

EMENTA

PLANO DE ENSINO

1. IDENTIFICAÇÃO

Campus: Avançado Quedas do Iguaçu

Eixo tecnológico: Informação e Comunicação

Curso: Técnico em Informática Integrado ao Ensino Médio

Componente curricular: Programação Orientada a Objetos

Docente: Danilo Giacobbo

Carga horária: 133 horas - 160 h/a

Turno: Matutino e vespertino

Número de aulas na semana: 4

Período letivo: 2020

Turma (s): 2º ano

Coordenador do curso: Odair Moreira de Souza

2. EMENTA

Conceitos da Orientação a Objetos. Programação Orientada a Objetos (POO). Elementos básicos de uma linguagem de desenvolvimento POO. Ambientes Integrados de desenvolvimento. Linguagem de programação Java. As APIs do Java. Tratamento de exceções. Desenvolvimento de interfaces gráficas com o usuário. Acesso a bancos de dados.

3. OBJETIVOS

3.1 Objetivo geral

Compreender os conceitos da programação orientada a objetos, adquirindo habilidades para o desenvolvimento de sistemas em camadas.

3.2 Objetivos específicos

- Entender os fundamentos do Paradigma Orientado a Objetos;
- Aprender uma linguagem de Programação Orientada a Objetos;

- Apresentar o processo de desenvolvimento orientado a objetos;
- Compreender os principais recursos da linguagem de programação;
- Conhecer os componentes para o desenvolvimento de software com interfaces gráficas e conexão com banco de dados;
- Modelar e implementar problemas utilizando Programação Orientada a Objetos; e
- Adquirir domínio básico de uma linguagem de programação orientada a objetos através da aplicação prática dos conceitos aprendidos.

4. CONTEÚDO PROGRAMÁTICO

- Conceitos de Programação
 - Algoritmo, Automação, Programa de Computador
 - O que é preciso para se fazer um programa de computador
 - Linguagem de programação, léxica, sintática
 - IDE - Ambiente Integrado de Desenvolvimento
- Introdução à linguagem Java
 - Entendendo as versões do Java
 - Histórico e edições de Java
 - JDK / JVM - Máquina Virtual do Java
 - Estrutura de uma aplicação Java
 - Instalando Java
 - Instalando o Eclipse / Netbeans
 - Primeiro programa em Java no Eclipse / Netbeans
- Estrutura sequencial
 - Expressões aritméticas
 - Variáveis e tipos básicos em Java
 - As três operações básicas de programação
 - Saída de dados em Java
 - Processamento de dados em Java, Casting
 - Entrada de dados em Java
 - Funções matemáticas em Java
- Estrutura condicional
 - Expressões comparativas
 - Expressões lógicas
 - Estrutura condicional (if-else)
 - Operadores de atribuição cumulativa
 - Estrutura switch-case
 - Expressão condicional ternária
 - Escopo e inicialização
- Estruturas repetitivas
 - Como utilizar o DEBUG no Eclipse / Netbeans
 - Estrutura repetitiva enquanto (while)
 - Teste de mesa com estrutura repetitiva enquanto
 - Estrutura repetitiva para (for)
 - Teste de mesa com estrutura repetitiva para
 - Exercícios de testes de mesa com for
 - Estrutura repetitiva faça-enquanto (do-while)
- Tópicos básicos sobre Java
 - Restrições e convenções para nomes
 - Operadores bitwise
 - Funções interessantes para String
 - Comentários em Java (básico)
 - Funções (sintaxe)
 - Trabalhando com datas - Date
- Introdução à Programação Orientada a Objetos
 - Object e toString

- Membros estáticos
 - Construtores
 - Palavra this
 - Sobrecarga
 - Encapsulamento
 - Geração automática de construtores, getters e setters com Eclipse / Netbeans
 - Modificadores de acesso
- Comportamento de memória, arrays, listas
 - Tipos referência vs. tipos valor
 - Desalocação de memória - garbage collector e escopo local
 - Vetores
 - Boxing, unboxing e wrapper classes
 - Laço for each
 - Listas
 - Matrizes
- Enumerações, composição
 - Enumerações
 - Composição
- Herança e polimorfismo
 - Herança
 - Upcasting e downcasting
 - Sobreposição, palavra super, anotação @Override
 - Classes e métodos final
 - Introdução ao polimorfismo
 - Classes abstratas
 - Métodos abstratos
- Tratamento de exceções
 - Discussão inicial sobre exceções
 - Estrutura try-catch
 - Pilha de chamadas de métodos (stack trace)
 - Bloco finally
 - Criando exceções personalizadas
- Interfaces
 - Inversão de controle e injeção de dependência
 - Herdar vs. cumprir contrato
 - Herança múltipla e o problema do diamante
 - Interface Comparable
 - Default methods
- Generics, Set, Map
 - Introdução aos Generics
 - Genéricos delimitados
 - Tipos curinga
 - Curingas delimitados
 - hashCode e Equals
 - Set
 - Map
- Acesso a banco de dados com JDBC
 - Visão geral do JDBC
 - Padrão de projeto DAO (Data Access Object)
- Interface gráfica com JavaFX
 - Visão Geral do JavaFX
 - Instalação do Scene Builder
 - Preparação do Eclipse / Netbeans
 - Criando um novo projeto JavaFX no Eclipse / Netbeans
 - Testando o FXML
 - Tratando eventos com JavaFX
 - Mostrando Alert
 - Usando TextField e Label

- Limitações para TextField, interface Initializable
- ComboBox
- Visão geral dos principais containers de layout

5. METODOLOGIA DE ENSINO E AVALIAÇÃO

A metodologia a ser utilizada engloba aulas expositivas dialogadas, contemplando a resolução e interpretação de exercícios, com a utilização de exemplos na lousa e no computador. Aplicação de listas de exercícios como fixação de conteúdo. Aplicação de palavras-cruzadas. Utilização de jogos de perguntas e respostas (passa ou repassa). Utilização de jogos digitais e não digitais. Uso de recurso áudio visual (data show). Discussão de atividades em grupo; debates em sala e constante interação entre o docente e os discentes em sala referente aos assuntos abordados. Aulas práticas em laboratório. Questões de concursos públicos. Simulados.

A avaliação se dará de forma contínua compreendendo a metodologia citada anteriormente. O professor manterá um registro de todas as atividades realizadas pelos alunos.

O presente componente curricular estará integrado às disciplinas de Língua Estrangeira Moderna - Inglês, Banco de Dados, Engenharia de Software, Tópicos Especiais em Tecnologia e Tópicos Especiais em Programação.

5.1 Recursos didáticos

Os recursos didáticos e materiais a serem adotados e utilizados ao longo da disciplina englobam o quadro branco, aparelho multimídia, computadores, softwares, textos impressos, conteúdo disponibilizado na ferramenta Classroom, vídeos do Youtube, entre outros.

6. CRITÉRIOS DE AVALIAÇÃO

A avaliação do aluno ocorrerá por meio de uma ou mais avaliações teóricas e práticas, trabalhos individuais e em grupo, atividades em sala de aula, apresentações de seminário, preenchimento de palavras-cruzadas, jogos didáticos, entre outros. O número de instrumentos avaliativos serão 5 (cinco): seminários, testes escritos e/ou orais/sinalizados, trabalhos individuais e/ou em grupos; outras atividades de ensino, pesquisa, extensão e inovação pertinentes aos cursos, neste caso a inclusão de jogos digitais na avaliação da aprendizagem e atividades em Ambiente Virtual de Aprendizagem (AVA).

A pesquisa e as leituras complementares serão incentivadas, podendo ser apresentadas como forma de avaliação. A participação do aluno ouvindo, questionando, argumentando, assim como sua postura respeitosa e acadêmica, são pontos fundamentais da avaliação.

Os resultados serão apresentados a cada trabalho, sendo explicitado o diagnóstico feito pelo docente. De acordo com as normas da Instituição, os alunos receberão os conceitos A, B, C ou D nos períodos determinados pelo IFPR e no final do conteúdo de cada área curricular.

O docente também irá fornecer aos discentes em cada atividade avaliativa os critérios necessários para obtenção dos conceitos utilizados na instituição.

7. RECUPERAÇÃO

A recuperação se dará de forma contínua para os alunos com dificuldades em assimilar os conteúdos vistos em sala de aula. O professor estará diagnosticando ao longo das aulas os alunos que precisam de auxílio e propor outras formas de abordar o conteúdo da disciplina para melhorar o seu rendimento. Para os

alunos com baixo rendimento será ofertado uma avaliação de recuperação ao final do bimestre para recuperação do conceito obtido. Nos horários de atendimento serão ofertadas aulas de reforço para os alunos com dificuldades também. O docente disponibilizará horários de atendimento para os alunos com dificuldades, sendo que a frequência será registrada por meio de um aplicativo.

Com base na Resolução nº 50/2017 do IFPR, a recuperação do estudante poderá ser realizada no decorrer do módulo da disciplina via atendimento mais direto e individualizado e com atividades diferenciadas que possam contemplar sua aprendizagem. Se a aprendizagem for ainda considerada insuficiente, o estudante cursará a disciplina novamente como progressão, em horários previamente combinados.

Para os estudantes que apresentarem dificuldades, será ofertada a realização de novos trabalhos que utilizem habilidades diferentes daquelas na qual ele apresenta dificuldade.

8. BIBLIOGRAFIA

8.1 Bibliografia Básica

DEITEL, H. M.; DEITEL, P. J. **Java: como programar**. 8. ed. São Paulo: Prentice Hall Brasil, 2010.

SANTOS, R. **Introdução à Programação Orientada a Objetos Usando Java**. 2. ed. Rio de Janeiro: Campus – Elsevier, 2013.

SCHILDT, H. **Java para iniciantes**. 5. ed. Porto Alegre: Bookman, 2013.

SIERRA, K.; BATES, B. **Use a cabeça! Java**. 2.ed. Rio de Janeiro, RJ: Alta Books, 2010.

TURINI, R. **Desbravando Java e Orientação a Objetos: Um guia para o iniciante da linguagem**. São Paulo: Casa do Código, 2014.

8.2 Bibliografia Complementar

BARNES, D. J.; KÖLLING, M. **Programação orientada a objetos com Java: uma introdução prática usando o BlueJ**. 4. ed. São Paulo, SP: Pearson Prentice Hall, 2009. xxii, 455 p. ISBN 9788576051879.

BORATTI, I. C. **Programação orientada a objetos em Java**. Florianópolis: Visual Books, 2007. 310 p. ISBN 978857502199-6.

CARDOSO, C. **Orientação a objetos na prática: aprendendo orientação a objetos com Java**. Rio de Janeiro: Ciência Moderna, 2006. 175 p. ISBN 8573935383.

HORSTMANN, C. S.; CORNELL, G. **Core Java**. 8. ed. São Paulo, SP: Pearson, 2010. 400 p. ISBN 9788576053576.

LARMAN, C. **Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**. 3. ed. Porto Alegre: Bookman, 2007. 695 p. ISBN 85-363-0358-1.

9. OBSERVAÇÕES

- A distribuição do conteúdo das aulas é uma previsão e poderá ser adequado durante o andamento do ano letivo para atender aos reajustes em função do desempenho da turma ou para tratar eventualidades.
- O Google Classroom será o canal de comunicação digital entre o docente e os discentes, para disponibilização dos materiais das aulas, listas de exercícios, implementações de exemplos, submissão de atividades, avisos para a turma, agendamento de atividades.
- O acesso dos estudantes aos laboratórios para o desenvolvimento de atividades em horários alternativos aos das aulas deverá ser solicitado ao professor e ter anuência do responsável pelo laboratório.



Documento assinado eletronicamente por **DANILO GIACOBO, Servidor Docente**, em 22/04/2020, às 08:44, 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://sei.ifpr.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0705617** e o código CRC **D61E9CEF**.