

FICHA2 - PLANO DE ENSINO

CÓDIGO: TE353	DISCIPLINA: PROGRAMAÇÃO ORIENTADA A OBJETO		TURMA: DA			
NATUREZA: Obrigatória ou Optativa		REGIME: Semestral	MODALIDADE: Presencial			
CH TOTAL: 60h		CH SEMANAL: 0h	CH Prática como Componente Curricular (PCC): 0h		CH Atividade Curricular de Extensão (ACE): 0h	
Padrão (PD): 60h	Laboratório (LB): 0h	Campo (CP): 0h	Orientada (OR): 0h	Estágio (ES): 0h	Prática Específica (PE): 0h	Estágio de Formação Pedagógica (EFP): 0h
FICHA 2 PREENCHIDA PELO DOCENTE: EDSON JOSÉ PACHECO						

EMENTA

Encapsulamento; Herança e Hierarquia; Composição e Derivação; Construtores; Polimorfismos; Modelos.

PROGRAMA

1. Fundamentos da Programação Orientada a Objetos

- Definição de OOP e seus princípios básicos
- Conceitos de classes, objetos, herança, polimorfismo e encapsulamento
- Comparação entre OOP e paradigmas de programação estruturada

2. Classes e Objetos

- Definição e uso de classes e objetos
- Atributos e métodos de classe e de instância
- Construtores e destrutores
- Modificadores de acesso (public, private, protected)

3. Herança e Polimorfismo

- Hierarquia de classes e herança
- Herança simples, múltipla e interfaces
- Polimorfismo estático e dinâmico
- Classes abstratas e interfaces

4. Encapsulamento e Modificadores de Acesso

- Encapsulamento e ocultação de informações
- Getters e setters



- Modificadores de acesso (public, private, protected)

5. Relacionamentos entre Classes

- Associação, agregação e composição
- Relacionamentos um-para-um, um-para-muitos e muitos-para-muitos
- Navegação e acesso a objetos relacionados

6. Tratamento de Exceções

- Conceito de exceções e erros
- Lançamento e captura de exceções
- Blocos try-catch-finally
- Criação de exceções personalizadas

7. Gerenciamento de Memória e Coleta de Lixo

- Alocação e desalocação de memória em OOP
- Coleta de lixo e garbage collectors
- Ciclo de vida de objetos e referências

OBJETIVO GERAL

O objetivo geral desta disciplina é proporcionar aos alunos uma compreensão sólida e abrangente dos conceitos, princípios e práticas da Programação Orientada a Objetos (OOP), capacitando-os a aplicar esses conhecimentos na análise, projeto e implementação de software de qualidade.

OBJETIVOS ESPECÍFICOS

1. Compreender os princípios fundamentais da OOP, incluindo classes, objetos, herança, polimorfismo e encapsulamento, e reconhecer as vantagens e benefícios dessa abordagem em relação a paradigmas de programação alternativos.
2. Projetar e implementar classes e objetos de forma eficiente, utilizando os conceitos de encapsulamento, modificadores de acesso e técnicas adequadas de modelagem e design.
3. Utilizar herança e polimorfismo de maneira eficaz para criar hierarquias de classes flexíveis e reutilizáveis, permitindo o desenvolvimento de software extensível e de fácil manutenção.
4. Aplicar os princípios de relacionamentos entre classes, como associação, agregação e composição, para modelar e implementar interações entre objetos e componentes de software.



5. Identificar e tratar exceções de forma adequada, implementando mecanismos de manipulação de erros e garantindo a robustez e a confiabilidade do software.

6. Compreender os conceitos de gerenciamento de memória em OOP, incluindo alocação e desalocação de memória, ciclo de vida de objetos e coleta de lixo, visando à eficiência e ao uso otimizado dos recursos do sistema.

PROCEDIMENTOS DIDÁTICOS

A disciplina será desenvolvida mediante aulas expositivas e atividades de laboratório, conforme o conteúdo curricular previsto pelo programa de ensino.

FORMAS DE AVALIACAO

1 prova (P), compondo 50% da nota da disciplina + atividades propostas, com teste de autoria, todas com mesmo peso, compondo os 50% restantes da nota da disciplina.

BIBLIOGRAFIA BÁSICA

Vide Ficha 01 - https://www.eletrica.ufpr.br/p/_media/graduacao:documentos:ppc-2018-diurno.rar

BIBLIOGRAFIA COMPLEMENTAR

Vide Ficha 01 - https://www.eletrica.ufpr.br/p/_media/graduacao:documentos:ppc-2018-diurno.rar

