



MINISTÉRIO DA EDUCAÇÃO  
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA  
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

PLANO DE DISCIPLINA			
IDENTIFICAÇÃO			
CAMPUS: Cajazeiras			
CURSO: Análise e Desenvolvimento de Sistemas			
DISCIPLINA: Programação Orientada a Objetos		CÓDIGO DA DISCIPLINA: TEC.0428	
PRÉ-REQUISITO: Estruturas de Dados			
UNIDADE CURRICULAR: Obrigatória [X] Optativa [ ] Eletiva [ ]		SEMESTRE/ANO: 2025.1	
CARGA HORÁRIA			
TEÓRICA: 50 h	PRÁTICA: 50 h	EaD¹: -	EXTENSÃO: -
CARGA HORÁRIA SEMANAL: 6 h/a			
CARGA HORÁRIA TOTAL: 100 h			
DOCENTE RESPONSÁVEL: Francisco Paulo de Freitas Neto			

EMENTA
--------

Fundamentos do paradigma de programação orientada a objetos. Linguagem de programação orientada a objetos: estrutura básica do programa, variáveis e tipos, expressões lógicas e aritméticas, manipulação de cadeias de caracteres, estruturas de decisão e de repetição, estruturas de dados, entrada e saída de dados. Orientação à objetos: conceitos de classes e instâncias, troca de mensagens entre objetos, encapsulamento, agregação, herança, sobrecarga e sobrescrita, polimorfismo. Tópicos avançados em orientação a objetos.

OBJETIVOS DA DISCIPLINA/COMPONENTE CURRICULAR (Geral e Específicos)
--

Objetivo Geral:

Capacitar os alunos a desenvolverem soluções de software utilizando o paradigma de programação orientada a objetos, com domínio das principais técnicas e conceitos, como classes, objetos, encapsulamento, herança, polimorfismo e estruturas de dados, para a construção de sistemas eficientes, reutilizáveis e de fácil manutenção.

Objetivos Específicos:

- Compreender os conceitos fundamentais do paradigma de programação orientada a objetos;
- Desenvolver habilidades na manipulação de variáveis, tipos de dados e estruturas de controle;
- Ensinar a construção e manipulação de programas em uma linguagem orientada a objetos, abordando a estrutura básica do código, boas práticas e a utilização de funções e métodos no contexto da POO
- Capacitar os alunos a implementar e utilizar estruturas de dados essenciais, como listas, pilhas, filas e outras, dentro do paradigma de objetos, visando o armazenamento e manipulação eficiente de dados
- Ensinar o conceito e a prática de criação e manipulação de classes e objetos, incluindo a compreensão da relação entre classes, instâncias e a troca de mensagens entre objetos.

- Aplicar os conceitos de encapsulamento, agregação, herança e polimorfismo, utilizando-os para criar sistemas modulares, reutilizáveis e flexíveis, com foco em eficiência e escalabilidade.
- Estimular a prática contínua e a resolução de problemas reais, por meio de projetos práticos, exercícios e estudos de caso, promovendo a aplicação dos conceitos aprendidos em contextos do mundo real.

## CONTEÚDO PROGRAMÁTICO

### 1. Introdução à Programação Orientada a Objetos

- Histórico e evolução do paradigma orientado a objetos
- Comparação entre paradigmas de programação: estruturada vs. orientada a objetos
- Características principais do paradigma de programação orientada a objetos
- Abstração, encapsulamento, herança e polimorfismo

### 2. Estrutura Básica de um Programa Orientado a Objetos

- Sintaxe e organização de código em linguagens orientadas a objetos
- Definição de classes e instâncias
- Métodos e atributos: definição e utilização
- Modificadores de acesso (public, private, protected, package)

### 3. Tipos de Dados e Variáveis em Programação Orientada a Objetos

- Tipos primitivos e compostos
- Tipos de dados em linguagens orientadas a objetos
- Variáveis de instância e de classe
- Inicialização e declaração de variáveis em objetos

### 4. Expressões Lógicas e Aritméticas

- Operadores aritméticos e lógicos
- Manipulação de expressões para controlar o fluxo de execução
- Precedência e associatividade de operadores

### 5. Estruturas de Controle de Fluxo

- Estruturas de decisão: if, else, switch
- Estruturas de repetição: for, while, do-while
- Controle de fluxo com break, continue e return

### 6. Manipulação de Cadeias de Caracteres

- Definição e manipulação de strings
- Funções para manipulação de strings em POO
- Concatenamento, busca e modificação de strings

### 7. Estruturas de Dados em Programação Orientada a Objetos

- Introdução às estruturas de dados: listas, pilhas, filas, árvores, etc.
- Implementação de estruturas de dados utilizando objetos
- Algoritmos de manipulação de dados (inserção, remoção, busca)

### 8. Agregação e Composição

- Diferença entre agregação e composição
- Implementação de relações entre objetos
- Exemplos práticos de agregação e composição em sistemas reais

### 9. Herança

- Definição e propósito da herança
- Herança simples e múltipla
- Sobrescrita de métodos (method overriding)
- Relação entre classes pai e classe filha

### 10. Polimorfismo

- Definição e aplicação do polimorfismo
- Polimorfismo de método (overloading e overriding)
- Polimorfismo em tempo de compilação e em tempo de execução
- Interfaces e classes abstratas

## 11. Tópicos Avançados em Orientação a Objetos

- **Design Patterns:** conceitos e tipos mais comuns (Singleton, Factory, Observer, etc.)
- **Interfaces:** definição e uso de interfaces em POO
- **Classes Abstratas:** diferenças em relação a interfaces e herança
- **Exceções e Tratamento de Erros:** mecanismos de captura e lançamento de exceções
- **Reflexão e Anotações:** introspecção de objetos em tempo de execução

## 12. Prática de Programação e Projetos

- Desenvolvimento de projetos práticos utilizando os conceitos aprendidos
- Aplicação de POO na solução de problemas reais
- Estudo de caso: análise e desenvolvimento de sistemas com POO
- Apresentação de projetos e revisão de código

### METODOLOGIA DE ENSINO

Aulas Expositivas e Teóricas; Exercícios Práticos e Laboratórios; Projetos de Desenvolvimento; Aulas Práticas com Estudos de Caso;

### RECURSOS DIDÁTICOS

[X] Quadro  
[X] Projetor  
[ ] Vídeos/DVDs  
[X] Periódicos/Livros/Revistas/Links  
[ ] Equipamento de Som  
[X] Laboratório  
[X] Softwares<sup>2</sup>: Java, IntelliJ, Github Classroom, Google Classroom  
[ ] Outros<sup>3</sup>

### CRITÉRIOS DE AVALIAÇÃO

*(Especificar quantas avaliações e formas de avaliação – avaliação escrita objetivo, subjetiva, trabalho, seminário, artigo, etc. - para integralização da disciplina/componente curricular, incluindo a atividade de recuperação final.)*

Serão realizadas três avaliações:

- Listas de exercícios - LE (100 pontos) - Peso 2
- Prova - PR (100 pontos) - Peso 4
- Projeto Final - PF (100 pontos) - Peso 4

A nota na disciplina corresponderá à média ponderada das notas:

Nota =  $(2*LE + 4*PR + 4*PF)/10$

### ATIVIDADE DE EXTENSÃO<sup>4</sup>

Não há

### BIBLIOGRAFIA<sup>5</sup>

Bibliografia Básica:

SIERRA K.; BATES, B. Use a Cabeça! Java. 2. ed. Rio de Janeiro: Alta Books, 2007.

DEITEL, H. M.; DEITEL, P. J. Java: como programar. 10. ed. São Paulo: Pearson Education, 2017.

FURGERI, S. Java Ensino Didático: Desenvolvimento e Implementação de Aplicações. São Paulo: Érica, 2018.

Bibliografia Complementar:

HORSTMANN, C. S. & CORNELL, G. Core Java, Volume 1. Pearson, 8ª edição, 2010;

CADENHEAD, R.; LEMAY, L. Aprenda Java em 21 Dias. Campus, 4ª edição, 2005.

BARNES, D. J.; KÖLLING, M. Programação Orientada a Objetos com Java: Uma Introdução Prática Usando o BlueJ. 4. ed. Porto Alegre: Pearson, 2009.

BLOCH, J. Java Efetivo. 3. ed. Rio de Janeiro: Alta Books, 2019.

JUNIOR, P. J. Java Guia do Programador. 4. ed. São Paulo: Novatec, 2021.

SCHILDT H. Java a Referência Completa. Rio de Janeiro: Alta Books, 2020. SCHILDT H. Java para Iniciantes. 6. ed. Porto Alegre: Bookman, 2015.

### OBSERVAÇÕES

*(Acrescentar informais complementares ou explicativas caso o docente(s) considere importantes para a disciplina/componente curricular)*

- 1 Para a oferta de disciplinas na modalidade à distância, integral ou parcial, desde que não ultrapassem os limites definidos em legislação.
- 2 Nesse item o professor deve especificar quais softwares serão trabalhados em sala de aula.
- 3 Nesse item o professor pode especificar outras formas de recursos utilizadas que não estejam citada.
- 4 Nesse item deve ser detalhado o PROJETO e/ou PROGRAMA DE EXTENSÃO que será executado na disciplina. Observando as orientações do Art. 10, Incisos I, II, III, IV, V, VI, VII e VIII, da Instrução Normativa que trata da construção do **Plano de Disciplina**.
- 5 Observar os mínimos de 3 (três) títulos para a bibliografia básica e 5 (cinco) para a bibliografia complementar.

Documento assinado eletronicamente por:

■ **Francisco Paulo de Freitas Neto**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 26/03/2025 16:48:24.

Este documento foi emitido pelo SUAP em 26/03/2025. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código 689108  
Verificador: b92c51bfb6  
Código de Autenticação:



Rua José Antônio da Silva, 300, Jardim Oásis, CAJAZEIRAS / PB, CEP 58.900-000  
<http://ifpb.edu.br> - (83) 3532-4100