

Docente: Cássio David Borralho Pinheiro
Discente: João Pedro Braga de Carvalho
Curso: Sistema de Informação - Bacharelado
Disciplina: Laboratório de Programação

2ª Atividade Extra

1) Programação Funcional:

Justificativa: A programação funcional é um paradigma de grande interesse devido às suas propriedades teóricas e práticas, como modularidade, testabilidade e componibilidade. Além disso, ela enfatiza a imutabilidade e funções puras, promovendo uma abordagem mais segura e fácil de entender para o desenvolvimento de software.

Questão: Quais são os benefícios teóricos e práticos da programação funcional em relação à modularidade, testabilidade e componibilidade do código?

Resposta: A programação funcional promove a quebra do problema em pequenos pedaços, tornando o código mais modular e fácil de entender. Além disso, suas funções puras facilitam a testabilidade, pois cada função é um alvo potencial para testes unitários, e a componibilidade é facilitada pela reutilização de funções em diferentes contextos, resultando em uma biblioteca de utilitários personalizada.

2) Evitação de efeitos colaterais:

Justificativa: A evitação de efeitos colaterais é um conceito fundamental na programação funcional, contribuindo para a segurança e previsibilidade do código. Ao evitar a modificação do estado interno e enfatizar funções puras, a programação funcional promove uma abordagem mais robusta e fácil de depurar.

Questão: Como a evitação de efeitos colaterais na programação funcional contribui para a segurança e previsibilidade do código?

Resposta: Evitar efeitos colaterais significa que as funções não modificam o estado interno ou produzem alterações não visíveis no valor de retorno. Isso torna o código mais seguro, pois reduz a possibilidade de comportamentos inesperados, e mais previsível, já que o resultado de uma função depende apenas de suas entradas.

3) Modularidade e Componibilidade:

Justificativa: A modularidade e componibilidade são vantagens práticas da programação funcional, facilitando a organização e reutilização de código em diferentes contextos. Ao escrever pequenas funções especializadas e formar uma biblioteca de utilitários, os programadores podem construir novos programas combinando e adaptando funcionalidades existentes.

Questão: Como a modularidade e componibilidade na programação funcional contribuem para a reutilização e extensibilidade do código?

Resposta: A modularidade na programação funcional permite que os programadores escrevam pequenas funções especializadas, facilitando a reutilização em diferentes contextos. Além disso, a componibilidade permite a construção de novos programas combinando e adaptando funcionalidades existentes, resultando em um código mais flexível, reutilizável e extensível.