no início do vetor	iVet = [1, 2, 3, 4, 5], iK = 1	0 (índice do elemento 1)
CT03	Busca um elemento no final do vetor	iVet = [1, 2, 3, 4, 5], iK = 5	4 (índice do elemento 5)
CT04	Busca um elemento que não existe	iVet = [1, 2, 3, 4, 5], iK = 6	-1
CT05	Busca em vetor com um único elemento	iVet = [3], iK = 3	0
CT06	Busca em vetor vazio	iVet = [], iK = 1	-1
CT07	Busca um elemento em vetor com dois elementos	iVet = [1, 3], iK = 1	0
CT08	Busca um elemento em vetor com dois elementos	iVet = [1, 3], iK = 3	1
CT09	Busca um elemento que está entre dois outros	iVet = [1, 2, 4], iK = 3	-1
CT10	Busca o menor elemento em vetor	iVet = [10, 20, 30, 40, 50], iK = 10	0
CT11	Busca o maior elemento em vetor	iVet = [10, 20, 30, 40, 50], iK = 50	4

- **Critérios de Aceitação**

1. A função deve retornar o índice correto do elemento buscado se este existir no vetor.
2. A função deve retornar -1 caso o elemento não esteja presente.
3. A função deve ser testada em diversos cenários, incluindo vetores vazios e de tamanho 1.

- **Execução dos Testes**

Registrando o resultado real e comparando com o resultado esperado.

## Exercício prático 3:

Plano de Testes;

Realizar um plano de testes para o exercício prático 2 da aula 04;

1. Receber, via teclado, na janela do sistema, o login do usuário no campo “Login”;
2. Receber, via teclado, na janela do sistema, a senha do usuário no campo “Senha”;
3. Aguardar, via mouse, na janela do sistema, a seleção do botão “Entrar” ;
4. Validar o login e a senha digitados com o login e a senha previamente armazenados na base de dados do sistema;
5. Se o login e/ou senha digitados não coincidirem com os previamente cadastrados, apresentar, na janela do sistema, a mensagem “Login e/ou Senha incorretos”;
6. Se o login e senha digitados coincidirem com os previamente cadastrados, gerar o código dinâmico para a validação em duas etapas;
7. Enviar o código de validação em duas etapas gerado pelo sistema, por SMS, para o número do celular do usuário previamente cadastrado no sistema;
8. Receber, via teclado, na janela do sistema, o código de validação em duas etapas enviado ao usuário por SMS;
9. Se o código de validação em duas etapas digitado não coincidir com o código gerado, apresentar, na janela do sistema, a mensagem “Login não autorizado!”;
10. Se o código de validação em duas etapas digitado coincidir com o código gerado, apresentar, na janela do sistema, a mensagem “Login realizado com sucesso” e liberar o acesso ao programa.

## Plano de Teste de Software

### Introdução

- **Identificação do Projeto:** Sistema de Login com Validação em Duas Etapas
- **Objetivos do Documento:** Estabelecer um plano de teste para garantir a funcionalidade e segurança do sistema de login.
- **Público-alvo:** Equipe de desenvolvimento, equipe de testes, gerentes de projeto e stakeholders.
- **Escopo:** Testar o fluxo de login e a validação em duas etapas, incluindo a interação do usuário com o sistema.

### Requisitos do Teste

- **Requisitos Funcionais:**
  1. Login e senha devem ser recebidos e validados.
  2. Mensagens de erro devem ser exibidas para credenciais incorretas.
  3. Geração e envio de código de validação em duas etapas.
  4. Recepção e validação do código de SMS.

- **Requisitos Não Funcionais:**
  1. O sistema deve responder em menos de 2 segundos para cada ação de validação.
  2. O sistema deve ser seguro e proteger os dados do usuário.

**Estratégias e Ferramentas**

- **Tipos de Testes:**
  1. Testes Funcionais: Verificar se todas as funcionalidades do sistema estão funcionando conforme o esperado.
  2. Testes de Usabilidade: Avaliar a experiência do usuário na interação com a interface.
  3. Testes de Segurança: Testar a proteção dos dados e a validação de acessos.
- **Técnicas Empregadas:**
  1. Testes de Caixa Preta.
  2. Testes Manuais e Automatizados.
- **Critérios de Finalização de Teste:**
  1. Todos os casos de teste devem ser executados com sucesso.
  2. Não devem existir falhas críticas não resolvidas.

**5. Cronograma de Atividades**

Atividade	Data de Início	Data de Término
Criação de Casos de Teste	30/09/2024	01/10/2024
Execução dos Testes Funcionais	30/09/2024	01/10/2024
Execução dos Testes de Usabilidade	30/09/2024	01/10/2024
Testes de Segurança	30/09/2024	01/10/2024
Análise dos Resultados	30/09/2024	01/10/2024
Relatório Final	30/09/2024	01/10/2024

**Casos de Teste**

**Caso de Teste 1: Validação de Login e Senha**

- **Objetivo:** Verificar se o sistema aceita credenciais corretas.
- **Entradas:** Login e senha corretos.
- **Resultados Esperados:** Acesso liberado e mensagem "Login realizado com sucesso".



### **Caso de Teste 2: Credenciais Incorretas**

- **Objetivo:** Verificar se o sistema rejeita credenciais incorretas.
- **Entradas:** Login ou senha incorretos.
- **Resultados Esperados:** Mensagem "Login e/ou Senha incorretos".

### **Caso de Teste 3: Validação de Código de SMS**

- **Objetivo:** Verificar se o sistema aceita o código de validação correto.
- **Entradas:** Código de validação correto.
- **Resultados Esperados:** Acesso liberado e mensagem "Login realizado com sucesso".

### **Caso de Teste 4: Código de Validação Incorreto**

- **Objetivo:** Verificar se o sistema rejeita um código de validação incorreto.
- **Entradas:** Código de validação incorreto.
- **Resultados Esperados:** Mensagem "Login não autorizado!".

## Exercício prático 4:

Roteiro de Testes;

Realizar um roteiro de testes para o exercício prático 2

1. Receber, via teclado, na janela do sistema, o login do usuário no campo "Login";
2. Receber, via teclado, na janela do sistema, a senha do usuário no campo "Senha";
3. Aguardar, via mouse, na janela do sistema, a seleção do botão "Entrar" ;
4. Validar o login e a senha digitados com o login e a senha previamente armazenados na base de dados do sistema;
5. Se o login e/ou senha digitados não coincidirem com os previamente cadastrados, apresentar, na janela do sistema, a mensagem "Login e/ou Senha incorretos";
6. Se o login e senha digitados coincidirem com os previamente cadastrados, gerar o código dinâmico para a validação em duas etapas;
7. Enviar o código de validação em duas etapas gerado pelo sistema, por SMS, para o número do celular do usuário previamente cadastrado no sistema;
8. Receber, via teclado, na janela do sistema, o código de validação em duas etapas enviado ao usuário por SMS;
9. Se o código de validação em duas etapas digitado não coincidir com o código gerado, apresentar, na janela do sistema, a mensagem "Login não autorizado!";
10. Se o código de validação em duas etapas digitado coincidir com o código gerado, apresentar, na janela do sistema, a mensagem "Login realizado com sucesso" e liberar o acesso ao programa.

## Roteiro de Teste de Software

**Objetivo:** Validar o processo de login e a autenticação em duas etapas do sistema.

### Pré-condições

- O sistema deve estar em funcionamento.
- O usuário deve ter um login e senha previamente cadastrados no sistema.
- O número de celular do usuário deve estar cadastrado para o envio do SMS.

<b>Caso de Teste</b>	<b>Descrição</b>	<b>Passos</b>	<b>Resultado Esperado</b>
CT01	Verificar mensagem de login e senha incorretos	1. Acessar a janela do sistema. 2. Inserir um login e senha incorretos. 3. Clicar no botão "Entrar".	A mensagem "Login e/ou Senha incorretos" deve ser exibida.
CT02	Verificar sucesso no login com senha correta	1. Acessar a janela do sistema. 2. Inserir um login e senha corretos. 3. Clicar no botão "Entrar".	Código dinâmico para validação em duas etapas deve ser gerado.
CT03	Verificar envio do código por SMS	1. Após inserir login e senha corretos, aguardar o envio do código por SMS. 2. Checar o celular para receber o código.	O código deve ser recebido no celular cadastrado.
CT04	Validar código de autenticação em duas etapas correto	1. Receber o código via SMS. 2. Inserir o código na janela do sistema. 3. Clicar em "Enviar".	A mensagem "Login realizado com sucesso" deve ser exibida. Acesso deve ser liberado ao programa.
CT05	Validar código de autenticação em duas etapas incorreto	1. Após o passo 3 do CT02, inserir um código incorreto. 2. Clicar em "Enviar".	Mensagem "Login não autorizado!" deve ser exibida.

## **Seção de Localização do Roteiro de Teste de Software**

**Objetivo:** Definir as localizações específicas no sistema onde os casos de teste serão executados.

### **Localização dos Objetos de Teste**

#### **Tela de Login**

- Localização: Tela Inicial do Sistema
  1. Campo "Login"
  2. Campo "Senha"
  3. Botão "Entrar"

#### **Tela de Autenticação em Duas Etapas**

- Localização: Tela de Validação em Duas Etapas (após clicar em "Entrar")
  1. Campo para Código de Validação
  2. Botão "Enviar"

#### **Observações:**

- Garantir que a mensagem de erro seja exibida de forma clara e em destaque.
- Confirmar que o código SMS seja único e não reutilizável.
- Verificar logs do sistema para auditoria de tentativas de login e envio de SMS.