

CHECKPOINT I – JUNIT Matheus Nathan Modanez Silva RM 86065

Junit Test Cases

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

public class CPFTeste {

    @Test
    public void testagemPositivaCPF() {
       CodigoPessoaFisica cpf = new CodigoPessoaFisica();
        String cpfTeste = "461.670.818-59";
        boolean resultadoEsperado = true;
        boolean resultadoObtido = cpf.validaCPF(cpfTeste);
        assertEquals(resultadoEsperado, resultadoObtido);
    }

    @Test
    public void testagemNegativaCPF() {
       CodigoPessoaFisica cpf = new CodigoPessoaFisica();
        String cpfTeste = "176.942.776-12";
        boolean resultadoEsperado = false;
        boolean resultadoObtido = cpf.validaCPF(cpfTeste);
        assertEquals(resultadoEsperado, resultadoObtido);
    }

    @Test
    public void testeRemoveCaracteresEspeciais() {
       CodigoPessoaFisica cpf = new CodigoPessoaFisica();
        String cpfTeste = "461.670.818-59";
        String resultadoEsperado = "46167081859";
        String resultadoObtido = cpf.removeCaracteresEspeciais(cpfTeste);
        assertEquals(resultadoEsperado, resultadoObtido);
    }
}
```

Resultados dos Testes

The screenshot displays an IDE with two panels. The left panel shows the JUnit test runner output, indicating that all three tests passed successfully. The right panel shows the source code of the `CPFTeste.java` file, which contains three test methods: `testagemPositivaCPF`, `testagemNegativaCPF`, and `testeRemoveCaracteresEspeciais`. Each test method uses `assertEquals` to verify the expected results against the actual results obtained from the `CodigoPessoaFisica` class.

JUnit X

Finished after 0,03 seconds

Runs: 3/3 Errors: 0 Failures: 0

CPFTeste [Runner: JUnit 4] (0,002 s)

- testeRemoveCaracteresEspeciais (0,000 s)
- testagemPositivaCPF (0,001 s)
- testagemNegativaCPF (0,000 s)

Failure Trace

```
1 import static org.junit.Assert.assertEquals;
2 import org.junit.Test;
3
4 public class CPFTeste {
5
6     @Test
7     public void testagemPositivaCPF() {
8         CodigoPessoaFisica cpf = new CodigoPessoaFisica();
9         String cpfTeste = "461.670.818-59";
10        boolean resultadoEsperado = true;
11        boolean resultadoObtido = cpf.validaCPF(cpfTeste);
12        assertEquals(resultadoEsperado, resultadoObtido);
13    }
14
15     @Test
16     public void testagemNegativaCPF() {
17         CodigoPessoaFisica cpf = new CodigoPessoaFisica();
18         String cpfTeste = "176.942.776-12";
19        boolean resultadoEsperado = false;
20        boolean resultadoObtido = cpf.validaCPF(cpfTeste);
21        assertEquals(resultadoEsperado, resultadoObtido);
22    }
23
24     @Test
25     public void testeRemoveCaracteresEspeciais() {
26         CodigoPessoaFisica cpf = new CodigoPessoaFisica();
27         String cpfTeste = "461.670.818-59";
28         String resultadoEsperado = "46167081859";
29         String resultadoObtido = cpf.removeCaracteresEspeciais(cpfTeste);
30         assertEquals(resultadoEsperado, resultadoObtido);
31    }
32 }
33
```