



## PLANO DE ENSINO

### 1. IDENTIFICAÇÃO

Componente Curricular:	PROGRAMAÇÃO ORIENTADA A OBJETOS						
Unidade Ofertante:	FACULDADE DE COMPUTAÇÃO						
Código:	FACOM39202	Período/Série:	2º PERÍODO		Turma:	U	
Carga Horária:				Natureza:			
Teórica:	30	Prática:	30	Total:	60	Obrigatória( ):	Optativa( )
Professor(A):	Eliana Pantaleão				Ano/Semestre:	2023/2	
Observações:							

### 2. EMENTA

Métodos e técnicas de programação sob o paradigma de programação da Orientação a Objetos.

### 3. JUSTIFICATIVA

O estudo de programação faz com que o estudante desenvolva seu raciocínio lógico e dedutivo e aumente sua capacidade de abstração. A orientação a objetos é o paradigma mais difundido atualmente e favorece o aprendizado de boas técnicas de programação, como o reaproveitamento de código, e questões de segurança, devido aos princípios de encapsulamento e herança. O estudo de arquivos, bancos de dados e interface gráfica com o usuário permite que o estudante desenvolva, em outras disciplinas ou em seu projeto final, aplicativos com boa qualidade. A máquina virtual Java está presente em diversos dispositivos eletrônicos da atualidade, de modo que o aprendizado da linguagem Java fornece uma base para que o estudante desenvolva inclusive sistemas embarcados, ou para dispositivos móveis, especialmente baseados em Android.

### 4. OBJETIVO

Ao final do curso o estudante deverá ser capaz de:

1. Identificar conceitos e características importantes relativos ao paradigma de programação da Orientação a Objetos;
2. Demonstrar capacidade de dedução, raciocínio lógico e abstração; e
3. Implementar aplicações de engenharia utilizando uma linguagem orientada a objetos.

Entre as competências a serem desenvolvidas no estudante destacam-se:

1. Ser capaz de modelar os fenômenos, os sistemas físicos e químicos, utilizando as ferramentas matemáticas, estatísticas, computacionais e de simulação,

entre outras;

2. Prever os resultados dos sistemas por meio dos modelos;
3. Conceber experimentos que gerem resultados reais para o comportamento dos fenômenos e sistemas em estudo;
4. Verificar e validar os modelos por meio de técnicas adequadas;
5. Ser capaz de conceber e projetar soluções criativas, desejáveis e viáveis, técnica e economicamente, nos contextos em que serão aplicadas;
6. Projetar e determinar os parâmetros construtivos e operacionais para as soluções de Engenharia;
7. Ser capaz de expressar-se adequadamente, seja na língua pátria ou em idioma diferente do Português, inclusive por meio do uso consistente das tecnologias digitais de informação e comunicação (TDICs), mantendo-se sempre atualizado em termos de métodos e tecnologias disponíveis;
8. Ser capaz de interagir com as diferentes culturas, mediante o trabalho em equipes presenciais ou a distância, de modo que facilite a construção coletiva;
9. Gerenciar projetos e liderar, de forma proativa e colaborativa, definindo as estratégias e construindo o consenso nos grupos;
10. Aprender a aprender.

## **5. PROGRAMA**

### **1. Introdução a Programação Orientada a Objetos**

1.1 Conceitos e princípios básicos: abstração de objetos e representação de classes, relacionamentos de herança, troca de mensagens, encapsulamento, polimorfismo

1.2 Introdução à Modelagem Lógica de Classes com UML

1.3 Linguagens de programação Orientadas a Objetos

1.4 Genealogia das linguagens de programação

1.5 Exemplos de linguagens orientadas a objetos

### **2. A Linguagem Java**

2.1 Classes em Java (atributos, métodos, modificadores de acesso)

2.2 Herança em Java

2.3 Polimorfismo em Java (polimorfismo, classes abstratas, interface, métodos estáticos)

### **3. Interfaces Gráficas com o Usuário (GUI)**

3.1 Implementação de GUI com bibliotecas gráficas do Java

### **4. Manipulação de arquivos em Java**

4.1 Tratamento de Exceções

4.2 Leitura e gravação de arquivo-texto

4.3 Serialização de objetos em Java com arquivos binários

### **5. SGBDs (Sistema de Gerenciamento de Banco de Dados)**

5.1 Introdução aos SGBDs

5.2 Acesso ao BD utilizando a linguagem Java

5.3 Linguagem SQL

## 6. METODOLOGIA

### • Conteúdo Programático para Atividades Teóricas Presenciais

As aulas teóricas serão realizadas às quartas-feiras das 8h50min às 10h40min no Laboratório de Informática Prata, Unipam, bloco G, sala 103.

<b>Aula</b>	<b>Data</b>	<b>Conteúdo Teórico</b>
1-2	10/01	Apresentação da disciplina. Conceitos básicos de orientação a objetos.
3-4	17/01	Classes e objetos.
5-6	24/01	Herança.
7-8	31/01	Desenvolvimento em camadas.
9-10	07/02	Avaliação.
11-12	21/02	Interfaces com o usuário.
13-14	28/02	Polimorfismo.
15-16	06/03	Classes abstratas.
17-18	13/03	Tratamento de exceções. Arquivo texto.
19-20	20/03	Avaliação.
21-22	27/03	Introdução aos SGBDs.
23-24	03/04	Linguagem SQL.
25-26	10/04	Linguagem SQL.
27-28	17/04	Avaliação.
29-30	24/04	Apresentação do trabalho final. Avaliação de recuperação de aprendizagem.

### • Conteúdo Programático para Atividades Práticas Presenciais

As aulas práticas serão realizadas às quartas-feiras das 10h40min às 12h20min no Laboratório de Informática Prata, Unipam, bloco G, sala 103.

<b>Aula</b>	<b>Data</b>	<b>Conteúdo Prático</b>
1-2	10/01	Apresentação do ambiente de programação.
3-4	17/01	Implementação usando classes e objetos.

5-6	24/01	Implementação usando herança.
7-8	31/01	Implementação usando camadas.
9-10	07/02	Avaliação.
11-12	21/02	Construção de interfaces com o usuário.
13-14	28/02	Implementação usando polimorfismo.
15-16	06/03	Implementação usando classes abstratas.
17-18	13/03	Leitura e escrita de arquivo texto.
19-20	20/03	Avaliação.
21-22	27/03	Projeto de SGBD.
23-24	03/04	Implementação usando Linguagem SQL.
25-26	10/04	Implementação usando Linguagem SQL.
27-28	17/04	Avaliação.
29-30	24/04	Apresentação do trabalho final. Avaliação de recuperação de aprendizagem.

• **Conteúdo Programático para Atividades Acadêmicas Extras (AAE)**

<b>Aula</b>	<b>Data</b>	<b>Conteúdo - Atividades Acadêmicas Extras (AAE)</b>
1-2	12/01	Prática: instalação e configuração do ambiente de programação.
3-4	02/02	Prática: desenvolvimento de um sistema acadêmico.
5-6	08/03	Prática: leitura e escrita de arquivo binário.
7-8	15/03	Teórica: diferentes formas de interface com o usuário.
9-10	05/04	Teórica: armazenamento de dados em nuvem.
11-12	19/04	Teórica: normalização de banco de dados.

	<b>Teórica</b>	<b>Prática</b>
<b>C.H Presencial Total</b>	30	30
<b>C.H. Atividades Acadêmicas Extras Total</b>	6	6

<b>C.H. Total da disciplina</b>	36	36
---------------------------------	----	----

- **Atendimento**

O atendimento aos alunos da disciplina será realizado de forma presencial no Unipam, bloco Alfa, terceiro andar, sala 8, de acordo com o seguinte planejamento: terças-feiras entre 14h50min e 16h30min, ou outro dia desde que agendado com a professora previamente.

## 7. AVALIAÇÃO

- **Aproveitamento**

A avaliação de desempenho dos discentes será feita por apresentação e entrega de atividades, três provas e um trabalho final. O cronograma de atividades avaliativas e a distribuição da pontuação é apresentada a seguir.

Os resultados das avaliações serão divulgados no mural do curso, sendo que as notas serão apresentadas pelos números de matrícula dos alunos. A divulgação das notas deve acontecer em até 15 dias úteis após a sua realização e a vista de prova será marcada com os alunos, a partir da data de divulgação das notas, respeitando-se o prazo de no máximo 5 dias úteis, como previsto na Resolução do CONGRAD (Nº46/2022).

<b>DATA</b>	<b>ATIVIDADE AVALIATIVA</b>	<b>PONTUAÇÃO</b>
07/02	Prova 1	20 pontos
20/03	Prova 2	20 pontos
17/04	Prova 3	20 pontos
-	Apresentação e entrega de atividades ao longo do período.	20 pontos
24/04	Trabalho Final	20 pontos
TOTAL		100 pontos

- **Frequência**

A frequência para aulas presenciais será aferida por chamada oral durante as aulas. Já as horas vinculadas à parte assíncrona serão aferidas com a entrega dos trabalhos.

- **Recuperação**

A prova de recuperação somente será aplicada àquele estudante que não obtiver o rendimento mínimo para aprovação e com frequência mínima de 75% no

componente curricular, conforme Resolução do CONGRAD (Nº46/2022). Esta prova substituirá a menor nota obtida entre as provas 1, 2 ou 3, abrangendo o conteúdo total da disciplina.

## 8. BIBLIOGRAFIA

### Básica

1. CORNELL, G.; HORSTMANN, C. S. **Core Java**: fundamentos. 8.ed. São Paulo: Prentice Hall, 2010.
2. DEITEL, H. M.; DEITEL P. J. **Java**: como programar. Porto Alegre: Prentice Hall, 2010.
3. ZIVIANI, N. **Projeto de algoritmos com implementações em Java e C++**. São Paulo: Cengage Learning, 2007.

### Complementar

1. ANSELMO, F. **Aplicando Lógica Orientada a Objetos em Java**. Florianópolis: Visual Books, 2005.
2. BERZTISS, A. T. **Data Structures: Theory and Practice**. 2nd ed. New York: Academic Press, 1975.
3. HEUSER, C. A. **Projeto de banco de dados**. 4. ed. Porto Alegre: Bookman, 2008.
4. SEDGEWICK, Robert. **Algorithms in Java**. 3rd ed. Boston: Addison-Wesley, 2002.
5. SIERRA, K. **Use a cabeça!**: Java. 2.ed. Rio de Janeiro: Alta Books, 2005.
6. SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. **Sistema de banco de dados**. 6ed. Rio de Janeiro: Elsevier, 2012.

## 9. APROVAÇÃO

Aprovado em reunião do Colegiado conforme Decisão Administrativa do Colegiado anexada ao processo referenciado.

Coordenação do Curso de Graduação: \_\_\_\_\_



Documento assinado eletronicamente por **Eliana Pantaleão, Professor(a) do Magistério Superior**, em 02/02/2024, às 09:23, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Daniel Costa Ramos, Coordenador(a)**, em 15/02/2024, às 08:54, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site [https://www.sei.ufu.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **4943139** e o código CRC **65B8ACCO**.