

**UNIVERSIDADE FEDERAL DO CEARÁ****Departamento de Computação**

Prof. Lincoln Souza Rocha

CK0442- Técnicas de Programação para Ciência de Dados - T01

TP05
2024.2

1. Implementando Testes Unitários Automatizados para o SisBanco

Descrição: Considerando o módulo `sisbanco.py` disponível no ArquivosTP05.zip no SIGAA/SI3, implemente os testes unitários para cada uma das classes do sistema utilizando o framework `unittest`¹. Os testes unitários devem ser implementados no arquivo `sisbanco_testes.py`. Ao final, execute todos testes e verifique se todos eles estão passando. (**Dica.** Use o padrão `from sisbanco import ContaAnstrata, ..., Banco` para importar as classes de `sisbanco.py` que serão testadas pelos testes unitários escritos no módulo `sisbanco_testes.py`).

2. Identificando Bugs com Testes Unitários

Descrição: Considerando o módulo `sisbanco_bugs.py` disponível no ArquivosTP05.zip no SIGAA/SI3, altere a importação das classes no arquivo `sisbanco_testes.py` para: `from sisbanco_bugs import ContaAnstrata, ..., Banco`. Em seguida, execute todos os seus testes. Quantos deles falharam? Indique as possíveis causas das falhas, mas não as corrija, faremos isso na próxima tarefa.

3. Escrevendo Testes de Efetivos

Descrição: Foram injetados 12 bugs no módulo `sisbanco.py` para gerar o módulo `sisbanco_bugs.py`. Esses bugs foram criados por meio de alterações sintáticas no código fonte do módulo `sisbanco.py`. Essas alterações sintáticas estão descritas no arquivo `bugs.txt`, disponível no arquivo ArquivosTP05.zip no SIGAA/SI3. Você deve criar testes adicionais para detectar a presença de todos os 12 bugs criados.

¹<https://docs.python.org/pt-br/3/library/unittest.html>