

DIRETORIA DE ENSINO/DEPARTAMENTO DE ENSINO
COORDENAÇÃO DO CURSO TÉCNICO INTEGRADO EM INFORMÁTICA
PROGRAMA DE UNIDADE DIDÁTICA – PUD

DISCIPLINA: PROGRAMAÇÃO ORIENTADA A OBJETOS		
Código: 31.101	Carga horária total: 80h	Créditos: 04
Nível: Médio-Técnico	Ano: 2º	Pré-requisitos: Não
CARGA HORÁRIA	Teórica: 40h	Prática: 40h
	Presencial: Sim	Distância: Não
	Prática Profissional: Não se aplica.	
	Atividades não presenciais: Não se aplica.	
	Extensão: Não se aplica.	
EMENTA		
Fundamentos da Programação Orientada a Objetos: classes, objetos e pilares fundamentais (abstração, encapsulamento, herança e polimorfismo). Estrutura, relacionamentos entre classes, sobrecarga e sobrescrita de métodos. Estruturação de dados orientada a objetos (listas, filas, pilhas e árvores). Utilização de classes abstratas e interfaces. Construção de aplicações orientadas a objetos em linguagem de programação de alto nível.		
OBJETIVO		
<ul style="list-style-type: none"> -Compreender os princípios teóricos e práticos da programação orientada a objetos, incluindo seus pilares fundamentais. -Desenvolver aplicações utilizando uma linguagem de programação orientada a objetos. -Empregar estruturas de dados em um contexto orientado a objetos. -Analisar problemas computacionais e modelá-los por meio de classes, objetos, abstrações e relações entre componentes. -Relacionar os conceitos estudados com outras disciplinas da área de computação. 		
PROGRAMA		
<p>UNIDADE 1 – Conceitos Fundamentais da Programação Orientada a Objetos</p> <p>1.1 Conceito de objeto e classe</p> <p>1.2 Atributos e métodos</p> <p>1.3 Abstração</p> <p>1.4 Encapsulamento</p> <p>1.5 Estrutura todo–parte (composição e agregação)</p> <p>UNIDADE 2 – Estrutura e Elementos de uma Linguagem Orientada a Objetos</p> <p>2.1 Palavras reservadas</p> <p>2.2 Constantes e variáveis</p> <p>2.3 Modificadores de acesso (public, private, protected)</p> <p>2.4 Membros estáticos</p> <p>2.5 Construtores e destrutores</p> <p>UNIDADE 3 – Herança e Polimorfismo</p> <p>3.1 Conceito de herança</p> <p>3.2 Tipos de herança</p> <p>3.3 Sobrecarga de métodos</p> <p>3.4 Sobrescrita de métodos</p> <p>3.5 Polimorfismo</p> <p>UNIDADE 4 – Classes Abstratas, Interfaces e Estruturas de Dados OO</p> <p>4.1 Conceito de classes abstratas</p>		

- 4.2 Conceito de interfaces (no contexto de OO)
- 4.3 Listas (estáticas e dinâmicas)
- 4.4 Pilhas e filas
- 4.5 Árvores

METODOLOGIA DE ENSINO

Exposição dialogada dos conteúdos teóricos para introdução dos conceitos fundamentais da Programação Orientada a Objetos.

Desenvolvimento de atividades práticas em laboratório de informática, utilizando computadores e ferramentas de programação.

Utilização de projetor multimídia para apresentação de conteúdos e exemplos de código.

Uso do quadro branco para explicações, esquemas de classes e modelagens.

Resolução de exercícios práticos que integrem teoria e aplicação.

Realização de atividades colaborativas, como trabalhos em grupo e seminários.

RECURSOS

Laboratório de Informática;
Projetor Multimídia;
Computador e/ou Notebook;
Lousa e Pincel.

AVALIAÇÃO

A avaliação ocorrerá segundo caráter contínuo, diagnóstico e cumulativo, acompanhando o desenvolvimento do estudante ao longo do semestre.

- As avaliações serão distribuídas em, no mínimo, quatro notas (N1, N2, N3 e N4), utilizando instrumentos como provas escritas e orais, relatórios, trabalhos individuais e em grupo, seminários e exercícios práticos.
- Serão realizadas aferições periódicas para acompanhamento da aprendizagem.
- Assiduidade e participação comporão parte da avaliação.
- Estudantes que não atingirem os resultados esperados participarão de recuperação paralela, por meio de grupos de estudo, monitoria e/ou atividades específicas.
- Após a recuperação, o estudante poderá realizar prova e/ou lista de exercícios para nova avaliação.

BIBLIOGRAFIA BÁSICA

1. SAUDATE, Alexandre. **APIs REST: seus serviços prontos para o mundo real**. São Paulo, SP: Casa do Código, 2021. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 10 out 2025.
2. OSÓRIO, Victor. **Roadmap back-end: conhecendo o protocolo HTTP e arquiteturas REST**. São Paulo, SP: Casa do Código, 2022. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 10 out 2025.
3. PEREIRA, Caio Ribeiro. **Construindo APIs REST com Node.js**. São Paulo, SP: Casa do Código, 2016. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 10 out 2025.

BIBLIOGRAFIA COMPLEMENTAR

1. VILARINHO, Leonardo. **Front-end com Vue.js: da teoria à prática sem complicações**. São Paulo, SP: Casa do Código, 2021. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 10 out 2025.
2. ESCUDELARIO, Bruna; PINHO, Diego. **React Native: desenvolvimento de aplicativos mobile com React**. São Paulo, SP: Casa do Código, 2020. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 10 out 2025.
3. TURINI, Rodrigo. **PHP e Laravel: crie aplicações web como um verdadeiro artesão**. São Paulo, SP: Casa do Código, 2015. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 10 out 2025.
4. BENTO, Evaldo Junior. **Desenvolvimento web com PHP e MySQL**. São Paulo, SP: Casa do Código, 2013. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 10 out 2025.

5. CARVALHO, Vinícius. **MySQL**: comece com o principal banco de dados open source do mercado. São Paulo, SP: Casa do Código, 2015. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 10 out 2025.