



DISCIPLINA:	PROGRAMAÇÃO ORIENTADA A OBJETOS	
Código:	POO	
Carga Horária Total: 80	CH Teórica: 40	CH Prática: 40h
Número de Créditos:	4	
Pré-requisitos:	-	
Semestre:	2º	
Nível:	Técnico	
EMENTA		
Introdução à POO, classes e objetos, relacionamento entre classes, tratamento de exceções.		
OBJETIVO		
Propiciar ao discente as ferramentas necessárias para o desenvolvimento de projetos aplicando os conceitos de orientação a objetos, incluindo classes e objetos, relacionamento entre classes, tratamento de exceções.		
PROGRAMA		
Unidade I - Introdução à POO - Definições: Abstração, Classes x objetos - Introdução ao desenvolvimento de sistemas reusáveis de software - Introdução à uma Linguagem OO (sintaxe básica e apresentação de estruturas de código) - Considerações teóricas sobre projeto e pacotes Unidade II - Classes e objetos - Classes: sintaxe para declaração - Membros de classes: atributos, métodos, construtores [e destrutores, caso sejam presentes na linguagem] - Encapsulamento: membros public, private, protected e default - Acesso ao encapsulamento: métodos getters e setters - Operadores new, this e self - Pacotes [e/ou namespaces, caso presentes na linguagem]: sintaxe Unidade III - Relacionamento entre classes - Associação, dependência e multiplicidade: considerações práticas; - Agregação e composição: considerações práticas; - Herança, polimorfismo e métodos e classes finais: considerações práticas - Sobrecarga e sobrescrita de métodos em herança - Construtores [e destrutores] nas subclasses - Interface, Implementação e classes abstratas: considerações práticas Unidade IV - Tratamento de exceções - Conceito de exceções - Instruções try, catch, finally		
METODOLOGIA DE ENSINO		
Aulas teóricas:		

- Ministradas em sala, ou outro ambiente que facilite o processo de ensino-aprendizagem, por meio expositivo-dialógico e com discussões com resolução de exercícios, onde a ênfase está em demonstrações conceituais e fundamentos essenciais;
- Como recursos de apoio, tem-se a utilização do quadro branco, projetor de slides e livro(s) de referência(s)

Aulas práticas:

- Ministradas em laboratório de informática, ou outro ambiente que facilite a consolidação dos conceitos fundamentais.
- Como recursos de apoio, tem-se a utilização de APIs para programação orientada a objetos, de plataformas online de ensino aprendizagem

AVALIAÇÃO

O processo avaliativo deve ser contínuo e constante durante todo o processo de ensino-aprendizagem, com o propósito de analisar o progresso do aluno, criando indicadores capazes de apontar meios para ajudá-lo na construção do conhecimento.

Desta forma, para início do processo ensino-aprendizagem, sugere-se avaliações diagnósticas, como forma de se construir um panorama sobre as necessidades dos alunos e, a partir disso, estabelecer estratégias pedagógicas adequadas e trabalhar para desenvolvê-los, inclusive evidenciando os casos que necessitarão de métodos diferenciados em razão de suas especificidades, tais como a necessidade de inclusão. Essas avaliações deverão seguir, preferencialmente, métodos qualitativos, todavia, também seguirão métodos quantitativos quando cabíveis dentro dos contextos individuais e coletivos da turma.

Durante toda a continuidade do processo ensino-aprendizagem, sugere-se a promoção, em alta frequência, de avaliações formativas capazes de proporcionar ao docente um feedback imediato de como estão as interferências pedagógicas em sala de aula, e permitindo ao aluno uma reflexão sobre ele mesmo, exigindo autoconhecimento e controle sobre a sua responsabilidade, frente aos conteúdos já vistos em aula, privilegiando a preocupação com a satisfação pessoal do aluno e juntando informações importantes para mudanças na metodologia e intervenções decisivas na construção de conhecimento dos discentes, inclusive com subsídios para propostas de atividades de recuperação paralela na(s) reunião(ões) de colegiado de curso, coordenadoria de curso e demais setores ligados ao ensino.

Ao final de cada etapa do período letivo, pode-se realizar avaliações somativas, com o objetivo de identificar o rendimento alcançado tendo como referência os objetivos previstos para a disciplina. Há nesses momentos a oportunidade de utilizar recursos quantitativos, tais como exames objetivos ou subjetivos, inclusive com recursos de TIC, todavia, recomenda-se a busca por métodos qualitativos, baseados no planejamento de projetos práticos, práticas interdisciplinares ou atuação em experimentos de laboratório, dentre outros.

BIBLIOGRAFIA BÁSICA

BARNES, David J.; KÖLLING, Michael. **Programação orientada a objetos com Java**: uma introdução prática usando o BlueJ. 4. ed. São Paulo: Pearson Prentice Hall, 2009. ISBN 9788576051879. Disponível em: <https://plataforma.bvirtual.com.br/Acervo/Publicacao/434>. Acesso em: 17 jul. 2020.

FÉLIX, Rafael. **Programação orientada a objetos**. São Paulo: Pearson Education do Brasil, 2016. ISBN 9788543020174. Disponível em: <https://plataforma.bvirtual.com.br/Acervo/Publicacao/128217>. Acesso em: 21 nov. 2021.

DEITEL, Paul; DEITEL, Harvey. **Java como programar**. 8. ed. São Paulo: Pearson Prentice Hall, 2010. ISBN 9788576055631. Disponível em: <https://plataforma.bvirtual.com.br/Acervo/Publicacao/1142>.

BIBLIOGRAFIA COMPLEMENTAR

SANTOS, Rafael. **Introdução à programação orientada a objetos usando Java**. Rio de Janeiro: Elsevier, 2013. ISBN: 9788535274332.

SINTES, Anthony. **Aprenda programação orientada a objetos em 21 dias**. São Paulo: Pearson Education do Brasil, 2002. E-book. (720 p.). ISBN 9788534614610. Disponível em: <https://plataforma.bvirtual.com.br/Acervo/Publicacao/8>. Acesso em: 17 Jul. 2020.

PAGE-JONES, Meilir. **Fundamentos do desenho orientado a objeto com UML**. São Paulo: Pearson, 2001. ISBN 9788534612432. Disponível em: <https://plataforma.bvirtual.com.br/Acervo/Publicacao/33>. Acesso em: 17 Jul. 2020.

HORSTMANN, Cay S.; CORNELL, Gary. **Core Java**: volume I: fundamentos. 8. ed. São Paulo: Pearson Prentice Hall, 2010. ISBN 9788576053576. Disponível em: <https://plataforma.bvirtual.com.br/Acervo/Publicacao/1238>. Acesso em: 17 Jul. 2020.

DEITEL, Harvey M.; DEITEL, Paul J. **C++ como programar**. 5. ed. São Paulo: Pearson Prentice Hall, 2006. ISBN 9788576050568. Disponível em: <https://plataforma.bvirtual.com.br/Acervo/Publicacao/338>. Acesso em: 17 Jul. 2020.

MIZRAHI, Victorine Viviane. **Treinamento em linguagem C++**: módulo 2. 2. ed. São Paulo: Pearson Prentice Hall, 2006. ISBN 9788576050469. Disponível em: <https://plataforma.bvirtual.com.br/Acervo/Publicacao/343>. Acesso em: 17 Jul. 2020.

Coordenador do Curso

Setor Pedagógico