

**MINISTÉRIO DA EDUCAÇÃO**  
**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA**  
**DIRETORIA DE ENSINO (DIREN)**  
**DEPARTAMENTO DE ENSINO SUPERIOR (DEPES)**  
**DEPARTAMENTO DE INFORMÁTICA (DEPIN)**  
**CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA INTERNET (CST-SI)**

DEPARTAMENTO
<b>DEPIN - Departamento Acadêmico de Informática</b>

PLANO DE CURSO DA DISCIPLINA
<b>PROGRAMAÇÃO ORIENTADA A OBJETOS</b>

CÓDIGO
<b>GTSI 1233</b>

PERÍODO
3º

ANO
2012

SEMESTRE
2

PRÉ-REQUISITOS
GTSI 1213 PROJETO DE ALGORITMOS COMPUTACIONAIS

CRÉDITOS
4

AULAS/SEMANA		
TEÓRICA	PRÁTICA	ESTÁGIO
4	0	0

TOTAL DE AULAS NO SEMESTRE
72

### EMENTA

Conceitos básicos de orientação a objetos. Aplicações práticas destes conceitos utilizando a linguagem Java. Inicialmente há a ambientação na linguagem por meio de conceitos básicos como variáveis, tipos, literais, instruções, operadores, controle de fluxo. Posteriormente os conceitos gerais de orientações a objetos são abordados na linguagem, incluindo encapsulamento, entrada e saída, listas, herança, polimorfismo, tratamento de exceções e acesso a Banco de Dados com JDBC.

### BIBLIOGRAFIA

#### Bibliografia básica

1. BLOCK, Joshua, Java Efetivo, Rio de Janeiro: Alta Books.
2. DEITEL, P. e DEITEL, H., Java como programar, São Paulo: Prentice Hall/Pearson.
3. SCHILD, H., A arte do Java, São Paulo: McGraw-Hill.

#### Bibliografia complementar

1. CADENHEAD, R., Aprenda em 21 dias Java 2, Rio de Janeiro: Campus/Elsevier.
2. SIERRA, Kathy e BATES, Bert, Certificação Sun Para Programador Java 6 Guia de Estudo, Rio de Janeiro: Alta Books. ISBN: 9788576083030.
3. ASCENCIO, A. F. G. e CAMPOS, E. A. V., Fundamentos da Programação de Computadores – algoritmos, Pascal, C/C++ e Java. São Paulo: Pearson Education – Prentice Hall.
4. PREISS, Bruno R., Estruturas de Dados e Algoritmos, Rio de Janeiro: Campus.
5. BARNES D. & KÖLLING M, Programação Orientada a Objetos com Java: uma introdução prática usando o BlueJ, Rio de Janeiro: Editora Pearson.

### OBJETIVO GERAL

Iniciar os alunos nos princípios da programação orientada a objetos. Praticar fundamentos do paradigma

orientado a objetos em uma linguagem que implemente de forma natural estes conceitos.

## METODOLOGIA

- Aulas expositivas, eventualmente contando com recursos audiovisuais.
- Resolução de exercícios de fixação e propostos.

## CRITÉRIO DE AVALIAÇÃO

A avaliação semestral envolve duas provas escritas (P1 e P2). As datas das provas são agendadas entre o professor e a turma. A média parcial (MP) será calculada pelo cômputo da média aritmética simples entre a nota P1 e P2:

$$MP = (P1 + P2) / 2$$

O aluno que faltar a uma das duas provas terá direito a uma avaliação alternativa, denominada segunda chamada, versando sobre todos os tópicos abordados no curso, e cuja data também é agendada entre docente e discentes. A nota obtida nessa 2ª chamada substituirá a da avaliação P1 ou P2 onde o aluno não esteve presente. Caso ele falte às duas avaliações, terá atribuído o grau ZERO em uma delas.

Segundo o regimento do CEFET-RJ, caso o aluno obtenha média parcial inferior a 3,0 (três e zero) estará reprovado diretamente. Graus MP maiores ou iguais a 7,0 (sete e zero) aprovam diretamente o aluno. Em situações onde o aluno tenha grau MP entre 3,0 inclusive e 7,0 exclusive, terá direito a uma prova final (PF), que, juntamente com a média parcial gerará uma nova média, denominada média final (MF). Essa média é calculada da seguinte forma:

$$MF = (MP + PF) / 2$$

Para ser aprovado, o aluno deve alcançar uma MF maior ou igual a 5,0 (cinco e zero). Caso contrário, estará reprovado, devendo repetir a componente curricular.

## PROGRAMA

1. Conceitos introdutórios
  - 1.1. Introdução à Tecnologia Java
  - 1.2. Introdução ao IDE a ser utilizado
2. Conceitos básicos de Java
  - 2.1. Declaração de Variáveis
  - 2.2. Tipos Primitivos
  - 2.3. Literais, Identificadores, Palavras Reservadas e Tipos em Java
  - 2.4. Operadores
  - 2.5. Conversão de Tipo Primitivos e Promoção Aritmética
  - 2.6. Controle de Fluxo de Programas
    - if/else, switch, while, do/while, for, for avançado
    - break, continue
3. Conceitos intermediários de Java
  - 3.1. Conceito de Modelagem de Classes

- 3.2. Definição de Classes
- 3.3. Instanciação de objetos
- 3.4. Package e Import
- 3.5. Garbage Collection
- 3.6. Atributos e Constantes
- 3.7. Métodos, construtores, passagem de parâmetros
- 3.8. Modificadores de acesso
- 3.9. JavaBeans: regras para se escrever JavaBeans e propriedades de um JavaBeans
- 3.10. Arrays
- 3.11. Herança
- 3.12. Overhiding (Sobrescrita) e Overloading (Sobrecarga)
- 3.13. Uso de this e super
- 3.14. Classes abstratas e interfaces
- 3.15. Tipos enumerados
- 3.16. Tratamento de Exceções
- 3.17. Principais classes dos pacotes java.util e java.lang
- 3.18. Autoboxing e Unboxing
- 3.19. Tipos parametrizados (Generics)
- 3.20. Threads
- 4. Acesso a Banco de Dados com JDBC
  - 4.1. Introdução
  - 4.2. A estrutura JDBC
  - 4.3. Conexão com um Banco de Dados
  - 4.4. A interface java.sql.Driver
  - 4.5. A classe java.sql.DriverManager
  - 4.6. A interface Connection
  - 4.7. As interfaces Statement e PreparedStatement
  - 4.8. *Realizando consultas no banco de dados.*
  - 4.9. *A interface ResultSet*
- 5. Testes
  - 5.1. Introdução a testes unitários com junit
- 6. Anotações