

	INSTITUTO FEDERAL DO RN Campus Natal-Central	
	Disciplina: Teste de Software	
	Professor(a): Plácido A. Souza Neto	
	Discente:	Matrícula:
	Curso: TADS	Semestre: 2024.1
	Revisão – Prova 01	

1. Imagine um sistema bancário simples em Python, onde cada agência possui diversas contas. As contas possuem um saldo e podem realizar operações de depósito e saque. Crie uma classe *Conta* em Python para representar uma conta bancária. Implemente os seguintes métodos:

- **init(self, numero, agencia, saldo):** Construtor da classe, inicializando o número da conta, agência e saldo inicial.
- **depositar(self, valor):** Adiciona um valor ao saldo da conta.
- **sacar(self, valor):** Retira um valor do saldo da conta. Lança uma exceção `SaldoInsuficienteException` caso o valor a ser sacado seja maior que o saldo.
- **saldo(self):** Método que retorna o saldo da conta.

Em seguida, utilizando TDD, escreva testes unitários para cobrir os seguintes cenários:

- Criação de conta: Verificar se a conta é criada com os valores corretos.
- Depósito: Verificar se o saldo da conta é atualizado corretamente após um depósito.
- Saque: Verificar se o saldo é atualizado corretamente após um saque e se a exceção `SaldoInsuficienteException` é lançada quando o saldo é insuficiente.
- Saldo negativo: Verificar se o saldo não pode ser negativo (lançar uma exceção adequada).

Considerações adicionais:

- TDD: Aplique o ciclo TDD (Test-Driven Development) para desenvolver os testes antes do código da classe *Conta*.
- Exceções: Defina uma classe `SaldoInsuficienteException` para indicar quando o saldo é insuficiente.
- Testes claros e concisos: Utilize nomes de testes que reflitam o comportamento sendo testado.
- Cobertura de código: Certifique-se de que seus testes cobrem todas as possíveis entradas e saídas da classe *Conta*.

OBS: Cada membro da dupla deve implementar 2 casos de teste e 2 operações (uma vez que há 4 casos de testes possíveis). Quando 1 membro implementa um caso de teste, o outro implementa a função, seguindo a metodologia do TDD. Essa abordagem será verificada a partir dos commits no repositório. Assim, é sempre necessário realizar primeiro o commit com o teste (por um dos membros), e depois o commit da implementação (por outro membro)