



## Plano de Ensino

---

### 1) Identificação

**Disciplina:** INE5402 - Programação Orientada a Objetos I  
**Turma(s):** 01208A  
**Carga horária:** 108 horas-aula      Teóricas: 30      Práticas: 78  
**Período:** 1º semestre de 2017

### 2) Cursos

- Ciências da Computação (208)

### 3) Requisitos

- Não há

### 4) Professores

- Luciana de Oliveira Rech (luciana.rech@ufsc.br)

### 5) Ementa

Algoritmo. Programação de computador. Resolução de problemas usando computador. Linguagem de Programação. Elementos de programação imperativa: variáveis simples, coleções uni e n dimensionais; tipos das variáveis; comandos; procedimentos e funções. Conceitos fundamentais do paradigma de Programação Orientada a Objetos: objeto, atributo, método, classe. Prática de programação usando alguma linguagem de programação orientada a objetos.

### 6) Objetivos

**Geral:** Apresentar as noções básicas de programação de computadores capacitando os alunos a analisar problemas de complexidade básica e projetar/desenvolver soluções de software sob a perspectiva de orientação a objetos.

**Específicos:**

- Apresentar os conceitos fundamentais da programação orientada a objetos.
- Capacitar o aluno a analisar problemas de complexidade básica, abstraindo, modelando e implementando soluções sob o enfoque de programação orientada a objetos.
- Desenvolver fluência em uma linguagem de programação orientada a objetos.

### 7) Conteúdo Programático

- 7.1) CONTEXTUALIZAÇÃO [8 horas-aula]
  - Modelo conceitual
  - Processos de abstração e representação
  - Histórico sobre linguagens de programação
- 7.2) CONCEITOS BÁSICOS DA ORIENTAÇÃO A OBJETOS [10 horas-aula]
  - Classes e objetos
  - Atributos
  - Métodos, argumentos e parâmetros
- 7.3) CONCEITOS BÁSICOS DE PROGRAMAÇÃO IMPERATIVA [40 horas-aula]
  - Algoritmos e programas
  - Processo de edição, compilação e execução
  - Variáveis e tipos de dados
  - Comando de atribuição
  - Operadores aritméticos e lógicos
  - Estruturas de seqüenciação, decisão (simples e compostas) e de repetição (condicionais e contadas)
- 7.4) MODELO DE OBJETOS [20 horas-aula]
  - Comunicação por troca de mensagens
  - Encapsulamento e ocultamento de informações

- Hierarquia de agregação/decomposição
- Hierarquia de especialização/generalização
- Herança e polimorfismo

#### 7.5) COLEÇÕES [30 horas-aula]

- Cadeias de caracteres (String)
- Coleções unidimensionais (Array)
- Coleções bidimensionais (Matrizes)

### 8) Metodologia

O conteúdo teórico será apresentado em sala por meio de aulas expositivas. As experimentações (desenvolvimento de programas) serão realizadas em laboratório. É esperado que cada estudante envolva-se ativamente na resolução das listas de exercícios propostas. As possíveis soluções para alguns dos exercícios das listas serão discutidos durante as aulas no laboratório.

Algumas aulas práticas serão acompanhadas pelo aluno de mestrado do PPGCC Ray Neiheiser, em seu Estágio de Docência.

### 9) Avaliação

A avaliação é feita por meio de 3 provas individuais (P1,P2,P3) e 1 Trabalho Prático (TP). As provas são individuais, sem consulta e predominantemente práticas, onde o aluno deverá escrever programas ou partes de programas usando a linguagem Java. Entretanto, algumas questões teóricas (conceituais) poderão ser cobradas. O Trabalho consiste de questões práticas de programação envolvendo os conceitos trabalhados no semestre.

A média final (MF) de cada estudante é calculada pela fórmula

$$MF = ((P1) + (P2 * 2) + (P3 * 3) + (TP * 2) ) / 8$$

Dado que a disciplina apresenta pelo menos 50% da carga horária consistindo de aulas práticas, conforme deliberação do Colegiado do Curso de Ciências da Computação de 18 de março de 2008, ela não prevê a realização de avaliação no final do semestre (recuperação) de que trata o parágrafo 2º do artigo 70 da Resolução 17/CUn/97.

### 10) Cronograma

As provas serão realizadas, aproximadamente, ao final de cada mês do semestre. As datas exatas serão divulgadas no Moodle.

### 11) Bibliografia Básica

- BORATTI, Isaias C. Programação Orientada a Objetos em Java. Florianópolis: VisualBooks. 2007.
- BORATTI, Isaias C. e OLIVEIRA, A. B. Introdução a Programação – Algoritmos. Visual Books, 3 Ed. 2007
- CAMARÃO, C. e FIGUEIREDO, L. Programação de Computadores em Java. Rio de Janeiro: LTC. 2003.
- DEITEL, Harvey M.; DEITEL, Paul J. Java: como programar. 3. ed. Porto Alegre: Bookman, 2001.

### 12) Bibliografia Complementar

- BOACH, Grady. Object-oriented analysis and design: with applications. 2nd ed. Reading: Addison-Wesley, 1994.
- BOOCH, G., Object-Oriented Design. Benjamin/Cummings Pub. 1998.
- BUDD, Timothy. Understanding object-oriented programming with Java. Updated ed. Reading: Addison Wesley, 2000.
- HORSTMANN, C. S. e CORNELL, G. Java 2 Volume I – Fundamentos. São Paulo, Pearson Education, 2003.
- JIA, Xiaoping. Object-oriented software development using Java: principles, patterns, and frameworks. Reading: Addison-Wesley, 2000.
- MEYER, Bertrand. Object-oriented software construction. 2nd. ed. Upper Saddle River, NJ: Prentice-Hall PTR, 1997.
- NEWMAN, Alexander. Usando Java: o guia de referência mais completo. Rio de Janeiro: Campus, 1997.
- RUMBAUGH, James et alii. Modelagem e Projetos Baseados em Objetos. Ed. Campus, 1994
- SANTOS, R. Introdução à Programação Orientada a Objetos usando Java. São Paulo: Campus, 2003.
- SOUZA, Marco F. de Souza; et al. Algoritmos e Lógica de Programação. São Paulo: Thomson Learning, 2005.
- WALNUM, Clayton. JAVA em exemplos. São Paulo: MaKron Books do Brasil, 1997.
- WAZLAWICK, Raul S. Análise e Projeto de Sistemas de Informação Orientados a Objetos. São Paulo: Campus. 2004.