



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PARANÁ
SETOR DE PALOTINA

Departamento de Engenharia e Exatas

Ficha 2 (variável)

Disciplina: Programação Orientada a Objetos I				Código: DEE358			
Natureza: (X) Obrigatória () Optativa		(X) Semestral () Anual () Modular					
Pré-requisito: DEE346 - Algoritmos e Est. de Dados I		Co-requisito:		Modalidade: (X) Presencial () Totalmente EAD () CH em EAD: 20%			
CH Total: 60 CH Semanal: 4h	Padrão (PD): 30	Laboratório (LB): 30	Campo (CP): 0	Estágio (ES): 0	Orientada (OR): 0	Prática Específica (PE): 0	Estágio de Formação Pedagógica (EFP): 0

EMENTA

Programação orientada a objetos. Métodos e Classes. Encapsulamento, interface e implementação. Composição, agregação e associação. Herança e polimorfismo. Tratamento de exceções. Técnicas de Desenvolvimento de Programas: Projeto Prático de Programação. Boas práticas de projeto e implementação de programas orientados a objetos. Introdução a padrões de projetos orientados a objetos.

PROGRAMA/CRONOGRAMA

1. Programação orientada a objetos. Métodos e Classes. Encapsulamento, interface e implementação. Composição, agregação e associação. Herança e polimorfismo.

- 1.1. Conceitos Básicos
- 1.2. Classes e Objetos
- 1.3. Atributos
- 1.4. Métodos, argumentos e parâmetros
- 1.5. Comunicação por troca de mensagens
- 1.6. Encapsulamento e ocultamento de informações
- 1.7. Associação
- 1.8. Hierarquia de agregação/decomposição
- 1.9. Hierarquia de especialização/generalização
- 1.10. Herança e polimorfismo
- 1.12. Aplicação dos Conceitos básicos no ambiente de desenvolvimento integrado (IDE) BlueJ

2. Linguagem Java. Tratamento de exceções.

- 2.1. IDE NetBeans e Eclipse
- 2.2. Histórico, Certificação e Java no Mercado de Trabalho
- 2.3. Fundamentos da Sintaxe e Semântica da Linguagem Java
- 2.4. Aplicação dos Conceitos de Orientação a Objetos com Java
- 2.5. Tratamento de exceções em Java.
- 2.6. Comparação de Java com outras linguagens orientadas a objetos.

3. Introdução aos Padrões de Projeto Orientados a Objetos

- 3.1. Padrões de Criação
- 3.2. Padrões Estruturais

OBJETIVO GERAL

Possibilitar ao aluno conhecer os conceitos de orientação a objetos e aplicá-los por meio de uma linguagem orientada a objetos bem difundida.

OBJETIVO ESPECÍFICO

Ao final da disciplina o aluno deverá ser capaz de:

- Entender e aplicar os fundamentos do paradigma de programação orientada a objetos;
- Planejar software antes de implementação direta dos conceitos de orientação a objetos, adotando técnicas de desenvolvimento e boas práticas de implementação de programas orientados a objetos;
- Conhecer as diferentes ferramentas disponíveis para facilitar o planejamento e desenvolvimento de software orientado a objetos;
- Compreender e identificar os principais padrões de projeto orientados a objetos.

PROCEDIMENTOS DIDÁTICOS

Apresentar os conceitos fundamentais da orientação a objetos, bem como aplicá-los na prática de resolução de problemas do cotidiano por meio de software, enfatizando a importância da aplicação de técnicas e boas práticas provenientes da Engenharia de Software e Indústria. Expor e exercitar os principais conceitos e ferramentas básicas relacionadas ao tema.

I) Métodos e Técnicas de Ensino

Aula expositiva e resolução de exemplos por meio de vídeo aulas gravadas e disponibilizadas previamente pelo docente; fixação do tema por meio de textos e vídeos complementares, listas de exercícios, participação em fórum e atividades práticas assíncronas.

- a. Quadro;
- b. Material multimídia, como projetor, laptop e computadores (laboratório);
- c. Ambiente de Desenvolvimento Integrado (IDE) NetBeans e Eclipse;
- d. Linguagens de programação orientadas a objetos;
- e. Artigos e Materiais didáticos (Bibliografia básica e complementar);
- f. Ambiente Virtual de Aprendizagem (AVA).

Todos os materiais (bibliográficos e extras), atividades e informações da disciplina serão disponibilizados no AVA UFPR Virtual da disciplina, bem como a submissão de todas as atividades extraclasse pelos alunos, com a exceção das que utilizarem outros recursos educacionais a serem previamente explicados também via UFPR Virtual.

Atendimentos síncronos presenciais ou remotos deverão ocorrer mediante a solicitação prévia dos alunos via, única e exclusivamente, e-mail institucional do docente: anderson.marcolino@ufpr.br.

II) Recursos Didáticos

Computador ou laptop; AVA UFPR Virtual; ambientes de desenvolvimento integrado (BlueJ e NetBeans) e software para modelagem de projetos orientados a objetos com UML (Astar ou diagrams.net).

FORMAS DE AVALIAÇÃO

A avaliação do aprendizado ao conteúdo proposto na disciplina será realizada por meio de médias aritméticas ponderadas e posterior multiplicação de pesos, a saber:

Parte Teórica (NT):

$$NT = (AVAL1 + AVAL2)/2$$

Onde NT corresponde à média das notas da avaliação 1 (AVAL1), realizada no primeiro trimestre e avaliação 2 (AVAL2), realizada no segundo trimestre.

Parte Prática (NP):

$$NP = (NP1 + NP2 + NPn)/n$$

A avaliação será composta pelos trabalhos desenvolvidos e apresentados durante as aulas. Em destaque para desenvolvimento trabalhos escritos; como lista de exercícios e atividades (presenciais ou em EaD), uma apresentação de seminário, a exposição oral do trabalho para a turma e implementações nos respectivos tópicos. Cada trabalho (NP) terá sua nota somada e posteriormente dividida pela quantidade de trabalhos realizados, resultando na média da nota prática (NP).

Nota final (NF):

$$NF = NT \cdot 0,6 + NP \cdot 0,4$$

A nota final da disciplina (NF) será a soma das médias ponderadas da parte teórica (NT) e parte prática (NP), multiplicados pelos pesos de 0,6 e 0,4, respectivamente.

Os alunos que obtiverem média de aproveitamento (NF) inferior a 70,0 e igual ou superior a 40,0 e frequência igual ou superior a 75% deverão prestar exame final, o qual constará de uma prova escrita acerca de todo o conteúdo da disciplina. Para ser aprovado no exame, o aluno deve obter média final igual ou superior a 50,0. Caso não tenha frequência igual ou superior a 75%, média de aproveitamento superior (NT) ou igual a 40,0 e média final (NT considerando a nota do exame) igual ou superior a 50,0 o aluno será reprovado.

BIBLIOGRAFIA BÁSICA (mínimo 03 títulos)

DEITEL, Paul J. **Java: como programar**. 10. ed São Paulo: Pearson Prentice Hall, 2017. xxxiv, 934p., il. ISBN 9788543004792.

SANTOS, Rafael. **Introdução à programação orientada a objetos usando JAVA**. 2. ed Rio de Janeiro: Elsevier, 2013. 313 p., il.

BARNES, David J. **Programação orientada a objetos com Java: uma introdução prática usando o BlueJ**. 4. ed. São Paulo: Prentice Hall, c2009. 455 p., il.

BIBLIOGRAFIA COMPLEMENTAR (mínimo 05 títulos)

JANDL JUNIOR, Peter. **Java: guia do programador : atualizado para Java 8**. 3. ed São Paulo: Novatec, c2015. 704 p., il. I

ANDERSON, Julie. **Java 6: uma abordagem ativa de aprendizado**. 2. ed Rio de Janeiro: LTC, 2010. 783 p., il.

PUGA, Sandra; RISSETTI, Gerson. **Lógica de programação e estruturas de dados com aplicações em java**. 2. ed. São Paulo: Prentice Hall, 2009. xiv, 262 p., il., 25 cm.

GAMMA, Erich. **Padrões de projeto: soluções reutilizáveis de software orientado a objetos**. Porto Alegre: Bookman, 2000 [reimpressão 2005 e 2008]. 364p., il. (broch.).

WEST, D. **Use a Cabeça! Análise & Projeto Orientado ao Objeto**. Alta Books, 2007.

INFORMAÇÕES COMPLEMENTARES

Docente responsável: Anderson da Silva Marcolino

E-mail: anderson.marcolino@ufpr.br

Vagas: 30

Esta ficha e seus respectivos itens seguem as especificações da Resolução CEPE nº 04/2022.



Documento assinado eletronicamente por **ANDERSON DA SILVA MARCOLINO**,
PROFESSOR DO MAGISTERIO SUPERIOR, em 07/04/2022, às 20:19, conforme art. 1º, III,
"b", da Lei 11.419/2006.



A autenticidade do documento pode ser conferida [aqui](#) informando o código verificador **4407001** e o código CRC **FD3F743C**.