

CEST: MISSÃO E VISÃO

Missão	<i>Promover educação superior de excelência, formando profissionais competentes e éticos, cidadãos comprometidos com o desenvolvimento da sociedade e com o segmento das pessoas com deficiência, conscientes de sua função transformadora.</i>
Visão	<i>Ser referência regional em educação superior, reconhecida pela excelência de sua atuação e compromisso social.</i>

DADOS DA DISCIPLINA

CÓDIGO	NOME	CH TOTAL	PERÍODO
	PROGRAMAÇÃO ORIENTADA A OBJETOS	80h	4º

EMENTA

Fundamentos de orientação a objetos: termos básicos, objetos, campos e métodos, encapsulamento, mensagens, classes, hierarquia de classes, herança, composição, delegação, polimorfismo, interface, classes abstratas, acoplamento X coesão. Uso da orientação a objetos em uma linguagem de programação OO: definição de classes, visibilidade dos membros de uma classe, redefinição de métodos, compatibilidade entre objetos, construtores e destrutores, operadores de classe, interfaces. Tratamento de exceções. Linguagem orientada a objetos: Java.

OBJETIVOS

GERAL

Compreender a Linguagem Orientada a Objetos, tendo em vista sua aplicação na modelagem e resolução de problemas.

ESPECÍFICOS

Descrever a solução de problemas na forma de algoritmos e Programas de computador.

Utilizar ferramentas e ambientes de desenvolvimento de sistemas.

Criar e manipular estruturas de armazenamento de dados.

CONTEÚDO PROGRAMÁTICO

UNIDADE I – INTRODUÇÃO (10h)

1.1 Introdução à Orientação a Objetos

1.2 Linguagens, ambientes, padrão de codificação, estrutura de pastas, compilação de classes.

1.3 Objetos e Classes – do caos a era dos objetos. Base da Orientação a Objetos. Classes, métodos e atributos.

Classe vs Objeto, instância, assinatura de métodos

1.4 Packages: Organização, implementação, imports, árvore de diretórios, exercícios.

UNIDADE II – ENCAPSULAMENTO E HERANÇA (14h)

2.1 Tipo de variáveis (primitivos e objetos). Níveis de acesso (public, default, protected, private), aplicabilidades

2.2 Encapsulamento

2.3 Herança. Conceito, funcionalidades e aplicações

UNIDADE III – JAVA (14)

3.1 Controle de execução (if..then..else, loop for, interação, true or false) 3.2 Arrays e Vetores

3.2 Lista Lineares

3.3 Foreach, break e continue, switch e Random

UNIDADE IV – JAVA E OO (14)

4.1 Abstração e interfaces (extends vs implements). Aplicação de conceitos usando Abstração, Herança, Constructor e Interfaces

4.2 Polimorfismo, Sobrecarga de funções

4.3 Reuso de classes

4.4 Reuso de classes, exercícios

4.5 Exceções em Java. Usando, implementando, extend

UNIDADE V – DESIGN PATTERNS (16h)

5.1 Design Patterns: Explicação geral

5.2 Design Patterns: Singleton

5.3 Design Patterns: Iterator

5.4 Robocode – Aplicando conceitos Representações de árvores em Linguagem C

UNIDADE VI – ARQUIVOS E BD (12h)

6.1 Introdução

6.2 Acesso a Arquivos

6.3 Acesso a Banco de Dados

PROCEDIMENTOS DIDÁTICOS

Aulas expositivas dialogadas; Atividades individuais e em equipes; Exposição de vídeos; Estudos dirigidos (individuais e em grupos).

TRABALHOS DISCENTES EFETIVOS – TDE's

1o TDE: 1a Lista de Exercícios (04h)

2o TDE: 2a Lista de Exercícios (04h)

3o TDE: 3a Proposta de uma aplicação em Java (08h)

RECURSOS DIDÁTICOS

Projektor de multimídia; Vídeos, Livros, Textos escritos em língua inglesa, Quadro e Pincel.

SISTEMA DE AVALIAÇÃO

A avaliação tem caráter processual e diagnóstico, objetivando o acompanhamento do desempenho do aluno no decorrer da disciplina, para tanto, devem ser considerados aspectos qualitativos como a participação ativa nas aulas e atividades acadêmicas, o relacionamento aluno-professor e aluno-aluno, a cooperação, a competência fundamentada na segurança dos conhecimentos adquiridos, a autonomia para aprofundar os conhecimentos, a pontualidade, o cumprimento de prazos na entrega de trabalhos, dentre outros.

Cotidianamente, a cada aula, a avaliação ocorrerá com base em procedimentos como: discussão de temas relacionados aos conteúdos; exercícios escritos; produção textual; estudo dirigido, estudo de casos, análises de peças jurídicas etc.

Além dos aspectos qualitativos, serão observados os critérios objetivos regimentais:

a) frequência mínima de 75% da carga horária da disciplina;

b) três notas parciais (uma por mês, a cada 1/3 do conteúdo trabalhado, cumulativamente ou não) que serão compostas da seguinte forma: avaliação escrita individual com peso de 100% quando admitir, em seu conteúdo, questões referentes ao texto acadêmico; avaliações escritas individuais na proporção mínima de 70%, quando associadas a atividades acadêmicas individuais, cuja proporção será de até 30% para a composição da nota.

Estão previstas como atividades acadêmicas:

a) leitura, interpretação e fichamento de artigo com peso de 30% na composição da nota da 2ª avaliação, conforme correlação entre artigo e conteúdo trabalhado;

b) outras atividades que se fizerem necessárias de acordo com a complexidade dos conteúdos.

Em consonância às normas institucionais, não obtendo média para aprovação a partir das três notas parciais, o discente automaticamente estará inscrito para realizar a prova substitutiva e, caso ainda não seja aprovado, mas tendo alcançado a média 4,0 (quatro), se submeterá à prova final.

BIBLIOGRAFIA

BÁSICA

HORSTMANN, Cay S.; Cornell, Gary. **Core Java. 8. ed.** São Paulo: Editora Pearson. 2013. v. 1. *E-book*

MANZANO, J. A. N. G.; COSTA JÚNIOR, R. F. da. **Java 8: programação de computadores – guia prático de introdução, orientação e desenvolvimento.** São Paulo: Érika, 2014.

SIERRA, K.; BATES, B. **Java. 2. ed.** Rio de Janeiro: Alta Books, 2010.



FACULDADE SANTA TEREZINHA - CEST
COORDENAÇÃO DO CURSO DE SISTEMAS DE INFORMAÇÃO
PLANO DE ENSINO

COMPLEMENTAR

DEITEL, Paul J.; Deitel, Harvey M. **Java: como programar**. 8 ed. São Paulo: Editora Pearson, 2013. *E-book*
FÉLIX, Rafael. **Programação orientada a objetos**. São Paulo: Editora Pearson. 2017. *E-book*
JORGE, Marcos. **Java: passo a passo Lite**. São Paulo: Editora Pearson, 2013. *E-book*
MORRISON, M. **JavaScript**. Rio de Janeiro: Alta Books, 2008.
SANTOS, R. **Introdução à programação orientada a objetos usando Java**. Rio de Janeiro: Elsevier, 2012.

Coordenação do Curso de Sistemas de Informação