

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: Odair Moreira de Souza

Carga horária: 120 horas/aula e 100 horas/relógio

Turno: Matutino

Número de aulas na semana: 3 aulas - 2,45 horas

Período letivo: 2020

Turma (s): DP B (progressão)

Coordenador do curso: Odair Moreira de Souza

2. EMENTA

Linguagem Java. Variáveis e Tipos Primitivos. Orientação a Objetos. Encapsulamento. Herança e Polimorfismo. Classe Abstrata. Interfaces. Pacotes. Arrays e Exceptions. As APIs do Java. Collection Framework. A História da linguagem Java.

3. OBJETIVOS

3.1 Objetivo geral

Compreender os conceitos da programação orientada a objetos, adquirindo habilidades para o desenvolvimento de sistemas em camadas e desenvolvimento de soluções computacionais por meio de linguagem de programação.

3.2 Objetivos específicos

- Entender os fundamentos do Paradigma Orientado a Objetos;
- Aprender uma linguagem de Programação Orientada a Objetos;
- Documentar 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 empregada no processo de ensino-aprendizagem consiste em aulas expositivas e dialogadas, incentivando a participação dos discentes durante a explanação do conteúdo, resolução de problemas reais, construção e contextualização dos conhecimentos abordados.

O conteúdo teórico será abordado utilizando como referencial teórico a bibliografia sugerida, por meio de exemplos e estudos de caso e ao final de cada assunto, ou conjunto de assuntos relacionados, será proposto aos discentes o desenvolvimento de exercícios individuais ou em dupla para fixação da teoria apresentada, os alunos irão realizar apresentações das soluções desenvolvidas.

Nas aulas práticas serão ministradas no laboratório de informática e para todos os conteúdos abordados, serão apresentados exemplos desenvolvidos em ambiente computacional e desenvolvido exemplos e exercícios. Em seguida solicita-se aos discentes que apliquem os conceitos expostos, com o intuito de incentivar a reflexão e a habilidade de raciocínio para resolução de problemas.

As atividades de desenvolvimentos de trabalhos compreenderão a construção de soluções computacionais para os problemas propostos e que propiciem a fixação dos conteúdos previamente identificados, será utilizado problemas de outras áreas do conhecimento, tais como, matemática, física e entre outras, para desenvolver soluções computacionais aplicadas.

5.1 Recursos didáticos

Será utilizado laboratório de informática, projetor multimídia e plataformas online de ensino-aprendizagem de programação.

6. CRITÉRIOS DE AVALIAÇÃO

A avaliação será realizada seguindo as orientações da Resolução nº 50/2017, que estabelece as normas de avaliação dos processos de ensino e aprendizagem no IFPR. Em cada bimestre, o conceito do aluno será composto por meio de sua participação e desempenho nos seguintes pontos:

- Participação e desempenho nos trabalhos individuais;
- Desenvolvimento de aplicações práticas;
- Desempenhos em avaliações individuais teóricas e práticas;
- Participação, assiduidade e proatividade.

Considera-se que para cada bimestre serão aplicados, no mínimo quatro atividade avaliativa prática de desenvolvimentos (40% do conceito bimestral) e uma ou duas avaliações práticas de desenvolvimento de software (60% do conceito bimestral). Ressalta-se que os trabalhos devem ser apresentados para a turma e/ou somente para o professor em formato de arguição.

Os resultados serão apresentados a cada atividade avaliativa, 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.

Interdisciplinaridade

Essa disciplina poderá ter atividades avaliativas interdisciplinares, envolvendo conteúdos relacionados com as disciplinas de Banco de Dados, Física e Matemática por meio de trabalhos práticos de desenvolvimentos.

Critérios de Aprovação

Conforme previsto na Resolução nº 50/2017 do IFPR, Art. 16. A aprovação dos estudantes ocorrerá considerando os seguintes critérios:

I – obtenção de conceito A, B ou C no componente curricular e frequência igual ou superior a 75% (setenta e cinco por cento) da carga horária total no período letivo dos cursos técnicos de nível médio.

7. RECUPERAÇÃO

I - Programa de Atividades e de Orientação: A recuperação do estudante poderá ser realizada no decorrer do módulo da disciplina através de atendimento mais direto e individualizado e com atividades diferenciadas que possam contemplar sua aprendizagem, baseada na Resolução nº 50/2017 do IFPR. O aluno com conceito insuficiente em um bimestre será convocado a participar de recuperação paralela contínua durante o próximo bimestre, exceto no último bimestre, pois isso possibilitará ao alunos tempo e atendimento para compreender a conteúdo antes da avaliação de recuperação. Caso a aprendizagem ainda for considerada insuficiente, o estudante cursa a disciplina novamente como progressão, em horários previamente combinados.

II - Formas de Avaliação: As atividades avaliativas de recuperação serão ofertadas em contraturno dos estudantes. Além disso, os alunos contam com atendimentos individualizados pelo professor. As recuperações paralelas serão agendadas com no mínimo 15 (quinze) dias de antecedência da aplicação.

III - Direito de Realiza as Avaliações de Recuperação: Conforme Resolução CONSUP/IFPR nº 50/2017, serão oferecidos estudos de recuperação paralela ou retomada dos conteúdos a todos os estudantes, independente do conceito atingido ser B, C ou D.

8. BIBLIOGRAFIA

8.1 Bibliografia Básica

1. DEITEL, H. M.; DEITEL, P. J. Java: como programar. 8. ed. São Paulo: Prentice Hall Brasil, 2010.
2. SANTOS, R. Introdução à Programação Orientada a Objetos Usando Java. 2. ed. Rio de Janeiro: Campus – Elsevier, 2013.
3. SCHILDT, H. Java para iniciantes. 5. ed. Porto Alegre: Bookman, 2013.
4. SIERRA, K.; BATES, B. Use a cabeça! Java. 2.ed. Rio de Janeiro, RJ: Alta Books, 2010.
5. 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

1. 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.
2. BORATTI, I. C. Programação orientada a objetos em java. Florianópolis: Visual Books, 2007. 310 p. ISBN 978857502199-6.
3. CARDOSO, C. Orientação a objetos na prática: aprendendo orientação a objetos com Java. Rio de Janeiro: Ciência Moderna, 2006. 175 p. ISBN 8573935383.
4. HORSTMANN, C. S.; CORNELL, G. Core Java. 8. ed. São Paulo, SP: Pearson, 2010. 400 p. ISBN 9788576053576.
5. 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 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.
- O local e os horários de atendimentos do docente e do monitor de programação serão divulgados nos murais de avisos do campus e na seção de avisos da turma do Google Classroom.



Documento assinado eletronicamente por **ODAIR MOREIRA DE SOUZA, Servidor Docente**, em 08/03/2020, às 13:55, 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 **0696023** e o código CRC **E1CA50C2**.

INSTITUTO FEDERAL DO PARANÁ | QUEDAS/SENS/QUEDAS/DG/IFPR/QUEDAS/IFPR/CASCADEL-
SENS/QUEDAS
Rua Emilio Bertolini, nº 54, Curitiba - PR | CEP CEP 82920-030 - Brasil