Tarefa teste de Software com JUnit

Códigos-fonte das classes implementadas.

Classe Calculadora

```java
public class Calculadora {

    public int somar(int a, int b) {
        return a + b;
    }

    public int subtrair(int a, int b) {
        return a - b;
    }

    public int multiplicar(int a, int b) {
        return a * b;
    }

    public double dividir(double a, double b) {
        if (b != 0) {
            return a / b;
        } else {
            throw new ArithmeticException("Nao divide zero");
        }
    }
}
```

Classe NumeroUtil

```java
public class NumeroUtil {

    public static boolean ehPar(int numero) {
        return numero % 2 == 0;
    }

    public static boolean ehPrimo(int numero) {
        if (numero <= 1) {
            return false;
        }
        for (int i = 2; i <= Math.sqrt(numero); i++) {
            if (numero % i == 0) {
                return false;
            }
        }
        return true;
    }
}
```

Classe ValidadorSenha

```java
public class ValidadorSenha {
    public boolean validar(String senha) {
        if (senha == null || senha.length() < 8) {
            return false;
        }
        boolean temMaiuscula = false;
        boolean temMinuscula = false;
        boolean temDigito = false;
        for (char c : senha.toCharArray()) {
            if (Character.isUpperCase(c)) temMaiuscula = true;
            if (Character.isLowerCase(c)) temMinuscula = true;
            if (Character.isDigit(c)) temDigito = true;
        }
        return temMaiuscula && temMinuscula && temDigito;
    }
}
```

Classe CarrinhoDeCompras

```java
import java.util.HashMap;
import java.util.Map;

public class CarrinhoDeCompras {
    private Map<String, Double> produtos = new HashMap<>();

    public void adicionarProduto(String nome, double preco) {
        produtos.put(nome, preco);
    }

    public void removerProduto(String nome) {
        produtos.remove(nome);
    }

    public double calcularTotal() {
        double total = 0.0;
        for (double preco : produtos.values()) {
            total += preco;
        }
        return total;
    }
}
```

Códigos de teste com cobertura adequada.

```java
import static org.junit.Assert.assertEquals;
import static org.junit.Assert.assertThrows;

import org.junit.Test;

public class CalculadoraTest {
    @Test
    public void testDividir() {
        Calculadora calc = new Calculadora();
        assertEquals(expected:2.0, calc.dividir(a:10, b:5), delta:0.0001);
        assertEquals(-2.5, calc.dividir(-5, b:2), delta:0.0001);
        assertEquals(expected:0.0, calc.dividir(a:0, b:10), delta:0.0001);
        assertThrows(expectedThrowable:ArithmeticException.class, () -> {
            calc.dividir(a:10, b:0);
        });
    }

    @Test
    public void testMultiplicar() {
        Calculadora calc = new Calculadora();
        assertEquals(expected:6, calc.multiplicar(a:2, b:3));
        assertEquals(expected:0, calc.multiplicar(a:0, b:5));
        assertEquals(-15, calc.multiplicar(-3, b:5));
        assertEquals(expected:9, calc.multiplicar(-3, -3));
    }

    @Test
    public void testSomar() {
        Calculadora calc = new Calculadora();
        assertEquals(expected:5, calc.somar(a:2, b:3));
        assertEquals(expected:0, calc.somar(a:2, -2));
        assertEquals(-1, calc.somar(-2, b:1));
        assertEquals(-5, calc.somar(-2, -3));
    }

    @Test
    public void testSubtrair() {
        Calculadora calc = new Calculadora();
        assertEquals(-1, calc.subtrair(a:2, b:3));
        assertEquals(expected:4, calc.subtrair(a:2, -2));
        assertEquals(-3, calc.subtrair(-2, b:1));
        assertEquals(expected:1, calc.subtrair(-2, -3));
    }
}
```

NumeroUtilTest.java > NumeroUtilTest > testEhPrimoComNumerosNaoPrimos()

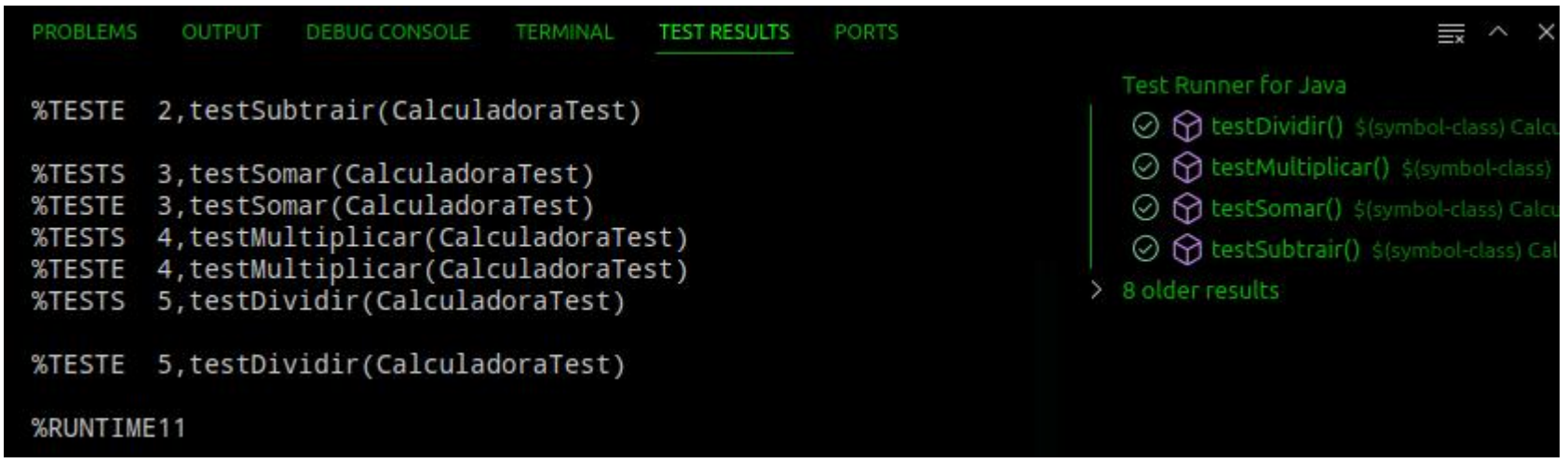
```java
import static org.junit.Assert.assertEquals;
import org.junit.Test;

public class NumeroUtilTest {

    @Test
    public void testEhParComNumeroPar() {
        assertEquals(expected:true, NumeroUtil.ehPar(numero:2));
        assertEquals(expected:true, NumeroUtil.ehPar(numero:0));
        assertEquals(expected:true, NumeroUtil.ehPar(-4));
    }

    @Test
    public void testEhParComNumeroImpar() {
        assertEquals(expected:false, NumeroUtil.ehPar(numero:3));
        assertEquals(expected:false, NumeroUtil.ehPar(-1));
        assertEquals(expected:false, NumeroUtil.ehPar(numero:7));
    }

    @Test
    public void testEhPrimoComNumerosPrimos() {
        assertEquals(expected:true, NumeroUtil.ehPrimo(numero:2));
        assertEquals(expected:true, NumeroUtil.ehPrimo(numero:3));
        assertEquals(expected:true, NumeroUtil.ehPrimo(numero:5));
        assertEquals(expected:true, NumeroUtil.ehPrimo(numero:13));
        assertEquals(expected:true, NumeroUtil.ehPrimo(numero:17));
    }

    @Test
    public void testEhPrimoComNumerosNaoPrimos() {
        assertEquals(expected:false, NumeroUtil.ehPrimo(numero:1));
        assertEquals(expected:false, NumeroUtil.ehPrimo(numero:0));
        assertEquals(expected:false, NumeroUtil.ehPrimo(-3));
        assertEquals(expected:false, NumeroUtil.ehPrimo(numero:4));
    }
}
```

```java
import static org.junit.Assert.assertFalse;
import static org.junit.Assert.assertTrue;
import org.junit.Before;
import org.junit.Test;

public class ValidadorSenhaTest {
    private ValidadorSenha validador;

    @Before
    public void setUp() {
        validador = new ValidadorSenha();
    }

    @Test
    public void testSenhaValida() {
        assertTrue(validador.validar(senha:"Senha123"));
        assertTrue(validador.validar(senha:"Abcdef1g"));
    }

    @Test
    public void testSenhaCurta() {
        assertFalse(validador.validar(senha:"Abc12"));
    }

    @Test
    public void testSemMaiuscula() {
        assertFalse(validador.validar(senha:"senha1234"));
    }

    @Test
    public void testSemMinuscula() {
        assertFalse(validador.validar(senha:"SENHA1234"));
    }

    @Test
    public void testSemDigito() {
        assertFalse(validador.validar(senha:"Senhaaaaa"));
    }

    @Test
    public void testSenhaNula() {
        assertFalse(validador.validar(senha:null));
    }
}
```

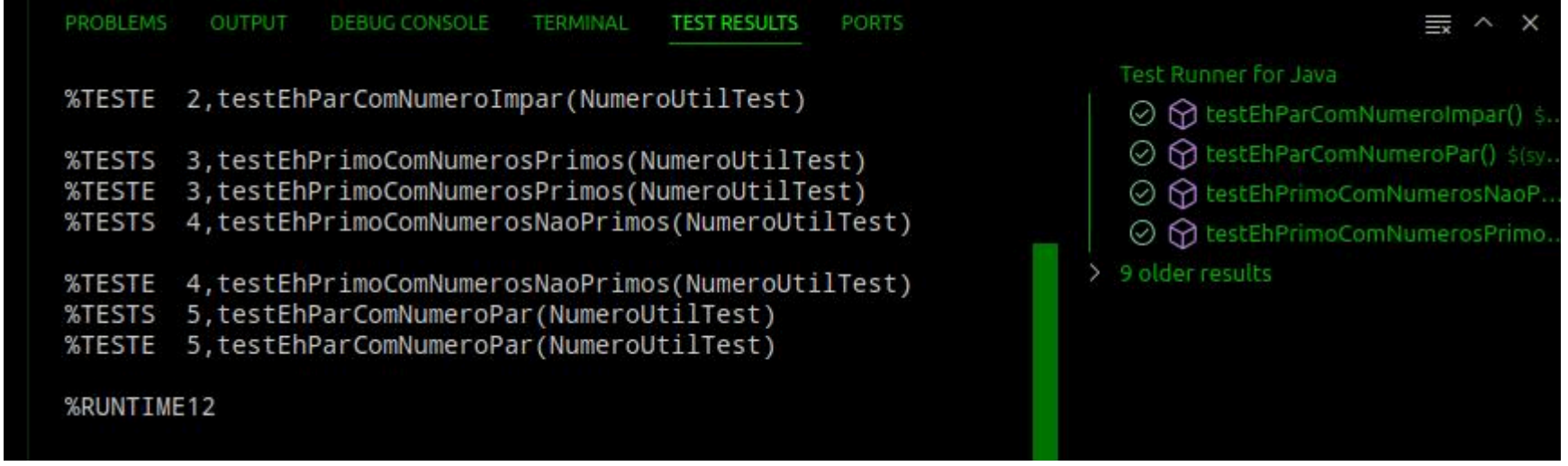
```java
import static org.junit.Assert.assertEquals;
import org.junit.Before;
import org.junit.Test;

public class CarrinhoDeComprasTest {
    private CarrinhoDeCompras carrinho;

    @Before
    public void setUp() {
        carrinho = new CarrinhoDeCompras();
    }

    @Test
    public void testAdicionarProduto() {
        carrinho.adicionarProduto(nome:"Arroz", preco:10.0);
        assertEquals(expected:10.0, carrinho.calcularTotal(), delta:0.0001);
    }

    @Test
    public void testAdicionarVariosProdutos() {
        carrinho.adicionarProduto(nome:"Arroz", preco:10.0);
        carrinho.adicionarProduto(nome:"Feijao", preco:8.5);
        carrinho.adicionarProduto(nome:"Macarrao", preco:5.0);
        assertEquals(expected:23.5, carrinho.calcularTotal(), delta:0.0001);
    }

    @Test
    public void testRemoverProduto() {
        carrinho.adicionarProduto(nome:"Arroz", preco:10.0);
        carrinho.adicionarProduto(nome:"Feijao", preco:8.5);
        carrinho.removerProduto(nome:"Arroz");
        assertEquals(expected:8.5, carrinho.calcularTotal(), delta:0.0001);
    }

    @Test
    public void testRemoverProdutoInexistente() {
        carrinho.adicionarProduto(nome:"Arroz", preco:10.0);
        carrinho.removerProduto(nome:"Feijao");
        assertEquals(expected:10.0, carrinho.calcularTotal(), delta:0.0001);
    }

    @Test
    public void testCalcularTotalVazio() {
        assertEquals(expected:0.0, carrinho.calcularTotal(), delta:0.0001);
    }
}
```

Prints gerado pelas ferramentas de cobertura (opcional).

Testes classe Calculadora



Testes classe Numero Util



Testes classe Validador Senhas

Teste Carrinho De Compras